Legal Knowledge and Information Systems
G. Sileno et al. (Eds.)

doi:10.3233/FAIA230992

Human-Centred Explanation of Rule-Based Decision-Making Systems in the Legal Domain: Demonstration

Suzan ZUURMOND [a,b], AnneMarie BORG [b], Matthijs VAN KEMPEN [a], Remi WIETEN [a]

[a] *Dutch Tax and Customs Administration, The Netherlands*
[b] *Utrecht University, The Netherlands*

Abstract. We demonstrate a human-centred explanation method for rule-based automated decision-making systems in the legal domain. This explanation method uses a graph database to enable question-driven explanations and multimedia display. This way, we can tailor the explanation to the user. We show the implementation of our explanation method in a real-world scenario at the Dutch Tax and Customs Administration.

Keywords. Explainability, legal reasoning, Dutch Tax and Customs Administration

Introduction

As the ongoing digitisation of governance raises concerns globally, there is a pressing need for innovation and digital technologies that improve transparency [1]. An example of an innovation that improves the transparency of legislative automation is the Agile Law Execution Factory (ALEF) [2], in use at the Dutch Tax and Customs Administration (DTCA). ALEF is a tool for developing and generating service applications that can perform specific legal tasks like tax calculations. Among others, it uses the legal analysis schema in Figure 1. While tools like ALEF could contribute to explaining automated decisions [3], they often lack adequate explanation mechanisms.

In this paper, we introduce and demonstrate a human-centred explanation method for rule-based decision-making systems within the legal domain and its application at the DTCA.[1] As a result, (i) we introduce an explanation mechanism for systems, like the ones created using ALEF, offering a way for generating and conveying explanations that are tailored to the needs of individuals involved with these systems generated by such tools and (ii) we improve the explainability of automated decisions made at the DTCA and ensure that both the decision-making and the applied legislation become transparent to the individuals involved with the system.

[1] See [5] for an extended version of this paper, including more details on the design of the explanation method.

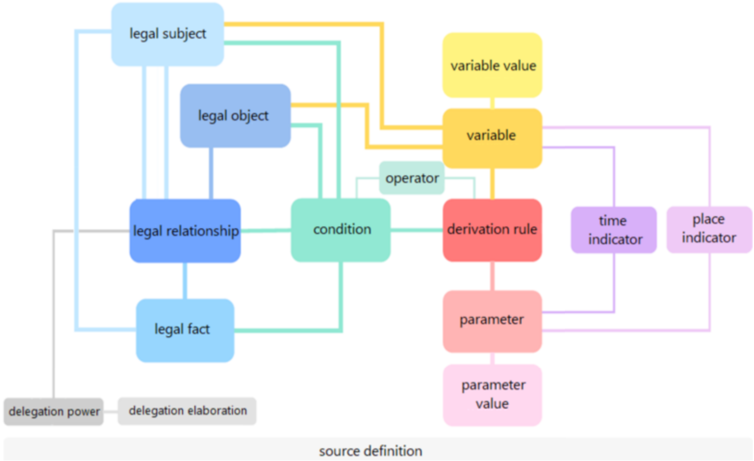

Figure 1. Diagram representing a framework for legal analysis, translated from [4].

1. Explanation Method

Graph Database Management Systems. We propose an explanation method using Graph Database Management Systems (GDBMS), specifically Neo4j [6]. These make it possible to store data in graph structures and allow for intuitive questioning of relational data and visualising relationships between data points.

In our explanation method, we utilise GDBMS to structure our explanations according to the legal analysis schema (Figure 1) and we combine it with question-driven user-system interaction and multimedia display to tailor the explanation to the receiver. More specific, in GDBMS, nodes can, for example, represent legal subjects, legal objects, requirements, rules and variables, while the connections represent their legal relationships. Recipients can question both the decision model and a specific decision (focus), filter information by, for instance, focusing on a specific part, node or relationship (level of detail) and extract causal relations between conditions, rules and derivations of these rules (causality), thereby varying the content of the explanation. Explanations are answers to predefined questions that can be communicated through visual graphs, verbal text answers and customised filtering options. Finally, GDBMS enables the creation of different perspectives for different recipient categories, thereby adapting the explanation to the recipient, by assuming a certain level of knowledge (both domain and reasoning knowledge) and linking a specified goal to a set of specific questions.

Implementation. We start by importing all the model elements in a structured form, creating an abstract syntax graph containing the knowledge from all the decision models. From this graph, we create a simplified graph representing the decision models' key structural and semantic information. To represent a specific decision, we can instantiate the simplified graph with the values of a model instance.[2]

[2]For the full implementation, see https://github.com/sjzuurmond/mps_explaineo.

In practice, a decision system uses multiple models representing its reasoning. ALEF uses three models: the Object Model, representing the entities and their attributes involved in decision-making; the Rule Model, representing rules and their dependencies for making decisions; and the Service Model, representing services used by legal professionals for decision-making by providing input and output variables.

2. Case Study

We concentrate on an application within the domain of tax law – the tax interest calculation system. This application plays a crucial role in determining interest rates for tax assessments in the Dutch tax landscape, where tax professionals rely on it to calculate rates with significant financial implications for taxpayers. Therefore, explanations are essential for all users involved; we consider two.

Model experts implement tax regulations into decision models. These experts are responsible for ensuring that the application accurately reflects the complex web of tax laws and regulations. When creating the model, our explanation method offers guidance in this modelling process by providing a visual representation of each model to aid the textual editor of ALEF. Such a visualisation contains the variables, objects, rules and input/output messages as nodes as well as the relationships between these nodes. What type of nodes and relations are included depends on the type of question. For example, when asking *What elements has the Object Model?* the nodes are the variables and objects, while when asking *What elements has the Rule Model?* there are nodes for rules as well.

For example, in order to understand the created system better, the model expert can ask *Can all output be created?*. If this is not the case, the system shows which variables cannot be derived (e.g., the start and end date of the interest period) in text, in a table it shows all the variables and whether there is a path from input to output of the service. The visualisation allows the expert to understand the relations and see how the variables could be assigned.

Legal support professionals bridge the gap between complex tax decisions and taxpayers seeking clarity. When communicating with taxpayers, they require a comprehensive understanding of the underlying decision-making process to provide concise and accurate explanations. The explanation method needs to answer a variety of questions for legal support professionals, we consider two types in our case study. A data explanation is about *what data was used in the explanation*, while a rationale explanation is about the *reasoning behind a decision that has been made.*

For example, when a taxpayer asks a legal support professional why they have to pay tax interest, the system shows the professional, in text, the applicable rules, including a link to the specific law (e.g., taxpayer paid taxes too late, link), and the conditions that are satisfied (e.g., the required payment specification was made at a specific date and at least one payment was overdue) as well as a visualisation of the conclusion, the rules and conditions and their relations, see Figure 2. The professional can then communicate this with the taxpayer.

Figure 2. Visualisation of the explanation for *why do I have to pay tax interest?*, with all the steps from input messages to decision.

3. Discussion and Conclusion

In this paper, we have demonstrated a human-centred explanation method that offers explanations for rule-based decision-making systems in the legal domain. Using graph databases we enhanced the explanatory value of the method, facilitating more effective communication and adaptation to certain individuals' needs. Since our explanation method is connected to the legal analysis diagram, explanations provide insight into the decision-making system and the legislation. We also implemented the proposed method in a real-world scenario: an automated decision-making system used by the DTCA.

While our primary focus has been on DTCA and ALEF, our explanation method holds significant value for a broader spectrum of automated decision-making systems. Moreover, integration of graph databases, combined with question-driven user-system interaction and multimedia display, exemplifies how our method harnesses both traditional principles of human explanation and emerging technologies to craft more effective and human-friendly explanations.

There are many possible directions for future work, from extending our explanation method to allow for, e.g., dynamics in the target group and other explanations forms, to improving the visualisation. Additionally, implementing the explanation method for the DTCA provides a good starting point for applying the method to other domains. We can then further explore the flexibility of the method and its ability to cater to the unique needs of different user groups.

References

[1] Open Govenment Partnership. The open government partnership's implementation plan 2020-2022; 2020. Available from: https://www.opengovpartnership.org/wp-content/uploads/2020/03/OGPs-Implementation-Plan-2020-2022-FINAL.pdf.
[2] JetBrains. MPS case study - Agile Law Execution Factory; 2019. Available from: https://resources.jetbrains.com/storage/products/mps/docs/MPS_DTO_Case_Study.pdf.
[3] Lokin M, van Kempen M. Van wet naar loket: bedrijfsregels en agile werken voor een transparante wetsuitvoering. RegelMaat. 2019;34(1):35–57.
[4] Lokin M, Ausems A, Bulles J. Wetsanalyse. Boom juridisch; 2021.
[5] Zuurmond S, Borg A, van Kempen M, Wieten R. Human-centred explanation of rule-based decision-making systems in the legal domain. CoRR. 2023;Available from: https://arxiv.org/abs/2310.16704.
[6] Lal M. Neo4j graph data modeling : design efficient and flexible databases by optimizing the power of Neo4j. Packt Publishing; 2015.