ESC-Rules: Explainable, Semantically Constrained Rule Sets

Martin Glauer[1,*], Robert West[2], Susan Michie[3] and Janna Hastings[1,3,*]

[1] *Institute for Intelligent Interacting Systems, Otto-von-Guericke University Magdeburg, Germany*
[2] *Department of Behavioural Science and Health, University College London, UK*
[3] *Department of Clinical, Educational and Health Psychology, University College London, UK*

Abstract

We describe a novel approach to explainable prediction of a continuous variable based on learning fuzzy weighted rules. Our model trains a set of weighted rules to maximise prediction accuracy and minimise an ontology-based 'semantic loss' function including user-specified constraints on the rules that should be learned in order to maximise the explainability of the resulting rule set from a user perspective. This system fuses quantitative sub-symbolic learning with symbolic learning and constraints based on domain knowledge. We illustrate our system on a case study in predicting the outcomes of behavioural interventions for smoking cessation, and show that it outperforms other interpretable approaches, achieving performance close to that of a deep learning model, while offering transparent explainability that is an essential requirement for decision-makers in the health domain.

Keywords

explainable machine learning, rules learning, semantic loss, evidence synthesis, behaviour change

1. Introduction

The rate evidence is generated outstrips the rate at which it can be synthesised, necessitating automated approaches [1]. The Human Behaviour-Change Project (HBCP) is an interdisciplinary collaboration between behavioural scientists, computer scientists and information systems architects that aims to build an end-to-end automated system for evidence synthesis in behavioural science [2]. For this objective, explanations of the predictions are as important as the accuracy of the system, since the intended users are practitioners and policy-makers who will use the insights gained from the evidence in order to make recommendations thus require suitable transparency and accountability [3].

Deep neural networks typically operate as black boxes without giving intrinsic insights into why specific predictions have been made.[1] Thus, there is a need for "glass-box" explainable machine learning frameworks for making predictions and recommendations that can transpar-

NeSy 22: 16th International Workshop on Neural-Symbolic Learning and Reasoning, 28-30 September, London, UK

[*] Corresponding author.
✉ martin.glauer@ovgu.de (M. Glauer); j.hastings@ucl.ac.uk (J. Hastings)
🌐 https://jannahastings.github.io/ (J. Hastings)
iD 0000-0001-6772-1943 (M. Glauer); 0000-0001-6398-0921 (R. West); 0000-0003-0063-6378 (S. Michie); 0000-0002-3469-4923 (J. Hastings)

CEUR Workshop Proceedings (CEUR-WS.org)

[1] Although, methods to give explanations for such networks are advancing [4, 5].

ently provide complete explanations in a form that matches the semantic expectations of the users [6, 7].

We aimed to develop an explainable system for the prediction of behaviour change intervention outcomes, based on a corpus of annotated literature together with features from an ontology - the Behaviour Change Intervention ontology [8] - and their logical relationships. Straightforward application of semantic approaches is not well suited for the quantitative task of predicting intervention outcomes, and moreover traditional symbolic learning approaches such as rules or decision trees lead to explanations that are overly complex and not ranked by their quantitative impact on the outcome variable. A deep neural network approach had better quantitative performance, but was not acceptable for our users due to the lack of transparency and explainability. Thus, we aimed to develop a 'best of both worlds' hybrid predictive approach that combined aspects of the symbolic and neural approaches.

Rules-based systems are inherently explainable because the features appear transparently in the rules. Our approach builds on systems that are able to learn rules from data. One of the earliest learning rule neural network systems was the Knowledge Based Artificial Neural Network - KBANN [9]. This approach translates domain knowledge into rules which are encoded into the structure of a neural network, for which weights are then learned. The approach that we developed has furthermore been inspired by traditional 'neuro-fuzzy' systems. It is, in particular, based on the principles of Tagaki-Sugeno controllers [10]. These controllers have been developed to account for the fact that expert knowledge, albeit valuable, is often too vague to be turned into rigid logical rules, thus necessitating fuzzy and weighted approaches to rules learning.

2. Explainable Semantically Constrained Rule Sets

We describe our approach in terms of feature preparation, architecture and optimisation, and semantic penalties. The system is implemented in Python using PyTorch. Source code is available at `https://github.com/HumanBehaviourChangeProject/semantic-prediction`.

2.1. Feature preparation

A precondition for our rules-based approach is that all input data features are binarized. Thus, categorical variables in the dataset are exploded into separate columns per value, and continuous (quantitative) variables are binarized by selecting ranges using one of several different approaches depending on the meanings of the values:

- Separation into meaningful semantic categories, e.g. for our case study we transform mean age values into child, young adult, older adult, and elderly, delineated with a fuzzy membership operator since the boundaries between categories are not rigid.
- Fixed-width categories, delineated with a fuzzy membership operator. For example, we divide number of times tobacco smoked into groups of width 5 corresponding to <5, <10, <15, ... <50. Note that this formulation creates an ordering because if the value is e.g. 6, then all of <10, <15, ..., <50 will be set on, and if the value is 46, only <50 will be set on.

- Categories selected based on quantiles in the dataset (i.e. a fixed proportion of the available data in each grouping, rather than fixed range of values), again using the <x formulation to maintain ordering.
- Fixed numeric values, for flagging exact values with a specific meaning, for example 100% female within the population percentage female continuous variable.
- Data-driven clustering can be used to determine clusters of data associated with specific value ranges.

2.2. Architecture and Optimisation

The general design of the architecture was inspired by the construction of Takagi-Sugeno controllers. But, while that approach assumes that a set of rules is given as a prior, the goal of our system is to automatically derive the rules from the dataset. We assume that rules are represented as conjunctive formulae of all features in the dataset and their negations. A weight is then attached to each rule and each feature in the rule as well as their negated counterparts:

$$(w_{i,1}, A_1) \land \cdots \land (w_{i,n}, A_n) \land (w_{i,n+1}, \neg A_1) \land \cdots \land (w_{i,2n}, \neg A_n) \Rightarrow r_i \tag{1}$$

A weight $w_{i,j} \in \mathbb{R}$ is attached each literal A_j. The sigmoid $\sigma(w_{i,j})$ of these weights denotes the degree to which literal A_j impacts the i-th rule. Each rule is also attached with a rule weight $r_i \in \mathbb{R}$ that denotes the impact of this rule on the prediction. For the sake of readability, we will further refer to the literals $A_1, ..., A_n, \neg A_1, ..., \neg A_n$ (i.e. features and negated features) as just $B_1, ..., B_{2n}$. The basic idea of our approach is to hide the parts of the conjunction that are not relevant to a particular rule. That is, variables with a small weight should not influence the rule as much as ones with larger weights. Given a fuzzy evaluation function that assigns a fuzzy membership value to each feature $\mu : \{B_1, ..., B_{2n}\} \to [0, 1]$, we therefore calculate the weighted conjunctive contribution of a literal B_j in rule i as a linear combination between $\mu(B_j)$ and 1. This allows us to finally define the fit of a rule as $\text{fit}_i = \max_{j=1}^{2n}(\sigma(w_{i,j})\mu(B_j) + (1 - \sigma(w_{i,j})))$. Similar to the Takagi-Sugeno controller approach, this fit is an indicator of how much the right-hand side of the rule should impact the final prediction of the system. Our system uses this fit as a simple scaling factor $y_{\text{pred}} = \sum_{i=1}^{m} \text{fit}_i \cdot r_i$ to aggregate predictions from multiple rules.

2.3. Semantic penalties

While the system described so far does learn rules, those rules are not meaningfully explainable. This is mainly due to two effects: A) The sigmoids of the weights often take arbitrary values from [0,1]. B) The left-hand sides of the rules are often very long, which makes it difficult to analyse the system easily. To address these points, we introduce additional regularisation terms that penalise such behaviour. The first one $\lambda_{\text{long}} = \sum_{i=1}^{m} \max\left(\sum_{j=1}^{n} \sigma(w_{i,j}), \theta\right)$ penalises rules with length exceeding the threshold θ. The second term $\lambda_{\text{fuzzy}} = \sum_{i=1}^{m} \sum_{j=1}^{n} (1 - \sigma(w_{i,j}))\sigma(w_{i,j})$ uses a quadratic equation for non-crisp literals. This penalty is only non-zero for those feature weights that are non-crisp.

These penalties improve the learned rules. However, the rules learned are still somewhat arbitrary and lack semantic coherence. Thus, as a final measure we add semantic ontology-based penalties that aim to increase the coherence and sense of the rules that are learned using the background domain knowledge as embodied in an ontology. For our use case we derive the rules from the Behaviour Change Intervention Ontology [8], using several semantic features.

Implied features The ontology contains a class hierarchy, which is used in two different ways. First, to pre-complete the data table: if a lower level feature is on, we ensure that the higher level feature that subsumes it is also set on in the input dataset. Hierarchical implications and other implications deriving from other types of relationships between entities in the ontology are also used to reduce rule length and enhance rule interpretability. Intuitively, a rule that has two features, such as 'Somatic$(X) \wedge$ PharmacologicalSupport$(X) \Rightarrow r$', where one feature implies the other, can be reduced to a rule with one feature, e.g. 'PharmacologicalSupport$(X) \Rightarrow r$', because of the implication given by the ontological relationship between the entities, as every intervention with pharmacological support is also one with a somatic delivery mode. This results in shorter, more readable rules. Therefore, we introduce an additional penalty $\lambda_{\text{implied}} = \sum_{i=1}^{m} \sum_{(j,j') \in \mathcal{I}} \min(\sigma(w_{i,j}), \sigma(w_{i,j'}))$ for the co-occurence of implied features in each rule.

Mutually exclusive features The ontology also contains axioms regarding negative dependencies between features. For example, an intervention that has the feature 'buproprion' (a pharmaceutical that is administered in form of a pill) cannot have the feature 'not pill'. These dependencies can, for instance, be expressed as disjointness axioms. Rules that include features that contradict each other semantically are penalised using the same mechanism that has been employed for the implications. This penalty $\lambda_{\text{exclusive}}$ is also applied to logically contradictory features within rules, e.g. $A \wedge \neg A \rightarrow r$.

These penalties ensure the logical soundness and usability of the final rules. Of course, it would be possible to use purely symbolic techniques to achieve similar behaviour. Using penalties does however have the added benefit that the neural network can learn which parts of a conflicting rule can be dropped with minimal impact on predictive performance, and thereby find a better fit for the data.

Additional penalties increase the complexity of the training task considerably, as a multitude of different goals have to be optimised. We therefore use an additional scaling factor α that fades these penalties in as the training progresses to later epochs. The final loss function is therefore calculated using the binary cross entropy (bce):

$$\text{loss}(y_{\text{pred}}, y_{\text{target}}) = \text{bce}(y_{\text{pred}}, y_{\text{target}}) + \alpha\left(\lambda_{\text{long}} + \lambda_{\text{fuzzy}} + \lambda_{\text{impl}} + \lambda_{\text{exclusive}}\right)$$

The resulting system learns human-readable rules from data. These rules are constrained to be simple and semantically meaningful (and explainable) through the semantic regularisation terms.

3. Evaluation

3.1. Dataset

The dataset consists of features extracted from randomised controlled trials of smoking cessation interventions by manual annotation as described in [11]. Features - annotated using ontology

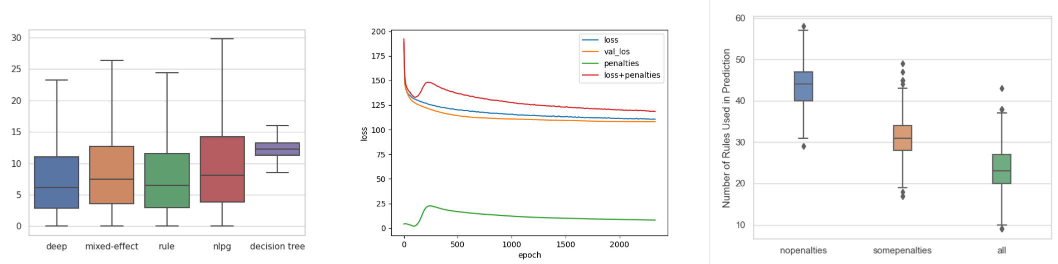

Figure 1: Comparisons of the results of different models. Left: Box plot showing the distribution and mean of the absolute errors in each model. Centre: Training/validation loss and penalties. Penalties are faded it at the beginning of the training. Right: Numbers of rules used in predictions with no, some or all penalties.

classes - cover several different aspects of intervention trial reports, including population attributes (e.g. mean age), delivery attributes such as who delivered it (e.g. nurse) and how it was delivered (e.g. face to face), and types of intervention (e.g. pharmacological support or goal setting). In the corpus, these features are associated with the text specifying the presence of the feature and any associated value, as well as the wider surrounding contextual text, and the whole composite is encoded in JSON. From the JSON, additional processing is used to extract tabular data with a column for each feature and a row for each intervention arm, and some additional data cleaning ensures that numeric values are appropriately parsed and that units are regularised. The resulting table has 77 feature columns and 1198 rows corresponding to intervention arms.

3.2. Model and Results

We evaluated our system against four other approaches. Firstly, we compare to the model that was previously developed within the HBCP [12], which we call 'NLPG' (short for NLP+Graph, as it learns from a combination of the textual feature contexts and the annotations encoded as a graph). This model was the previous best approach for this problem and has been applied as-is as it works directly from the data encoded in the JSON corpus. The remainder of the approaches work from the tabular feature dataset described above, which includes additional data cleaning steps such as ensuring consistent units for numeric features. Thus, for additional points of comparison, we also applied three different models from different families of machine learning approach.

decision trees A random forest of 100 trees with a maximal depth of 4 layers and 7 leaf nodes.

deep A feed-forward neural network with 77 input neurons and 3 hidden layers with 154, 77 and 38 neurons, trained for 100 epochs using the Adam optimizer.

mixed-linear As baseline we use a mixed-effect linear regression model with a random effect for study and fixed effects for all other variables.

Our model was initialised with a set of 100 rules. Singleton rules for each (non-negated) feature were initialised based on the weights from a simple linear regression, and the remaining

23 were initialised randomly using two uniform distributions: $\mathcal{U}(-1, 1)$ for feature weights and $\mathcal{U}(-10, 10)$ for rule weights. The resulting model was trained for at least 300 epochs, after which the training was stopped as soon as the loss (including penalties) did not further improve on the validation set. The final evaluation was then conducted on a test set that had not been seen before.

Figure 1 shows the performances of each model in terms of the absolute errors of the predictions on the unseen test set data. Our model outperforms the original approach for prediction, as well as decision trees and the mixed linear model. It is still slightly outperformed by the pure deep neural network, however, that approach is not explainable. Additional evaluations have shown that training on a smaller subset of the dataset leads to less robust results.

The final state of our system can be used to extract human-readable rules that apply for each prediction. Some examples of rules that are used in predictions are illustrated in Listing 1.

```
"Pharmaceutical company competing interest"[1.0]
  => raise predicted outcome by 0.2623 (fit: 0.9999)

"1.2 Problem solving"[1.0]
  => raise predicted outcome by 3.2550 (fit: 0.9999)

"Mean age (adult)"[1.0] & "doctor"[0.46218] & "Pill"[0.3893]
  => raise predicted outcome by 4.6709 (fit: 0.5382)

"doctor"[0.4442] & "Biochemical verification"[1.0]
  => lower predicted outcome by 2.7093 (fit: 0.5563)

"aggregate patient role"[0.6734] & "Abstinence: Continuous"[1.0]
  & not "1.4 Action planning"[1.0] & not "Somatic"[0.7504]
  & not "Proportion identifying as female gender" (<= 35)[0.7507]
  => lower predicted outcome by 2.9024 (fit: 0.3272)
```

Listing 1: Excerpt of a learned rule base that has been applied to a set of input values. Feature names are in quotations, followed by their attached weight. The denoted increases and decreases are already scaled according to the fit of the corresponding rule.

As can be seen, the rules that are learned vary in length from simple weights for individual features, to complex interactions between features, and vary in their impact on the outcome value, which may be positive or negative.

The rules and their fits are completely transparent, allowing experts to inspect the functioning of the system and add additional semantic constraints where needed to improve the overall performance of the system, leading to an iterative process which may also have the benefit of improving the associated ontology from which the semantic constraints are drawn.

4. Related Work

Our work is related to several ongoing research areas within neuro-symbolic computing, which we discuss in turn.

4.1. Learning fuzzy logic operations in deep neural networks

Our approach has aspects of a fuzzy inference system, which are less general than neural networks but more explainable. Most fuzzy neural systems are restricted to just a few logical operations, usually `and` and `or`. They present an activation function that can learn additional logical operations and allow learning of complex logical expressions with accuracy comparable to a standard deep neural network with `tanh` activation functions. A deep learning architecture has been proposed for learning fuzzy logic expressions in [13]. More generally, Logic Tensor Networks [14] are able to represent arbitrary first-order formulae and learn the semantic interpretation of constants, predicate and function symbols during training by minimising a specific semantic loss function. The semantics for logical connectives are based on existing fuzzy semantics - usually based on Łukasiewicz logic. Another notable approach to rule learning are differentiable logic machines[15]. These networks allow the expression and learning of logical formulae and use a hiding mechanism similar to the one presented in this work. They are however more suitable for purely symbolic applications and less for the regression task that was our use-case.

4.2. Logically constraining deep neural networks

There is a significant body of work on constraining neural networks with decision rules (reviewed in [16]) or with logical constraints (reviewed in [17]). A recent addition to this family of approaches is Deep Neural Networks with Controllable Rule Representations (DeepCTRL) [18], an approach that incorporates a rule encoder into a deep neural network model together with a rule-based objective. It allows specification of rules for inputs and outputs, with rule 'strength' adjustable at inference time via a parameter (not requiring retraining). The rules constrain the model search space to reduce under-specification and improve generalisability, which is similar to the effect of the semantic ontology-based constraints in our approach. However, their rules are not based on an ontology thus must be manually specified.

4.3. Learning weighted rule sets

The RuleFit algorithm [19] learns sparse linear models that include automatically detected interaction effects in the form of rules. Similarly to our approach, in this approach base variables take values in (0,1) with categorical variables one-hot encoded and numerical variables discretised into ranges. However, this approach uses pre-specified rule lengths rather than optimising the rule length together with the rule weights as we do. In RuleFit, the sizes of the rules are governed by a random variable selected from an exponential distribution so that smaller rules are favoured. Thus, most of the rules will be simple capturing main effects while some will be larger capturing interaction effects. Moreover, it does not incorporate semantic constraints on the rules as our system does.

A two-layer neural network for learning rules for binary classification rather than regression, Decision Rules-net (DR-net) [20] includes a first layer consisting of a set of neurons that each map to decision rules, and a second layer that performs a disjunctive combination of the rules from the first layer. Input features are binarized in a similar fashion to our system, and then the first layer operates as a conjunction over features in the input using a binary step activation function. This operation is not differentiable, thus it is approximated with the straight-through estimator using gradient clipping. Learnable weights are associated with each decision rule in the first layer of the network, with positive weights corresponding to a positive association of the input feature, negative weights with a negative association, and a zero weight corresponding to exclusion of the input feature. Stochastic gradient descent is used for training. Sparsity-based regularisation implements a tradeoff between accuracy of predictions and simplicity of explanations in terms of the length of the rules. However, they include no additional semantic penalties.

5. Conclusions

We presented a system for rule learning on sparse data for a regression problem, and evaluated the performance on a use-case from the domain of behaviour change interventions. Our system out-performed other explainable methods, and was not far from the predictive power of a black-box deep neural network. Our system is able to generate human-readable rules that can be used to easily explain predictions. Moreover, it is possible to enhance the system through semantic constraints that can either be defined by experts or extracted from an ontology. This ensures that the system is able to generate rules that not only fit the dataset but also reflect domain knowledge, are semantically correct, explainable, and do not contain unnecessarily complex (and likely overfitted) conditions.

In the future, we aim to conduct a more in-depth evaluation study with users and domain experts, in particular with respect to the explainability of the rules. One particular aspect which we would like to explore more deeply with the domain experts is the explainability associated with the negated features, i.e. the explanatory value of *absences* of features, where these appear in rules. We will also conduct a more in-depth comparison to newer neuro-symboli approaches. Another aspect we plan to explore is whether there is a distinction or preference among domain experts for rules that contain interaction terms from different semantic sub-domains (e.g. intervention, population, setting) or those that show interactions within a semantic domain (e.g. different types of intervention in combination).

We also plan to apply the system on different datasets drawn from a wider range of behavioural domains, including physical activity.

Acknowledgments

Thanks to all members of the Human Behaviour-Change Project team and in particular to Alison Wright and James Thomas. The HBCP is funded by a Wellcome Trust collaborative award led by Susan Michie.

References

[1] J. Elliott, R. Lawrence, J. C. Minx, O. T. Oladapo, P. Ravaud, B. Tendal Jeppesen, J. Thomas, T. Turner, P. O. Vandvik, J. M. Grimshaw, Decision makers need constantly updated evidence synthesis, Nature 600 (2021) 383–385. URL: https://www.nature.com/articles/d41586-021-03690-1. doi:10.1038/d41586-021-03690-1, bandiera_abtest: a Cg_type: Comment Number: 7889 Publisher: Nature Publishing Group Subject_term: Research management.

[2] S. Michie, J. Thomas, P. Mac Aonghusa, R. West, M. Johnston, M. P. Kelly, J. Shawe-Taylor, J. Hastings, F. Bonin, A. O'Mara-Eves, The Human Behaviour-Change Project: An artificial intelligence system to answer questions about changing behaviour, Wellcome Open Research 5 (2020) 122. URL: https://wellcomeopenresearch.org/articles/5-122/v1. doi:10.12688/wellcomeopenres.15900.1.

[3] C. J. Kelly, A. Karthikesalingam, M. Suleyman, G. Corrado, D. King, Key challenges for delivering clinical impact with artificial intelligence, BMC medicine 17 (2019) 195. doi:10.1186/s12916-019-1426-2.

[4] M. Moradi, M. Samwald, Post-hoc explanation of black-box classifiers using confident itemsets, Expert Systems with Applications 165 (2021) 113941. URL: https://www.sciencedirect.com/science/article/pii/S0957417420307302. doi:10.1016/j.eswa.2020.113941.

[5] A. Madsen, S. Reddy, S. Chandar, Post-hoc Interpretability for Neural NLP: A Survey, arXiv:2108.04840 [cs] (2021). URL: http://arxiv.org/abs/2108.04840, arXiv: 2108.04840.

[6] A. Holzinger, M. Dehmer, F. Emmert-Streib, R. Cucchiara, I. Augenstein, J. D. Ser, W. Samek, I. Jurisica, N. Díaz-Rodríguez, Information fusion as an integrative cross-cutting enabler to achieve robust, explainable, and trustworthy medical artificial intelligence, Information Fusion 79 (2022) 263–278. URL: https://www.sciencedirect.com/science/article/pii/S1566253521002050. doi:10.1016/j.inffus.2021.10.007.

[7] A. Holzinger, R. Goebel, R. Fong, T. Moon, K.-R. Müller, W. Samek, xxAI - Beyond Explainable Artificial Intelligence, in: A. Holzinger, R. Goebel, R. Fong, T. Moon, K.-R. Müller, W. Samek (Eds.), xxAI - Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020, July 18, 2020, Vienna, Austria, Revised and Extended Papers, Lecture Notes in Computer Science, Springer International Publishing, Cham, 2022, pp. 3–10. URL: https://doi.org/10.1007/978-3-031-04083-2_1. doi:10.1007/978-3-031-04083-2_1.

[8] S. Michie, R. West, A. N. Finnerty, E. Norris, A. J. Wright, M. M. Marques, M. Johnston, M. P. Kelly, J. Thomas, J. Hastings, Representation of behaviour change interventions and their evaluation: Development of the Upper Level of the Behaviour Change Intervention Ontology, Wellcome Open Research 5 (2020) 123. URL: https://wellcomeopenresearch.org/articles/5-123/v1. doi:10.12688/wellcomeopenres.15902.1.

[9] G. G. Towell, J. W. Shavlik, Knowledge-based artificial neural networks, Artificial Intelligence 70 (1994) 119–165. URL: https://www.sciencedirect.com/science/article/pii/0004370294901058. doi:10.1016/0004-3702(94)90105-8.

[10] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, IEEE transactions on systems, man, and cybernetics (1985) 116–132.

[11] F. Bonin, M. Gleize, A. Finnerty, C. Moore, C. Jochim, E. Norris, Y. Hou, A. J. Wright, D. Ganguly, E. Hayes, S. Zink, A. Pascale, P. Mac Aonghusa, S. Michie, HBCP Corpus: A New

Resource for the Analysis of Behavioural Change Intervention Reports, in: Proceedings of the 12th Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France, 2020, pp. 1967–1975. URL: https://aclanthology.org/2020.lrec-1.242.

[12] D. Ganguly, M. Gleize, Y. Hou, C. Jochim, F. Bonin, A. Pascale, P. Tommasi, P. Mac Aonghusa, R. West, M. Johnston, et al., Outcome prediction from behaviour change intervention evaluations using a combination of node and word embedding, in: AMIA Annual Symposium Proceedings, volume 2021, American Medical Informatics Association, 2021, p. 486.

[13] L. B. Godfrey, M. S. Gashler, A parameterized activation function for learning fuzzy logic operations in deep neural networks, arXiv:1708.08557 [cs] (2017). URL: http://arxiv.org/abs/1708.08557, arXiv: 1708.08557.

[14] L. Serafini, A. d. Garcez, Logic tensor networks: Deep learning and logical reasoning from data and knowledge, arXiv preprint arXiv:1606.04422 (2016).

[15] M. Zimmer, X. Feng, C. Glanois, Z. Jiang, J. Zhang, P. Weng, L. Dong, H. Jianye, L. Wulong, Differentiable logic machines, 2021. URL: https://arxiv.org/abs/2102.11529. doi:10.48550/ARXIV.2102.11529.

[16] Y. Okajima, K. Sadamasa, Deep Neural Networks Constrained by Decision Rules, Proceedings of the AAAI Conference on Artificial Intelligence 33 (2019) 2496–2505. URL: https://ojs.aaai.org/index.php/AAAI/article/view/4095. doi:10.1609/aaai.v33i01.33012496, number: 01.

[17] E. Giunchiglia, M. C. Stoian, T. Lukasiewicz, Deep Learning with Logical Constraints, arXiv:2205.00523 [cs] (2022). URL: http://arxiv.org/abs/2205.00523, arXiv: 2205.00523.

[18] S. Seo, S. O. Arik, J. Yoon, X. Zhang, K. Sohn, T. Pfister, Controlling Neural Networks with Rule Representations, 2021. URL: https://openreview.net/forum?id=owQmPJ9q9u.

[19] J. H. Friedman, B. E. Popescu, Predictive learning via rule ensembles, The Annals of Applied Statistics 2 (2008). URL: http://arxiv.org/abs/0811.1679. doi:10.1214/07-AOAS148, arXiv: 0811.1679.

[20] L. Qiao, W. Wang, B. Lin, Learning Accurate and Interpretable Decision Rule Sets from Neural Networks, arXiv:2103.02826 [cs] (2021). URL: http://arxiv.org/abs/2103.02826, arXiv: 2103.02826.