

Causality Prediction with Neural-Symbolic Systems: A Case Study in Smart Grids

Katrin Schreiberhuber^{1,2,*}, Marta Sabou^{1,2}, Fajar J. Ekaputra^{1,2}, Peter Knees²,
Peb Ruswono Aryan², Alfred Einfalt³ and Ralf Mosshammer³

¹Vienna University of Economics and Business, Welthandelsplatz 1, 1020 Vienna, Austria

²TU Wien – Faculty of Informatics, Favoritenstr. 9-11, 1040 Vienna, Austria

³Siemens AG Österreich, Siemensstr. 90, 1210 Vienna, Austria

Abstract

In complex systems, such as smart grids, explanations of system events benefit both system operators and users. Deriving causality knowledge as a basis for explanations has been addressed with rule-based, symbolic AI systems. However, these systems are limited in their scope to discovering causalities that can be inferred by their rule base. To address this gap, we propose a neural-symbolic architecture that augments symbolic approaches with sub-symbolic components, in order to broaden the scope of the identified causalities. Concretely, we use Knowledge Graph Embeddings (KGE) to solve causality knowledge derivation as a link prediction problem. Experimental results show that the neural-symbolic approach can predict causality knowledge with a good performance and has the potential to predict causalities that were not present in the symbolic system, thus broadening the causality knowledge scope of symbolic approaches.

Keywords

Knowledge Graph, Knowledge Graph Embedding, Explainability, Causality, Smart Grid

1. Introduction

Causality knowledge enables advanced applications such as explaining events in complex (engineering) systems. For example, in the smart grid domain, the event of an electrical vehicle failing to load at full capacity could be due to: (i) an increased energy demand in the area due to low temperatures or (ii) reduced energy production due to low solar radiation in the area. Explanations of such reasons depend on understanding *actual causalities* [1], that is, concrete events that happen in the underlying energy grid and their causal relations (e.g., low levels of solar radiation at a concrete time/location cause low power generation at PV-station A). We focus on the problem of deriving such actual causality knowledge. Typically, this means starting out with a system (e.g., grid) topology and time-series data describing the evolution of variables

NESY 2023: 17th International Workshop on Neural-Symbolic Learning and Reasoning, Certosa di Pontignano, Siena, Italy

*Corresponding author.

✉ katrin.schreiberhuber@wu.ac.at (K. Schreiberhuber); marta.sabou@wu.ac.at (M. Sabou);

fajar.ekaputra@wu.ac.at (F.J. Ekaputra); peter.knees@tuwien.ac.at (P. Knees); peb.aryan@tuwien.ac.at

(P. R. Aryan); alfred.einfalt@siemens.com (A. Einfalt); ralf.mosshammer@siemens.com (R. Mosshammer)

📄 0000-0003-1815-8167 (K. Schreiberhuber); 0000-0001-9301-8418 (M. Sabou); 0000-0003-4569-2496 (F.J. Ekaputra);

0000-0003-3906-1292 (P. Knees); 0000-0002-1698-1064 (P. R. Aryan)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

in that system (e.g., solar radiation, power production) as input and deriving individual events and causal relations between them as output.

Such task of actual causality knowledge derivation was addressed with *deterministic (symbolic) approaches*, such as graph-traversal algorithms [2][3] and rule/logic-based solvers [4][5][6]. These approaches are limited in their scope by the rules they rely on, being unable to discover/learn causalities beyond those that can be deduced by rule inference. At the same time, we are not aware of any machine learning (sub-symbolic) approach that addresses such a complex problem spanning the large knowledge gap from time-series based variables as input to actual causalities as output. The recent trend in Artificial Intelligence towards the development of neural-symbolic (NeSy) approaches [7] focuses on the complementary combination of symbolic and sub-symbolic AI approaches. For example, the Semantic Web community proposed techniques such as KGE and deductive reasoning [8], and has seen rapid growth in systems that combine Semantic Web and Machine Learning components (a particular type of neural-symbolic systems) [9]. Therefore, in this paper we investigate:

- (RQ1) What is the causality prediction performance of a neural-symbolic approach compared to existing knowledge?
- (RQ2) To what extent can the neural-symbolic system predict new knowledge?

We address our research questions in the context of a smart grid use case (detailed in Sect. 2). We rely on the use of a domain-specific simulation environment to generate input data (i.e., a smart grid topology and time-series data of a number of key parameters generated over a set period of time). As a baseline, we consider a symbolic, knowledge graph-based approach to solving this problem [10], described in Sect. 3. We extend this system with a sub-symbolic component by applying KGEs for link prediction, where predicted links are causal relations (see Sect. 4). We set up an experiment (i) to compare the performance of the various KGE algorithms and (ii) to investigate the derivation of causality knowledge with a neural-symbolic approach (Sect. 5). As such, the contributions of this paper are:

- a neural-symbolic architecture for deriving causality knowledge;
- experimental evidence in a concrete case study about the performance of KGE as a particular implementation of a NeSy approach for causality derivation.

Our empirical investigations show that the proposed neural-symbolic system predicts causality links with an average hits@5 of 0.24 using TransE. Performance differs between event types, achieving up to 0.6 for some event types. Moreover, the trained models capture the latent semantics of the causality relations being able to predict (a) causalities for event types that they were not trained on; (b) indirect causality links that were not explicitly present in the training data.¹

¹implementation of our investigations is released at <https://github.com/Kat-rin-sc/ExpCPSKGE>

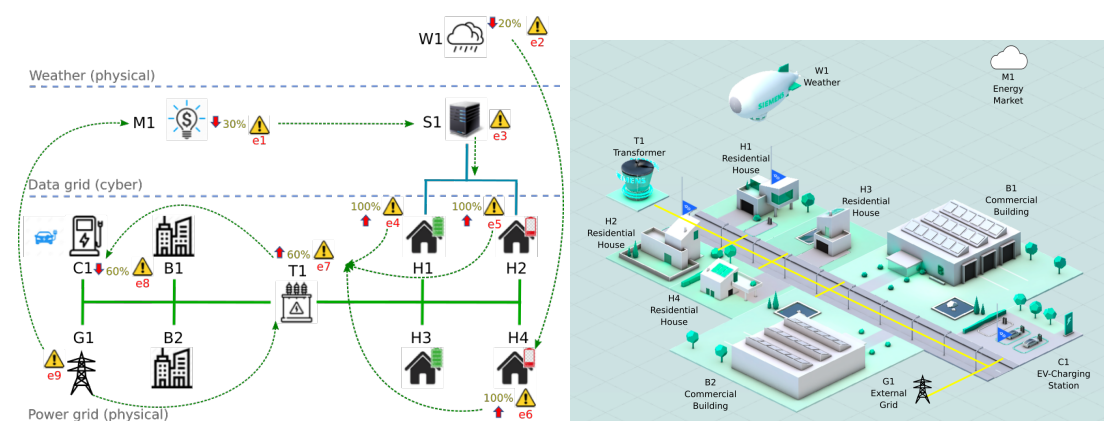


Figure 1: (a) Example energy community, including key events and their causal relations. (b) Corresponding BIFROST simulation.

2. Causality-based Event Explanations in Smart Grids

In the use case domain of smart grids, being able to explain complex events is crucial for saving energy and costs, but also to ensure the grid’s stability. Consider for example, the *energy community* (EC) (also referred to as an *(energy) village*) depicted in Fig. 1(a) consisting of houses H1-H4 (equipped with PV cells, batteries) connected to a flexibility operator S1 which informs EC members about best trading actions (selling/buying energy) depending on the energy price on the market M1. The EC is served by transformer T1, to which other office buildings (B1, B2) and eCar charging stations (C1, C2) are connected.

In this setting complex event explanations can be derived consisting of chains of causality links between concrete events that happened previously. For example, the event of slow charging at C1 (event e8) is caused because of reducing an overload at transformer T1 (e7). On its term, e7 was caused by the EC members (i) consuming a high amount of energy (e4-6) because a “buy” command of S1 (e3) in line with low energy prices (e1) and discharged batteries of some houses and (ii) producing less energy as usual due to reduced solar radiation (e2). Other event types (see Table 1) that require explanations include the overloading or normalisation of a transformer, or lowering/peaking in energy demand of customers.

We employ BIFROST [11], a smart grid simulator, to construct such scenarios, as in Fig. 1(b), and to simulate the behavior thereof over a period of time (typically, a day). As a result, the evolution of several physical measures (e.g., voltage levels) is computed leading to time-series data with a chosen time frequency (usually 1 hour).

3. ExpCPS: Symbolic AI Solution

We build on an existing approach to derive system event explanations using symbolic AI, called ExpCPS [2, 10] and depicted in Fig. 2. The input is provided by BIFROST. Core to the system is a semantic knowledge structure, the *ExpCPS KG*, which stores (a) a semantic representation of the simulated scenario and (b) additional knowledge necessary for deriving explanations

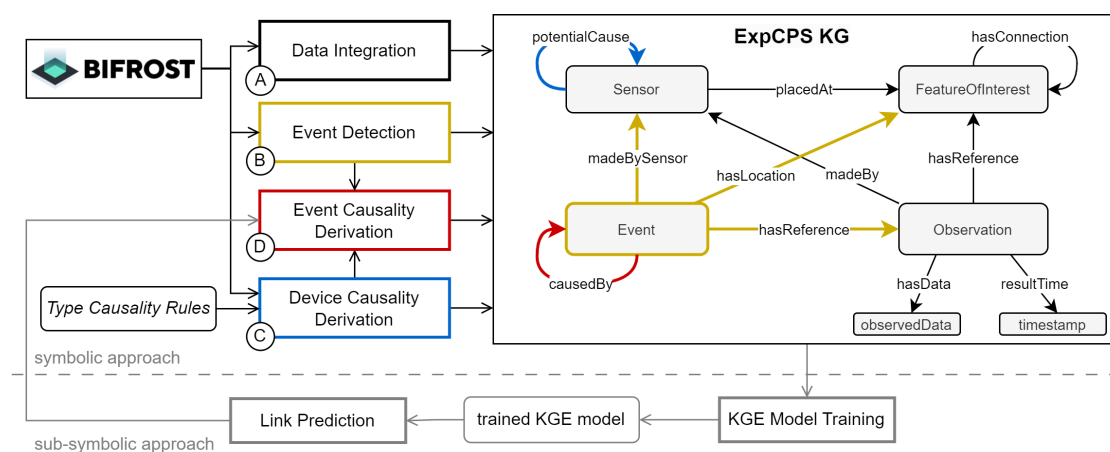


Figure 2: Neural-symbolic architecture for causality detection combining the ExpCPS *symbolic* approach (top layer) and *sub-symbolic* models for learning causalities from ExpCPS KG (bottom layer).

and which is inferred using symbolic techniques such as rules. The workflow for deriving the ExpCPS KG and explanation of events (see symbolic approach in Fig. 2) consists of the following components (color coded and labelled in Fig. 2):

- (A) In a *Data Integration* process, *static topology* data including physical components of the simulated village (i.e. building, EV-charging station, underground cable) are represented as instances of the *FeatureOfInterest* concept. Sensors are placed at these features to measure their state at different times during a simulation (i.e. loading of a transformer). *Dynamic* data corresponding to the sensor measurements during the simulation are stored as *Observations* (i.e., *observedData* at a *timestamp*).
- (B) An *Event Detection* process identifies anomalies in the dynamic measurement data which are stored as instances of different types of *Events* (see Table 1) in ExpCPS KG.
- (C) *Device Causality Derivation* uses *type causality rules* defined by domain experts to derive *device causality* in the energy community (see an example in Fig 5, Appendix). *Type causality* [1] represents general relations between entity types in a system (i.e. the power consumed by a residential house influences the loading of a transformer). Based on this knowledge, potential causes between specific instances of sensors in the energy community can be derived, called *device causality*. It captures the relation between two devices in a physical context (i.e. “power consumption at powermeter M1 influences transformer load at loadingsensor L6” –device causality– is derived from “power consumption at a powermeter influences transformer load” –type causality).
- (D) The rule-based *Event Causality Derivation* algorithm considers device causalities between sensors and events in the system to derive a set of actual causalities. *Actual causality* focuses on particular events that happened in a system at a specific time and place (i.e. *PeakingDemand* E3 causes *Overloading* E4), which make up an event explanation. To find explanations of a certain event (i.e. *Overloading* E4), multiple steps need to be taken: (i)

Event Type	Definition
Overloading/Normalizing	transformer is overloading/normalizing
Lowering/Peaking Demand	active power demand of energy consumers is lowering/peaking
Flex Request Approved/Rejected, Flex Unavailable State	states of a flexibility contribution program, where energy consumers adhere to requests to use more or less energy

Table 1

Definitions of event types relevant for actual causality prediction

find the sensor, which registered the event (i.e. loadingsensor L6), (ii) find sensors, which are connected to this sensor registering the event by device causality (i.e. powermeter M1 influences loadingsensor L6), (iii) check for events, which happened at these sensors (i.e. PeakingDemand E3 is registered at powermeter M1) and (iv) add the causality relation to the KG (i.e. PeakingDemand E3 cause Overloading E4). For a full explanation path, these steps need to be repeated recursively for each event added in (iv).

Limitation. While ExpCPS successfully addresses the causality-based explanation generation problem, its scope is limited by the rule-based mechanism it relies on which leads to a *deterministic behavior*, i.e., identifying only those causalities that can be derived by rules, but not being able to *learn* other plausible causalities by itself.

4. Proposed Sub-Symbolic Extension

Our *hypothesis* is that the *symbolic* ExpCPS system can be extended with *sub-symbolic* components in order to increase the scope of the learned causalities. To test this hypothesis, we propose a *neural-symbolic architecture* (Fig.2) which complements the symbolic ExpCPS approach with a sub-symbolic layer trained on the existing ExpCPS KG to derive new actual causalities, leading to extended system explanations. Concretely we make use of KGE methods, which use relations between KG entities to embed the components of a KG into a continuous vector space as a basis for downstream tasks such as KG completion, relation extraction or entity classification [12]. We formalise the problem as a *link prediction* problem: in the ExpCPS KG causalities between events are represented as RDF-triples, e.g.: (OverloadEventC, causedBy, PeakingDemandEventA). The role of KGE is to predict the tail entities (i.e., objects) in such causality triples (i.e., the potential causes for the head entity. i.e., the subject of the triple). We focus on evaluating the performance of this prediction task (RQ1) and whether the neural-symbolic system can predict new knowledge beyond what ExpCPS provides (RQ2).

Rather than replacing the original ExpCPS architecture, the aim of using KGE is to enhance the existing system. The proposed sub-symbolic approach should be an extension of the symbolic system, training the models on the KG, which was created using the ExpCPS approach (see Fig. 2)

Training Data Architecture. We make use of two villages, representing energy community setups in BIFROST with different topologies, to create the training/test and validation environment. For each village the corresponding ExpCPS KG is derived using the ExpCPS system

	Topology	Observations	Events	Device Causality	Actual Causality
Village A	1 004 sensors 379 FOIs 23 feature types	25 067 observations in 24h	107 events	57 potential causes between sensors	85 actual causes between events
Village B	1 203 sensors 379 FOIs 24 feature types	30 075 observations in 24h	191 events	77 potential causes between sensors	validation data: 133 actual causes between events

Table 2

Overview of village A and village B - actual causality in Village B (grey background) is used as validation data, while the rest is randomly split for training and testing (FOI = feature of interest)

with characteristics captured in Table 2. The *training/test data* consists of the entire KG derived for village A. Since KGEs cannot predict any relations of instances which are unknown in the training phase, the topology, observation, events and device causality of village B are included in the training set as well. The *validation data* (i.e., *ground truth*) consists of the “Actual Causality” knowledge of village B, which we try to predict.

In this setup, a KGE model can learn actual causality embeddings from all information in the village A ExpCPS KG. Moreover, it is equipped with information about the topology, observations and device causalities of village B, but no actual causalities. By testing the performance of embedding models on predicting actual causalities for a set of events in Village B, we can test their ability to represent actual causality. As the buildings and events are different in the two villages, we can ensure that village A data is only used to train actual causalities overall, but no direct information on actual causality in village B is used in training. Thus, there is no data leak between training and validation data.

Knowledge Graph Embeddings We chose four embedding methods based on their computational intensity and their ability to represent various types of relations. TransE [13], TransH [14] and TTransE [15] are translational embedding models, meaning that they aim to use relations as a “translation” from one entity to another. TransE is one of the earliest proposed embedding approaches. TransH extends the idea of TransE by using separate Hyperplanes for different types of relations. TTransE adds temporal information in the form of timestamps to each RDF triple [12]. We chose ComplEx [16], a high-performing [17] semantic matching model which leverage latent semantics present in the vector embeddings of entities and relations to measure the plausibility of triples.

For each KGE method, hyperparameter optimization (HPO) using grid search was conducted on loss function, model embedding dimensions, optimizer, negative sampler and number of epochs (see Table 3 in the Appendix for HPO search space and optimized setup of each embedding model). For HPO, the training/test data was split randomly into training and test set, training each setup on the training data and testing the performance on test data. For final evaluation, each model was trained on the full set of training and test data using the optimized setup and was tasked to predict actual causalities for events in the validation data. The results were then evaluated on their prediction performance of actual causalities in the validation data.

For Village B, explanations for 17 events (of event types shown in Table 1) exist in the valida-

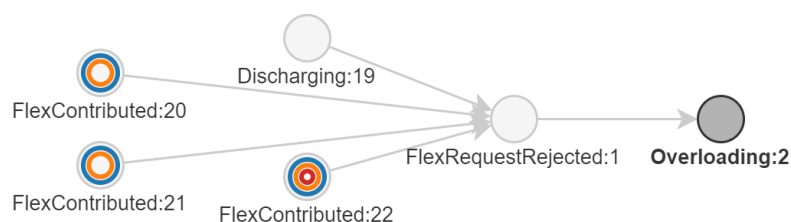


Figure 3: Event explanation of `OverloadEvent2`. Green node is the explained event. Colored circles show the predictions of the respective models (● TransE, ● TransH, ● ComplEx, ● TTransE)

tion data. Each explanation contains one to twelve cause events. For example, for the event `OverloadEvent2` there is only one direct cause in the validation data (`FlexRequestRejectedEvent1`), but four events are indirectly causing the event (`DischargingEvent19`, `FlexContributedEvent20`, `21` and `22`)(see Fig. 3).

Evaluation Setup The proposed system predicts the most fitting causes for an event in the validation data according to the embedding model trained on the training and test set of the ExpCPS KG (see Table 2). For evaluation, we measure the performance of causality link prediction on validation events with the *hits@5* metric, which captures the percentage of true predictions in the five top-ranked predictions of a model.

True predictions are determined in two ways. First, the set of actual causalities in the validation data, which were also determined by the symbolic system, (i.e., actual causality data for village B in Table 2) are considered definitely true. Second, the models’ predictions that were not present in the validation data were checked by experts in an evaluation workshop and labelled on their likelihood to be true on a scale from 1 to 4 (1 - definitely false, 2 - probably false, 3 - probably true, 4 - definitely true). From the above example, `FlexRequestRejected:1` is a true cause of `Overloading:2` according to the validation data, thus it is true. Moreover, a KGE model might also suggest `Discharging:4` to be a cause of `Overloading:2`, which was analysed by experts and labeled as probably true. Overall, each prediction which is not definitely true is evaluated by experts to determine its likelihood to be a missing link in the graph. Accordingly, two performance metrics are defined: (a) *defHits@5* includes predictions which are definitely true because they were present in the validation data; (b) *probHits@5* includes predictions, which are either probably or definitely true, based on expert evaluation and ground truth data. Therefore, the improvement from *defHits@5* to *probHits@5* shows the potential of a KGE method to increase the knowledge base created by the symbolic ExpCPS approach.

5. Evaluation Results

Performance of KGE methods. In Fig. 4, the comparison between *defHits@5* and *probHits@5* is visualised overall and per event type (see Table 4 in the appendix for a tabular representation of these values). Overall, TransE performed best on link prediction for *defHits@5* and *probHits@5*. TransH and ComplEx benefited the most from including probable hits in the performance metric as their result improved by 0.09 from *defHits@5* to *probHits@5*.

While *probHits@5* increase model performances for some types of events, general performance

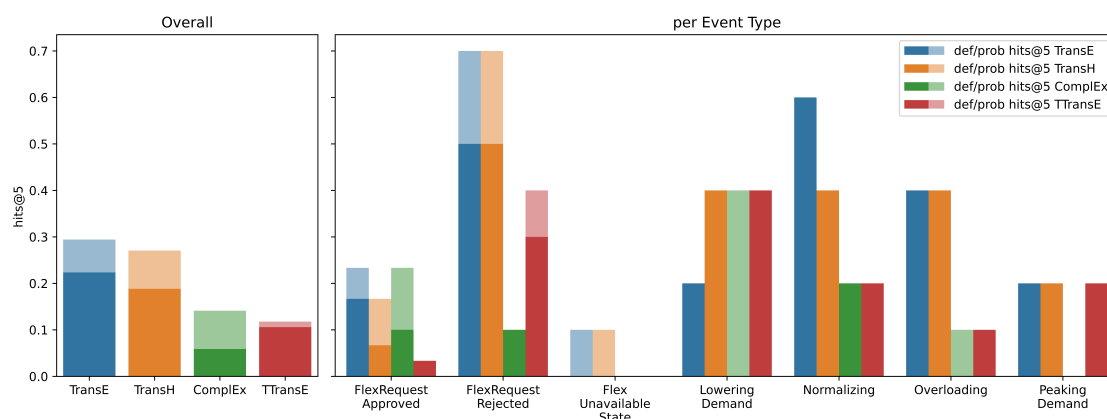


Figure 4: Link Prediction Performance of all four models overall(left) and per event type(right). Solid colors represent defHits@5, transparent colors represent probHits@5.

rankings between the four models hardly ever change between *defHits@10* and *probHits@5*. Even though TransE does not perform best for each event type individually, it does outperform other models overall, being able to always predict at least a probable cause for each event type.

Predicting event types absent from the training data. The event types `FlexRequestRejected` and `FlexUnavailableState` did not occur in Village A. Therefore, there was no explanation of these event types known to the trained KGE models. As expected, all KGE models perform poorly when trying to find an explanation for `FlexUnavailableState` event types. Surprisingly, predictions on `FlexRequestRejected` event types are good, especially with TransE and TransH. As there was no event of type `FlexRequestRejected` present in the training data of Village A, it is impressive that performance on this event type is among the best compared to other event types. Furthermore, as this causality was not learned by the KGE models explicitly, these results suggest that the KGE models “picked up” on the latent semantic implications of causality sufficiently to be able to predict causal events of event types unseen during training.

Predicting indirect causality. Indirect causalities are causes of a cause of an event, e.g., in Fig 3, for the `OverloadingEvent:2` the direct cause is `FlexRequestRejectedEvent:1`; while the other 4 events in the figure are indirect causalities. We observed that the KGE models were able to predict such indirect causalities. Fig. 3 shows that none of the KGE models could predict the immediate cause event for `OverloadingEvent:2`, but three of the indirect causes were predicted by TransE and TransH. Indirect causalities are ground-truth data which was never directly connected to the effect events in the original KG. Therefore, the behavior of predicting such causalities shows that the KGE models could represent actual causality relations and apply these semantics to the data beyond simple reconstruction of existing direct relations.

6. Related Work

Topics related to system explainability include anomaly detection and root-cause analysis (RCA), both of which can be tackled by methods such as Failure Mode and Effect Analysis [18] and Fault Tree Analysis [19]. These methods apply a human-oriented approach, where causes of potential anomalies need to be specified by domain experts. However, they are at risk of inconsistent terminology and ambiguous knowledge. As an extension, the use of ontologies has been proposed to remove ambiguity from causality knowledge [20]. Moreover, semantic technologies enable researchers to use existing general-purpose knowledge about a system for analysing its behaviour [21]. RCA methods primarily employ deterministic and probabilistic techniques [22] and are hampered by the increasing complexity of the growing graphs.

KGEs could address this situation by better scaling to such large graphs. Yet, work on using KGEs for predicting causality relations is limited to rather recent endeavours. For example, Khatiwada et al [23] have investigated the use of KGE for predicting causal relations for news events based on Wikidata Knowledge. However, the sparsity of causal relations in the Wikidata KG limits the capabilities of KGE predictions.

7. Conclusions

We propose a neural-symbolic approach for causality link prediction which we implement and test in a smart grid setting, where we extended a symbolic system (ExpCPS) with KGE, investigating four KGE methods. Performance evaluations showed that links are predicted with an average hits@5 of 0.24 using TransE (and up to 0.6 for some event types). More interestingly, the approach was able to discover new knowledge such as (i) explanations of event types which were not present in the training data and (ii) indirect causalities (only present implicitly in the training data).

Acknowledgements. The master thesis work of Katrin Schreiberhuber has been funded by the Career Grant proposals scheme of the Faculty of Informatics in TU Wien. This work has been funded by the FFG SENSE project (project Nr. FO999894802) and the Austrian Science Fund (FWF) [P33526].

References

- [1] J. Y. Halpern, *Actual causality*, MIT Press, 2016.
- [2] P. R. Aryan, F. J. Ekaputra, M. Sabou, D. Hauer, R. Mosshammer, A. Einfalt, T. Miksa, A. Rauber, Simulation support for explainable cyber-physical energy systems, in: 8th Workshop on Modeling and Simulation of Cyber-Physical Energy Systems, IEEE, 2020, pp. 1–6.
- [3] J. Qiu, Q. Du, K. Yin, S.-L. Zhang, C. Qian, A causality mining and knowledge graph based method of root cause diagnosis for performance anomaly in cloud applications, *Applied Sciences* 10 (2020) 2166.

- [4] J. Ploennigs, A. Schumann, F. Lécué, Adapting semantic sensor networks for smart building diagnosis, in: *Proceedings of ISWC, Part II 13*, Springer, 2014, pp. 308–323.
- [5] F. Wotawa, O. Tazl, D. Kaufmann, Automated diagnosis of cyber-physical systems, in: *Proceedings of IEA/AIE, Part II 34*, Springer, 2021, pp. 441–452.
- [6] T. Mari, T. Dang, G. Gössler, Explaining safety violations in real-time systems, in: *Proceedings of FORMATS, Part 19*, Springer, 2021, pp. 100–116.
- [7] H. A. Kautz, The third AI summer: AAAI Robert S. Engelmore Memorial Lecture, *AI Magazine* 43 (2022) 105–125. doi:<https://doi.org/10.1002/aaai.12036>.
- [8] P. Hitzler, F. Bianchi, M. Ebrahimi, M. K. Sarker, Neural-symbolic integration and the semantic web, *Semantic Web* 11 (2020) 3–11.
- [9] A. Breit, L. Waltersdorfer, J. F. Ekaputra, M. Sabou, A. Ekelhart, A. Iana, H. Paulheim, J. Portisch, A. Revenko, A. Ten Teije, F. van Harmelen, Combining Machine Learning and Semantic Web -A Systematic Mapping Study, *ACM Computing Surveys* (2023).
- [10] P. R. Aryan, F. J. Ekaputra, M. Sabou, D. Hauer, R. Mosshammer, A. Einfalt, T. Miksa, A. Rauber, Explainable cyber-physical energy systems based on knowledge graph, in: *Proceedings of the 9th Workshop on Modeling and Simulation of Cyber-Physical Energy Systems*, 2021, pp. 1–6.
- [11] R. Mosshammer, K. Diwold, A. Einfalt, J. Schwarz, B. Zehrfeldt, Bifrost: A smart city planning and simulation tool, in: *Intelligent Human Systems Integration 2019: Proceedings of the 2nd International Conference on Intelligent Human Systems Integration (IHSI 2019): Integrating People and Intelligent Systems*, February 7-10, 2019, San Diego, California, USA, Springer, 2019, pp. 217–222.
- [12] Q. Wang, Z. Mao, B. Wang, L. Guo, Knowledge graph embedding: A survey of approaches and applications, *IEEE Transactions on Knowledge and Data Engineering* 29 (2017) 2724–2743.
- [13] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, *Advances in neural information processing systems* 26 (2013).
- [14] Z. Wang, J. Zhang, J. Feng, Z. Chen, Knowledge graph embedding by translating on hyperplanes, in: *Proceedings of the AAAI conference on artificial intelligence*, volume 28, 2014.
- [15] Z. Han, G. Zhang, Y. Ma, V. Tresp, Time-dependent entity embedding is not all you need: A re-evaluation of temporal knowledge graph completion models under a unified framework, in: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 8104–8118.
- [16] T. Trouillon, J. Welbl, S. Riedel, É. Gaussier, G. Bouchard, Complex embeddings for simple link prediction, in: *International conference on machine learning*, PMLR, 2016, pp. 2071–2080.
- [17] Y. Dai, S. Wang, N. N. Xiong, W. Guo, A survey on knowledge graph embedding: Approaches, applications and benchmarks, *Electronics* 9 (2020) 750.
- [18] M. Ben-Daya, S. O. Duffuaa, A. Raouf, J. Knezevic, D. Ait-Kadi, *Handbook of maintenance management and engineering*, volume 7, Springer, 2009.
- [19] C. A. Ericson, *Fault tree analysis primer*, CreateSpace Incorporated, 2011.
- [20] L. Dittmann, T. Rademacher, S. Zelewski, Performing FMEA using ontologies, in: *18th*

- International Workshop on Qualitative Reasoning. Evanston USA, 2004, pp. 209–216.
- [21] B. Steenwinckel, Adaptive anomaly detection and root cause analysis by fusing semantics and machine learning, in: *Extended Semantic Web Conference (ESWC)*, Springer, 2018, pp. 272–282.
- [22] M. Solé, V. Muntés-Mulero, A. I. Rana, G. Estrada, Survey on models and techniques for root-cause analysis, *ArXiv* (2017).
- [23] A. Khatiwada, S. Shirai, K. Srinivas, O. Hassanzadeh, Knowledge graph embeddings for causal relation prediction, in: *Workshop on Deep Learning for Knowledge Graphs (DL4KG)*, 2022.

A. Appendix

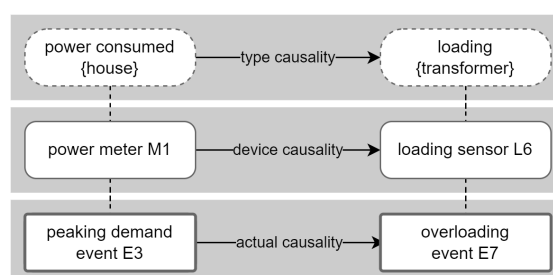


Figure 5: Inference of actual causality from type causality

Parameter	Potential Values	TransE	TransH	Complex	TTransE
dimensions	32,64,128	64	64	64	32
loss function	margin ranking, softplus, cross entropy loss	margin ranking	margin ranking	softplus	cross entropy loss
optimizer	Adam, SGD	Adam	Adam	Adam	Adam
learning rate	0.001, 0.01, 0.1	0.001	0.001	0.001	0.01
epochs	max 100, early stopping	100	30	80	45

Table 3
hyperparameter optimization search space and final model setups

event type	model	defHits@5	probHits@5
FlexRequest Approved	TransE	0.17*	0.23*
	TransH	0.07	0.17
	ComplEx	0.10	0.23
	TTransE	0.03	0.03
FlexRequest Rejected**	TransE	0.50*	0.70*
	TransH	0.50*	0.70*
	ComplEx	0.10	0.10
	TTransE	0.30	0.40
Flex Unavailable State**	TransE	0.00	0.07
	TransH	0.00	0.13*
	ComplEx	0.00	0.00
	TTransE	0.00	0.00
Lowering Demand	TransE	0.20	0.20
	TransH	0.40*	0.40*
	ComplEx	0.00	0.40*
	TTransE	0.40*	0.40*
Normalizing	TransE	0.60*	0.60*
	TransH	0.40	0.60*
	ComplEx	0.20	0.20
	TTransE	0.20	0.20
Overloading	TransE	0.40*	0.40*
	TransH	0.40*	0.40*
	ComplEx	0.00	0.10
	TTransE	0.10	0.10
Peaking Demand	TransE	0.20*	0.20*
	TransH	0.20*	0.20*
	ComplEx	0.00	0.00
	TTransE	0.20*	0.20*
Overall	TransE	0.24*	0.30*
	TransH	0.20	0.29
	ComplEx	0.06	0.15
	TTransE	0.11	0.13

* best result in this category

** event type is not present in the training data

Table 4

Link Prediction Performance over all event types and models. Best-performing models per event type are in bold.