Modeling Administrative Discretion Using Goal-Directed Answer Set Programming

Joaquín Arias[1], Mar Moreno-Rebato[1], José A. Rodríguez-García[1] and Sascha Ossowski[1]

[1]*CETINIA, Universidad Rey Juan Carlos, Madrid, Spain*

Abstract

This paper is an extended abstract of: J. Arias, M. Moreno-Rebato, J. A. Rodriguez-García, S. Ossowski, Modeling Administrative Discretion Using Goal-Directed Answer Set Programming, in: Advances in Artificial Intelligence, CAEPIA 20/21, Springer International Publishing, Cham, 2021, pp. 258–267. doi:10.1007/978-3-030-85713-4_25 [1].

Keywords

Answer Set Programming, Goal-Directed Evaluation, Administrative Discretion

The formal representation of a legal text to automatize reasoning about them is well known in literature, and is recently gaining much attention thanks to the interest in the so-called smart contracts, and to autonomous decisions by public administrations [2, 3, 4]. For deterministic rules there are several proposals, often based on logic-based programming languages [5, 6]. However, none of the existing proposals are able to represent the ambiguity and/or administrative discretion present in contracts and/or applicable legislation, e.g., *force majeure*.

In this work we present a framework, called s(LAW) [1], that allows for modeling legal rules involving ambiguity, and supports reasoning and inferring conclusions based on them. Additionally, thanks to the goal-directed execution of s(CASP) [7], the underlying system used to implement our proposal, s(LAW) provides justification [8] of the resulting conclusions (in natural language). To evaluate the expressiveness of our proposal we have translated (using a set of patterns) part of the rules of the procedure for awarding school places for the "Educación Secundaria Obligatoria" (ESO) of centers supported with public funds in the Comunidad de Madrid.

Patterns to translate law into ASP The first contribution is a set of patterns to translate ambiguity and/or discretion concepts, that in previous proposals required the help of an expert the field of application, to specify only one interpretation and/or decision.[1]

2nd Workshop on Goal-directed Execution of Answer Set Programs (GDE'22), August 1, 2022

joaquin.arias@urjc.es (J. Arias); mar.rebato@urjc.es (M. Moreno-Rebato); joseantonio.rodriguez@urjc.es (J. A. Rodríguez-García); sascha.ossowski@urjc.es (S. Ossowski)

http://www.ia.urjc.es (J. Arias)

0000-0003-4148-311X (J. Arias); 0000-0002-4177-9239 (M. Moreno-Rebato); 0000-0002-6362-9880 (J. A. Rodríguez-García); 0000-0003-2483-9508 (S. Ossowski)

[1] On January 14th, 2021, Dr. Robert Kowalski explained how they bypassed in [6] the representation of vague concepts such as *without undue delay* [9, 1:20:15, 1:26:00].

1. **Requirement For Applying** These are the most common constructions in legal articles. There are two patterns: (i) Disjunction, and (ii) Conjunction.
2. **Exceptions For Applying** They are encoded using negation as failure.
3. **Ambiguity** Ambiguity occurs when some aspects of the law can be interpreted in different ways. For example, "proximity to the family or work address" is a specific and defined requirement based on the distribution by educational districts. However, in case of *force majeure*, students from a education district may be reassigned to a school from another district. The encoding below allows evaluation without having to determine a priori the force majeure circumstances necessary to justify the reassignment of students.

```
school_proximity :-   same_education_district.
school_proximity :-   not same_education_district, force_majeure.
force_majeure :-      not n_force_majeure.
n_force_majeure :-    not force_majeure.
```

   This pattern generates a model where `force_majeure` is assumed to hold and another model where there is *no* evidence that `force_majeure` holds.
4. **Discretion To Act** The discretion to act introduces different possible interpretations of the law and/or the contract that we intent to model by generating multiple models. Implementations based on Prolog compute a single, canonical model, and therefore, bypass this non determinism by selecting one interpretation. Using s(LAW), we obtain two possible models: In one model the discretion to act is applied (according to the purpose / intention of the law and it is not unlawful) and in the other it does not.
5. **Unknown Information** The use of default negation may introduce unexpected results in the absence of information (positive and/or negative). Therefore, in many cases the desirable behavior should capture the absence of information by generating different models depending on the relevant information. To state that some information is certain we would use the predicate `evidence/1`, and to specify that we have evidences supporting it falsehood we would use *strong* negation, i.e., `-evidence/1`.

The framework: The second contribution is s(LAW)[2], built on top of s(CASP), and composed by three modules: the first contains the *articles*, the second contains *explanations* to generate readable justifications, and the third one contains the *evidences* for each candidate.

- **A priori Deduction** Table 1 shows the data corresponding to six candidates and the conclusion generated by s(LAW) for the query `?- obtain_place`. Students 1, 3, 4, and 5 obtain a place at the school while students 2 and 6 do not. Fig. 1 shows the justification in natural language for student 1.
- **A posteriori Deduction** s(LAW) generates justifications not only for positive but also for negative information, so we can analyze the reason for a specific inference and/or to determine which are the requirements needed to obtain a specific conclusion. E.g, the query `?- not force_majeure, obtain_place` avoids the assumption of force majeure and the student 3 would not obtain a place.

[2] Available at http://platon.etsii.urjc.es/~jarias/papers/slaw-caepia21.

Table 1
Case of different students evaluated using s(LAW).
Note: '+' is a positive evidence, '—' is a negative evidence, '?' means unkown.

	st_1	st_2	st_3	st_4	st_5	st_6
`large_family`	+	+	+	—	—	—
`renta_minima_insercion`	+	+	+	?	—	—
`sibling_enroll_center`	+	+	—	+	—	—
`same_education_district`	+	+	—	+	—	—
`b1_certificate`	+	—	+	?	—	—
`foreign_student`	—	—	—	—	+	—
`specific_etnia`	—	—	—	—	—	+
`?- obtain_place`	**yes**	no	**yes**	**yes**	**yes**	no

```
s/he may obtain a school place, because
    a common requirement is met, because
        s/he is part of a large family.
    a specific requirement is met, because
        s/he has siblings enrolled in the center.
    there is no evidence that an exception applies, because
        s/he came from a non-bilingual public school, and
        s/he wish to study 2nd ESO in the Bilingual Section, and
        s/he accredit required level of English for 2nd ESO, because
            in the four skills certificate level b1.
```

Figure 1: Justification in Natural Language for the evaluation of `student01.pl`.

Additionally, we can collect the partial models, in which the school place is or is not obtained, together with their justification and analyze "Epistemic Specifications" [10], that is, what is true in all/some models, which partial models share certain assumptions, etc. This reasoning makes it possible to detect the missing information that would change the decision from "not obtained" (or "obtained" under some assumptions) to "obtained".

In conclusion, we have shown that using goal-directed answer set programming, s(LAW) is capable of modeling discretion and ambiguity. The deduction based on s(LAW) allows: the consideration of different conclusions (multiple models) which can be analyzed by humans thanks to the justification generated in natural language; and the reasoning about the set of these conclusions/models. We would like to emphasize that explainable AI techniques for black-box AI tools, most of them based on machine learning, are not able to explain how variation in the input data changes the resulting decision [11]. To the best of our knowledge, s(LAW) is the only system that exhibits the property of modelling vague concepts.

References

[1] J. Arias, M. Moreno-Rebato, J. A. Rodriguez-García, S. Ossowski, Modeling Administrative Discretion Using Goal-Directed Answer Set Programming, in: Advances in Artificial Intelligence, CAEPIA 20/21, Springer International Publishing, Cham, 2021, pp. 258–267. doi:10.1007/978-3-030-85713-4_25.

[2] A. Cerrillo i Martínez, El derecho para una inteligencia artificial centrada en el ser humano y al servicio de las instituciones: Presentación del monográfico., IDP: Revista de Internet, Derecho y Politica (2019).

[3] J. Cobbe, Administrative law and the machines of government: judicial review of automated public-sector decision-making, Legal Studies 39 (2019) 636–655.

[4] J. P. Solé, Inteligencia artificial, derecho administrativo y reserva de humanidad: algoritmos y procedimiento administrativo debido tecnológico, Revista general de Derecho administrativo 50 (2019).

[5] S. Ramakrishna, Ł. Górski, A. Paschke, A dialogue between a lawyer and computer scientist: the evaluation of knowledge transformation from legal text to computer-readable format, Applied Artificial Intelligence 30 (2016) 216–232.

[6] M. J. Sergot, F. Sadri, R. A. Kowalski, F. Kriwaczek, P. Hammond, H. T. Cory, The british nationality act as a logic program, Communications of the ACM 29 (1986) 370–386.

[7] J. Arias, M. Carro, E. Salazar, K. Marple, G. Gupta, Constraint Answer Set Programming without Grounding, Theory and Practice of Logic Programming 18 (2018) 337–354. doi:10.1017/S1471068418000285.

[8] J. Arias, M. Carro, Z. Chen, G. Gupta, Justifications for goal-directed constraint answer set programming, in: Proceedings 36th International Conference on Logic Programming (Technical Communications), volume 325 of *EPTCS*, Open Publishing Association, 2020, pp. 59–72. doi:10.4204/EPTCS.325.12.

[9] R. A. Kowalski, Logical English = Logic + English + Compupting, https://utdallas.app.box.com/s/ngsyloscj5sk24uh3axexxz451o74z0u, 2021. HackReason Opening Ceremony. Last accessed 19 April 2021.

[10] M. Gelfond, Logic programming and reasoning with incomplete information, Annals of mathematics and artificial intelligence 12 (1994) 89–116.

[11] DARPA, Explainable Artificial Intelligence (XAI), Defense Advanced Research Projects Agency, 2017. https://www.darpa.mil/program/explainable-artificial-intelligence.