A Multifunctional, Interactive DMN Decision Modelling Tool

Simon Vandevelde* and Joost Vennekens

KU Leuven, De Nayer Campus, Dept. of Computer Science
Leuven.AI - KU Leuven Institute for AI, B-3000 Leuven, Belgium
{s.vandevelde, joost.vennekens}@kuleuven.be

Abstract. In this demo we showcase DMN-IDP, a user-friendly tool which combines the readability of the Decision Model and Notation (DMN) standard with the power of the IDP system through an interactive interface.

The Decision Model and Notation (DMN) standard is a table-based way of representing decision logic, with a focus on readability and user-friendliness. Designed by the Object Management Group, it was quickly adopted in various industries. In academia, interest in DMN to represent knowledge is also growing, because of its accessibility as a modelling language for domain experts [6]. To use DMN models, tools exist which can compute a suitable assignment of values to the decision variables, given the values of the environmental variables, by means of forward propagation.

In [4], it was argued that the knowledge expressed in a DMN model can be used for much more. For instance, value propagation can also be done in other directions, such as from decision to environmental variables. Other examples are reasoning on incomplete data, and applying different inference tasks, such as optimization. To illustrate their approach, the authors made use of the IDP knowledge base system [5]. By manually translating DMN models into first-order logic knowledge bases (KBs), users could interact with the KB in a user-friendly way via a browser-based interface. While this results in a powerful and flexible way of working, there are two main downsides. Firstly, the DMN models need to be created in a separate tool. Secondly, the translation from DMN to IDP KB is done manually, for which knowledge of the representation language of the IDP system is required.

In this demo, we present DMN-IDP, a full-fledged DMN tool which combines the dmn-js DMN editor [2] and the IDP-based Interactive Consultant interface [3]. Using this tool, a user can upload or create DMN models, which are then translated into IDP KBs. Users can interact with these models via the Interactive Consultant interface. The translation from DMN to IDP is done by the same transformation used in the cDMN framework [1]. The interface supports propagating values in any direction, reasoning on incomplete data, optimization of values and explanation of decisions. In this way, DMN models become useable in more situations, removing the need to build specific models for every target output in a use case.

* This research received funding from the Flemish Government under the “Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen” programme.

Calculate Body Mass Index			
U	**Weight(kg)**	**Height(m)**	**BMI**
1	—	—	weight / (height * height)

Decide BMI Level			
U	**BMI**	**BMI Level**	**Risk Level**
1	< 18.5	Underweight	Increased
2	[18.5..24.8]	Normal	Low
3	[25..29.9]	Overweight	Increased
4	[30..34.9]	Obese I	High
5	[35..39.9]	Obese II	Very High
6	> 39.9	Extreme Obesity	Extremely High

Fig. 1: A DMN model for deciding a patient's BMI level and risk level.

As an example, consider the DMN model in Figure 1, which calculates a BMI level and risk level based on a patient's weight and height. The table on the left consists of one rule, which is read as "For every possible weight and height, the BMI is weight divided by height squared." The table on the right then decides what the value for *BMI Level* and *Risk Level* is, based on the BMI. Using this model, standard DMN tools could for example calculate that for a weight of 100kg and height of 1.8m, the BMI is 30.9, resulting in a high risk level. However, say we now want to know the opposite, i.e., what weight would give a 1.8m patient a low risk level. Standard DMN tools cannot infer this information from this model. Our tool on the other hand is capable of reasoning backwards, even with incomplete data. This allows us to enter the height and set the value of *Risk Level* to "Low" while leaving the weight unknown. By now maximizing the *Weight* variable, we find that a weight less than 80.7kg results in a low risk for the height.

The tool also includes some basic functionality for detecting common errors in DMN specifications. We plan to develop this further in future work, along with a functionality to improve the traceability of decisions.

During the demo, participants will get to interact with our tool via multiple use cases, allowing them to explore the capabilities of the system freely. They will also be encouraged to experiment with the DMN models themselves, so that they can learn the connections between the components. An online version of the tool is available at https://autoconfig-dmn.herokuapp.com/.

References

1. Aerts, B., Vandevelde, S., Vennekens, J.: Tackling the dmn challenges with cdmn: A tight integration of dmn and constraint reasoning. Springer (2020)
2. bpmn.io: Dmn viewer and editor. https://bpmn.io/toolkit/dmn-js/ (2015)
3. Carbonnelle, P., Aerts, B., Deryck, M., Vennekens, J., Denecker, M.: An interactive consultant. In: Proceedings of the 31st Benelux Conference on Artificial Intelligence (BNAIC 2019) and the 28th Belgian Dutch Conference on Machine Learning (Benelearn 2019), Brussels, Belgium, November 6-8, 2019 (2019)
4. Dasseville, I., Janssens, L., Janssens, G., Vanthienen, J., Denecker, M.: Combining dmn and the knowledge base paradigm for flexible decision enactment (2016)
5. De Cat, B., Bogaerts, B., Bruynooghe, M., Janssens, G., Denecker, M.: Predicate logic as a modeling language: The idp system. pp. 279–329. ACM Books (2018)
6. Deryck, M., Hasic, F., Vanthienen, J., Vennekens, J.: A case based inquiry into the decision model and notation (dmn) and the knowledge base (kb) paradigm. In: Proceedings of RuleML+RR 2018. vol. 11092 LNCS, pp. 248–263. Springer (2018)