Bus Inf Syst Eng
https://doi.org/10.1007/s12599-024-00886-3

RESEARCH PAPER

Ontology-Driven Automated Reasoning About Property Crimes

Francisco Navarrete · Ángel L. Garrido · Carlos Bobed · Manuel Atencia · Antonio Vallecillo

Received: 4 May 2023 / Accepted: 11 June 2024
© The Author(s) 2024

Abstract The classification of police reports according to the typification of the criminal act described in them is not an easy task. The reports are written in natural language and often present missing, imprecise, or even inconsistent information, or lack sufficient details to make a clear decision. Focusing on property crimes, the aim of this work is to assist judges in this classification process by automatically extracting information from police reports and producing a list of possible classifications of crimes accompanied by a degree of confidence in each of them. The work follows the design science research methodology, developing a tool as an artifact. The proposal uses information extraction techniques to obtain the data from the reports, guided by an ontology developed for the Spanish legal system on property crimes. Probabilistic inference mechanisms are used to select the set of articles of the law that could apply to a given case, even when the evidence does not allow an unambiguous identification. The proposal has been empirically validated in a real environment with judges and prosecutors. The results show that the proposal is feasible and usable, and could be effective in assisting judges to classify property crime reports.

Keywords Property crimes · Ontologies · Knowledge graphs · Information extraction · Uncertainty

Accepted after two revisions by Pnina Soffer.

F. Navarrete
Junta de Andalucia and ITIS Software, Universidad de Málaga, Málaga, Spain
e-mail: franciscoj.navarrete@juntadeandalucia.es

Á. L. Garrido · C. Bobed
Universidad de Zaragoza, Zaragoza, Spain
e-mail: garrido@unizar.es

C. Bobed
e-mail: cbobed@unizar.es

M. Atencia · A. Vallecillo (✉)
ITIS Software, Universidad de Málaga, Málaga, Spain
e-mail: av@uma.es

M. Atencia
e-mail: manuel.atencia@uma.es

1 Introduction

In the Spanish judicial system, criminal complaints (police reports) filed with the police or security forces are submitted to the duty court in the city where the criminal act took place to be processed for possible prosecution. These reports are text documents written using natural language following a basic structure.

When a report arrives at the court, it is classified according to the typification of the criminal act described. This classification is carried out by a judge and, optionally, by a prosecutor, in order to determine one of three possible actions: (a) it is accepted and scheduled for a trial; (b) the investigation should continue if there is not enough evidence for a trial to be held; (c) the report is filed if the evidence is very weak, e.g., the author of the crime or the stolen property could not be identified.

This classification process is often slow and difficult, and it usually has to deal with missing, inaccurate, or even inconsistent information, or a lack of sufficient detail to make a clear decision. Another problem is the high number of police reports that the courts have to handle. For example, the city of Malaga alone receives more than 82,000 criminal cases per year. The city has 14 examining courts, each with one magistrate, making a total of 5857 criminal cases per judge per year (CGPJ 2023). Any assistance in the initial classification of these reports can significantly impact the workload of judges and courts.

Our goal is to assist judges in this classification of property crimes by automatically extracting information from police reports and producing a list of possible classifications of crimes, accompanied by a degree of confidence in each one.

The work follows the design science research methodology (Johannesson and Perjons 2014), with the provision of a tool as the artifact to solve the problem. The proposal uses information extraction techniques to obtain the data from the reports. The extraction process is guided by an ontology we developed for the Spanish legal system on property crimes, called SPCO (Spanish Property Crime Ontology). The extracted information is used to populate the ontology, which is stored as a knowledge graph using a graph database. This ensures scalable access and efficient reasoning mechanisms to support decision-making. These decisions involve issues such as identifying the article of the law that an offense violates, resolving possible conflicts when more than one norm is violated, and typifying the offending actions with appropriate punishments. In particular, we show how in this domain it is possible to decide which norm should be selected, based on the evidence provided in police reports.

Another relevant contribution of our work is the algorithm used to select the set of norms that could be applied to a given case when the evidence is not sufficient to unambiguously identify one, a very common situation in practice. In this case, the algorithm is able to quantify the uncertainty associated with each norm, so that candidate options can be ranked according to their probability of being correct, or simply discarded when their associated uncertainty is high enough.

The proposal has been validated using real police reports on property crimes from Spanish courts. Such documents present a homogeneous structure with a restricted domain language – language economy is important as they are a means to communicate between different law agents. This has been exploited to analyze them using Natural Language Processing (NLP) techniques. Their information has been extracted to populate a Neo4j database, using the SPCO ontology to provide the terminology. The resulting decisions produced by our set of rules have been contrasted with real decisions to evaluate our proposal's accuracy. The results show that our proposal can assist judges in the classification of police reports, helping them determine the decision to be made on the report.

The paper is organized as follows. After this introduction, Sect. 2 briefly describes the background of our proposal and relates it to existing similar works. The problem addressed by this work and the requirements for a tool to solve it are described in Sect. 3. The next three sections describe our proposal. First, Sect. 4 presents the ontology we have defined to represent and manage all the information related to property crimes according to the Spanish legal system. It also describes the representation of the ontology as a knowledge graph and the corresponding Neo4j implementation we have developed to store it. Next, Sect. 5 describes the process followed to populate the ontology from the police statements that report the offenses, using NLP and information extraction techniques. Then, Sect. 6 shows the rules defined to reason about property crimes, in particular, how to classify each offense based on the available evidence, and the algorithm we have developed to identify the potential set of articles of the law when there is uncertainty due to missing or redundant evidence. After that, Sect. 7 presents the results of the evaluation exercise we have conducted to assess our proposal and discusses its main advantages and limitations. Finally, Sect. 8 concludes with an outline of future work.

2 Background and Related Work

2.1 Computational Law and Legal Ontologies

Our proposal follows the computational law approach (Love and Genesereth 2005). Computational law, also called legal computing, is an interdisciplinary research field that advocates the use of knowledge representation and reasoning techniques to develop computer systems that automate judicial tasks, such as crime classification, legal argumentation, simulation of legal decisions, and indexing and organization of legal cases. The ultimate goal is to assist judges, lawyers and other actors in the judicial system in their day-to-day decision-making. In our case, we aim to assist judges in making decisions on police reports of property crimes.

Unlike black-box AI systems (Chalkidis et al. 2019; Yang et al. 2022; Sivaranjani and Jayabharathy 2022), systems based on knowledge and reasoning can natively provide explanations to inferred decisions, which is of particular interest in the legal domain. This is the case with our tool too.

Ontologies are typically part of systems based on knowledge and reasoning (Studer et al. 1998), as they allow to formally describe the knowledge of an application domain. The language with which they are written is usually based on a logic for which there exist efficient inference mechanisms, such as OWL 2 DL (W3C OWL Working Group 2012). Ontologies are essential to computational law.

Different ontologies have been proposed in the legal field and, particularly, in criminal law. Some of them are legal core ontologies intended to develop specialized legal domain ontologies. This is the case of the LRI-Core ontology (Breuker et al. 2004), written in OWL, proposed to build ontologies for the management of criminal trial documents of EU countries, such as the Dutch Criminal Law Ontology (OCL.NL).

Another example of a legal core ontology is LKIF-Core, which plays a central role in the Legal Knowledge Interchange Format, LKIF (Hoekstra et al. 2009). LKIF is an architecture that serves as a translation between legal knowledge bases using different formats, and as a unifying formalism, based on OWL 2 DL and rules, that can be the basis for reasoning services of knowledge-based legal systems. LKIF-Core, as a central part of LKIF, enables the acquisition, exchange, and representation of legal knowledge. LKIF-Core has been specialized in a legal domain ontology that is used by OWL Judge (van de Ven et al. 2008), a tool that assists users in the normative assessment of individual cases by resolving conflicting norms based on the principle of *lex specialis* (the most specific norm takes precedence). LKIF rules have been illustrated with an example about support obligations, based on German family law, and the Carneades argumentation system (Gordon et al. 2007) has been used to load the example's rule base, make queries, and visualize results (Gordon 2008). A more recent example of legal core ontology is UFO-L (Griffo et al. 2015). UFO-L is grounded on the unified foundational ontology UFO (Guizzardi et al. 2022), and includes basic concepts of law based on Robert Alexy's Theory of Fundamental Rights.

In addition to the aforementioned OCL.NL, there are other legal domain ontologies in the literature focused on criminal laws. The most related to our work is OntoCrime (de Oliveira Rodrigues et al. 2016), which is an ontology for the description of property crimes based on the Brazilian Penal Code. OntoCrime is grounded on UFO-B (Guizzardi et al. 2022), an event-centered fragment of UFO, and covers crimes such as theft, robbery, robbery with death, damage, extortion and misappropriation. It aims to support decision-making processes such as the classification of agents' behavior and the inference of punishments. Like OntoCrime, SPCO was designed to describe property crimes. However, SPCO is based on the Spanish Criminal Code, which differs from the Brazilian Code. Thus, although some classes of both ontologies may have the same names, their definitions are different.

Soh et al. (2017) have proposed a general model for designing criminal law ontologies and rules. It consists of a general-purpose ontology covering common aspects of criminal law systems, and a methodology for designing judgment rules using the Semantic Web Rule Language, SWRL (Horrocks et al. 2004). The proposed model has been applied to a case of the Korean anti-corruption law.

Asaro et al. (2003) introduce an Italian ontology of crime that can be used for the development of tools that support the activity of judges. It describes concepts such as offender, behavior, event, circumstances, punishment, sanction and safety measures.

In the specific area of cybercrime, Park et al. (2009) developed a cyber-forensics ontology for cybercrime investigation and mining. Moreover, Bezzazi (2007) proposed a small ontology for case resolution, which is modeled as a classification problem. Both cases and articles are described using classes, and the task is to determine whether an article class subsumes a case class. Bezzazi also proposes the use of non-monotonic reasoning and external ontologies to clarify technical concepts that are not explicitly defined in criminal law articles, but may have an impact on final judge decisions. Instead of class subsumption, we resolve whether an article applies to a police report by instance classification, where the police report is the instance and the article is the class.

Finally, Bak and Jedrzejek (2009) propose an ontology-based model for economic crime of fraudulent disbursement. The model uses a minimal ontology, expressed in OWL DL, and rules, expressed in SWRL, translating fraudulent disbursement activities and their associated sanctions as defined in the Polish Penal Code.

Our SPCO ontology is, to the best of our knowledge, the first property crime ontology based on the Spanish penal code. Existing criminal code ontologies are based on the criminal codes of other countries, as mentioned above. These codes are different from the Spanish one, so we could not reuse any of them in their entirety. Importing specific fragments of these ontologies was not the best option either, since, in addition to not being a simple task in general (Suárez-Figueroa 2012), and not as common a practice as one might think (Fernández-López et al. 2019), in the case of legal domain ontologies, the risk of collision is high and can be fatal in practice, because the same offense may be classified differently by different legal systems, and also penalties may vary from country to country.

We also did not want to use basic legal ontologies, as we were concerned about the performance of the tool. Therefore, we followed a bottom-up approach, starting with property crimes, making sure that the tool worked for this case, and leaving the extension to other types of crimes for the future. Furthermore, most of the reviewed papers propose legal ontologies that can be used for the development of knowledge-based legal systems, but they do not go beyond examples of proof-of-concept applications. Moreover, none addresses the uncertainty inherent in legal documents, as well as the divergence of judges' views in interpreting them, which also distinguishes our work.

2.2 Knowledge Graphs

Knowledge Graphs (KG) have been used in Computer Science for quite a long time under different names (e.g., Semantic Networks); however, they have attracted a lot of attention lately since Google adopted the term in 2012. While there is not a consensual definition of what a Knowledge Graph is (Gutiérrez and Sequeda 2021), authors Hogan et al. (2022) adopt the following definition without binding the definition to any particular data model; they define a Knowledge Graph as "*a graph of data intended to accumulate and convey knowledge of the real world, whose nodes represent entities of interest and whose edges represent relations between these entities.*"

Being more specific, a common definition considers KGs as labeled directed graphs (E, R, L), where the nodes of E represent entities, the edges of R represent relations between the entities, and L is a labeling function that maps each element in the graph to its name/type (Chaudhri et al. 2022). Thus, elements of a KG can be simply regarded as triples $\langle s, p, o \rangle$—the so-called *subject-predicate-object* (SPO) triples—where $s, o \in E$, $(s, o) \in R$ and $p \in L$. Such triples provide structured representations of real-world entities and relations that describe relational facts, making it possible to capture knowledge from heterogeneous sources, and reason about the information stored in the KG. For example, KG triples can be directly considered as binary facts of first-order logic, which evaluate to true or false, e.g., *capitalOf(Madrid,Spain)*. This encoding allows the use of first-order logic methods to reason about the knowledge described in the KG, such as inferring properties and relations, or developing Q &A systems.

A KG is usually stored and managed in a Graph database (Robinson et al. 2015), upon which a reasoning layer can be deployed to interpret and manipulate it. Basically, graph databases are databases that use graph structures to perform semantic queries. Two main data organizations are typically used: *triplestores*, where triples are first-class citizens and the KG information is directly stored as sets of triples; and *property graphs*, where, in contrast, nodes and edges are the main data elements, storing the data as labeled *nodes* and *edges*, which can have key/value *properties* associated with them. Examples of triplestores include GraphDB, Virtuoso, or Fuseki; while property graphs include, e.g., Neo4j, JanusGraph or ArangoDB. They count on specialized graph query languages such as Gremlin, Cypher, SPARQL, or GraphQL to interrogate the Graph database and manipulate its information.

There are several works on the use of KGs to represent legal ontologies. For example, Filtz (2017) models the Austrian legal norms and court decisions by means of a KG, usable in various applications for lawyers, attorneys, citizens, or journalists. The GitHub repository by the Liquid Legal Institute (2021) provides a list of resources, methods, and tools dedicated to legal ontologies, data schemes, and knowledge graphs.

2.3 Information Extraction

Information Extraction (IE) is the area of Natural Language Processing (NLP) that allows obtaining automatically structured data from documents in natural language. Focusing on the legal domain, according to Sansone and Sperlí (2022) these IE systems can be classified into three main categories: NLP, Deep Learning-based approaches, and ontology-based systems.

1. NLP-based systems: NLP techniques have recently improved the analysis, indexing and retrieval of large document repositories in the legal domain (Zhong et al. 2020). Some approaches generate and select features to guide the supervised classification of documents (Biagioli et al. 2005; Bommarito et al. 2018). Other approaches extract relevant concepts or terms from the set of legal documents using rules and restrictions (de Maat and Winkels 2010; Mok and Mok 2019), as well as lexical patterns (Brighi and Palmirani 2009; Sleimi et al. 2021).
2. Deep Learning-based systems: Machine and deep learning models are currently used for legal document classification, translation, summarization, contract review, case forecasting, and information retrieval (Chalkidis and Kampas 2019). Semantic search has also benefited from the use of machine learning techniques for context identification and word analysis (Bansal et al. 2019). These approaches are based on the assumption that words used in similar contexts are very likely to be semantically similar (Mikolov et al. 2013). Language models based on the transformer architecture (Tunstall et al. 2022) are also being used in the legal domain (Mokanov et al. 2019; Nguyen et al. 2020; Chalkidis et al. 2020; Vuong et al. 2023).
3. Ontology-based systems: Ontologies are especially useful as they make it possible to capture the knowledge to guide the extraction Chandrasekaran et al. (1999), and are widely used in this context (Corcho et al. 2005; Getman and Karasiuk 2014; Palmirani et al. 2018; Humphreys et al. 2021). The IE systems that use ontologies to extract information are known as OBIE (Ontology-Based Information Extraction) systems (Wimalasuriya and Dou 2010). Using a more general approach, Gutierrez et al. (2016) present the Ontology-based Components for Information Extraction (OBCIE) architecture, which promotes reusability through modularity, and enables orthogonal extensions

to allow the construction of hybrid OBIE systems with higher extraction accuracy and newer functionalities.

Our work could be considered as an OBIE system. A prominent example of an OBIE system for the legal domain is AIS (Buey et al. 2016), which allows extracting relevant information from legal documents written in natural language. The extraction process is guided by an ontology that stores the knowledge about the structure and the content of different types of documents, and by a set of appropriate extraction operations. Then, Buey et al. (2019) showed how ontologies could be effectively used to validate and improve the quality of the results of IE processes. Both proposals were validated in real settings. Recently, Sovrano et al. (2020) showed how legal knowledge extracted from heterogeneous legal sources can be represented using ontology design patterns for mapping the textual information into the knowledge base. The knowledge base is stored in a KG that can be queried by legal experts to retrieve the relevant information. In the same line of work, Carnaz et al. (2019) present an ontology population work related to SEM (Simple Event Model) with instances retrieved from crime-related documents, supported by an SVO (Subject, Verb, Object) algorithm using hand-crafted rules to extract events.

The main difference between these works and ours is threefold: the alignment of norms and criminal acts, the use of information extraction techniques based on real police reports, and the use of a specific algorithm to select the set of norms that could be applied to a given case, even when the evidence does not allow unambiguous identification.

3 Problem Description and Proposed Solution

3.1 Methodology

In this work we have followed the Design Science Research (DSR) methodology (Johannesson and Perjons 2014; Wieringa 2014) because our research focuses on the development and performance of an artifact with the explicit intention of providing a solution to a practical problem of interest to a particular group of users or practitioners. DSR ensures that the artifacts that solve the problem are built correctly (Dresch et al. 2015).

The DSR methodology defines a set of activities (Johannesson and Perjons 2014) that should be performed.

1. Explicating the problem.
2. Defining the requirements for a solution artifact.
3. Designing and developing the artifact.
4. Demonstrating the developed artifact by showing how it can be applied to solve a problem instance.
5. Evaluating the artifact to validate the extent to which the artifact solves the problem and satisfies the requirements.

As a result of the evaluation phase, some iterations may be necessary before the final artifact is produced (Wieringa 2014).

The rest of the paper is organized according to these activities. Next, Sect. 3.2 describes the problem and its context. Then, Sect. 3.3 presents the requirements for the artifact that we propose to solve the problem, while Sect. 3.4 describes its architecture and constituent components. The next three sections describe these components in detail. First, Sect. 4 presents the SPCO ontology used to represent and manage all the information related to property crimes. Sect. 5 describes the information extraction process followed to populate the ontology from the police reports. Sect. 6 shows the rules defined to reason about property crimes and demonstrates how the developed artifact can be applied to solve particular problem instances. Finally, Sect. 7 shows the evaluation exercises conducted and discusses the advantages and limitations of our approach.

3.2 The Spanish Criminal Law and Property Crimes

Unlike other countries whose legal system is based on similar cases, the Spanish legal system is based on civil law. Each law is composed of two separate parts: the conduct and the penalty. Thus, criminal law comprises a set of legal norms (or rules) that define possible offenses, their types, mitigating and aggravating circumstances, and associated criminal sanctions.

Crimes against property are defined in Organic Law 10/1995, of 23 November, of the Spanish Criminal Code. Beyond its economic value, for criminal purposes, the value of the property also includes the affective or moral value for the owner. Crimes against property include Theft, Robbery, Extortion, Vehicle theft and robbery, and Usurpation. In this paper we focus only on the two first types of property crimes, theft and robbery.

1. Theft: Taking a chattel (movable property), for himself or for others, without the will of its owner. The offender will be punished with a prison sentence of one to three months if the amount stolen does not exceed 400 euros; six to eighteen months if it exceeds 400 euros; or up to three years if the stolen goods are of special interest, e.g., artworks or public infrastructures (Arts. 234–236).
2. Robbery: Taking a chattel, for himself or for others, by force, violence or intimidation. The offender will be punished with a prison sentence of one to three years.

It can be of two to five years if the stolen goods are of special interest (Arts. 237–242).

Penalties can be modified depending on the particular circumstances, such as a robbery outside business opening hours or in the case of first-time offenders.

To illustrate how articles are defined in that Law, the following three paragraphs correspond to the definition of Article 234, about thefts:

234-1. Anyone who, for profit, takes another's chattel without the owner's consent shall be punished, as a theft offender, with a prison sentence of six to eighteen months if the amount of the stolen property exceeds 400 Euros.

234-2. A fine of one to three months shall be imposed if the amount of the stolen goods does not exceed 400 Euros unless any of the circumstances of Article 235 apply. However, if the offender has been convicted of at least three offenses under this Title, even of a minor nature, provided that they are of the same nature and that the cumulative amount of the offenses exceeds 400 Euros, the penalty of paragraph 1 of this Article shall be imposed.

234-3. The penalties established in the previous sections shall be imposed in their upper half when, during the commission of the offense, the alarm or security devices installed on the stolen property have been neutralized, removed, or rendered useless by any means.

3.3 Requirements for a Solution Artifact

Our goal is to assist judges in this classification process by providing a tool (the DSR *artifact*) that is able to extract information from police reports and produce a list of possible classifications for the offenses and their associated punishments, accompanied by a degree of confidence in each one. As prescribed by the design science methodology, the first step is to formulate the design problem using the DRS template (Wieringa 2014):

- To assist the current classification of police reports in Spain
- by providing a tool
- that is able to read the police reports and produce a list of possible classifications for the offenses accompanied by a degree of confidence in each one
- so that judges can decide if the report is accepted and scheduled for a quick trial, the investigation should continue, or the report is filed.

To meet these design goals, the solution has to satisfy the following requirements:

R1. It is able to extract information from the original police reports on thefts and robberies, identifying the actors, elements, and relevant details of the reported crimes.

R2. It is able to identify the Articles of the law that might apply to the offense, and how each matches the information extracted.

R3. It is able to assign a probability to each identified Article, which represents its likelihood based on the information not only present in the report but also absent from it.

In addition, three requirement satisfaction questions about the designed artifact should be addressed:

S1. Are the results produced by the tool reliable?
S2. Are the results produced by the tool understandable?
S3. Is the proposed solution usable by judges?

Reliable results mean that they can be trusted. Given that there is no ground truth to compare with because both the facts in the reports and the laws are subject to interpretation, *reliability* will be assessed in two ways (see Sect. 7.2). First, reliability will be empirically assessed by directly asking the judges whether they consider the results produced by the tool in a series of experiments to be correct. In addition, reliability will be evaluated by measuring the degree of agreement between the ratings produced by the tool and those produced by several judges on the same reports. For this, we will use the Fleiss Kappa (Fleiss 1971) index, a measure of inter-rater reliability. After conducting controlled experiments with judges, the degrees of *understandability* of the results and the *usability* of the tool will be measured empirically. They are subjective measures, based on the judges' opinions. *Understandability* of the results means that they are presented so that judges can easily comprehend them. *Usability* refers to the degree to which judges can use the tool to classify crime reports with effectiveness, efficiency and satisfaction (ISO/IEC 2011).

3.4 Proposed Solution

To achieve our goals, our solution artifact is based on three main components as depicted in Fig. 1. In a first step, a semantic information extraction based on the ICIX (Insynergy Consulting Information eXtraction) architecture and using the SPCO ontology is performed on the police reports. ICIX (see Sect. 5.2) is an OBIE system in which the typology and structure of the documents, the entities to be extracted, and the extraction rules are specified using an ontology. This results in a first RDF graph describing the

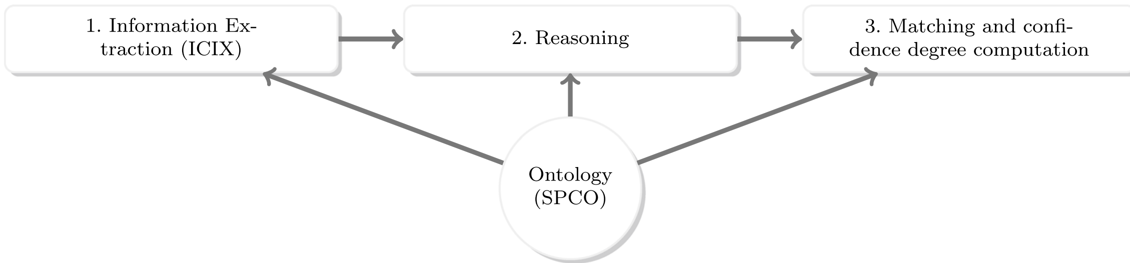

Fig. 1 Sequence of steps of the proposed solution and their functional dependence on the property crime ontology (SPCO)

information contained in the reports and expressed in terms of classes and properties of the SPCO ontology.

In a second step, a reasoner is fed with the RDF graph from the first step and the SPCO ontology to obtain an extended graph that includes a pre-classification of the alleged offense. If the information from the police reports is complete, the pre-classification is final, i.e. the crime is classified into an article class that carries a penalty; if incomplete, the crime is classified into a superclass of several possible article classes, and a third step is required. In the third and final step, the RDF graph from the second step is extended and labeled, for each of the possible articles of the pre-classification, with the missing, redundant and disregarded information. Then, a matching algorithm computes the similarity between the new RDF graph and each of the articles, expressed as graph patterns. This results in matching probabilities, which translate into degrees of confidence in applying each article.

These components will be described in Sect. 5 and 6. However, since the SPCO is the cornerstone of our approach, for readability's sake we will describe it first in the next section. Note that all these components form an integral part of the solution artifact that constitutes our contribution.

4 An Ontology for Property Crimes

4.1 The SPCO Ontology

This section describes the domain ontology that we have developed to represent the articles of the Spanish law on crimes against property, namely Titles I and II on theft and robbery, i.e., Articles 234–242. The ontology has been implemented using Prótége 5.5.0, and Pellet 2.2.0 as the reasoning engine. It contains 408 axioms over 112 classes and 35 properties, and its expressivity is $\mathcal{ALCHIQ}(\mathcal{D})$ (Baader et al. 2003).[1]

Figure 2 shows the upper layer of the SPCO ontology, with the most general concepts:

- `Report` is an abstract concept that represents the document base that triggers a judicial action at the court when a police report is registered. The refined concept, `PropertyCrimeReport`, represents a report of a property crime.
- `OffenseElement` represents the different components that constitute an offense or its effects. Derived concepts include the stolen objects, the characteristics of such objects, the actors involved in the report (physical and/or legal persons) and the factors that aggravate or mitigate the effects of the offense.
- `OffenseCharacteristic` and its subconcepts represent the qualities that individually or in combination will determine the typology of an offense.
- Finally, `Punishment` is used to derive the punishment to be applied depending on the offense.

Note that some of the concepts are *defined*, i.e., we have stated necessary and sufficient conditions for an instance to belong to them, so such a fact will be inferred when its condition holds. For example, `RobberyCrimeReport` is a `PropertyCrimeReport` that describes something stolen by a person and has any of the properties that characterize a robbery. In OWL, the axiom that defines this concept is the following:

```
RobberyCrimeReport ::=
  PropertyCrimeReport
  and (stolenThing some (stolenBy some Accused))
  and (hasOffenseCharacteristic
                some RobberyCharacteristic)
```

OWL, based on Description Logics (DL), makes the open world assumption (OWA). According to this assumption, given a state of affairs, the absence of information is treated as unknown, as opposed to the closed-world assumption (CWA), where it is interpreted as negative information. The OWA is the most reliable assumption for knowledge representation, as long as it is not possible to guarantee that all information has been provided, or is not yet available.

[1] $\mathcal{ALCHIQ}(\mathcal{D})$ allows negation, intersection and union of classes, existential, universal and qualified cardinality restrictions, subproperties, inverse and datatype properties.

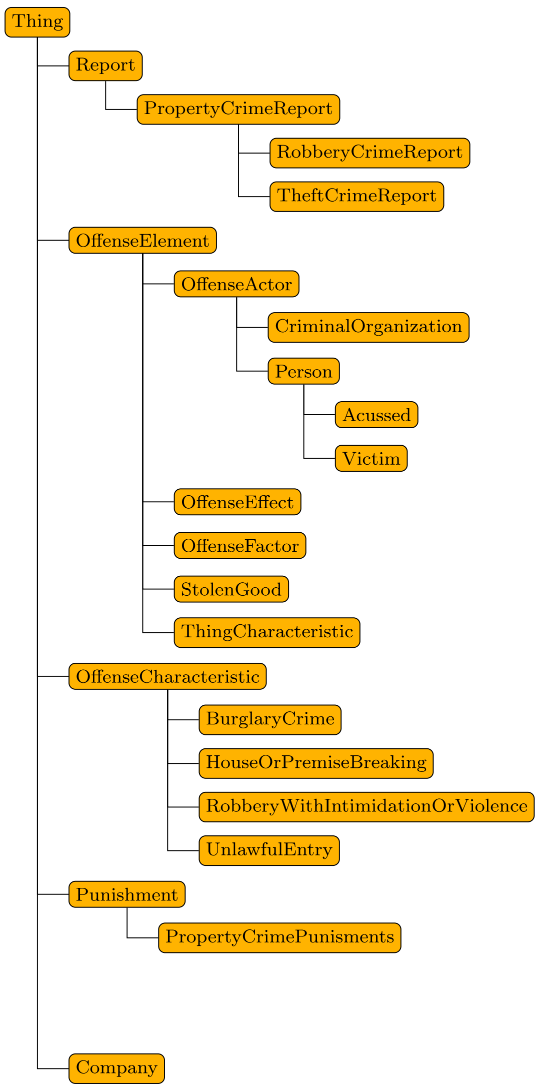

Fig. 2 Upper layer of the SPCO ontology

Figure 3 shows the middle layer of the SPCO ontology, with the concepts that represent robbery and theft crimes in the Spanish Law (Titles I and II). This hierarchy illustrates the classification of concepts covering all theft and robbery crimes. SPCO contains the elements that determine the characteristics of the theft (as properties, not shown in Fig. 3), including the value of the stolen good, if it was taken by force, or if there was violence/intimidation of the persons possessing or guarding it. Our model enables the use of a DL reasoner to derive some of these concepts automatically.

When modeling this part of the ontology, note the slight naming abuse as, obviously, `Articles` are not subclasses of crime reports; however, for the sake of usability, we opted for shortening the concept names (otherwise, e.g., `Article234_1` could read as `TheftCrimeReport TheftByOwnerArticle234_1Applies`).

The articles defining property crimes also include the penalties they entail, which stipulate the possible consequences and punishments for violating the rules, called legal effects. Although the article to be applied can be derived directly using information from their predecessors, their penalties must be calculated differently because they depend on many other factors. To capture this, we model these broader rules within DL by using General Concept Inclusion (GCI) axioms, i.e., `subClassOf` axioms where we have a complex class expression on the left-hand-side. This model allows us to establish that, whenever an offense meets all the conditions (in this case, stated as belonging to different concepts on the left-hand-side), it entails an associated punishment (expressed as an existential assertion of the `hasPunishment` relationship). For example, a theft which is classified as a crime against property under Article 234.2 is punishable by a prison sentence of between one and three months.

Finally, the SPCO includes a set of annotations for each relationship, stating whether or not they are *necessary* for the application of each article, as well as their *relevance* to the article. Note that, at first sight, all facts in the GCIs should be necessary; however, sometimes there are either missing or redundant facts in the report, and the degree of relevance of those facts needs to be established. While this knowledge is not considered by the DL reasoner, it is fundamental to reason with imprecise and incomplete information (see Sect. 6).

4.2 A Motivating Example

To illustrate how police reports about property crimes are represented using the SPCO ontology, this section describes one example from a real police report (pseudonymised to respect the privacy of the persons involved and to comply with EU Regulation 2018/1725).

Typical police reports consist of a declaration form describing the details of the complainant and the victim, the location, date and time of the offense, the stolen properties, their value, and the circumstances of the inci-

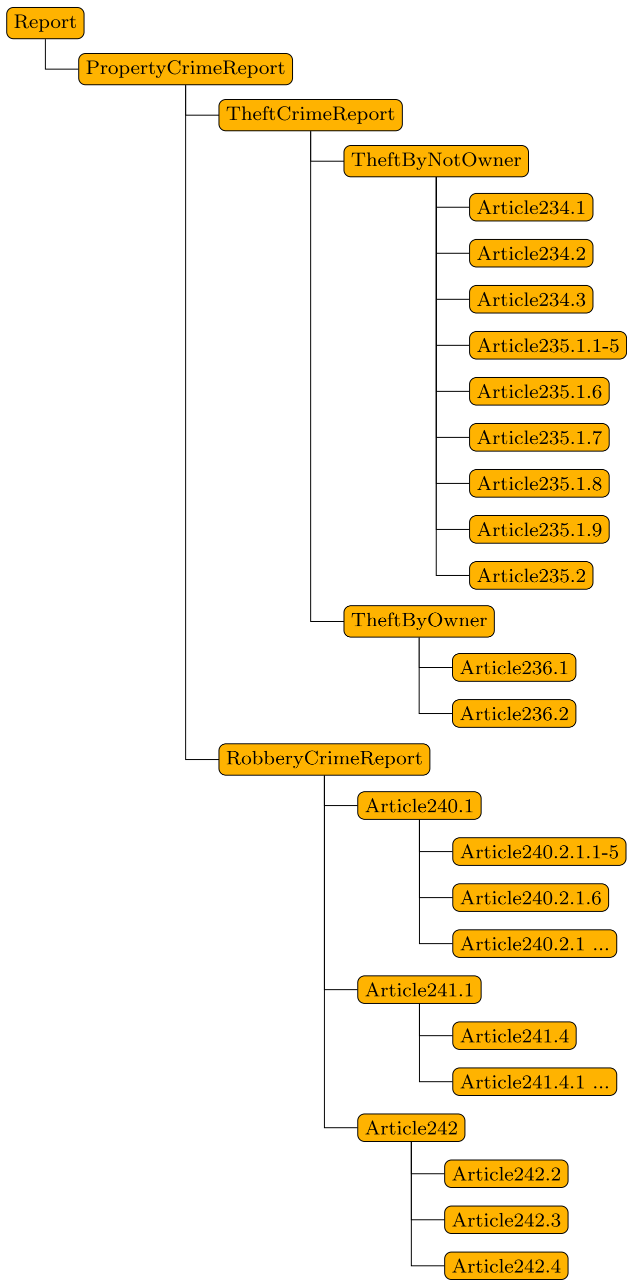

Fig. 3 Classes related to `PropertyCrimeReport`

dent. This form is either completed at the place of the theft or at the police station where the report is filed. The reports may subsequently be supplemented by other documents in which the police officers record further information, such as the details of the accused if identified, their criminal records, etc.

In this case, a person identified as GHEH presents himself at a local police station in the city of Málaga in November 2021, stating that a person has stolen his wallet with 70 Euros and documentation. The complainant also gives a physical description of the alleged thief, indicating that she is a woman approximately 1.65 meters tall, with blonde hair, slim, light eyes, and details the clothes she is wearing. Following the statement, a report is generated. From the description of the thief, the police identify a person who is a regular in the area, and who is known to have carried out similar acts in the past. After a routine check by other police officers in the area, she is identified and arrested.

Below we show some of the `RDF` triples in `Turtle` format that define this report in our ontology. They are also graphically shown in Fig. 4.

```
@prefix spco: <http://www.../ontologies/spco>

### spco#R_26877_21
spco:R_26877_21 rdf:type owl:NamedIndividual ,
spco:Report ,
[ rdf:type owl:Class ;
  owl:complementOf [ rdf:type owl:Restriction;
   owl:onProperty spco:hasoffenseCharacteristic ;
   owl:someValuesFrom spco:RobberyCharacteristic
   ]
] ;
spco:stolenThing spco:wallet .

### spco#wallet
spco:wallet rdf:type owl:NamedIndividual ,
  spco:StolenGoods ;
  spco:belongsTo spco:GHEH ;
  spco:stolenBy spco:MV ;
  spco:usedBy spco:GHEH ;
  spco:valueCost 70.0 .

spco:GHEH rdf:type owl:NamedIndividual ,
  spco:Victim ;
  spco:Age "58"^^xsd:int .

### spco#MV
spco:MV rdf:type owl:NamedIndividual ,
    spco:Accused ;
    spco:Age "38"^^xsd:int
```

Once we have these facts, we can infer the concepts and relations that can be applied to the report example with the help of a DL reasoner. Below we show the same RDF triples after applying the DL reasoner to the proposed example on the SPCO ontology. They are also graphically shown in Fig. 5.

```
@prefix spco: <http://www.../ontologies/spco>

### spco#R_26877_21
spco:R_26877_21 rdf:type owl:NamedIndividual ,
 spco:Report ,
 spco:PropertyCrimeReport ,
 spco:TheftCrimeReport ,
 spco:TheftByOwner ,
 spco:Article234_2 ;
 [ rdf:type owl:Class ;
   owl:complementOf [ rdf:type owl:Restriction ;
   owl:onProperty spco:hasoffenseCharacteristic ;
   owl:someValuesFrom spco:RobberyCharacteristic
 ]
] ;
spco:stolenThing spco:wallet .

### spco#wallet
spco:wallet rdf:type owl:NamedIndividual ,
     spco:OffenseElement ,
      spco:StolenGood ;
      spco:belongsTo spco:GHEH ;
      spco:stolenBy spco:MV ;
      spco:usedBy spco:GHEH ;
      spco:valueCost 70.0 .

spco:GHEH rdf:type owl:NamedIndividual ,
    spco:OffenseElement ,
    spco:OffenseActor ,
    spco:Person ,
      spco:Victim ;
      spco:age "58"^^xsd:int .

### spco#MV
spco:MV rdf:type owl:NamedIndividual ,
    spco:OffenseElement ,
    spco:OffenseActor ,
    spco:Person ,
    spco:Accused ;
    spco:age "38"^^xsd:int
```

The chain of derivation rules to define the concept that determines that this example attestation derives to Article 234.2 is the following:

```
Article234_2 ::=
 TheftByNotOwner
 and (stolenthing some
 (valueCost some xsd:decimal[<= 400.0]))

TheftByNotOwner ::=
 Theft
 and (stolenthing some (usedByOwner some Person))

TheftCrimeReport ::=
 PropertyCrimeReport
 and (not (hasOffenseCharacteristic
 some RobberyCharacteristic))
 and (stolenthing some (stolenBy some Accused))

PropertyCrimeReport ::=
 Report
 and (stolenthing some (belongsTo some Victim))
```

However, this would only be the first stage of our two-step reasoning procedure (we sketch it here and will provide the details in Sect. 6): in order to reason about the whole set of reports, we decided to use a graph database as they provide very expressive query languages, which allowed us to express the types of statements we needed to perform the ad-hoc reasoning we needed to deal with imprecise and incomplete reports, and, additionally, easily navigating the knowledge base to visualize all the information.

5 Populating the Ontology Using Information Extraction Techniques

So far we have described the SPCO ontology and how it is used to represent both the information about property crimes and the articles of the law that define their associated penalties. In this section we describe the process that is able to extract the information from the police report documents, and populate the corresponding individuals of the ontology.

5.1 Characterizing the Documents: Reports

To enable automatic reasoning using our ontology, we need to populate the individual facts about the offenses (the ABox of the ontology). Currently, police officers fill in the reports using either a free text entry form (for most of the offenses) or, in the case of shoplifting and other minor crimes, a set of predefined templates, normally hand-written and scanned. In this paper we focus only on the first type of report, as the second comprises a limited set of templates that can be directly handled by OCR and a fixed set of ad-hoc extractors, cleaning the contents using an approach similar to that of Buey et al. (2019). Therefore, our input data will be a set of police reports written in natural language.

These police reports contain information on the date and time when the report was written, the persons appearing, the date, time and place of the criminal act, a detailed description of the criminal facts and the objects stolen, as well as the actions that have been taken since the original report (possible investigations, arrests, citations, etc.). At first glance, these police reports consist of completely free text. However, upon analysis, some regularities emerge. Due to the daily work of the officers, the contents have converged in such a way that most of them follow some unwritten style guidelines. Furthermore, if we group the documents by source precinct, we can identify more precinct-specific regularities; this allows us to adopt rule-based IE mechanisms, as the effort of writing rules is worthwhile given their precision and high level of reusability. Note that, in principle, this is applicable to the rest of Spain since the linguistic register used in the preparation of police reports should be the same across the country as long as the reports are in Spanish.[2] In the case of using co-official languages such as Catalan or Galician, the relevant adaptations would have to be made in the Spacy language configuration.

[2] Recall that the rule extraction can be done in a semiautomatic way, adapting the pattern detection from PATTY (allowing output patterns with just one named entity) to bootstrap the rule detection.

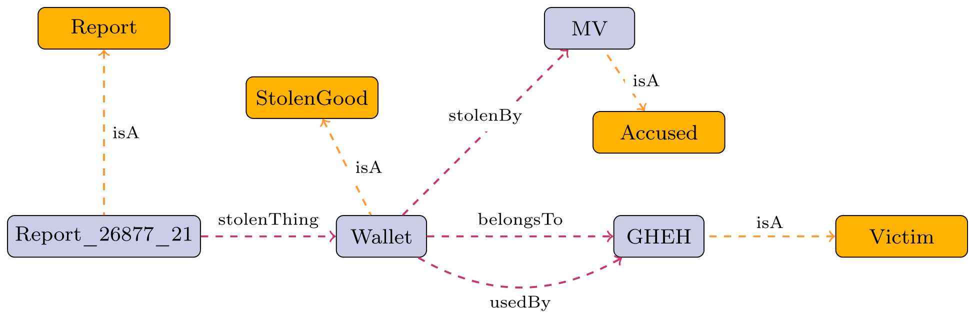

Fig. 4 Graphical representation of the example report(`R_26877_21_A`) provided by the IE tool. The classes identified by the IE subsystem are represented in orange and instances in indigo

To guide the extraction, we decided to adapt the techniques defined in (Buey et al. 2016; Garrido et al. 2021) to this particular domain. Given the criticality of the decisions to be made, we also decided to perform a semi-automatic IE step, keeping the officer in the loop, as suggested in (Opasjumruskit et al. 2022, 2020) (Fig. 6).

Regarding the texts, the most relevant characteristics that affect the IE process are the following:

- The documents have a flat structure but normally contain two main sections: the list of stolen objects and the different tasks associated with the report (in Spanish, "Diligencias"). These subsections are optional, but when present, they always have the same format. The list of objects starts with different variants of "Relación de objetos" (in Spanish) and each object is separately described in its own line of text, with all its related information. The list of tasks can span several paragraphs, but always starts with the formula "Task of" ("Diligencia de") and ends with "confirm and certify" ("Conste y certifico"). Thus, in both cases, we can identify the tokens that delimit the beginning and the end of these sections.
- Each paragraph is usually self-contained and is used to present a particular fact or element. For example, all information about the offender is normally contained within one single paragraph. This greatly alleviates the potential problems of co-references between paragraphs.
- In the paragraphs describing the plaintiff's statement, the discourse is usually articulated from the point of view of an objective bystander (the officer writing the report) who speaks about the complainant in the third person (the plaintiff being the focus of the narrative), which makes it easier to detect who is who and who does what because of the verbal tenses used.
- All persons appearing in the text are usually well identified (e.g., by identity cards or driving licenses). These identifications can be easily captured by standard rule-based matchings. For example, "*<PERSON>*, con DNI $[0-9]8[A-Z]$" would capture and bind the ID number of a person, where *<PERSON>* would be a text span tagged by a Named Entity Recognizer (NER). The rest of the entities/objects in the text (e.g., the stolen goods) do not need to be bound to concrete entities in the ontology (at most, they should be classified, but our ontology model is robust enough to reason with limited knowledge about them). This way, the usual complexity of entity linking is minimized.
- The same identification facilities apply to the report themselves. The reports might be standalone or refer to other previous reports (extending the information in them), but they always include the appropriate report identifiers in a closed format. Using the report identifiers directly in the KG as their node identifiers facilitates their linking when they are finally stored in the graph database.
- Special keywords and synonyms, as well as lexical patterns, are used to identify words or expressions that convey relevant meanings to the property crime. For example, they are used to identify particular offense characteristics such as the use of force or violence, or whether some devices installed on the property have been neutralized, removed, or disabled.

5.2 Extracting the Information

To populate the ontology from the reports, we adopted the ICIX architecture (Garrido et al. 2021), an approach already successfully applied to legal texts with the AIS system (Buey et al. 2016), the information extraction system integrated in *OnCustomer*[3], a commercial Content Relationship Management (CRM) system developed by the company *InSynergy Consulting*.[4]

In brief, ICIX is an OBIE system in which the typology and structure of documents, the entities to be extracted and the procedures/rules used to extract them are specified by

[3] http://www.isyc.com/es/soluciones/oncustomer.html (accessed 03 July 2024)

[4] http://www.isyc.com (accessed 03 July 2024)

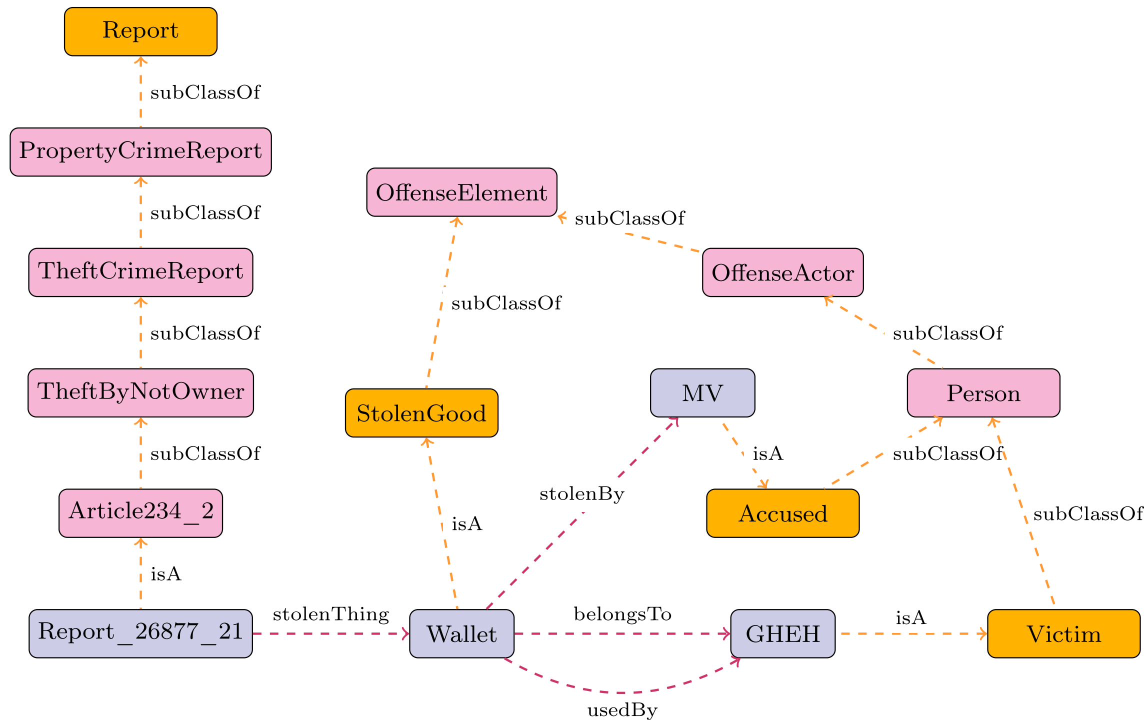

Fig. 5 Graphical representation of the exemplar report (R_26877_21_A) derived by the DL reasoner. Classes identified by the IE subsystem are colored in orange, instances in indigo, and classes derived by the DL reasoner in magenta

an external ontology. In our case, as we only have one type of document with a flat structure, the ICIX ontology extraction module is greatly simplified.[5] However, for the sake of completeness, an explanation of the main ICIX steps is included here. Figure 7 provides an overview of this architecture, which we have adapted to our current context. The dotted box limits the information extraction system, which is composed of four main services that perform the actual extraction process. The grey arrows represent the document data flows, and the white arrows represent the interaction of the users with the additional useful services. To apply ICIX in different contexts, it is only necessary to modify the knowledge base and the database.

1. Pre-processing service: This service cleans and corrects the text in the document as much as possible, regardless of its origin. The noise can come from, e.g., a scanned and OCR-processed document, or a pdf document with noisy formatting.
2. Classification Service: ICIX can handle different types of documents, and this step is necessary to classify them. Since we only deal with one type of document in this work, this step is not used.
3. Extraction Service: This is the core ICIX service, which queries the ontology to define the elements to search for in the text, and decides the extraction methods/rules to use to obtain the relevant information. To do so, it uses a two-step extraction process that requires a set of rules to detect the different sections of the document, and a set of section-dependent rules to extract the entities that may appear in each of them. To define both the sections and the elements to be extracted we use the SPACY rule-based matching mechanism (Honnibal et al. 2020).
4. Data Curation Service: Once the extraction process is completed, ICIX takes advantage of the knowledge stored in the domain ontology to perform a review of the extracted data, curating possible errors and improving the quality of the results by correcting and enriching them. In our case, given that the extracted information is specific to each report, the curation service is mostly delegated to the police officers, who are presented with the results and asked for validation, thus keeping them in the loop as suggested in Opasjumruskit et al. (2022, 2020).

The output of this process is a set of RDF triples according to the SPCO, along with the spans of text that have produced them. In this way, we can ask the officer at hand to validate the different facts that our system has detected, making it possible to detect errors and curate the input facts

[5] We refer the interested reader to Garrido et al. (2021) for an example of the ontology they developed in AIS for procurement documents.

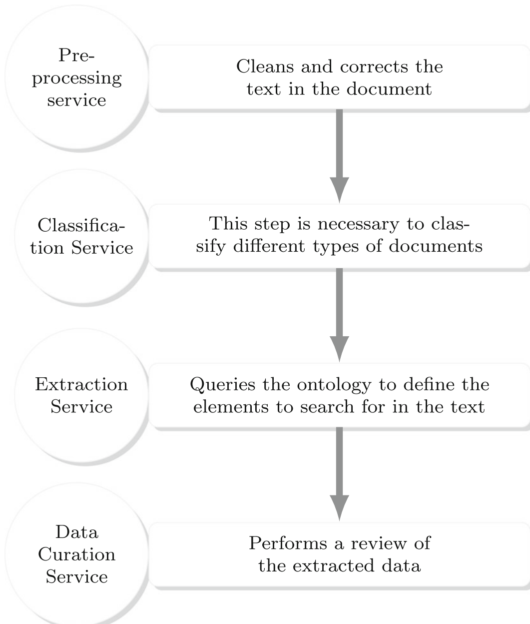

Fig. 6 ICIX architecture processes

before feeding the DL reasoner with them and start the reasoning step.

As a side note, given the regularities observed in the texts, we also considered using classical relation pattern detection techniques such as PATTY (Nakashole et al. 2012). At first, we implemented a lightweight version of PATTY in Spanish (without the full type system connected to the ontology, relying directly on the entity types detected by the NER model[6]) and some patterns emerged. However, the amount of anaphoras and omissions in the expressions used by the police officers hindered uncovering some interesting patterns that could easily be obtained by direct analysis of the texts: PATTY requires two entities in each sentence to find patterns in the expressions, and the style used – almost always omitting a particular central entity – obfuscated the finding of anchors in the text. Similarly, we discarded the use of other Machine Learning techniques as they require large amounts of data to achieve fine-tuned results. This said, as part of our future work, we plan to adapt these relation pattern extraction techniques to improve the extraction process by applying the knowledge we have gained about this particular type of text (hence reducing the amount of data required to make them work properly).

6 Reasoning About Property Crimes

In this section, we present our reasoning approach to obtain an explainable set of possible penalties given all the information about the offense and its participants. First, our solution artifact uses the knowledge modeled in the SPCO and the facts extracted in the report to infer new facts and classify initially the reported offense. Such materialized knowledge is integrated into the KG containing all the previous reports to bring together all the details that might be relevant for the offense classification but are missing in the single report. Finally, on top of the KG, we devised a reasoning algorithm to tackle incomplete or redundant information about the offense, producing a set of applicable articles along with their probability and explanations.

6.1 Inferring Facts from the Offense Report

As presented in Sect. 4, the way in which we model the offenses in SPCO allows our solution to directly use a DL reasoner in order to infer an initial classification of the offense according only to the information included in the police report. First, we have defined the offenses together with their articles as *defined* concepts to enable the DL reasoner to classify the offense as belonging to each particular article whenever the necessary and sufficient conditions are met. Second, we have extended these axioms with GCIs in other to establish the punishment via existential axioms (with GCIs, the reasoner can infer the right-hand-side – in this case, the applicable punishment – out from the left-hand-side of the axiom using classification).

In this reasoning step, we only work with the intensional knowledge of the SPCO ontology (its TBox) and the triples extracted from the report at hand. This allows us to obtain such an initial classification of the offense and a set of inferred facts, which we integrate in our KG. However, using only this information has a limitation: possible incompleteness. A report might not be self-contained and different previous details about the offense and the offenders might be scattered across different previous reports (which might as well influence the actual classification of the offense). More importantly, reports could also amend previous statements, which clashes with the monotonicity required by DL reasoning. Thus, to reason about the whole set of reports we needed to apply a reasoning algorithm capable of dealing with imprecise and incomplete (even contradictory) facts, which is presented next.

[6] The NER we have used is the one included in the SPACY Python library: https://spacy.io/api/entityrecognizer (accessed 03 July 2024).

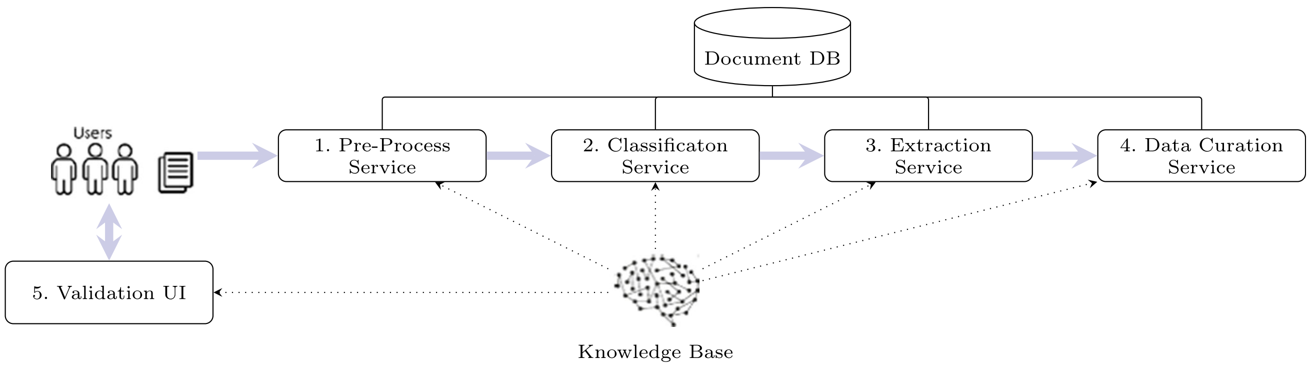

Fig. 7 Overview of the ICIX architecture

Motivating Example: Inferred Triples.

Fig. 8 shows the inferred knowledge graph of the report. The labels of node `Report` determine the different classes that the report belongs to (`PropertyCrimeReport`, `TheftCrimeReport`, etc.) as inferred by the DL reasoner following SPCO. Some of the RDF triples inferred are shown below in Turtle.

```
@prefix spco: <http://www...>

### spco:R_26877_21
spco:R_26877_21 rdf:type owl:NamedIndividual ,
 spco:Article234_2 ,
 spco:PropertyCrimeReport ,
 spco:Report ,
 spco:Theft ,
 spco:TheftByNotOwner ,
 [ rdf:type owl:Class ;
   owl:complementOf [ rdf:type owl:Restriction ;
   owl:onProperty spco:hasoffenseCharacteristic ;
   owl:someValuesFrom spco:RobberyCharacteristic
   ]
] ;
spco:stolenthing spco:Wallet .

### spco:Wallet
spco:Wallet rdf:type owl:NamedIndividual ,
  spco:offenseElement ,
  spco:StolenGoods ;
  spco:belongsTo spco:GHEH ;
  spco:referencedIn spco:R_26877_21 ;
  spco:stolenBy spco:MV ;
  spco:usedBy spco:GHEH ;
  spco:usedByOwner spco:GHEH ;
  spco:valueCost 70.0 .

### spco:GHEH
spco:GHEH rdf:type owl:NamedIndividual ,
  spco:offenseActor ,
  spco:offenseElement ,
  spco:Person ,
  spco:Victim ;
  spco:isOwnerOf spco:Wallet ;
  spco:isUserOf spco:Wallet ;
  spco:age "58"^^xsd:int .

### spco:MV
spco:MV rdf:type owl:NamedIndividual ,
  spco:Accused ,
  spco:offenseActor ,
  spco:offenseElement ,
  spco:Person ;
  spco:isThiefOf spco:Wallet ;
  spco:age "38"^^xsd:int .
```

We can see how the associated penalty follows from the relationship with Article 234.2, represented by a new node and a new relation `hasPunishment` between the report and the penalty in Fig. 9. For our report `R_26877_21`, the new penalty node corresponds to a prison sentence of 1 to 3 months.

The data extracted from the report, together with the information inferred from it, is called SUMMARY, and is presented to the judges both as a graph and as a table. Graphs are displayed using the graph representation tool provided by Neo4j. The tabular format contains the same information: the elements of the report, their types and attributes, as well as their relationships. Punishments are included if derived. For example, Table 1 below represents the SUMMARY of `R_26877_21`.

This information summarizes all aspects of the police report that are relevant to the property crime. It was considered very useful and easy to understand by the judges. However, as we shall see in Sect. 7.2, the judges did not find the summaries presented in graphical form to be as understandable and easy to use.

6.2 Classifying Reports with Incomplete or Redundant Information

So far we have been able to infer the article that applies to a given case and its associated punishment when all the evidence was available and all the required conditions were met. However, this is not as common as it should be. Very often some of the evidence required to classify an offense is missing or, conversely, we have additional evidence that may indicate that a different type of offense may have occurred instead. In this section, we discuss how to deal with these types of situations.

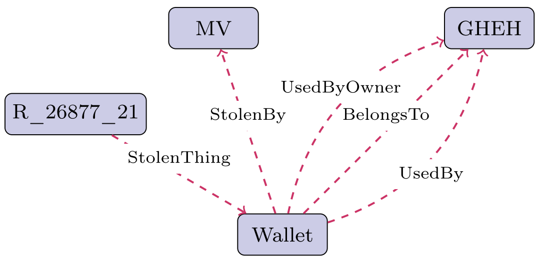

Fig. 8 Knowledge graph with the classes and relationships inferred for `R_26877_21`

First, the DL reasoner, when the information is incomplete, pre-classifies a report into a non-leaf class of the SPCO ontology, i.e., a class that does not correspond to any article associated with a penalty (e.g., `RobberyCrimeReport`). This pre-classification allows us to reduce the number of articles with which the algorithm will compare the report, as it will only compare it with articles whose associated classes are subsumed by the class inferred by the reasoner (e.g., Articles 240-242). This will be specially useful in the future when we consider more types of crimes (fraud, usurpation, etc.).

Our reasoning proposal comprises an algorithm that takes as input the whole KG, the report root node R, and the article A which we want to evaluate, and returns the probability that A applies to R. Intuitively, it uses the fact that our proposal represents articles by means of graph patterns, each of which defines the set of elements that must be present for the article to be applicable. Then, the algorithm creates the graph with the elements involved in the report R, and tries to match it with the graph pattern of the article. Roughly, it accounts for those elements in the report that match the pattern, those that are missing in the report but present in the article, and those that appear in the report but not in the article. More precisely, the algorithm involves three main steps as described in Fig. 10:

1. Materialize the KG around R: we compute the subgraph whose root is R by gathering all the facts affecting the report that were previously stored in the KG, i.e., coming from other related reports. We will refer to the original report graph as G_R, and this extended report subgraph as G'_R.
2. Annotate and enrich the relations in G'_R depending on the article A being evaluated:
   - We annotate the existing relationships as *necessary* if the article requires that information; as *surplus* if the relationship is not used in the article; and as *not considered* if the relationship is defined in the report, but not considered in our ontology. The necessary relationships are further annotated with their *weight* or *relevance* – initially $1/n$, with n being the number of relationships defined in the graph that represents the article.
   - Finally, we create those relationships that are required by the article but do not exist in our report. These newly added relationships in G'_R are annotated as *created*.
3. With all this information, the probability that A applies to R can be computed as follows:

$$P(G'_R, A) = \frac{1}{N}\sum_{r=1}^{N} W(t(r)) * F(r) \qquad (1)$$

In Equation (1), $N = n + c + s$, with n being the number of *necessary* relationships in G'_R; c the number of *created* ones; s the number of superfluous (*surplus*) ones; $t(r)$ identifies the type of relationship (*necessary*, *surplus* or *created*); $W(t)$ is the weight (ranging between 0 and 1) assigned to that type of relationship in the article; and $F(r)$ is the ratio of relations of the same type in the subgraph G'_R. Note how relationships of type *not considered* do not have any influence on the formula because they are not relevant to the article. For example, if we decide to focus on the necessary relations of the subgraph, we choose that both $W(t)$ and $F(r)$ are equiprobable for all *necessary* relations, 0

Fig. 9 Knowledge graph with the instances and relationships inferred for `R_26877_21` and its punishment

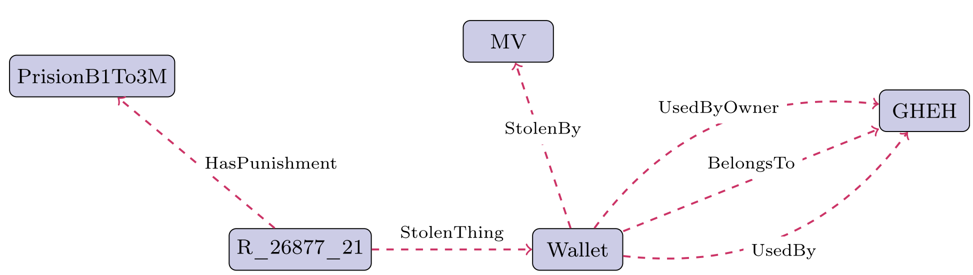

Table 1 Tabular representation for `R_26877_21` (Fig. 9)

Element	Type	Properties
R_26877_21	Report	
Wallet	StolenGoods	valueCost:70
GHEH	Victim	age:38
MV	Person	
Prison1-3M	Punishment	
Element	**Relation**	**Element**
R_26877_21	isStolen	Wallet
Wallet	belongsTo	GHEH
Wallet	usedByOwner	GHEH
Wallet	stolenBy	MV
R_26877_21	hasPunishement	Prison1-3M

otherwise. Other choices could be selected depending on how we want to assign weights to these relations.

The result is the probability of the article A being applicable to the offense reported in R, along with G'_R, which can be presented to the judge as an explanation of the elements taken into account to reach the particular decision.

There are two exceptional cases to consider. The first one occurs when either the authors of the crime or the stolen goods are missing in the report. In these cases, the probability will never be equal to 1, but it is crucial to inform the user of these circumstances because without them the report will likely be archived. The second case concerns information on the defendant's previous arrest record. The information kept by the police and usually reflected in the reports refers to the arrests, but it may happen that the accused was acquitted of those arrests, or the statute of limitations may have expired. Therefore, judges tend to disregard this information even if it is present in the reports. After consulting the judges, this relationship is classified as "not considered" by our algorithm, which corresponds more closely to the way the judges classify the reports.

Implementation Details. We have implemented the above algorithm as a plugin in Neo4j database. In particular, once we have loaded all the information of the reports, we first generate the subgraph G'_R using the following Cypher statement:

```
MATCH (rootA:owl__NamedIndividual
           {name:'R_26877_21'})
CALL apoc.path.subgraphAll(rootA,
           {relationshipFilter:'<|>'})
YIELD nodes, relationships
RETURN nodes, relationships;
```

The resulting subgraph coincides with G_R, and we store a copy of such subgraph in the KG. The article A we want to evaluate is associated to it in order to differentiate them. The extension of G_R to G'_R is achieved by a Java method called `addArticlePriorProbability()` that we have implemented. Its parameters are the root node of the subgraph and the related article. For example, the Cypher sentence that generates the subgraph for the report shown in Fig. 8 when compared to Article 234.2 is shown below.

```
MATCH  (rootA:owl__NamedIndividual
  {name:'R_26877_21',
  article:'ns0__Article234_2'})
CALL ontology.util.addArticlePriorProbability
  (rootA, 'Article234_2')
YIELD relations
RETURN relations;
```

Note that in this case the article perfectly matches the report, and therefore the probability is 1.0. More precisely, the relations of the resulting graph after the application of that method are the following.

```
{ "identity": 1182, "start": 506, "end": 555,
  "type": "ns0__stolenthing",
  "properties": { "probabilty_element": "1",
                  "factor": "1.0",
         "typeReportRelation": "necessary" }
},
{ "identity": 1183, "start": 555, "end": 506,
  "type": "ns0__referencedIn",
  "properties": {
    "typeReportRelation": "not_considered" }
} ,
{ "identity": 1212, "start": 555, "end": 598,
  "type": "ns0__usedBy",
  "properties": {
     "typeReportRelation": "not_considered" }
} ,
{ "identity": 1239, "start": 555, "end": 598,
  "type": "ns0__belongsTo",
  "properties": { "probabilty_element": "2",
                  "factor": "1.0",
         "typeReportRelation": "necessary"}
} ,
...
```

Finally, the third step of the algorithm has been implemented as a Java method called `subGraphProbability()`, which calculates the probability that a police report matches an article. Then, given an article A and a graph G'_R that represents the information of a report R, the probability that A applies to G'_R can be determined by the following Cypher expression:

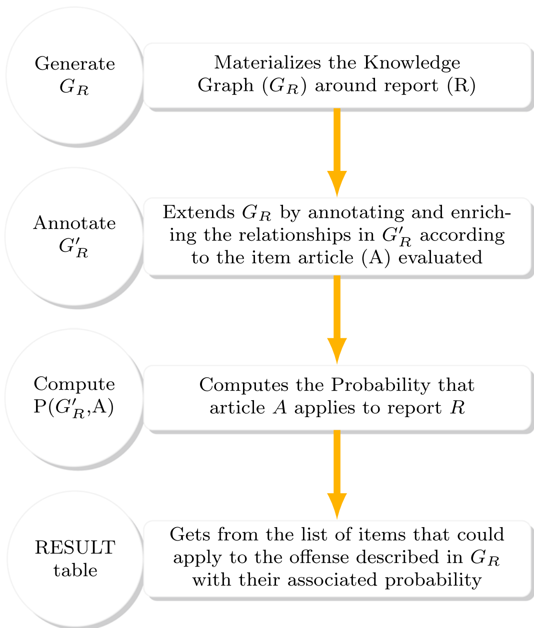

Fig. 10 Main steps in inferring facts from the offense report

```
MATCH  (rootA:owl__NamedIndividual
  {name:'R_<R>',
  article:'ns0__<A>'})
CALL apoc.path.subgraphAll
 (rootA, %{relationshipFilter:'<|>'})
YIELD nodes, relationships
RETURN ontology.util.subGraphProbability
  (relationships,'<A>');
```

Motivating Example: Final Result

To illustrate how this algorithm works, suppose that the author of the theft in Report `26877_21` is not identified and therefore it is not present in the graph. The new graph misses the author of the theft and its corresponding relations with the stolen object (the wallet), and also changes the qualification of the property crime. In this case, the DL reasoner determines that it is a generic property crime, without being able to specify a specific article that identifies a penalty.

The application of method `addArticlePriorProbability()` generates the missing relations for Article 234.2, which are listed below and shown in Fig. 11.

Table 2 RESULT table, with the Articles that can be applied to `R_26877_21_modified`, and their probabilities

Option	Article	Probability
#1	234.2	0.75
#2	234.3	0.60
#3	235.1	0.60
#4	235.2	0.60
#5	234.1	0.40
...	...	...
#14	242.4	0.33
#15	236.2	0.16


```
{ "identity": 325, "start": 612, "end": 613,
  "type": "ns0__stolenthing",
  "properties": { "probabilty_element": "1",
  "typeReportRelation": "necessary",
  "factor": "1.0" }
} ,
{ "identity": 412, "start": 613, "end": 616,
  "type": "ns0__usedByOwner",
  "properties": { "probabilty_element": "6",
  "typeReportRelation": "necessary",
  "factor": "1.0" }
} ,
{ "identity": 400, "start": 613, "end": 615,
  "type": "ns0__stolenBy",
  "properties": { "probabilty_element": "3",
  "typeReportRelation": "created",
  "factor": "1.0" }
} ,
{ "identity": 377, "start": 613, "end": 614,
  "type": "ns0__belongsTo",
  "properties": { "probabilty_element": "2",
  "typeReportRelation": "necessary",
  "factor": "1.0" }
}, ...
```


Note how in this case the Article does not exactly match the report graph and therefore the `subGraphProbability()` method produces a probability of 0.75.

The same process can be applied to the rest of the articles, obtaining the list of articles that could be applied to the crime described in the report with their associated probabilities. This list is called RESULTS and is shown in Table 2. Note that, in general, several articles could be applied to the same report.

Associated with each option in the RESULTS table, our proposal also produces its SUMMARY, i.e., the representation of its knowledge graph in two formats: visually as a graph and tabularly in textual form, as described in the previous section.

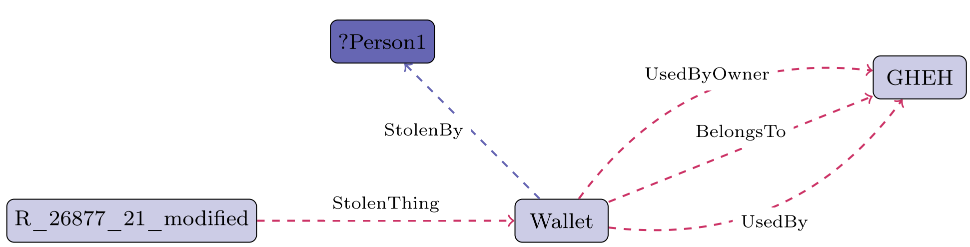

Fig. 11 Knowledge graph for `R_26877_21_modified`, after applying procedure `addArticlePriorProbability()`. Existing required instances are shown in indigo. Node and relation created are shown in blue

Table 3 Tabular representation of the SUMMARY for `R_26877_21_modified` (see Fig. 11). The node and relationship explicitly created by the algorithm have been highlighted to facilitate the user's observation

Element	Type	Properties
R_26877_21	Report	
Wallet	StolenGoods	valueCost:70
GHEH	Victim	age:58
?PERSON1	Person	

Element	Relation	Element
R_26877_21	IsStolen	Wallet
Wallet	usedByOwner	GHEH
Wallet	stolenBy	*?PERSON1*
Wallet	belongsTo	GHEH

** Warning: No author identified in the report.

For example, Option#1 means applying Article 234.2 to the report shown in Fig. 11, whose thief was not identified in the report. The tabular representation of that report, as presented to the judge in the SUMMARY, is shown in Table 3. Note the warning about the missing author, indicating that the report may have to be archived. The tool produces such warnings when the report does not identify the author of the crime or the stolen goods. The occurrence of special circumstances in the report, such as violence or intimidation, is also included in the warnings section, highlighting them so that the user can easily identify them.

7 Evaluation

The last step in the DSR methodology is the evaluation of the artifact, with the goal to validate the extent to which the artifact solves the problem and satisfies the requirements. This section presents the experiments that we designed and carried out to validate the proposal, namely a set of unit tests to check that all the components of the tool worked as expected, and an empirical study we carried out with judges to evaluate the proposal and respond to the research questions.

7.1 Initial Evaluation

The tool that supports our proposal was developed using an iterative development process, in which the functional correctness of each of its components (the IE component, the ontology and the reasoner) was evaluated. The IE component was tested with more than a hundred real police reports that the Council of the Judiciary and the Dean's Office of Judges authorized us to use (once anonymized). The ontology and the reasoner were also tested with the information extracted from these reports, verifying that the results made sense and matched those decided by the judges (all the reports contained the associated sentence). Interestingly, we detected some cases where our results and the verdict did not match, but the case was not really clear as we later checked in the empirical experiment.

7.2 Empirical Evaluation

To validate the proposal we also carried out an empirical experiment. We followed the basic methodology for conducting usability studies (Rubin and Chisnell 2008), which is derived from the classical approach for conducting controlled experiments, as well as the Empirical Standards for Software Engineering Research (Ralph et al. 2020). As recommended by Rubin and Chisnell (2008), instead of formulating a hypothesis, this experiment aims to answer our research questions.

7.2.1 Experiment Design and Setup

The experiment consisted of an on-site exercise with three parts (Sessions 1-3) and a duration of 2.5 hours, including breaks. Six anonymized *real* police reports, extracted from the Courts records, were used in the experiment. They covered different situations. One of them (*R*1) had enough information to classify it correctly, while another (*R*6)

Table 4 Classifications by judges and by the tool (in green)

	*R*1	*R*2	*R*3	*R*4	*R*5	*R*6
234.1	7+1	1			1+1	
234.2		5+1				
234.3						
235.1						
235.2						
236.1						
236.2						
240.1			4+1		3	
240.2						
241.1			2			
241.4						
242.1				4+1		
242.2						1
242.3						5+1
242.4						
none		1	1	2	2	

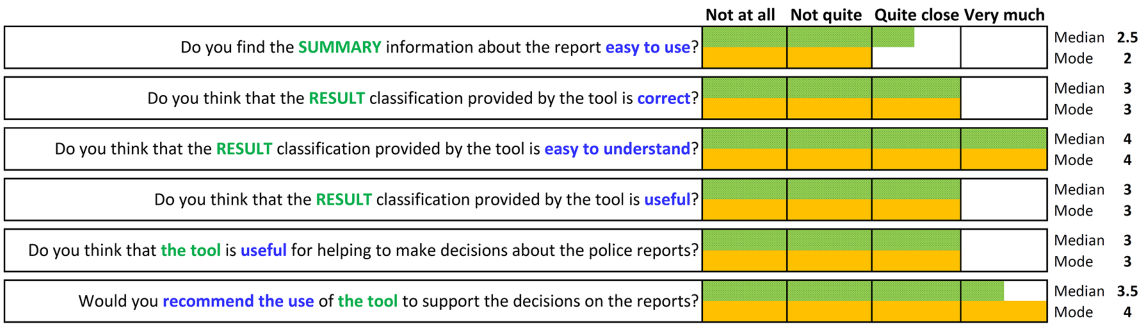

Fig. 12 Final questions and their responses

lacked too much information to be directly filed. So decisions about them were *clear*. Two of them (*R*2 and *R*3) were *missing* some details; namely, the value of the stolen goods and the offender's criminal record number, respectively. Finally, two other reports (*R*4 and *R*5) had *vague* information that could be interpreted in different ways. For example, report *R*4 could also be considered attempted theft, and not theft per se, so another law not contemplated in the ontology could apply. Likewise, report *R*5 did not mention the severity of the aggression and there was no injury report, something that could determine the application of other articles. Reports R1, R2 and R3 were classified by 7 judges each, whilst reports R4, R5 and R6 were classified by 6 judges. Table 4 shows the classifications made by judges. Columns correspond to reports and rows to law articles. Cells indicate the number of judges that assigned an article to a report. The last row (*none*) corresponds to cases where the judge could not classify a report according to any of the listed articles. The classification made by our tool is shown by adding 1 in the green cells.

The complete protocol, materials and exercises provided to the subjects, the questionnaires used and the anonymized data collected can be found on our companion website (Navarrete et al. 2023).

The goal of Session 1 (30 min) was to replicate the traditional way of work by judges. For this purpose, each subject was given three different reports and asked to determine the articles that applied.

The goal of Session 2 (60 min) was to evaluate our proposed process and compare the results with those obtained for each report in Session 1. It had 3 parts:

- First, each subject was given three police reports, different from those evaluated in Session 1, together with the tabular and graph representation of the information extracted by our tool (SUMMARY). Subjects were asked to check whether that information was understandable and complete (30 min).
- Second, each subject was given the list produced by our tool with the articles that could apply to each of the 3 reports (RESULTS). They were asked to check whether the proposed classification was correct or not – i.e., it coincides with the judge's classification (15 min).
- Third, for each of the same three reports, subjects were provided with three decisions made by other judges during Session 1 on the same reports and asked to check whether there were discrepancies between them and whether the results of our proposal were better or worse than those provided by the judges. In case of discrepancies, they should indicate which report they agreed with more. In some cases we included one report with the results of our tool (written as if it had been elaborated by a human), to check if they could identify it (15 min).

Session 3 concluded by collecting the subjects' general opinions on both our proposal and the experiment itself (5 min). Specifically, we asked them about the perceived correctness, usefulness and understandability of the SUMMARY and RESULTS documents provided by our proposal, the usefulness of the results, and whether they would recommend its use to support decision-making. These questions aimed at answering the research questions posed in Sect. 3.3. We used a Likert scale from 1 (Not at all) to 4 (Very much), see Fig. 12. In addition, we asked the subjects whether they preferred the summary information to be displayed as tables or graphs, their general opinion about our proposal and the process, as well as suggestions for future improvements.

A pilot experiment with two subjects was conducted in March 2023 to refine our protocol, materials, instructions, exercises and questionnaires. The full experiment was conducted in April 2023, with the same questions, since the pilot experiment worked well, and with eleven participants (different from the first two). They represented all profiles involved in the classification of police reports, including judges, prosecutors and court registrars. Although Nielsen and other authors maintain that five users are enough for usability testing (Turner et al. 2006; Nielsen 2020), other authors suggest the rule of 16 ± 4 participants (Alroobaea and Mayhew 2014). By running the pilot with 2 subjects and the experiment with 11, we tried to cover both situations. We carried out the experiment on three different dates to accommodate the participants' agendas. The first and last authors of the paper were always present and ensured that all the sessions were equally executed. The similarity of the results obtained in all three groups, and their correlation with the results of the three groups combined, seem to support Nielsen's theory.

7.2.2 Results and Lessons Learned

This section aims at answering the three requirements satisfaction questions about the solution artifact posed in Sect. 3.3. The first one was about the reliability of the results produced by our tool, i.e., whether they can be trusted or not. To assess it, we have measured the degree of agreement between the ratings produced by the tool and those produced by the judges on the same reports. For this, we used the Fleiss Kappa κ (Fleiss 1971), a measure of inter-rater reliability. If the raters are in complete agreement then $\kappa = 1$. If there is no agreement among the raters (other than what would be expected by chance) then $\kappa \leq 0$. In our case, we calculated the Fleiss kappa measure for the judges' ratings, using Table 4, obtaining a value of $\kappa = 0.463$. This value is usually interpreted as *moderate* agreement. It is not higher because the judges' opinions diverge for the reports with vague or missing information. We then added the classifications produced by our tool as a new rater and calculated κ with them. Should the tool's results diverge from the judges' ratings, the new κ value would decrease. Conversely, the value would increase if the tool confirmed the judges' opinions. In our case, the new value of κ is 0.471. This means that we can consider the tool's results to be *reliable* with respect to the judges' opinions. More importantly, the results of our tool fully agree with the judges' decisions in those cases with clear classifications.

In addition to the assessment of the reliability of the results produced by our tool, Fig. 12 shows the results of the general questions posed during the experiment, which aim at providing a *subjective* assessment of the reliability and understandability of the results, as well as the usefulness of the solution artifact, thus responding to the three requirements satisfaction questions, see Sect. 3.3. The results were mostly uniform across all subjects. The weakest point was the ease of use of the SUMMARY document. The general comment was that it should not be used on its own but always accompanied by the police report. Anyway, this document is basically for internal use of the tool and is not intended to make decisions on its own.

Overall, most participants found our proposal useful, expressive and usable. All of them preferred the tabular representation of the information extracted by our proposal, only two subjects liked the graphs too. The information provided by our proposal was considered informative, but not very useful in the cases of missing information, since no decision could be made about the crime (independently of the tool).

Finally, when asked about their opinion on the integration of our proposal in their daily work, they found it particularly interesting for detecting those reports with a clear verdict: either a 100% match or lack of a fundamental element (author or stolen property). According to the judges' own estimates, this type of report represents around 30% of those received by the courts, so our proposal could significantly reduce their work in these cases. A more in-depth review is needed of the rest of the reports, although all of them found the summary tables and results of our proposal to be quite useful. Nevertheless, we plan to investigate how we can better assist users in this step in the future.

7.2.3 Threats to Validity

Threats to validity are inherent to every empirical study. Threats are classified into four categories: internal, construct, external, and conclusion validity (Wohlin et al. 2012).

Internal Validity Threats related to the factors that could affect the results of our evaluation.

- Communication: The information provided by our proposal (summaries and results) could be misunderstood. *Mitigation:* We gave a presentation before the experiment, describing how the results produced by our proposal are presented. We also provided two different representations of the results, using tables and graphs.
- Background: The background of the participants was too homogeneous. *Mitigation:* We selected participants from all the different profiles involved in the report classifications, namely judges, prosecutors and court registrars.
- Separate days: We performed the experiment on three different days, trying to recreate the same conditions for each of them. However, there might be some differences in the explanations and the questions asked by the participants that could make the sessions not identical. *Mitigation:* We followed the same protocol for each of the sessions in order to reduce significant differences between them. Furthermore, two of the authors were present in all experiments, ensuring they all followed the same protocol.

Construct Validity These threats are related to those issues that might arise during research design, which are concerned with the relationship between theory and what is observed.

- Protocol: As for the methodology followed to conduct the experiment, it can never be guaranteed to provide sufficient detail for the success of the study. *Mitigation:* We followed the basic methodology for conducting usability studies (Rubin and Chisnell 2008) as well as the Empirical Standards for Software Engineering Research Ralph et al. (2020).
- Questionnaires: The questionnaires might not be able to cover our research questions. *Mitigation:* We tested the questionnaires with the pilot experiment to ensure the appropriate coverage.

External Validity These threats are related to the extent to which it is possible to generalize the findings and conclusions of this study beyond the experiment context.

- Selection: The conclusions may depend on the choice of particular reports. *Mitigation:* We selected six reports that tried to cover the more common situations found in reality. Moreover, all our cases recreate realistic situations.
- Population: The experiments have been conducted with participants with a judicial background and in a court. Although this is referred to as convenience sampling (Wohlin et al. 2012) and is common practice in controlled experiments, this imposes a threat. *Mitigation:* Participants with different profiles were selected for our experiment.

Conclusion Validity These are concerned with the issues that affect the ability to draw correct conclusions and whether the experiments can be repeated.

- Sample size: The number of participants in the experiments may be insufficient to draw correct conclusions. *Mitigation*: The experiments have included 13 participants. As previously, mentioned, considering 16 ± 4 participants (Alroobaea and Mayhew 2014) is considered sufficient in these types of studies.

7.3 Discussion

In this section, we will revisit the requirements and questions posed in Sect. 3.3 to check the extent to which we have covered them, and also discuss some of the main advantages and limitations of our proposal.

Firstly, we have managed to develop a tool that supports our proposal, which is capable of automatically extracting information from police reports on property crimes, identifying the actors, elements, and relevant details of the reported crimes (R1). It is also able to identify the Articles of the law that might apply to the offense, and how each matches the information extracted (R2). Last, it is able to assign a probability to each Article, which represents its likelihood based on the information not only present in the report but also absent from it (R3).

In addition, an empirical experiment was conducted to evaluate the three requirement satisfaction questions about the solution artifact. Namely, the results produced by our proposal can be considered reliable (S1), understandable (S2), and the proposed solution is usable by judges (S3). The initial tests and experiment results confirm that all these requirements are met. The judges considered that the results of our proposal were correct and useful.

Nevertheless, the proposal also has some limitations in its current state. First, it has been tested with a good number of police reports, but further testing is needed to gain more confidence about the behavior of the information extractor and the reasoner. For example, we have found that reports are normally written similarly and uniformly in all Spanish police stations, but it would be useful to check this claim with a larger sample of reports. Secondly, the supporting tool is in the prototype phase, with some rudimentary parts such as its user interface. More work is needed to make it usable for non-expert users. Third, the results provided by our proposal are considered useful in those cases of perfect matches with an Article, or when a fundamental element is missing in the report. How best to assist users in the rest of the cases, especially when information is vague or incomplete, remains an open issue. Finally, more feedback from judges and potential users of our proposal could help identify further limitations and potential improvements.

8 Conclusions

In this paper we have presented a proposal to assist judges in the classification of police reports about property crimes by automatically extracting information from the reports and producing a list of possible classifications of crimes, accompanied by a degree of confidence in each of them. Our proposal has been empirically validated by judges and prosecutors. The results show that the proposal is usable and can be very effective in helping judges classify property crime reports.

In the future, we would like to extend our work along several lines of research. First, in order to broaden the scope and usefulness of our proposal, we will cover other types of crimes beyond property theft and robbery, such as vehicle theft and robbery, fraud and usurpation. This will require extending our ontology, and, for this purpose, we will consider using a foundational ontology, e.g. UFO, to ensure consistency. Additionally, we will have to rethink the information extraction part, as the information to be taken into account will be more diverse and complex than that of property crime police reports. Second, when conducting the experiment with judges we realized that there is plenty of subjectivity in the classification process, mostly due to the uncertainty present in some of the police reports. For example, two experts may have different confidence in the sources or evidence provided, and, therefore, their opinions may not coincide. Explicitly dealing with subjective information and uncertainty is something we would like to explore further.

Acknowledgements We would like to thank the Council of the Judiciary and the Dean of Judges of Málaga city for giving us access to all the police reports that allowed us to develop and test our proposal. We are grateful to the participants in our experiment for their time and valuable comments. We would also like to acknowledge the paper's reviewers for their insightful comments and valuable suggestions, which helped us improve this work. This work was partially supported by the Spanish Government (FEDER/Ministerio de Ciencia e Innovación–Agencia Estatal de Investigación) under projects TED2021-130523B-I00, PID-2021-125527NB-I00, and PID2020-113903RB-I00, and by project T42_23R, funded by Gobierno de Aragón. Funding for open access charge: Universidad de Málaga / CBUA.

Funding Funding for open access publishing: Universidad Málaga/ CBUA.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

Alroobaea R, Mayhew PJ (2014) How many participants are really enough for usability studies? In: Proceedings of SAI'14, IEEE, pp 48–56

Asaro C, Biasiotti MA, Guidotti P, Papini M, Sagri MT, Tiscornia D (2003) A domain ontology: Italian crime ontology. In: Proceedings of MWAIL@ICAIL 2003

Baader F, Calvanese D, McGuinness D, Nardi D, Patel-Schneider PF (2003) The description logic handbook: theory, implementation, and applications. Cambridge University Press, Cambridge

Bak J, Jedrzejek C (2009) Application of an ontology-based model to a selected fraudulent disbursement economic crime. In: Proceedings of AICOL-I/IVR-XXIV and AICOL-II/JURIX, Springer, Heidelberg, LNCS, vol 6237, pp 113–132

Bansal N, Sharma A, Singh R (2019) A review on the application of deep learning in legal domain. In: Proceedings of AIAI'19, Springer, pp 374–381

Bezzazi EH (2007) Building an ontology that helps identify criminal law articles that apply to a cybercrime case. In: Proceedings of ICSOFT'07, INSTICC Press, pp 179–185

Biagioli C, Francesconi E, Passerini A, Montemagni S, Soria C (2005) Automatic semantics extraction in law documents. In: Proceedings of ICAIL'05, pp 133–140

Bommarito M, Katz DM, Detterman E (2018) Lexnlp: natural language processing and information extraction for legal and regulatory texts. Research Handbook on Big Data Law

Breuker J, Valente A, Winkels R (2004) Legal ontologies in knowledge engineering and information management. Artif Intell Law 12(4):241–277

Brighi R, Palmirani M (2009) Legal text analysis of the modification provisions: a pattern oriented approach. In: Proceedings of ICAIL'09, pp 238–239

Buey MG, Garrido ÁL, Bobed C, Ilarri S (2016) The AIS project: boosting information extraction from legal documents by using ontologies. In: Proceedings of ICAART'16, SciTePress, pp 438–445

Buey MG, Román C, Garrido ÁL, Bobed C, Mena E (2019) Automatic legal document analysis: improving the results of information extraction processes using an ontology. Intell Methods Big Data Indus Appl 40:333–351

Carnaz G, Nogueira VB, Antunes M (2019) Knowledge representation of crime-related events: a preliminary approach. In: Proceedings of SLATE'19, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik

CGPJ (2023) Reports by territory on the activity of judicial bodies (in Spanish). https://bit.ly/499nRFp, accessed 03 July 2024

Chalkidis I, Kampas D (2019) Deep learning in law: early adaptation and legal word embeddings trained on large corpora. Artif Intell Law 27(2):171–198

Chalkidis I, Androutsopoulos I, Aletras N (2019) Neural legal judgment prediction in English. In: Proceedings of ACL'19, pp 4317–4323

Chalkidis I, Fergadiotis M, Malakasiotis P, Aletras N, Androutsopoulos I (2020) Legal-bert: the muppets straight out of law school. arXiv:2010.02559

Chandrasekaran B, Josephson JR, Benjamins VR (1999) What are ontologies, and why do we need them? IEEE Intell Syst Appl 14(1):20–26

Chaudhri VK, Baru CK, Chittar N, Dong XL, Genesereth MR, Hendler JA, Kalyanpur A, Lenat DB, Sequeda J, Vrandecic D, Wang K (2022) Knowledge graphs: introduction, history and perspectives. AI Mag 43(1):17–29

Corcho O, Fernández-López M, Gómez-Pérez A, López-Cima A (2005) Building legal ontologies with methontology and webode. Law and the semantic web: legal ontologies, methodologies, legal information retrieval, and applications. Springer, Heidelberg, pp 142–157

Dresch A, Lacerda D, Valle Antunes Jr JA (2015) Design science research: a method for science and technology advancement. Springer, Heidelberg

Fernández-López M, Poveda-Villalón M, Suárez-Figueroa MC, Gómez-Pérez A (2019) Why are ontologies not reused across the same domain? J Web Semant 57

Filtz E (2017) Building and processing a knowledge-graph for legal data. In: Proceedings of ESWC'17, LNCS, pp 184–194

Fleiss JL (1971) Measuring nominal scale agreement among many raters. Psychol Bullet 75(5):378–382

Garrido ÁL, Peiró A, Román C, Bobed C, Mena E (2021) ICIX: a semantic information extraction architecture. In: Proceedings of IDEAS'21, ACM, pp 75–83

Getman AP, Karasiuk VV (2014) A crowdsourcing approach to building a legal ontology from text. Artif Intell Law 22(3):313–335

Gordon TF (2008) Constructing legal arguments with rules in the legal knowledge interchange format (LKIF). Computable models of the law, languages, dialogues, games, ontologies, LNCS, vol 4884. Springer, Heidelberg, pp 162–184

Gordon TF, Prakken H, Walton Walton D (2007) The carneades model of argument and burden of proof. Artif Intell 171(10–15):875–896

Griffo C, Almeida JPA, Guizzardi G (2015) Towards a legal core ontology based on alexy's theory of fundamental rights. In: Proceedings of MWAIL@ICAIL 2015

Guizzardi G, Benevides AB, Fonseca CM, Porello D, Almeida JPA, Sales TP (2022) UFO: Unified foundational ontology. Appl Ontol 17(1):167–210

Gutiérrez C, Sequeda JF (2021) Knowledge graphs. Commun ACM 64(3):96–104

Gutierrez F, Dou D, Fickas S, Wimalasuriya D, Zong H (2016) A hybrid ontology-based information extraction system. J Inf Sci 42(6):798–820

Hoekstra R, Breuker J, Bello MD, Boer A (2009) LKIF core: principled ontology development for the legal domain. In: IOS Press, Frontiers in Artificial Intelligence and Applications, pp 21–52. https://doi.org/10.3233/978-1-58603-942-4-21

Hogan A, Blomqvist E, Cochez M, d'Amato C, de Melo G, Gutierrez C, Kirrane S, Gayo JEL, Navigli R, Neumaier S, Ngomo AN, Polleres A, Rashid SM, Rula A, Schmelzeisen L, Sequeda JF, Staab S, Zimmermann A (2022) Knowledge graphs. ACM Comput Surv 54 (4) :71:1–71:37

Honnibal M, Montani I, Van Landeghem S, Boyd A (2020) spaCy: Industrial-strength natural language processing in Python

Horrocks I, Patel-Schneider PF, Boley H, Tabet S, Grosof B, Dean M (2004) SWRL: a semantic web rule language. Combining OWL and RuleML. W3C member submission. http://www.w3.org/submissions/SWRL/, accessed 03 July 2024

Humphreys L, Boella G, van der Torre L, Robaldo L, Di Caro L, Ghanavati S, Muthuri R (2021) Populating legal ontologies using semantic role labeling. Artif Intell Law 29:171–211

ISO/IEC 25010 (2011) Systems and software engineering–systems and software quality requirements and evaluation (SQuaRE)–system and software quality models. ISO/IEC

Johannesson P, Perjons E (2014) An introduction to design science. Springer, Heidelberg

Liquid Legal Institute (2021) Legal Ontologies. https://github.com/Liquid-Legal-Institute/Legal-Ontologies, Accessed 03 July 2024

Love N, Genesereth MR (2005) Computational law. In: Proceedings of AIL'05, ACM, pp 205–209

de Maat E, Winkels R (2010) Suggesting model fragments for sentences in dutch laws. In: Proceedings of LOAIT'10) pp 19–28

Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv:1301.3781

Mok WY, Mok JR (2019) Legal machine-learning analysis: first steps towards ai assisted legal research. In: Proceedings of ICAIL'19, pp 266–267

Mokanov I, Shane D, Cerat B (2019) Facts2law: using deep learning to provide a legal qualification to a set of facts. In: Proceedings of the seventeenth international conference on artificial intelligence and law, pp 268–269

Nakashole N, Weikum G, Suchanek FM (2012) PATTY: a taxonomy of relational patterns with semantic types. In: Proceedings of EMNLP-CoNLL'12, ACL, pp 1135–1145

Navarrete F, Garrido ÁL, Bobed C, Atencia M, Vallecillo A (2023) Ontology-based automated reasoning about property crimes: companion website. https://github.com/atenearesearchgroup/property-crime-classification, Accessed 03 July 2024

Nguyen HT, Vuong HYT, Nguyen PM, Dang BT, Bui QM, Vu ST, Nguyen CM, Tran V, Satoh K, Nguyen ML (2020) Jnlp team: deep learning for legal processing in coliee 2020. arXiv:2011.08071

Nielsen J (2020) How many test users in a usability study? https://www.nngroup.com/articles/how-many-test-users/, Accessed 03 July 2024

de Oliveira Rodrigues CM, de Freitas FLG, de Azevedo RR (2016) An ontology for property crime based on events from UFO-B foundational ontology. In: Proceedings of BRACIS', IEEE Computer Society, pp 331–336

Opasjumruskit K, Peters D, Schindler S (2020) DSAT: ontology-based information extraction on technical data sheets. In: Proceedings of ISWC'20, CEUR-WS, pp 251–256

Opasjumruskit K, Böning S, Schindler S, Peters D (2022) OntoHuman: Ontology-based information extraction tools with human-in-the-loop interaction. In: Proceedings of CVDE'22, Springer, LNCS, pp 68–74

Palmirani M, Martoni M, Rossi A, Cesare B, Livio R et al (2018) Legal ontology for modeling GDPR concepts and norms. Front Artif Intell Appl 313:91–100

Park H, Cho S, Kwon H (2009) Cyber forensics ontology for cyber criminal investigation. In: Proceedings of e-Forensics'09, Springer, Heidelberg, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, pp 160–165

Ralph P, Baltes S, Bianculli D, Dittrich Y, Felderer M, Feldt R, Filieri A, Furia CA, Graziotin D, He P, Hoda R, Juristo N, Kitchenham BA, Robbes R, Méndez D, Molleri J, Spinellis D, Staron M, Stol K, Tamburri DA, Torchiano M, Treude C, Turhan B, Vegas S (2020) ACM SIGSOFT empirical standards. CoRR ArXiv: abs/2010.03525

Robinson I, Webber J, Eifrem E (2015) Graph databases: new opportunities for connected data, 2nd edn. O'Reilly, Sebastopol

Rubin J, Chisnell D (2008) Handbook of usability testing. Second edition: how to plan, design, and conduct effective tests. Wiley, Hoboken

Sansone C, Sperlí G (2022) Legal information retrieval systems: state-of-the-art and open issues. Inf Syst 106(101):967

Sivaranjani N, Jayabharathy J (2022) Neural network towards uncertain legal case retrieval. J Uncertain Syst 15(2):1–14 (**2241,001**)

Sleimi A, Sannier N, Sabetzadeh M, Briand L, Ceci M, Dann J (2021) An automated framework for the extraction of semantic legal metadata from legal texts. Empir Softw Eng 26:1–50

Soh C, Lim S, Hong K, Rhim Y (2017) Ontology modeling for criminal law. In: Proceedings of AICOL workshops 2015–2017, Springer, Heidelberg, LNCS, pp 365–379

Sovrano F, Palmirani M, Vitali F (2020) Legal knowledge extraction for knowledge graph based question-answering. In: Proceedings of JURIX'20, IOS Press, Frontiers in Artificial Intelligence and Applications 334:143–153

Studer R, Benjamins VR, Fensel D (1998) Knowledge engineering: principles and methods. Data Knowl Eng 25(1–2):161–197

Suárez-Figueroa MC (2012) Neon methodology for building ontology networks: specification, scheduling and reuse. PhD thesis, Technical University of Madrid, Spain, https://d-nb.info/1029370028, Accessed 03 July 2024

Tunstall L, Von Werra L, Wolf T (2022) Natural language processing with transformers. O'Reilly, Sebastopol

Turner CW, Lewis JR, Nielsen J (2006) Determining usability test sample size, vol 3, 2nd edn. CRC, Boca Raton, pp 3084–3088

van de Ven S, Hoekstra R, Breuker J, Wortel L, Ali AE (2008) Judging amy: automated legal assessment using OWL 2. In: Proceedings of the OWLED workshop@ISWC'08), CEUR-WS.org, CEUR Workshop Proceedings, vol 432, http://ceur-ws.org/Vol-432/owled2008eu_submission_28.pdf, accessed 03 July 2024

Vuong YTH, Bui QM, Nguyen HT, Nguyen TTT, Tran V, Phan XH, Satoh K, Nguyen LM (2023) SM-BERT-CR: a deep learning approach for case law retrieval with supporting model. Artif Intell and Law 31(3):601–628

W3C OWL Working Group (2012) OWL 2 Web Ontology Language Document Overview (2nd Edition). http://www.w3.org/TR/owl2-overview/, Accessed 03 July 2024

Wieringa RJ (2014) Design science methodology for information systems and software engineering. Springer, Heidelberg

Wimalasuriya DC, Dou D (2010) Ontology-based information extraction: an introduction and a survey of current approaches. J Inf Sci 36(3):306–323

Wohlin C, Runeson P, Höst M, Ohlsson MC, Regnell B, Wesslén A (2012) Experimentation in software engineering. Springer, Heidelberg

Yang J, Ma W, Zhang M, Zhou X, Liu Y, Ma S (2022) LegalGNN: legal information enhanced graph neural network for recommendation. ACM Trans Inf Syst 40(2):33:1-33:29

Zhong H, Xiao C, Tu C, Zhang T, Liu Z, Sun M (2020) How does nlp benefit legal system: a summary of legal artificial intelligence. arXiv:2004.12158