

Studying Trustworthiness of Neural-Symbolic Models for Enterprise Model Classification via Post-Hoc Explanation

Alexander Smirnov^a, Anton Agafonov^b and Nikolay Shilov^c
SPC RAS, 14th Line, 39, St. Petersburg, Russia

Keywords: Neuro-Symbolic Artificial Intelligence, Deep Neural Networks, Machine Learning, Concept Extraction, Post-Hoc Explanation, Trust Assessment, Enterprise Model Classification.

Abstract: Neural network-based enterprise modelling support is becoming popular. However, in practical enterprise modelling scenarios, the quantity of accessible data proves inadequate for efficient training of deep neural networks. A strategy to solve this problem can involve integrating symbolic knowledge to neural networks. In previous publications, it was shown that this strategy is useful, but the trust issue was not considered. The paper is aimed to analyse if the trained neural-symbolic models just “learn” the samples better or rely on the meaningful indicators for enterprise model classification. The post-hoc explanation (specifically, the concept extraction) has been used as the studying technique. The experimental results showed that embedding symbolic knowledge does not only improve the learning capabilities but also increases the trustworthiness of the trained machine learning models for enterprise model classification.


1 INTRODUCTION


Recently, the application areas of machine learning methods based on artificial neural networks (ANN) have significantly extended. Nevertheless, the efficient application of ANNs is still highly dependent on training data that is required in significant volumes. Thereby absence of large volumes of training data is still a significant constraining factor for their application in a number of areas (Anaby-Tavor et al., 2020; Nguyen et al., 2022). On the other side, once defined symbolic knowledge can be tailored to new problems without the necessity of training on extensive datasets. Consequently, the fusion of symbolic knowledge and ANNs (sub-symbolic knowledge) can be considered as a promising research direction. The result of such a fusion is referred to as neural-symbolic artificial intelligence (Garcez & Lamb, 2020).


One of the areas that can benefit from sub-symbolic and symbolic knowledge fusion is enterprise modelling assistance. Application of machine learning techniques to enterprise modelling assistance has been addressed recently (Shilov et al.,

2021, 2023) demonstrating the potential efficiency of the enterprise modeller assistance based on the ANN paradigm. The assistance can include both suggestion and verification of node and relationship types and labels implementing such functions as auto-completion and error corrections. It was also shown that efficient assistance can only be achieved if the ANN-based models take into account the modelling context, e.g., class of the model (such as concept model, process model, etc.), its target users (engineers or top managers), and others.

In the previous publication (Smirnov et al., 2023) the authors analysed the application of the symbolic artificial intelligence to the enterprise model classification problem. It was demonstrated that its usage indeed improved the ANN-based model trained on a limited dataset. However, the publication did not consider the trust issue. It was not researched if the trained models just “learned” the samples or relied on the meaningful model class indicators. The research question to be answered in this work is “If the neural-symbolic machine learning model is more trustworthy than the pure ANN model?”. For this purpose, an approach from the area of explanation of

^a  <https://orcid.org/0000-0001-8364-073X>

^b  <https://orcid.org/0000-0002-7960-8929>

^c  <https://orcid.org/0000-0002-9264-9127>

trained ANNs (namely, post-hoc ANN explanation) has been used to understand if the ANNs rely on meaningful enterprise model class indicators.

The paper is structured as follows. Section 2 describes the state-of-the-art in the areas of symbolic and sub-symbolic knowledge integration and post-hoc ANN explainability. It is followed by the presentation of the data used and the research methodology. Section 4 presents the experimentation results and their discussion. The concluding remarks are given in Section 5.

2 STATE OF THE ART REVIEW

The section briefly considers approaches and architectures used for integration of symbolic and neural knowledge as well as techniques of post-hoc ANN explanations.

2.1 Approaches for Embedding Symbolic Knowledge into ANNs

Embedding of symbolic knowledge into ANNs can be achieved using various techniques. The paper (Ultsch, 1994) defined four distinct approaches: *neural approximative reasoning*, *neural unification*, *introspection*, and *integrated knowledge acquisition*. The approaches address different tasks and their choice is normally defined by the task being solved.

- The *neural approximative reasoning* combines methods in the area of approximate inference generation in intelligent systems (Guest & Martin, 2023) mostly aimed at building ANNs approximating existing rules.
- The *integrated knowledge acquisition* aims to extract knowledge from a limited set of examples (usually generated by an expert) and then to re-formulate discovered patterns into rules (Mishra & Samuel, 2021).
- The *neural unification* aims training ANNs to learn logical statement sequences leading to the original statement confirmation or refutation for generalizing argument selection strategies when proving assertions (Picco et al., 2021).
- The *introspection* assumes ANNs to monitor steps performed during logical inference thus learning to avoid erroneous pathways and to come to reasoning results faster (Prabhushankar & AlRegib, 2022).

Thus, the most appropriate approaches for embedding symbolic knowledge into ANN-based

classifier are the *neural approximative reasoning* and *integrated knowledge acquisition*.

2.2 Symbolic and Neural Knowledge Integration Architectures

The symbolic and neural knowledge integration architectures are classified based on the “location” of symbolic rules in an ANN (Wermter & Sun, 2000):

- The *Unified architecture* suggests to encode symbolic knowledge within the neural network. In this case, two ways are possible: (i) encoding symbolic knowledge in separate ANN fragments (Arabshahi et al., 2018; Pitz & Shavlik, 1995; Xie et al., 2019), or (ii) by network’s non-overlapping fragments (Hu et al., 2016; Prem et al., n.d.).
- The *Transformation architecture* assumes mechanisms translating neural knowledge into symbolic and/or back, e.g., extraction of rules from an ANN (Shavlik, 1994).
- *Hybrid modular architecture* suggests to encode symbolic knowledge into modules that are separate from ANN. This can be done in three ways: (i) *Loosely coupled architecture*: one-way interoperability (Dash et al., 2021; Li et al., 2022); (ii) *Tightly coupled architecture*: two-way interoperability (Xu et al., 2018; Yang et al., 2020); and (iii) *Fully integrated architecture*: two-way interoperability via several interfaces (Lai et al., 2020).

In contrast to the approaches (sec. 2.1), the architectures are not tailored to specific use cases but should be selected based on the unique problem under consideration. It can be noticed that when symbolic knowledge is stored within dynamic modules such as, for example, evolving ontologies, a *hybrid modular architecture* is preferable (the symbolic knowledge can be updated without affecting the neural knowledge). Conversely, when dealing with static knowledge, *unified* and *transformational architectures* might seem to be appealing due to their adaptability and the amount of techniques available. As a result, in (Smirnov et al., 2023) the *loosely coupled hybrid modular architecture* was selected to maintain the autonomy of symbolic knowledge with provisions for its extension and update.

2.3 Post-Hoc Approaches to the Explainability of ANNs

Post-hoc techniques (Confalonieri et al., 2019, 2020, 2021; Panigutti et al., 2020) are designed to explain

pre-existing models that have been trained without explicit provisions for interpretability. These approaches have the potential to be employed with any existing ANN. The majority of post-hoc methods involve approximating the ANN using a more understandable model (e.g., a decision tree).

An alternative approach for post-hoc explaining ANN’s predictions involves establishing a link between knowledge or concepts, typically represented in an ontology, and the activation of ANN’s layers (Agafonov & Ponomarev, 2022; de Sousa Ribeiro & Leite, 2021). This correspondence process, referred to as “concept extraction”, entails training a mapping ANN. The mapping network takes the output of specific neurons from the main network being explained and produces the probability that the sample processed by the main network corresponds to the specified ontology concept. Frequently, these mapping networks can attain a significantly high level of predictive accuracy, facilitating the dependable extraction of a set of concepts from a given sample.

In (Agafonov & Ponomarev, 2023), a library was presented that includes a number of concept extraction approaches based on the construction of mapping networks. In particular, an algorithm was implemented that simultaneously extracts all concepts using a single ANN. This approach uses all activations from the main network as input to the proposed mapping network. The proposed architecture of such a mapping network includes outputs corresponding to specific concepts.

In this paper, we consider how the concept extraction approach can be used to assess the reliability of networks in which symbolic knowledge has already been integrated. In particular, the scenario of using this approach is considered in the absence of an explicit ontological connection between the concepts of the subject area.

3 RESEARCH APPROACH

3.1 Problem and Dataset

The considered problem is enterprise model classification. The class of an enterprise model is determined by the quantity and types of the model’s nodes (the detailed dataset description can be found in (Smirnov et al., 2023)). The dataset comprising 112 models (insufficient for conventional ANN training) of 8 unbalanced classes (Table 1).

Among the 36 node types in the dataset, the analysis focuses only on 20 meaningful ones, including: *Attribute, Cause, Component, Concept,*

Constraint, External Process, Feature, Goal, Individual, Information Set, IS Requirement, IS Technical Component, Opportunity, Organizational

Table 1: Enterprise model classes in the dataset.

Enterprise model class	Number of samples
Business Process Model	43
Goal and Goal & Business Rule Model	13
Business Rule and Business Rule & Process Model	13
Actors and Resources Model	12
Concepts Model	10
Technical Components and Requirements Model	10
4EM General Model	7
Product-Service-Model	4

Unit, Problem, Process, Unspecific/Product/Service, Resource, Role, Rule. The mean node count per model is 27.3.

3.2 Methodology

3.2.1 Classification of Enterprise Models

The experiment reported in (Smirnov et al., 2023) was based on the usage of the ANN shown in Figure 1(a). It is aimed at classification of enterprise models based the presence and quantities of nodes of certain types in the model without accounting for the graph topology. It has three fully connected layers followed by the rectified linear unit (ReLU) activation function. The input data is presented as a vector of the size 20, with each element presenting a distinct node type. Intermediate layers have sizes 128 and 64 neurons respectively. The output data is the vector of size 8, with each value corresponding to enterprise model classes. The highest value position in the 8-number vector identifies the model class.

Pre-processing involves two steps. Initially, the quantity of nodes for each of the 20 types within an enterprise model is computed. This vector is normalized by dividing it by the highest node count, resulting in values ranging between 0 and 1.

Training is done with the learning rate of 10^{-3} , chosen after conducting multiple experiments with various learning rate values. The Adam optimizer (Kingma & Ba, 2014) is employed due to its superior performance in most scenarios, faster computational speed, and minimal parameter tuning requirements. Early stopping is implemented to stop training when the test set accuracy fails to improve for 20 consecutive epochs.

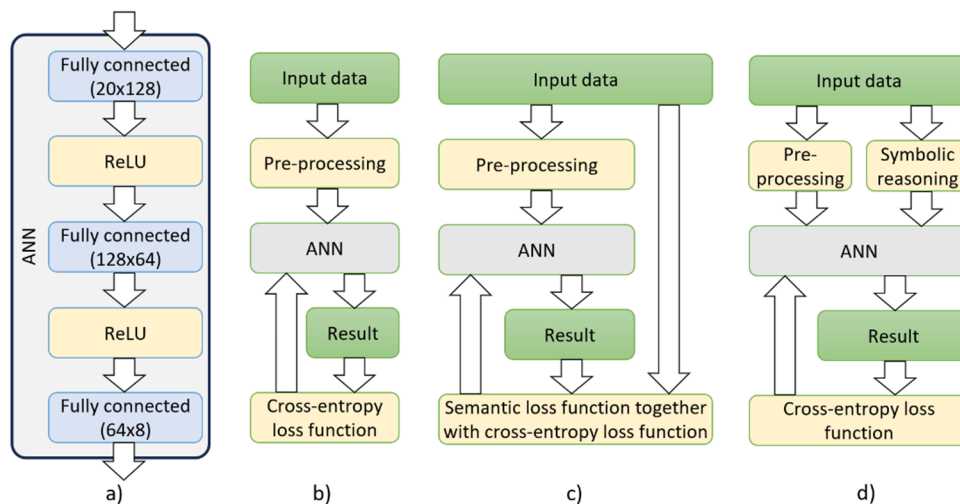


Figure 1: Studied ANN-based machine learning models (based on (Smirnov et al., 2023)).

Classification model evaluation employs a 5-fold cross-validation approach, which assumes partitioning the dataset into 5 subsets of similar sizes, with 5 experiments conducted where each subset serves as the test set once, while the remaining subsets are merged to form the training set.

The following approaches to network training were considered:

Basic ANN training. In the initial experiment, a normal ANN is employed for classification, with the Cross Entropy function being used as the loss function, as illustrated in Figure 1 (b).

Embedding symbolic knowledge using the semantic loss function. The semantic loss function proposed in (Xu et al., 2018) is combined with cross entropy loss via calculating the weighted sum (Figure 1(c)):

$$L = CELoss + \lambda \cdot SLoss, \tag{1}$$

where $CELoss$ and $SLoss$ represent the cross-entropy loss and the semantic loss respectively, $\lambda > 0$ is a hyperparameter that balances the constituents of the total loss function, representing the weight of the semantic loss. The initial semantic loss weight is 0.5. Each successive epoch the weight of the semantic loss is reduced by the value inverse to the total number of epochs. At each iteration, the final semantic loss weight is determined as the maximum between zero and the current semantic loss weight. The semantic loss function facilitates the integration of logical constraints into the ANN output vectors, leveraging such knowledge to enhance the training process. These constraints entail specifications where an enterprise model can be classified into a particular class if it has a node of a specific type. For example: "if the model includes a *Rule* node, it can only belong

to one of the following classes: *4EM General Model, Goal Model and Goal & Business Rule Model, or Business Rule Model and Business Rule & Process Model*". 20 rules have been defined for all 20 node types. The remaining training parameters were held as in the previous experiment.

Embedding symbolic knowledge using symbolic pre-processing. The third experiment included extension with extra inputs derived from the application of rules to the original input data, as illustrated in Figure 1(d). These rules align with those described in the previous experiment. The additional 8 inputs correspond to potential model classes (assigned a value of 1 if the class is viable, and -1 otherwise). Consequently, the initial layer of the ANN was expanded to the size of 28 nodes instead of the original 20. All other training parameters remained unaltered.

To conduct the experiment, 5 launches of the cross-validation procedure for each of the above approaches have been carried out. According to the results of each cross-validation, the mean accuracy value for all folds was saved along with the accuracy values of the best network (with the lowest loss value on the test set).

The results of training the main models reported in (Smirnov et al., 2023) showed that symbolic knowledge can indeed notably enhance the results of regular ANN for classifying enterprise models with small amount of training data. The most substantial enhancement was observed for the model incorporating symbolic data pre-processing: mean achieved accuracy was 0.973 vs. 0.929 achieved using regular ANN). At the same time, the usage of the semantic loss function did not yield any significant improvement (accuracy of 0.920).

3.2.2 Trustworthiness Assessment Using a Concept Extraction Approach

In the presence of trained networks for classifying enterprise models, it becomes possible to interpret their predictions using a post-hoc approaches to explanation. In particular, the concept extraction approach would allow to identify the types of nodes of a specific enterprise model, the presence or absence of which influenced the classification result. In other words, it will be possible to understand if the machine learning model relies on the node types as the significant sign of the enterprise model class and does not just learn the enterprise models in the training set.

As noted earlier, the extraction of concepts (node types) is carried out using mapping networks, which provide links between the internal representation of the sample by the main classification network and each of the concepts. The quality indicators of mapping networks can be used to compare the classification networks of enterprise models in terms of consistency of their internal representations with symbolic knowledge. Thus, the more consistent the internal representations are, the higher the trustworthiness of the network and its reliability.

The process of extracting concepts is carried out using the simultaneous extraction approach implemented in the RevelioNN library (Agafonov & Ponomarev, 2023). The simultaneous mapping network receives as input the values of the produced (when the sample is inferred) activation of all fully connected layers of the main network classifying enterprise models, and its outputs are the probabilities of each of the concepts (node types).

The following values of the architecture parameters of the simultaneous mapping network were set (Figure 2):

- 16 output neurons in decoder blocks;
- 8 output neurons in the internal representation block;

- Concept blocks are represented by layers containing 8 neurons at the input and 1 neuron at the output;
- 20 concept blocks (in our case, it is determined by the number of possible types of nodes of the enterprise model).

Each mapping network was trained three times for each already trained best classification network (with the lowest loss value on the test set). Thus, the number of mapping networks was 15 for each approach described in sec. 3.2.1.

The number of learning epochs was limited to 1000, and the patience value for the early stopping was 200. The Adam optimizer with a learning rate of 0.001 was used.

To assess the quality of mapping networks, the prediction accuracy of each of the concepts (node types) was calculated, as well as the mean prediction accuracy of all node types.

4 RESULTS AND DISCUSSION

The results of the carried out experiment are illustrated in Figures 3-5 and Table 2. The vertical line segments in the figures indicate the variation of the indicated value between different experiment launches.

Figure 3 shows the distribution of the mean accuracy of enterprise model classification networks based on the results of all launches of the cross-validation procedure. It can be seen that the best classification quality (accuracy about 0.98) is typical for the approach using symbolic pre-processing (no variation between different launches). The reason for this can be the strong correlation between the additional inputs in this scenario and the anticipated outcome, which positively impacts the efficiency of the machine learning model. The classification accuracy in basic network training turns out to be

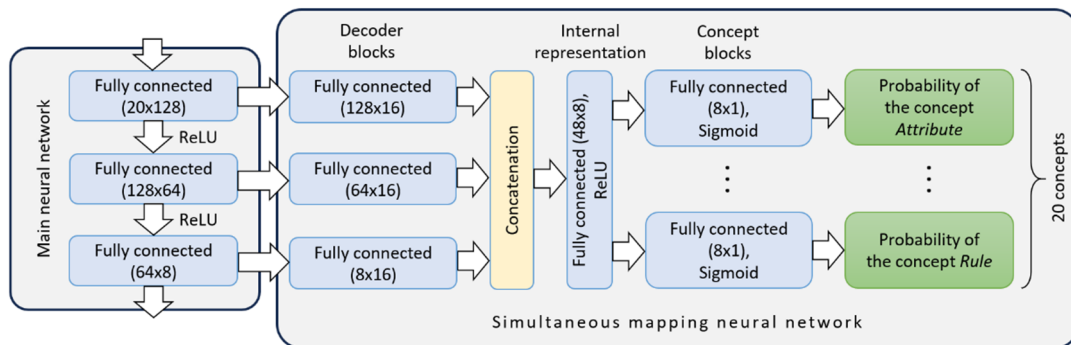


Figure 2: Simultaneous mapping network structure.

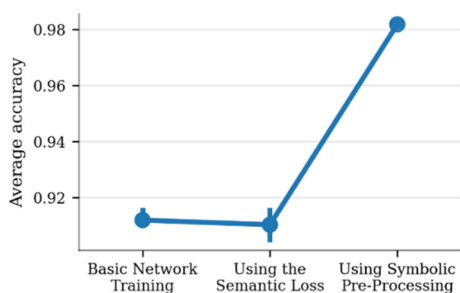


Figure 3: Distribution of the mean classification accuracy for all networks.

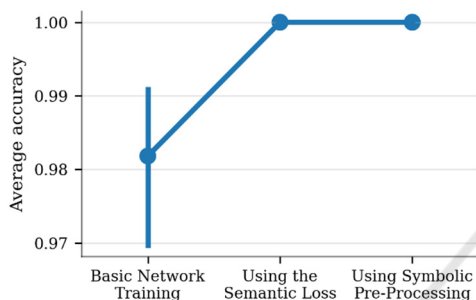


Figure 4: Distribution of the mean classification accuracy for networks with the lowest loss value at each cross-validation.

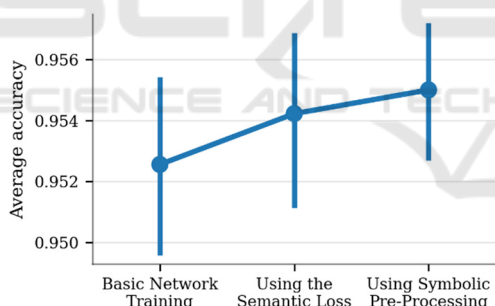


Figure 5: Distribution of the mean accuracy of concept extraction.

significantly lower, as well as when using semantic loss. It is also worth noting that in the case of using symbolic pre-processing, there is practically no quality variation.

An interesting observation can be done, if only the networks with the lowest loss value on the test set (the “best” ones) obtained during each cross-validation are considered. It turns out that when using the semantic loss function and the symbolic pre-processing approach (those with symbolic knowledge), the mean classification accuracy is 1.0 and there is no variation (see Figure 4). While the classical approach to ANN training produces a lower

accuracy with a significant variation in its values. It can be concluded that embedding of symbolic knowledge makes it possible to achieve better prediction results (though not always) with a higher stability.

Figure 5 shows the distribution of the mean accuracy of extraction of all types of nodes by mapping networks. Since 15 instances of mapping networks were trained for each approach to training the classification network, the results can be considered fairly representative. It can be noted that although numerically the values are quite close to each other, the greatest accuracy is achieved when extracting concepts from a network using symbolic pre-processing.

Table 2 shows the characteristics of the distribution of the accuracy of extracting concepts (each of the node types) by mapping networks. As noted earlier, the mean prediction accuracy of the entire set of concepts turns out to be approximately comparable when for each of the approaches to the classification network training. However, if consider the best accuracy values for each node type are considered, one can see that networks trained by different approaches may be more preferable for extracting some types of nodes. From this point of view, none of the approaches under consideration is clearly better than the other. But when considering the best accuracy for each node type and for each approach, it can be noted that the largest number of concepts extracted with the highest accuracy (indicated with bold) is extracted from a classification network that uses symbolic pre-processing (15 out of 20). Thus, it can be concluded that its internal representations are best aligned with symbolic knowledge, and, consequently, it inspires more trust.

5 CONCLUSION

The research question stated in the paper is “If the neural-symbolic machine learning model is more trustworthy than the pure ANN?”

In order to answer the question, a state-of-the-art analysis in the corresponding areas has been performed and several experiments have been carried out. Enterprise model classification based on the contained node types has been used as the use case for the experiments.

Three ANN-based architectures have been analysed: regular ANN without any symbolic knowledge, usage of the semantic loss function, and data pre-processing using symbolic rules. Earlier obtained results showed that embedding symbolic

Table 2: Characteristics of the distribution of the accuracy of extraction of concepts.

Node type \ Approach	Basic Network Training		Using the Semantic Loss Function		Using Symbolic Pre-Processing	
	Mean	SD	Mean	SD	Mean	SD
Rule	0.9422	0.0320	0.9378	0.0330	0.9511	0.0330
Goal	0.9244	0.0295	0.9422	0.0266	0.9467	0.0246
Organizational Unit	0.9711	0.0117	0.9667	0.0218	0.9711	0.0117
Process	0.9600	0.0187	0.9622	0.0172	0.9333	0.0000
Resource	0.9622	0.0117	0.9644	0.0086	0.9644	0.0086
IS Technical Component	0.9933	0.0187	0.9867	0.0276	0.9978	0.0086
IS Requirement	0.9711	0.0117	0.9711	0.0117	0.9711	0.0117
Unspecific / Product / Service	1.0000	0.0000	0.9978	0.0086	1.0000	0.0000
Feature	1.0000	0.0000	0.9978	0.0086	1.0000	0.0000
Concept	0.9022	0.0295	0.9178	0.0248	0.9289	0.0172
Attribute	0.9556	0.0163	0.9578	0.0153	0.9667	0.0000
Information Set	0.9200	0.0303	0.9222	0.0499	0.8800	0.0246
External Process	0.7533	0.0676	0.7733	0.0491	0.7711	0.0486
Problem	0.9667	0.0126	0.9644	0.0266	0.9889	0.0163
Cause	0.9933	0.0138	0.9978	0.0086	1.0000	0.0000
Role	0.9333	0.0000	0.9311	0.0086	0.9289	0.0172
Constraint	0.9667	0.0000	0.9667	0.0000	0.9667	0.0000
Component	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000
Opportunity	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000
Individual	0.9356	0.0086	0.9333	0.0000	0.9333	0.0000
Mean accuracy	0.9526		0.9546		0.9550	
Number of concepts extracted with the highest accuracy	9		8		15	

knowledge as data pre-processing rules gives a significant advantage in terms of the enterprise model classification accuracy.

In this paper the problem of trustworthiness of different ANN-based architectures has been analysed via post-hoc explanation (specifically, the concept extraction). This technique is aimed at searching for certain concepts within the ANN-based models, showing that the neural model indeed relies at these concepts as indicators for the classification instead of just learning the samples. The obtained results showed that the accuracy of concept extraction for the models with symbolic knowledge is higher than for the model without such knowledge, though the difference between the latter and the model with semantic loss is relatively small. Thus, it can be concluded that embedding semantic knowledge into ANN-based models increases their trustworthiness since they become more oriented to usage of proper features (node types in this particular experiment) for generating the output (enterprise model class).

Future research directions will be concentrated on exploring more use cases and larger datasets.

ACKNOWLEDGEMENTS

The research is funded by the Russian Science Foundation (project 22-11-00214).

REFERENCES

- Agafonov, A., & Ponomarev, A. (2022). Localization of Ontology Concepts in Deep Convolutional Neural Networks. *2022 IEEE International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON)*, 160–165. <https://doi.org/10.1109/SIBIRCON56155.2022.10016932>
- Agafonov, A., & Ponomarev, A. (2023). RevelioNN: Retrospective Extraction of Visual and Logical Insights for Ontology-Based Interpretation of Neural Networks. *Conference of Open Innovation Association, FRUCT*, 3–9. <https://doi.org/10.23919/fruct60429.2023.10328156>
- Anaby-Tavor, A., Carmeli, B., Goldbraich, E., Kantor, A., Kour, G., Shlomov, S., Tepper, N., & Zwerdling, N. (2020). Do Not Have Enough Data? Deep Learning to the Rescue! *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05), 7383–7390. <https://doi.org/10.1609/aaai.v34i05.6233>

- Arabshahi, F., Singh, S., & Anandkumar, A. (2018). *Combining Symbolic Expressions and Black-box Function Evaluations in Neural Programs*.
- Confalonieri, R., Prado, F. M. del, Agramunt, S., Malagarriga, D., Faggion, D., Weyde, T., & Besold, T. R. (2019). An Ontology-based Approach to Explaining Artificial Neural Networks. *ArXiv, 1906.08362*(January). <http://arxiv.org/abs/1906.08362>
- Confalonieri, R., Weyde, T., Besold, T. R., & Moscoso del Prado Martín, F. (2021). Using ontologies to enhance human understandability of global post-hoc explanations of black-box models. *Artificial Intelligence, 296*. <https://doi.org/10.1016/j.artint.2021.103471>
- Confalonieri, R., Weyde, T., Besold, T. R., & Moscoso Del Prado Martín, F. (2020). Trepan reloaded: A knowledge-driven approach to explaining black-box models. *Frontiers in Artificial Intelligence and Applications, 325*, 2457–2464. <https://doi.org/10.3233/FAIA200378>
- Dash, T., Srinivasan, A., & Vig, L. (2021). Incorporating symbolic domain knowledge into graph neural networks. *Machine Learning, 110*(7), 1609–1636. <https://doi.org/10.1007/s10994-021-05966-z>
- de Sousa Ribeiro, M., & Leite, J. (2021). Aligning Artificial Neural Networks and Ontologies towards Explainable AI. *35th AAAI Conference on Artificial Intelligence, AAAI 2021, 6A*(6), 4932–4940. <https://doi.org/10.1609/aaai.v35i6.16626>
- Garcez, A. d'Avila, & Lamb, L. C. (2020). *Neurosymbolic AI: The 3rd Wave*.
- Guest, O., & Martin, A. E. (2023). On Logical Inference over Brains, Behaviour, and Artificial Neural Networks. *Computational Brain & Behavior*. <https://doi.org/10.1007/s42113-022-00166-x>
- Hu, Z., Ma, X., Liu, Z., Hovy, E., & Xing, E. (2016). *Harnessing Deep Neural Networks with Logic Rules*.
- Kingma, D. P., & Ba, J. (2014). *Adam: A Method for Stochastic Optimization*.
- Lai, P., Phan, N., Hu, H., Badeti, A., Newman, D., & Dou, D. (2020). Ontology-based Interpretable Machine Learning for Textual Data. *2020 International Joint Conference on Neural Networks (IJCNN)*, 1–10. <https://doi.org/10.1109/IJCNN48605.2020.9206753>
- Li, Y., Ouyang, S., & Zhang, Y. (2022). Combining deep learning and ontology reasoning for remote sensing image semantic segmentation. *Knowledge-Based Systems, 243*, 108469. <https://doi.org/10.1016/j.knosys.2022.108469>
- Mishra, N., & Samuel, J. M. (2021). Towards Integrating Data Mining With Knowledge-Based System for Diagnosis of Human Eye Diseases. In *Handbook of Research on Disease Prediction Through Data Analytics and Machine Learning* (pp. 470–485). IGI Global. <https://doi.org/10.4018/978-1-7998-2742-9.ch024>
- Nguyen, D. T., Nam, S. H., Batchuluun, G., Owais, M., & Park, K. R. (2022). An Ensemble Classification Method for Brain Tumor Images Using Small Training Data. *Mathematics, 10*(23), 4566. <https://doi.org/10.3390/math10234566>
- Panigutti, C., Perotti, A., & Pedreschi, D. (2020). Doctor XAI: An ontology-based approach to black-box sequential data classification explanations. *FAT* 2020 - Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 629–639. <https://doi.org/10.1145/3351095.3372855>
- Picco, G., Lam, H. T., Sbodio, M. L., & Garcia, V. L. (2021). *Neural Unification for Logic Reasoning over Natural Language*.
- Pitz, D. W., & Shavlik, J. W. (1995). Dynamically adding symbolically meaningful nodes to knowledge-based neural networks. *Knowledge-Based Systems, 8*(6), 301–311. [https://doi.org/10.1016/0950-7051\(96\)81915-0](https://doi.org/10.1016/0950-7051(96)81915-0)
- Prabhushankar, M., & AlRegib, G. (2022). *Introspective Learning: A Two-Stage Approach for Inference in Neural Networks*.
- Prem, E., Mackinger, M., Dorffner, G., Porenta, G., & Sochor, H. (n.d.). Concept support as a method for programming neural networks with symbolic knowledge. In *GWAI-92: Advances in Artificial Intelligence* (pp. 166–175). Springer-Verlag. <https://doi.org/10.1007/BFb0019002>
- Shavlik, J. W. (1994). Combining symbolic and neural learning. *Machine Learning, 14*(3), 321–331. <https://doi.org/10.1007/BF00993982>
- Shilov, N., Othman, W., Fellmann, M., & Sandkuhl, K. (2021). Machine Learning-Based Enterprise Modeling Assistance: Approach and Potentials. *Lecture Notes in Business Information Processing, 432*, 19–33. https://doi.org/10.1007/978-3-030-91279-6_2
- Shilov, N., Othman, W., Fellmann, M., & Sandkuhl, K. (2023). Machine learning for enterprise modeling assistance: an investigation of the potential and proof of concept. *Software and Systems Modeling*. <https://doi.org/10.1007/s10270-022-01077-y>
- Smirnov, A., Shilov, N., & Ponomarev, A. (2023). Facilitating Enterprise Model Classification via Embedding Symbolic Knowledge into Neural Network Models. *Communications in Computer and Information Science, 1875*, 269–279. https://doi.org/10.1007/978-3-031-39059-3_18
- Ultsch, A. (1994). The Integration of Neural Networks with Symbolic Knowledge Processing. In *New Approaches in Classification and Data Analysis* (pp. 445–454). https://doi.org/10.1007/978-3-642-51175-2_51
- Wermter, S., & Sun, R. (2000). An Overview of Hybrid Neural Systems. *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science), 1778*, 1–13. https://doi.org/10.1007/10719871_1
- Xie, Y., Xu, Z., Kankanhalli, M. S., Meel, K. S., & Soh, H. (2019). Embedding Symbolic Knowledge into Deep Networks. *Advances in Neural Information Processing Systems, 32*.
- Xu, J., Zhang, Z., Friedman, T., Liang, Y., & Broeck, G. (2018). A Semantic Loss Function for Deep Learning with Symbolic Knowledge. *Proceedings of Machine Learning Research, 80*, 5502–5511.
- Yang, Z., Ishay, A., & Lee, J. (2020). NeurASP: Embracing Neural Networks into Answer Set Programming. *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, 1755–1762*. <https://doi.org/10.24963/ijcai.2020/243>