April 2022

Learning and Reasoning over Smart Home Knowledge Graphs

Ronald Siebes, [a,1] Victor de Boer [a], Roberto Reda [b], and Roderick van der Weerdt [a]

[a] *VU University Amsterdam, The Netherlands*

[b] *Università di Bologna, Italy*

Abstract. Vendors of Smart Home devices and research institutes come together in the EU funded InterConnect project to develop a shared data model and accompanying infrastructure that facilitates seamless integration where all devices and services communicate via a human and machine interpretable language: SAREF. The Linked Data approach where a knowledge graph captures the shared information about Smart Devices and services in the network offers the opportunity for novel AI research. In this paper we outline four research directions currently being undertaken by the authors and present the results of using the SAREF knowledge graphs to 1) improve the explainability of existing machine learning (ML) approaches, 2) uniform transformation of graph data to tabular input for these ML approaches, 3) applying graph Deep Learning to derive missing classes and values and 4) reasoning with SWRL rules for Smart Home control.

Keywords. Smart Homes, Explainability, SWRL, Linked Data

1. Introduction

The InterConnect project[2] gathers 50 European partners to develop and demonstrate advanced solutions for connecting and converging digital homes and buildings with the electricity sector. The key to this interoperability is the use of the SAREF IoT ontology[1], that allows the variety of devices and services to communicate using shared concepts. Machine Learning (ML) algorithms also play a significant role in Interconnect. Most prominent are the services that do some type of forecasting such as predicting energy consumption for (Smart) devices and households in general. In this paper we address two aspects of such machine learning algorithms.

Firstly, in Section 3, we look into the 'black box' aspects of ML approaches. *Explainability* of (the outcomes of) algorithms is fervently debated topic [2]. In the media we can find various examples where bias in the training data and implementations lead to undesirable results. Users want to know why an algorithm came to a certain result and European Union has expressed the need to address this important aspect through clear guidelines. Through the use of SAREF in Interconnect, we have already an adequate mechanism in place to make knowledge explicit on various parts of the ML process that

[1] Corresponding Author: Ronald Siebes, VU Amsterdam The Netherlands; E-mail: r.m.siebes@vu.nl.

[2] `http://interconnectproject.eu`

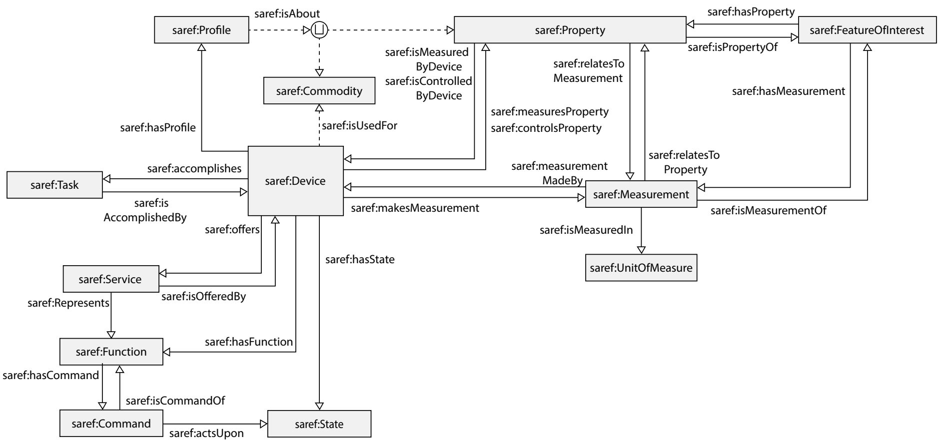

Figure 1. Overview of the SAREF ontology[4].

improve aspects of explainability, for example provenance. Having a standardized way to express who developed the algorithm, on which training data and where the privacy details and other aspects can be found contributes to explainability.

Secondly, in Section 4 we look into the *specificity* of both the type of predictions and the parameter settings. One important goal of Interconnect is standardization of the interaction between Smart devices and services. Therefore, including the interface to the services that are capable of applying machine learning in a standardized manner benefits the re-use and improve the adoption of them. SAREF and its extensions are the schemas and vocabularies that we use to express the capabilities, interaction details and other aspects of the Smart components and facilitating services. These include measurement parameters and values. Graph Patterns using the SAREF vocabulary allow us to standardize the input and output for the Interconnect services, including those that are based on Machine Learning. Thirdly, in Section 5 we demonstrate how the Semantic Web Rule Language (SWRL) applied on SAREF, enables IoT home automation via control logic.

2. SAREF

Different smart devices use different protocols and standards, often proprietary to each vendor, to communicate their data with other smart devices from that same vendor. SAREF, the Smart Applications REFerence ontology[3] [3], was created to be a new standard where every other device could translate their data into to share it with other devices, and translate data from to receive it from other devices [4].

Figure 1 shows how the main classes in the SAREF core ontology relate to each other. For our purposes we mainly focus on the `saref:Measurement` class and its related classes. For each different measurement a new `saref:Measurement` instance is created with two literals, the measurement value and the timestamp. It has a relation with a `saref:unitOfMeasure` instance (such as degrees Celsius), `saref:Property`

[3]`https://saref.etsi.org/`
[4]Image taken from: `https://saref.etsi.org/core/v3.1.1/#Figure_1`

instance (such as temperature), `saref:FeatureOfInterest` instance (such as the room where the measurement is taken) and to a `saref:Device` instance (being the device that has taken the measurement [5]. A more detailed description of the SAREF classes and properties can be found in [3].

3. Jupyter Notebooks for ML explainability

The effort in the Interconnect project to harmonize and standardize the communication between Smart devices and services, including AI technology as part of the services but also the Knowledge Engines, inherently will influence the life conditions of the people that will use these solutions. “Safety, Health and Fundamental Human Rights”, are identified by the European Union as categories with special responsibilities to the AI algorithms involved in the decision making process. In a complex and large project like Interconnect, where actual devices and AI technology will be implemented in office and home residences, we are obliged to have a look on the potential risks and prepare for this upcoming legislation drafted by the EU. Next to preparation for the legislation, the explainability of AI algorithms is a desired feature in many situations like *Error detection* and *End user feedback*.

3.1. Jupyter notebooks providing input for Explainability

Jupyter Notebooks[5] are run as a web application for creating and sharing computational documents in the Python programming language. The advantage of having a web interface where a user can monitor step by step the execution flow and visualize intermediate results is gaining insight into the algorithms. On this GitHub repository, [6],we developed an extensive example on using two popular ML algorithms (Linear Regression and Random Forests) to forecast the energy consumption of heat pumps based on the Open Power System Dataset Household data dataset (OPSD)[6] .

The in-line comments in the code together with the Jupyter environment makes the code easy to read even for those not familiar with ML algorithms, hence contributing significantly to the aspect of explainability.

4. SAREF to ML input

In this section two different methods will be addressed to transform the SAREF knowledge graph into data that is interpretable by ML methods that are commonly used to create forecasters.

4.1. Embeddings on SAREF data

The idea of an numerical representation, or an embedding, for a node is taken from the world of Natural Language Processing, where you have the literal word, and the semantic meaning of that word. Because the context of a word is relevant to the meaning of a

[5] https://jupyter.org/
[6] https://github.com/rsiebes/interconnect-explainability

specific word they created numerical representations for every word that is based on the words that are occur "near" that word in sentences in (large) texts used for training. This results in a multi dimensional embedding vector, with similar words having vectors that are closer together to each other and dissimilar words having vectors that are more apart. Using this approach on knowledge graphs, where the knowledge graph functions as the text, and walks through the text function as the sentences, where each word is a node from the graph and the next "word" is a node connected to the current node.

We provide an example in this repository[7], including a jupyter notebook, that demonstrates how such an approach would take the SAREF knowledge graph, create embeddings for relevant nodes, and use this embedding to train a Multi-Layered Perceptron (MLP) model. The MLP model can then be used as a forecaster.

4.2. Feature extraction on SAREF data

Feature extraction uses the numerical values that are available in the SAREF knowledge graph. We query the SAREF knowledge graph to retrieve all the measurement values from specific devices (e.g. all the devices in one home or one room) which are collected in a large table, with columns for each device and with each row collecting the measurements made for each specific time-stamp. This process benefits from the SAREFization process[8], since all the data from the different devices is collected and standardized within the knowledge graph, so we can reliably query the graph instead of having to combine the data from the different devices and datasets manually. Our example of how feature extraction can be implemented, in combination with a MLP forecaster can be found in this repository[9].

4.3. Preliminary results

Using the SAREFized OPSD dataset, an experiment was performed to examine whether the embeddings created from the measurements in the knowledge graph contained enough information to make useful classifications.

The task of the experiment was to predict whether the outside temperature is hot or cold based on the energy consumption measurements in the OPSD dataset, which was extended with the outside temperature measurements at the local municipality level, which was used to create the target classes of hot and cold. The 1000 warmest and 1000 coldest datapoints (energy consumption measurements and outside temperature measurements) were selected to create a dataset of 2000 energy consumption measurements labeled 'hot' or 'cold'.

We implemented a simple neural network with two hidden ReLU layers of size 512 using pyTorch trained for 20 epochs. This network was used to train a classifier for three different approaches:

- *RDF2Vec*, using RDF2Vec to create embeddings each energy consumption measurement node and using the subsequent embedding as input to predict the class label. The embeddings have a length of 100.

[7] https://github.com/RoderickvanderWeerdt/SAREF-RDF2vec-into-MLP
[8] SAREFization is the process of mapping a dataset to a SAREF knowledge graph
[9] https://github.com/RoderickvanderWeerdt/SAREF-Feature-Extraction-into-MLP

	RDF2Vec	Random	Feature Extraction
Test Accuracy	49.2%	48.4%	100.0%
Train Accuracy	49.9%	100.0%	99.9%

Table 1. Train and test accuracy of the classification of hot and cold weather based on the heat pump energy consumption (averaged over 5 runs).

- *Random*, using a vector of random values as an embedding for each measurement node and using the subsequent embedding as input to predict the class label. This will create a baseline to compare the result of the embedded approach with.
- *Feature Extraction*, using the value of the energy consumption measurement directly as input to predict the class label.

Table 1 shows the accuracy of these experiments. Feature Extraction clearly outperforms the RDF2Vec approach, and the random baseline results in a comparable result to the RDF2Vec produced embeddings, indicating that the information available in the graph is not being included in these learned embeddings. The RDF2Vec approach results in a model that classifies every instance as 'hot', and therefor get a result of almost 50%. But the random approach completely overfits on the available training data, seen from the 100% training accuracy, but it does not learn any generalized rules it can use for the test set.

5. SWRL for Smart Homes

Nowadays, IoT domotic devices have opened up a plethora of new opportunities in the home automation landscape. Crucial application areas of IoT-based smart home systems include energy management, home safety and assisted living. However, the real potential of IoT technology is still quite underexploited due to the high heterogeneity of the devices (which results in poor interoperability) and the scarce expressivity of the most common scenario programming paradigms.

The vast majority of available IoT home automation systems are based on the simple trigger-action model that consists of executing an action when certain conditions are met, thus scenarios are described by means of "if-then" rules.

Moreover, the poor interoperability among devices by different vendors makes it difficult to write rules device-independently. This means that rules cannot be reused for different vendor devices despite these performing the same functions.

To address the aforementioned issues we devised a novel approach to IoT home automation in which Semantic Web technologies are employed as a means to provide common semantics and to define the control logic. In particular, our approach leverages the capabilities of automated reasoning provided by the Semantic Web Rule Language (SWRL) and OWL. Automatic actions are thus the result of logical inferences over the sensor data combined with heterogeneous external information resources and formalised knowledge. The numerous experiments that we carried out in a simulated environment that makes use of SAREF as the main reference ontology proved the feasibility of the proposed approach and its applicability to a standard and well-validated context.

6. Summary and future work

Be representing Smart Home data as Linked Data, the resulting knowledge graphs are in a machine interpretable and standardized model. In this paper we presented ongoing work and results from the authors with the intention to facilitate a discussion for the workshop of the possibilities of the InterConnect project for the heterodox AI domain from various angles, unifying both the non-symbolic (e.g. Machine Learning) and symbolic (e.g. Description Logics) approaches. Within the Interconnect project, various pilots are planned to implement and demonstrate the usefulness of having Smart Home data available as Linked Data. These pilots provide the platform to test and improve the solutions described in this paper. Also we continue our research into the application of embeddings of the SAREFized knowledge graph, including creating a synthetic dataset generator that is able to adjust the amount of internal connections between the nodes in order to inverstigate whether the s̈hallow-nessöf a graph effects to learning of the embeddings. Further we will investigate how a more complete Smart Home knowledge graph could be used, which does not only contain energy consumption measurements, but a wide range of smart home data that different smart home devices can create (e.q. CO2 measurements, devices on/off, doors open/closed, washing machine full/empty, etcetera).

Acknowledgements

This work was supported in part by the European Union's Horizon 2020 project *InterConnect* under grant agreement 857237.

References

[1] Daniele L, den Hartog F, Roes J. Created in Close Interaction with the Industry: the Smart Appliances REFerence (SAREF) Ontology. In: International Workshop Formal Ontologies Meet Industries. Springer; 2015. p. 100-12.
[2] Murdoch WJ, Singh C, Kumbier K, Abbasi-Asl R, Yu B. Interpretable machine learning: definitions, methods, and applications. arXiv preprint arXiv:190104592. 2019.
[3] ETSI TS. 103 264 V3.1.1, SartM2M; Smart Applications; Reference Ontology and oneM2M Mapping. 2020.
[4] Daniele L, Strabbing W, Roelofsen B, Aalberts A, Stapersma P. Study on Ensuring Interoperability for Demand Side Flexibility. European Commission; 2018.
[5] van der Weerdt R, de Boer V, Daniele L, Nouwt B. Validating SAREF in a Smart Home Environment. In: Garoufallou E, Ovalle-Perandones MA, editors. Metadata and Semantic Research. Cham: Springer International Publishing; 2021. p. 35-46.
[6] Wiese F, Schlecht I, Bunke WD, Gerbaulet C, Hirth L, Jahn M, et al. Open Power System Data – Frictionless data for electricity system modelling. Applied Energy. 2019 feb;236:401-9. Available from: `https://doi.org/10.1016%2Fj.apenergy.2018.11.097`.