Reasoning about Judgement in GDPR Litigation by PROLEG (Demonstration Paper)

Ken SATOH [a,1], Marina DE VOS [b], Julian PADGET [b] and Sabrina KIRRANE [c]
[a] *National Insitute of Informatics*
[b] *The University of Bath*
[c] *the Vienna University of Economics and Business*

Abstract. In this paper, we show an application of PROLEG (PROlog-based LEGal reasoning support system) to GDPR (General Data Protection Regulation) litigation. PROLEG has been originally used to formalize reasoning in a civil litigation, but it is also generally applicable to laws which include general rules and exceptions. GDPR consists of general rules and exceptions so it is possible to formalize it in PROLEG. This paper discusses a concern about problems to represent GDPR in PROLEG.

Keywords. GDPR, PROLEG

1. Introduction

In this paper, we investigate possibility of applicability of PROLEG (PROlog-based LEGal reasoning support system) [2] to GDPR litigation. ROLEG has been originally used to formalize reasoning in a civil litigation. However, shown in [3], PROLEG is expected to be applicable to legal systems based on inference using general rules and exceptions, and GDPR is such a law. Therefore, it would be possible to implement reasoning about GDPR. In this paper, we show an example of reasoning about infringement related with data transfer around Article 44 in GDPR.

2. PROLEG

We firstly introduce PROLEG system [2]. The system consists of PROLEG rulebase and PROLEG factbase. We have shown that the representation power of PROLEG system is the same as PROLOG (with negation as failure) for an answer set semantics[3] so PROLEG system is very general.

[1] Corresponding Author: National Institute of Informatics, 2-1-2, Chiyoda-ku Tokyo, 101-8430, Japan; E-mail: ksatoh@nii.ac.jp

2.1. PROLEG Rulebase

PROLEG rulebase consists of the following expression; a rule and an exception. A **rule** of the form of Horn clauses (without negation as failure)

$$H \Leftarrow B_1, ..., B_n.$$

where H is called a head of the rules and B_i's is called a requisite. An **exception** is an expression of the form $exception(H, E)$ where H, E are atoms each of which is the head of rule.

The rule means that "H is satisfied in principle if all the requisites $B_1, ..., B_n$ are satisfied", whereas the exception means that "H is not satisifed if E is satisfied (even if all the requisites of a rule whose head is H are satisfied)".

2.2. PROLEG Factbase

A requisite in a general rule which does not have any rule in PROLEG rulebase whose head is the requisite itself is called a "**fact predicate**" and it is defined in a PROLEG factbase to represent concrete facts in the litigation. "$fact(XX)$" means that the fact XX has been asserted by either party and the judge's degree of confidence exceeds the degree of proof standard.

3. How to translate GDPR into PROLEG

Although the most of GDPR articles are of the form of rules, we need to consider the following.

3.1. Segmentation of Conditions into Predicate

The article is only shown in a natural language sentences, we need to segment these sentences into predicate form. This part actually introduces "interpretation" of rules. Therefore, we need to be careful about this segmentation. Currently, we use a commonsense of natural language interpretation and we separate the phrases if it is obvious. If the separation would be doubtful, we just put very long phrases without any segmentation.

3.2. Separating General Rules and Exceptions

To represent reasoning about GDPR, we need to separate parts of in the condition of rules in GDPR into general rule condition and exception condition. In a litigation, it would be possible that some condition cannot be decided due to lack of evidences. In this situation, a deductive answer is "I do not know". But in a litigation, it is not allowed. To solve this problem, in Japan, judges developed "ultimate fact theory" where from the view of AI, they put default truth value to each conditions and they use the default vale if the truth value of the corresponding condition is unknown. This idea is translated into PROLEG as follows. Suppose that we have a rule in GDPR as:

If B1,...,Bn, C1,...,Cm then H

where the default value of B1,...,Bn is false and C1,...,Cm is true. Then, we can have the following

```
H <= B1...Bn.
exception(H,E1)
...
exception(H,Em)
E1 <= -C1.
...
Em <= -Cm.
```

where E1,...,Em is the name of exception and -C1,...,-Cm is the negated expression of C1,...,Cm respectively. This translation is necessary since PROLEG has only default value of "false".

4. PROLEG Example

In accordance with the above implementation, an example of PROLEG program is shown below. This is an example of GDPR (Article 44 (General principle for transfers)[2]) which was presented in [1]

```
PROLEG rulebase:
infringement(
        transfer(
            ControllerOrProcessor,
            personal_data(DataSubject),
            CountryOrOrganisation)
    )<=
    transfer(
        ControllerOrProcessor,
        personal_data(DataSubject),
        CountryOrOrganisation),
    personal_data(DataSubject),
    undergoing_processing_or_intended_for_processing_after_transfer(
        personal_data(DataSubject),
        ControllerOrProcessor),
    third_country_or_international_organisation(CountryOrOrganisation).

% personal data
% under processing
undergoing_processing_or_intended_for_processing_after_transfer(
```

[2] Any transfer of personal data which are undergoing processing or are intended for processing after transfer to a third country or to an international organisation shall take place only if, subject to the other provisions of this Regulation, the conditions laid down in this Chapter are complied with by the controller and processor, including for onward transfers of personal data from the third country or an international organisation to another third country or to another international organisation. All provisions in this Chapter shall be applied in order to ensure that the level of protection of natural persons guaranteed by this Regulation is not undermined.

```
        personal_data(DataSubject),ControllerOrProcessor
    )<=
    process(ControllerOrProcessor,
        personal_data(DataSubject)).

% intend for processing after transfer
undergoing_processing_or_intended_for_processing_after_transfer(
        personal_data(DataSubject),
        ControllerOrProcessor
     )<=
     intend_for_processing_after_transfer(
        ControllerOrProcessor,
        personal_data(DataSubject)).

% a third country = not EAA country
third_country_or_international_organisation(Country)<=
     third_country(Country).
third_country(Country)<=
     not_EEA_country(Country).
% an international organisation = Article-4(26)
third_country_or_international_organisation(Organisation)<=
     international_organisation(Organisation).

% Exception by Article-46, 1
exception(
    infringement(
        transfer(
            ControllerOrProcessor,
            personal_data(DataSubject),
            CountryOrOrganisation)),
    subject_to_appropriate_safeguards(
        ControllerOrProcessor)).

subject_to_appropriate_safeguards(ControllerOrProcessor)<=
    has_provided(
        ControllerOrProcessor,appropriate_safeguards(Matter)),
    appropriate_safeguards(Matter),
    on_condition_that_enforceble_data_subject_right_and_effective_legal_remedies_for_data_s

% Various definition by Article-46, 2
% (a) a legally binding and enforceable instrument between public authorities or bodies;
appropriate_safeguards(a_legally_binding_and_enforceble_instrument)<=
    a_legally_binding_and_enforceble_instrument_between_public_authorities_or_bodies.

% (b) binding corporate rules in accordance with Article 47;
appropriate_safeguards('BCR')<=
    binding_corporate_rules_in_accordance_with_Article_47.
```

```
% (c) standard data protection clauses adopted by the Commission in
% accordance with the examination procedure referred to in Article
% 93(2);
appropriate_safeguards('SCC')<=
    standard_data_protection_clauses,
    'adopted_by_the_Commission_in_accodance_with_the_examination_procedure_referred_to_in_A

% (d) standard data protection clauses adopted by a supervisory
% authority and approved by the Commission pursuant to the examination
% procedure referred to in Article 93(2);
appropriate_safeguards('SCC')<=
    standard_data_protection_clauses,
    adopted_by_a_supervisory_authority,
    'approved_by_the_Commission_pursuant_to_the_examination_procedure_referred_to_in_Articl

% (e) an approved code of conduct pursuant to Article 40 together with
% binding and enforceable commitments of the controller or processor
% in the third country to apply the appropriate safeguards, including
% as regards data subjects' rights; or
appropriate_safeguards(code_or_conduct)<=
    approved_code_or_conduct_pursuant_to_Article_40,
    together_with_binding_and_enforceable_commitments_of_the_controller_or_processor_in_the
    "including_as_regards_data_subjects'_rights".

% (f) an approved certification mechanism pursuant to Article 42
% together with binding and enforceable commitments of the controller
% or processor in the third country to apply the appropriate
% safeguards, including as regards data subjects' rights.
appropriate_safeguards(certification_mechanism)<=
    an_approved_certification_mechanism_pursuant_to_Article_42,
    together_with_binding_and_enforceable_commitments_of_the_controller_or_processor_in_the
    "including_as_regards_data_subjects'_rights".

PROLEG factbase:
% Article 44 related fact
fact(transfer(companyA_Ireland,personal_data(personalData),thirdCountry)).
fact(personal_data(personalData)).
fact(not_EEA_country(thirdCountry)).
fact(process(companyA_Ireland,personal_data(personalData))).
fact(intend_for_processing_after_transfer(companyA_Ireland,personal_data(personalData))).

% Article 46(1) related fact
fact(has_provided(companyA_Ireland,appropriate_safeguards('BCR'))).
fact(on_condition_that_enforceble_data_subject_right_and_effective_legal_remedies_for_data_
```

Figure 1 shows the result of executing PROLEG (called PROLEG block diagram). In the output of the PROLEG block diagram, the conclusion is placed at the upper left, the boxes corresponding to the requisites of the general rule leading to the conclusion are connected by solid lines to the right of the conclusion. If a requisite is satisfied, "o" is shown in the bottom of the box, and if the requisite is not satisfied, "x" is shown. When exceptions have to be considered, boxes corresponding to the exceptions connected by broken lines below are shown, and the existence/nonexistence of the exception is indicated by "o" or "x". If one of the exception boxes has "o", the conclusion of the upper left becomes "x". In the case above, there is no exceptional information, the infringement is proved.

But if we add

```
% Article 46(2) related fact
% (b)
fact(binding_corporate_rules_in_accordance_with_Article_47).
```

then, the result is inversed(Fig.2).

5. Conclusion

In this paper, we discusss PROLEG implementation of GDPR litigation. As a future research, we would like to advance this implementation into GDPR compliance and consider how to handle normative terms such as `on_condition_that_enforceble_data_subject_right_and_effec` and `binding_corporate_rules_in_accordance_with_Article_47`.

Acknowledgments: This work was partially supported by JSPS KAKENHI Grant Number 19H05470.

References

[1] De Vos, M., Kirrane, S., Padget, J., Satoh, K., "ODRL Policy Modelling and Compliance Checking", Proceedings of Rules and Reasoning, the Third International Joint Conference (RuleML+RR 2019), pp. 36-51, `https://link.springer.com/book/10.1007/978-3-030-31095-0` (2019).

[2] Satoh, K., Asai, K., Kogawa, T., Kubota, M., Nakamura, M., Nishigai, Y., Shirakawa, K., Takano, C., "PROLEG: An Implementation of the Presupposed Ultimate Fact Theory of Japanese Civil Code by PROLOG Technology", New Frontiers in Artificial Intelligence: JSAI-isAI 2010 Workshops, Revised Selected Papers, LNAI 6797, pp. 153-164 (2012).

[3] Satoh, K., Kogawa, T., Okada, N., Omori, K., Omura, S., and Tsuchiya, K., "On Generality of PROLEG Knowledge Representation", Proceedings of the 6th International Workshop on Juris-informatics (JURISIN 2012), pp. 115 – 128 (2012).

November 2019

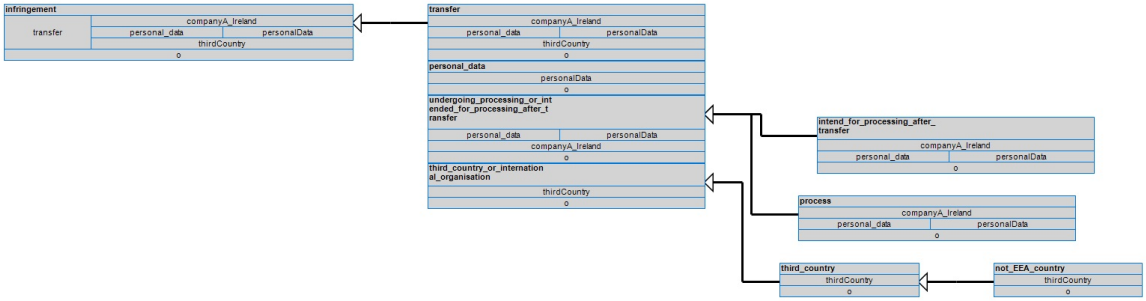

Figure 1. Example of Infrindgement

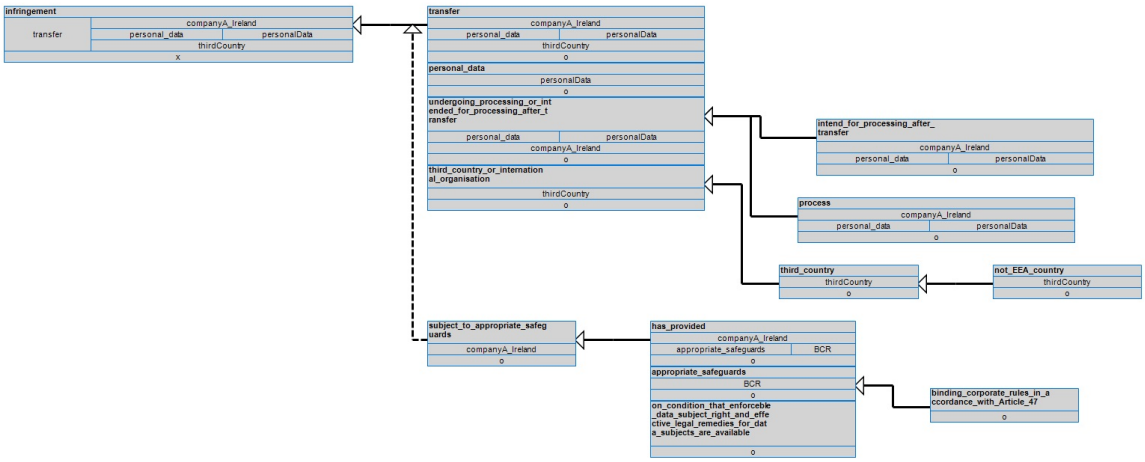

Figure 2. Example of Non-Infridgement