Multilingual Logical English for Double Taxation Conventions

Giuseppe Contissa[1,*,†], Alessia Fidelangeli[1,†] and Galileo Sartor[2,†]

[1]University of Bologna, Bologna, Italy
[2]Swansea University, Swansea, UK

Abstract

This paper presents the development of the NILDE research project, with the modelling of a logic rule-based system in the legal domain of double taxation. The translated legal provision is the French-Italian Convention against double-taxation, and it is modelled in Logical English, a controlled natural language layer to Prolog. In particular the rules will be written in Italian, as is the source text. This paper will present the challenges that arise when modelling a computable version of legal norms, and how Logical English can assist in this task. Furthermore the possibility of building expert system accessible in multiple languages will be presented, as a way of lowering the barrier of entry for citizens.

Keywords

Logical English, Law, Legal Knowledge Modelling, Logic Programming, Prolog, Tax Law.

1. Introduction

The objective of Artificial Intelligence (AI) from its early days has been to automate legal reasoning and problem-solving. Legal texts, norms, and arguments have a structured format that has encouraged the use of formal logic to capture legal knowledge, including legal rules, cases, and facts, and make inferences based on this knowledge.

Many researchers have attempted to develop rule-based systems for the law by assuming that legal provisions can usually be represented as rules, and that legal reasoning is based on the application of rules to facts.

Though logical knowledge can be represented in different ways, using different logical formalisms, the leading approach has consisted in modelling legal knowledge through conditional statements (rules), usually built using a subset of predicate logic.

In addition to several theorethical challenges relating to the representation of legal knowledge (isomorphism, deontic positions, temporal aspects, etc.), a set of relevant technical and practical issues have limited the success of such systems in the course of the last decades.

Namely, the fact that representing legal texts in computable models is a highly specialized skill, requiring both legal and technical expertise, and that the process of formalizing legal texts

3rd international workshop KM4LAW – Knowledge Management and Process Mining for Law
*Corresponding author.
†These authors contributed equally.
giuseppe.contissa@unibo.it (G. Contissa); alessia.fidelangeli2@unibo.it (A. Fidelangeli); galileo.sartor@swansea.ac.uk (G. Sartor)
0000-0002-0877-7063 (G. Contissa); 0000-0001-7116-9338 (A. Fidelangeli); 0000-0002-9421-8566 (G. Sartor)

requires a considerable expenditure of resources (time, work) to overcome the difficulties that systems face in scaling up from small prototypes to large applications. Further resources are needed to keep the system up to date.

More specifically, one possible obstacle towards the effective development and use of expert systems in complex systems is that of language limitations. It should not be assumed that the persons developing and interacting with the system are familiar with the language in which the tool is written, especially in complex domains such as that of EU law.

The objective of the NILDE project is to develop a computable model of the French-Italian Double Taxation Convention that is able to cope with the above mentioned issues. The system relies on Italiano Logico, a version of Logical English[1] based on the Italian language. This computable model can be used by means of a user-friendly interaction, both in Italian and other languages.

The main goal was thus to build a knowledge base from the Convention, and define a way for potential users to interact with the system, keeping in mind that the users may not be familiar with the language used in modelling the rules.

To address this linguistic issue we decided to enable the rules to be converted in multiple natural languages, separating the logic structure from the specific natural languages. By having each logic statement correspond to multiple natural languages, it is possible for the legal professionals to write the rules in their main language, and have them be translated, either automatically or with the intervention of legal experts familiar with other languages. The latter approach ensures that the rules use the correct terminology in use in the specific legal system, so we might envision the case in which the initial language is translated automatically, then validated by legal experts.

This approach, while not fully language independent, makes it possible for the different parties involved to interact through natural language, lowering the linguistic barriers, and enabling different interactions in the future.

1.1. Legal Background

The NILDE project focuses on the domain of tax conventions against double taxation, known as a Double Taxation Convention (DTC).

A DTC is an international bilateral treaty between two States. Through a DTC the two parties to the convention accept an international law obligation and commit themselves to relinquishing, completely or partially, the application of taxes in specific situations. The objective of a DTC is to ensure that people or companies, who are resident in one country but produce or receive income from another, are not taxed by both. Hence, the aim of a DTC is to avoid double taxation[2].

Currently over 2,500 DTCs exist. Each DTC is negotiated separately and the two contracting states are free to decide how to regulate the tax issues arising at the international level. Notwithstanding this freedom, many DTCs resemble each other, as international organisations first developed Model Tax Conventions which acted as the starting point for bilateral negotiations.

In the field of taxation, the Organisation for Economic Cooperation and Development (OECD) and the United Nations (UN) drafted the most famous Model Conventions, as well as their Commentaries, which serve as the multilateral basis for bilateral negotiation[2]. In most aspects the UN Model follows the OECD Model and deviations exist only with respect to certain rules.

Most developed countries follow the OECD Model Convention, while developing countries usually rely on the UN Model.

In the NILDE project we focused on the DTC between Italy and France, which follows the OECD Model.

The choice to model DTC rules is motivated by many factors. First, the production of transnational income is common, including for taxpayers who do not have an in-depth knowledge of taxation. Therefore, an application which provides indications to taxpayers concerning the place where they have to pay taxes could be of widespread interest. Second, the DTC represents a closed and self-sufficient system of rules, with a limited number of references to domestic sources. Third, the way in which DTCs based on the OECD Model Convention are written ensures a clear structure. Therefore, they are easy to understand and to model. Fourth, most DTCs are stable and their structure has remained the same over a long period of time (most of the rules of the OECD Model Convention date back to 1963). This reduces the risk of changes that alter the text and make the modelled rules less valuable. Finally, because of the similarity between conventions concluded on the basis of the OECD Model, the work done with NILDE could in the future be used for the translation of other conventions ratified by Italy. This would lead to its wider usability.

1.2. Logical English and Italiano Logico

Logical English is a controlled natural language interface, that translates to Prolog. It enables the description, modelling, and consulting of logic programs in natural language.

Programs in Logical English are written on the basis of a set of templates, sentences with predefined positions for variables, that can be automatically translated in Prolog predicates by a metainterpreter. The generate Prolog code is then evaluated by a standard Prolog reasoner.

The solution can be returned in Prolog, as if it were a regular program, or converted back in natural language, and enriched with connectors to define the relation between different predicates/sentences. This proof tree can be consulted as a human understandable explanation of the functioning of the system.

This drafting process makes both writing, reading, and understanding the rules easier for people without a technical background in logic or computer science. Furthermore the drafted rulebase is presented in a computable form, that is understandable, and auditable by domain experts, lowering the barrier to their direct involvement in the rule modelling. This approach helps overcoming the aforementioned technical issues with legal knowledge modelling.

Logical English is not limited to rule modelling in the English language, and, with the exception of the logic operators and the interface itself, is language agnostic, depending only on the language in which the templates are provided. In the following section we will explore the additions made in the course of the NILDE project to Logical English, to enable reasoning with the Italian language, and present the possibility of interacting with a codebase in multiple languages, thus further lowering the barrier of entry, and simplifying access to legal provisions modelled as computer programs.

Additions to the basic syntax of Logical English make it possible to write rules in multiple languages: English, Italian, Spanish, French, and Portuguese. Normally the language used to query the system is the same used to write the rules. One of the goals of the NILDE project is to

add the possibility to change the language, and verify that a program written in one language can be translated and queried in another.

An additional goal of the project was to see how experts of the legal domain, with no specific pre-existing experience in logic rule modelling or programming would fare in modelling rules. From this experience we will also see if the issues they might identify as major are consistent with the ones that the programmers may have.

2. Related works

Many researchers have focused on modeling rules and developing rule-based systems for the law, and here we will not mention all the proposed methodologies, rather introduce the context in which our research takes place, to better understand the context in which our research took place.

Particularly significant was the formalization of the British Nationality Act using logic programming[3] which paved the way for a number of rule-based models of different legal domains[4].

Many theoretical issues relating to the representation of legal knowledge have been raised and discussed, including aspects such as isomorphism[5] deontic positions, open texture[6], and how to deal with defeasibility in law[7].

More complex frameworks have been developed to capture the dialectical relation between arguments in legal contexts[8].

Some modern approaches have been addressing certain limitations, providing controlled natural languages for representing legal knowledge in a more human friendly way[1], combining symbolic logic approaches with data driven ones[9] or extending ASP without grounding and constraints [10].

3. Methodology

In this section we will go over the methodology undertaken for the rule modelling, describe the steps involved and the issues that arise. We will then describe our approach to the issues, and how modelling legal norms in logic may help in understanding their precise meaning.

3.1. Phases

The work in NILDE was carried out in three phases:

1. Analysis of the French-Italian Convention (the 'Convention').
2. Implementation and modelling of some of the rules in Italiano Logico.
3. Preparation of the interface, i.e. development of the interface to be used by the user.

During the first phase we identified the structure, the drafting techniques, and the syntax of the rules of the Convention. The objective was to build a knowledge base for the implementation of the rules. The structure of the Convention is quite simple. It consists of four sets of rules dealing with: the scope of the Convention (persons covered and taxes covered) and definitions;

the allocation of taxing rights either to Italy or France; the elimination of double taxation; and special provisions. The drafting technique consists in a sequence of rules and exceptions. This means that almost every article of the Convention has a first paragraph providing for a general rule. The following paragraphs provide for exceptions to the general rule.

Definitions of the concepts used in the rules are provided either by the rule itself or by Article 3, which concerns definitions. The syntax is generally straightforward for the general rules and more complex for the exceptions. The Convention's reference to definitions, rules, or concepts in other sources of law is infrequent, while the reference to other rules of the Convention is fairly frequent. The lexicon is legalistic but more simple than that used in national law.

Since the project lasted for only one year, we could not model all the rules of the Convention. Hence, we decided to select the rules based on two criteria: relevance, i.e. the rules whose application is more frequent (e.g. allocation of taxing rights concerning employment income); clarity and conciseness, i.e. we first modelled rules whose content was less complex and which could be applied to specific cases without further specifications. Most of the modelled rules concern the scope of the Convention, the allocation rules, and the rules for the elimination of double taxation.

The modelling of the rules took place following a precise sequence. First, a group made up of a lawyer and a computer scientist (an expert in the modelling of rules in Logical English) modelled a couple of rules. Afterwards, the lawyer modelled the other thirty rules alone. During the modelling phase the lawyer wrote down their doubts. These doubts were periodically shared with the computer scientist. Once the first draft of the rules was ready, they revised the rules together. The computer scientist made sure that Italiano Logico had the characteristics necessary to correctly express the logical relations contained in the rules. One of the most interesting aspects of NILDE is thus that the lawyer intervened in both the first and second phase of the project.

3.2. Logic modelling of legal norms

There have been multiple approaches to formal modelling of legal norms, starting in the '80s with the representation of statutes on UK citizenship[3] and tax benefits[11], to more recent approaches to address the needs of specific domains.

The development of PROLEG[12] and s(CASP)[13] focus, among other aspects, on dealing with incomplete information in reasoning on open concepts, while approaches like CATALA[14] focus on building efficient executable specifications using default logic.

The approach we present here is based on the use of controlled natural languages to encode the rules in a more accessible manner, and the additional possibility of interacting with the program in multiple languages.

The main challenges to consider when representing legal norms like the ones present in the Convention are: the relation between rules with different outcomes (exceptions), the issues of open texture and ambiguity, and the needed reference to other legal documents to model the rules effectively.

Rules-exceptions structure Conventions are structured as a set of general rules and exceptions. This structure concerns the relationship between both rules (i.e., Article N is an exception

to Article N+1) and paragraphs within the same rule (i.e., Paragraph N+1 of Article X is an exception to Paragraph N of the same Article).

The first hypothesis is evident for rules concerning the income of physical persons performing both independent and dependent personal services. Since some personal services have some specific features (e.g. they are strictly related to the territory of the State where they are performed, as with sport), it is to be expected that there are exceptions to the main two rules.

For example, Article 17, concerning the income of artists and athletes, provides that 'Notwithstanding the provisions of Articles 14 and 15, income derived by a resident of a State [...] may be taxed in that other State'. Article 14 concerns income deriving from independent personal services, while Article 15 concerns income deriving from dependent personal services.

The rules-exceptions structure within each article is quite easy to identify, especially in allocation rules. These rules often start with a paragraph containing the general rule which is followed by an exception. In addition, there are often exceptions that apply to the exception.

For example, we should understand the first sentence of Article 10 as a general rule: 'Dividends paid by a company which is a resident of one State to a resident of the other State may be taxed in that other State'. The rule is followed by an exception: 'However, such dividends may also be taxed in the State of which the company paying the dividends is a resident and according to the law of that State'. Nonetheless, there is also an exception to the exception: 'if the recipient is the beneficial owner of the dividends, the tax so charged may not exceed 5 per cent of the gross amount of the dividends if the beneficial owner is a company subject to corporation tax which holds, directly or indirectly, for a period of at least 12 months preceding the date of declaration of the dividends, at least 10 per cent of the capital of the company paying the dividends'.

Hence, the first difficulty to be faced by the lawyer in modelling the rules is to understand the relationship between general rules, exceptions, and exceptions to the exceptions. To this end, before drafting the rules in Italiano Logico, it is necessary to know the content of the DTC in detail and which rules are derogated by other rules.

In addition, the articles of the DTC frequently include so-called 'tie-breaker rules', 'conflict rules', 'inclusion rules', and 'exclusion rules'. Tie-breaker rules apply in cases of double residence. In fact, conventions frequently rely on national rules to assess residence for tax purposes. Based on national rules it is possible that a person is considered to be a resident of both the contracting states. The purpose of the tie-breaker rules is to attribute tax residency to just one country. An example of a tie-breaker rule can be found in Article 4 Paragraph 1: 'if a person is a resident of both states, his status will be determined as follows: he will be considered to be a resident of the state where he has a permanent dwelling at his disposal; if he has a permanent dwelling at his disposal in both states, he will be considered to be a resident of the state with which his personal and economic relations are closest (centre of vital interests)'. A 'conflict rule' is a rule which excludes the application of another rule in certain cases. An example of a 'conflict rule' is the so called PE proviso, i.e. an exception to the rules concerning the taxation of dividends, interest, royalties, and capital gains in cases in which the company receiving those types of income has a permanent establishment (PE) in the country of distribution. For example, according to Article 10 Paragraph 7: 'The provisions of paragraphs 1 and 2 do not apply if the beneficial owner of the dividends, being a resident of one state, carries on business in the other state of which the company paying the dividends is a resident, through a permanent establishment situated there'. Finally, there are many 'inclusion rules' or 'exclusion rules' within definition rules,

which are rules defining the notions used within the DTC. Since the DTC applies to countries having different legal traditions and different legal terminology, definition rules are important. 'Inclusion rules' or 'exclusion rules' provide that a certain situation falls within or outside the definition. For example, according to Article 7 Paragraph 2, 'immovable property *shall in any case encompass* property accessory to immovable property, livestock and equipment used in agriculture and forestry, as well as rights to which the provisions of general law respecting landed property apply', while according to Article 5 Paragraph 3 'permanent establishment *shall not be deemed to include*: (a) the use of facilities solely for the purpose of storage, display or delivery of goods or merchandise belonging to the enterprise; etc.'.

Open texture and ambiguity One of the main difficulties in implementing rules in Italiano Logico concerns general clauses. A 'general clause' is a legal term or concept which is not precisely formulated (e.g. 'good faith' in Article 10) and does not even have a clear core. In this case it is necessary to give content to textual elements involving an evaluation that is very difficult to make with logical modelling. In fact the content of general clauses is to be determined case by case by the public authority or judge applying the rule to the facts.

Similar issues arise in relation to: broad expressions, such as 'centre of vital interests' (Article 4); vague/ambiguous concepts which are not (properly) defined by the Convention, such as 'fixed place of business' (Article 5); or 'relational concepts', i.e. concepts which are defined in relation to the core features of other concepts. For example, Article 2 Paragraph 4, with regard to the taxes covered by the Convention, provides that it applies to 'any identical or substantially similar taxes' (Paragraph 4). In this case modelling this rule entails either identifying the most important features of the taxes listed and writing a rule which helps to assess the similarity between taxes, or identifying the existing taxes which 'have a similar nature'. The first proposal is very complex to pursue in practice. The second solution is likely to be updated soon, as the tax system undergoes frequent changes.

In NILDE, due to time constraints, we had to assume that the user knew the content and meaning of most general clauses, broad, vague and relational concepts which were not defined within the DTC. A way to address this issue is to include information presented to the user at question time, or to encode additional rules, useful to clarify details not contained in the source of the Convention.

Reference to national law Some of the rules of the Convention refer to national rules or definitions contained within national law.

For instance, Article 3 Paragraph 2, which defines the terms used in the DTC, provides that any term not defined by the Convention shall 'unless the context otherwise requires, have the meaning which it has under the law of that state concerning the taxes to which the Convention applies'. Article 6 Paragraph 2, regulating the allocation rule for income from immovable property, provides that immovable property 'shall be defined in accordance with the law of the State in which the property in question is situated'. In these cases, the person modelling the rules must first have access to the law of the countries involved. Moreover, they must know the legal system and language of those countries. One of the reasons why we decided to model the French-Italian DTC was that we knew both languages, and French legislation is easily accessible

online. We modelled most of the rules to which the DTC referred but did not implement all of them, due to time constraints.

In the future, the knowledge base could be enriched with other rules of the DTC and useful national provisions (e.g. those used to determine the place of residence and domicile). It is worth keeping in mind that national rules have to be added carefully, as different national rules might impact the DTC in different ways.

4. Implementation

4.1. Logical English translation

As mentioned in the previous sections, the rules were modelled in Italian. In order to do so small modifications to the Logical English base were made.

Italiano Logico Previous uses of *Italiano Logico*, the Italian version of Logical English, were examples from the Italian Citizenship law [1]. In the development the NILDE rulebase, changes were made to the builtin operators and linguistic formulations, such as negation (e.g. from the English *it is not the case that* to the Italian *non è il caso che*).

Currently Logical English provides formulations for multiple languages: English, Italian, Spanish, French, and Portuguese. This is made possible by the fact that the specific additions for the different languages are limited to a small lexicon, expressed in the LE codebase as a DCG (Listing 1).

The rules were modelled as close as possible to the original text, in order to preserve the intended meaning.

As an example, consider Article 6, that determines that a certain income derived from an *immovable property* (real estate) has to be taxed in the state in which the property is located, even if the person receiving the income is resident in another country.

The formulation in Italiano Logico is as follows:

> un reddito di una persona è tassabile in uno stato S ai sensi dell'articolo 6 se la persona riceve il reddito per la proprietà e la persona è residente in uno stato R e la proprietà è di tipo bene immobile e la proprietà si trova in uno stato S.

In the encoded version we can see that there are concepts not explicitly defined in the document, such as that of residence. This may have different definitions in the national legislation of the countries involved. By adding that information to the rulebase the system would become more useful and complete.

Using LogicalEnglish the code is converted to Prolog internally, but the interaction is all done in natural language, from the point of view of the user and the knowledge engineer.

The first way to interact with the system is to provide scenarios with the facts relevant to a specific case, and query the system.

If we tell the system that

```
ind_det --> [a].
ind_det --> [an].
ind_det --> [un].        % spanish, italian, and french
ind_det --> [una].       % spanish, italian
ind_det --> [une].       % french
ind_det --> [qui].       % french which
ind_det --> [quel].      % french which masculine
ind_det --> [quelle].    % french which femenine
ind_det --> [che].       % italian which
ind_det --> [quale].     % italian which
ind_det --> [uno].       % italian
ind_det --> ['cuál'].    % spanish

def_det --> [the].
def_det --> [el].        % spanish
def_det --> [la].        % spanish, italian and french
def_det --> [le].        % french
def_det --> [l], [A], {atom_string(A, "'")}.  % french, italian
def_det --> [il].        % italian
def_det --> [lo].        % italian
```

Listing 1: Snippet of the LE input DCG

> scenario federico: federico riceve reddito 1 per affitto su ufficio di via galliera 3. federico è residente in francia. ufficio di via galliera 3 è di tipo bene immobile. ufficio di via galliera 3 si trova in italia.

can be queried in both Italian and French (reddito 1 being here an identifier for the income, although it could be changed to represent a numerical value if necessary), telling the system to 'rispondere reddito 1 di federico è tassabile in quale stato ai sensi di quale articolo con scenario federico' (answer reddito 1 of federico is taxable in which state according to which article with scenario federico). The system will load the facts corresponding to the selected scenario, and answer replacing the 'which ...' statements with the solution.

This is the normal behaviour of Logical English, and already guarantees that an answer will be presented in natural language, and as we will see, also including an explanation derived from the rules used to reach the goal.

In the following section we will also show querying the system in different languages, and thus how the solution can be presented to users.

4.2. Explanation

The Logical English explanations have builtin phrases and reserved words used to display the answer, such as *because*, and *it is the case that*. These were translated to the Italian formulations.

reddito 1 di federico è tassabile in italia ai sensi di articolo 6 perché

- ufficio di via galliera 3 è di tipo bene immobile e
- federico riceve reddito 1 per ufficio di via galliera 3 perché
  - federico riceve reddito 1 per affitto su ufficio di via galliera 3

  e
- federico è residente in germania e
- ufficio di via galliera 3 si trova in italia

Listing 2: Answer with Explanation

The explanation is built from the execution trace, with the different solved predicates presented in natural language. It is possible (though not yet available in the upstream Logical English codebase) to ask for the explanation in a specific language, and as long as the templates have been written in that language, the LE/Prolog reasoner will do so.

In our case the main goal was to enable querying the system in Italian and French, so the templates were converted from Italian to French through automated translation. Since the program relies on templates, it is possible for the legal expert to validate the translation and make the necessary changes.

Through this methodology multiple formulations in natural language can be used to query and interact with the knowledge base.

The goal

```
di_è_tassabile_in_ai_sensi_di(reddito_1, federico, italia,
                              'articolo 6').
```

can be presented in natural language as both Italian and French (or other languages), thus be more useful for the user of the system:

```
Italian answer:
    reddito_1 di federico è tassabile in italia ai sensi di articolo 6
French answer:
    reddito_1 de federico est imposable en italia sous article 6
English answer:
    reddito_1 of federico is taxable in italia under article 6
```

The linguistic variants have a specific lexicon, similar as the one above, for the connectors and keywords, such as because (perché, parce-que, ...). This additional structure enables us to present the solution with an explanation, shown here in italian, but translatable to any one of the languages included:

additional commands to execute during container initialization The controlled natural language representation is still structured, and presented as such to the user via indentation, to clarify visually which conditions depend on which others.

5. Future work and conclusions

The representation and answers we obtain from the system could be enhanced or further edited through the use of Large Language Models, while keeping the rules expressd in a stricter form, without this hindering the rule drafting, as could be the case translating pure logic statements through LLMs.

Another possible future development that has been explored in the context of the project is the construction of an interactive question-answer system, that converts the natural language statements in questions, and through a Prolog metainterpreter, asks the user the necessary questions depending on the available facts. This interface could leverage the different localized versions of the templates to ask questions in the language the user is more familiar with.

In this paper we demonstrated the use of Logical English and the linguistic variants to encode legal rules in a multilinguistic domain, where it is important to remove obstacles hampering the access to the system for potential users with different linguistic and legal backgrounds.

As mentioned in the introduction, we decided to model the Italian-French Convention against double taxation because it (and similar conventions) is used in the context of University collaborations. A tool like the one proposed in this project would prove to be very useful for the administrative offices in the University, and other administrations that rely on foreign collaborations.

The representation and its usefulness are further enhanced by adding the possibility of linking multiple linguistic variants, enabling a more natural interaction for users who are not familiar with the language in which the rulebase is written.

This also removes some of the responsibility for consistency from the hands of the original drafters of the logic rulebase. Often the direct translation is not the correct one and, especially in the legal domain, the specific words used carry important meaning. While the initial modelled controlled natural language structure can be automatically translated, it is possible to tie the program with official translation services, like the one used by the EU courts, or to have legal experts from other countries focus not on rewriting the rules, rather on checking the natural language concepts used.

Logical English and the methodology adopted in this paper can possibly be adapted to approach the modeling of legal knowledge in many areas areas of the law, in particular in multilinguistic domains, and in those contexts where one of the main requirements is to ensure esplicability of the outcomes produced by the system.

References

[1] R. Kowalski, J. Dávila, G. Sartor, M. Calejo, Logical english for law and education, in: D. S. Warren, V. Dahl, T. Eiter, M. V. Hermenegildo, R. Kowalski, F. Rossi (Eds.), Prolog: The Next 50 Years, Lecture Notes in Computer Science, Springer Nature Switzerland, Cham, 2023, pp. 287–299. doi:10.1007/978-3-031-35254-6_24.

[2] M. Lang, Introduction to the law of double taxation conventions, Linde Verlag GmbH, 2021.

[3] M. J. Sergot, F. Sadri, R. A. Kowalski, F. Kriwaczek, P. Hammond, H. T. Cory, The british nationality act as a logic program 29 (1986) 370–386. doi:10.1145/5689.5920.
[4] T. F. Gordon, G. Governatori, A. Rotolo, Rules and norms: Requirements for rule interchange languages in the legal domain, in: Lecture Notes in Computer Science, Springer Berlin Heidelberg, ????, pp. 282–296. doi:10.1007/978-3-642-04985-9_26.
[5] T. J. M. Bench-Capon, F. P. Coenen, Isomorphism and legal knowledge based systems, Artificial Intelligence and Law 1 (1992) 65–86. URL: https://doi.org/10.1007/BF00118479. doi:10.1007/BF00118479.
[6] T. Bench-Capon, P. Visser, Open texture and ontologies in legal information systems, in: Database and Expert Systems Applications. 8th International Conference, DEXA '97. Proceedings, 1997, pp. 192–197. doi:10.1109/DEXA.1997.617268.
[7] G. Governatori, A. Rotolo, E. Calardo, Possible world semantics for defeasible deontic logic, in: T. Ågotnes, J. Broersen, D. Elgesem (Eds.), Deontic Logic in Computer Science, Springer, Berlin, Heidelberg, 2012, pp. 46–60. doi:10.1007/978-3-642-31570-1_4.
[8] H. Prakken, G. Sartor, Law and logic: A review from an argumentation perspective, Artificial Intelligence 227 (2015) 214–245. URL: https://www.sciencedirect.com/science/article/pii/S0004370215000910. doi:10.1016/j.artint.2015.06.005.
[9] R. Calegari, G. Ciatto, A. Omicini, On the integration of symbolic and sub-symbolic techniques for xai: A survey, Intelligenza Artificiale 14 (2020) 7–32.
[10] J. Arias, M. Carro, E. Salazar, K. Marple, G. Gupta, Constraint answer set programming without grounding, 2018. URL: https://arxiv.org/abs/1804.11162. arXiv:1804.11162.
[11] T. J. M. Bench-Capon, G. O. Robinson, T. W. Routen, M. J. Sergot, Logic programming for large scale applications in law: A formalisation of supplementary benefit legislation, in: Proceedings of the first international conference on Artificial intelligence and law - ICAIL '87, ACM Press, 1987. doi:10.1145/41735.41757.
[12] K. Satoh, K. Takahashi, T. Kawasaki, Interactive system for arranging issues based on PROLEG in civil litigation, in: Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law, ICAIL '21, Association for Computing Machinery, New York, NY, USA, 2021, pp. 273–274. URL: https://doi.org/10.1145/3462757.3466096. doi:10.1145/3462757.3466096.
[13] J. Arias, M. Moreno-Rebato, J. A. Rodriguez-García, S. Ossowski, Automated legal reasoning with discretion to act using s(LAW), Artificial Intelligence and Law (2023). URL: https://doi.org/10.1007/s10506-023-09376-5. doi:10.1007/s10506-023-09376-5.
[14] L. Huttner, D. Merigoux, Traduire la loi en code grâce au langage de programmation Catala, 2020. URL: https://hal.inria.fr/hal-03128248.