TECHNISCHE UNIVERSITÄT DRESDEN

DRESDEN concept

René Schöne, Johannes Mey, Boqi Ren and Uwe Aßmann

Bridging the Gap between Smart Home Platforms and Machine Learning using Relational Reference Attribute Grammars

Models@run.time 2019

September 17, 2019

Context and Goal

https://www.youtube.com/watch?v=HSFO1aD_w8o

Context and Goal

https://www.youtube.com/watch?v=HSFO1aD_w8o

Two Important Trends

More “Smart” Devices

— *“20 billion internet-connected things by 2020”* [Gartner, Inc., 2017]

— Middleware platforms manage large number of devices, abstract hardware details, and provide static rules

Commercial Machine Learning

— *“Machine learning has progressed dramatically over the past two decades, from laboratory curiosity to a practical technology in widespread commercial use”* [Jordan and Mitchell, 2015]

Two Important Trends

💡 More “Smart” Devices

— *“20 billion internet-connected things by 2020”* [Gartner, Inc., 2017]

— Middleware platforms manage large number of devices, abstract hardware details, and provide static rules

🎰 Commercial Machine Learning

— *“Machine learning has progressed dramatically over the past two decades, from laboratory curiosity to a practical technology in widespread commercial use”* [Jordan and Mitchell, 2015]

🤗 Combination of Smart Home Platforms and Machine Learning

— Middleware only use static rules, machine learning alone can not be used easily at runtime

— Very few approaches integrating both, and none uses Open Source Middleware Platforms

State-of-the-art: Middleware Platforms

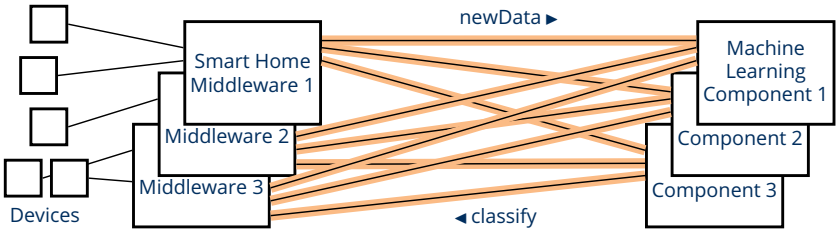
openHAB
Written in Java, based on Eclipse Smarthome. Concepts: Thing, item, type
Home Assistant
Written in Python. Concepts: state object, event, zone, scene
ioBroker
Written in JavaScript. Concepts: Adapter, instance, (UI) object
FHEM
Written in Perl. Concepts: Devices, events

Research Questions

RQ1 How to integrate multiple smart home middleware platforms with multiple machine learning components?

RQ2 Which model elements and relations between them are necessary for such an integration?

RQ3 How to ease the selection of relevant inputs for different instances of machine learning approaches?

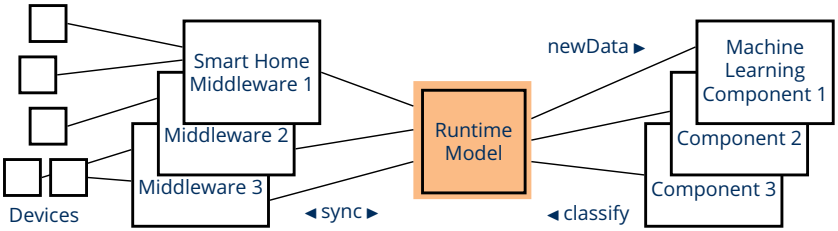


Research Questions

RQ1 How to integrate multiple smart home middleware platforms with multiple machine learning components?

RQ2 Which model elements and relations between them are necessary for such an integration?

RQ3 How to ease the selection of relevant inputs for different instances of machine learning approaches?

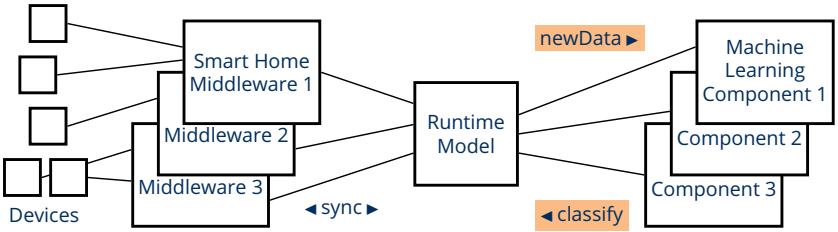


Research Questions

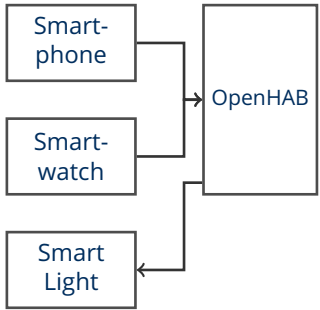
RQ1 How to integrate multiple smart home middleware platforms with multiple machine learning components?

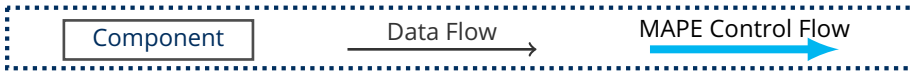
RQ2 Which model elements and relations between them are necessary for such an integration?

RQ3 How to ease the selection of relevant inputs for different instances of machine learning approaches?

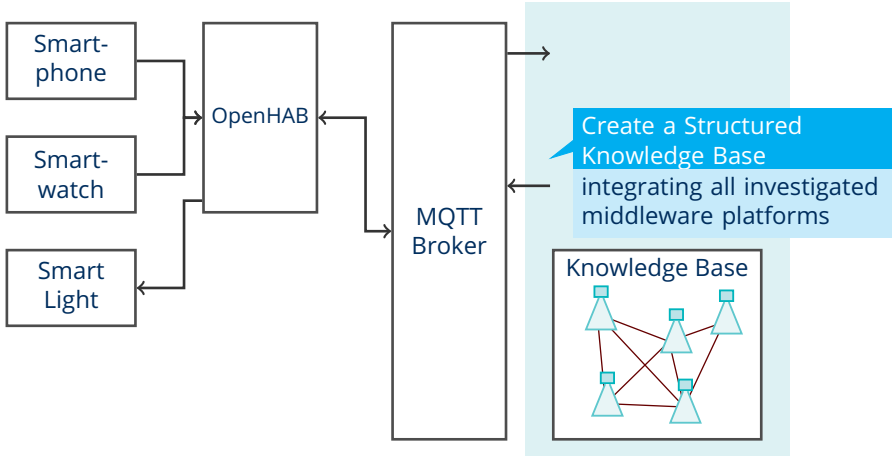


Solution: A Structured Model

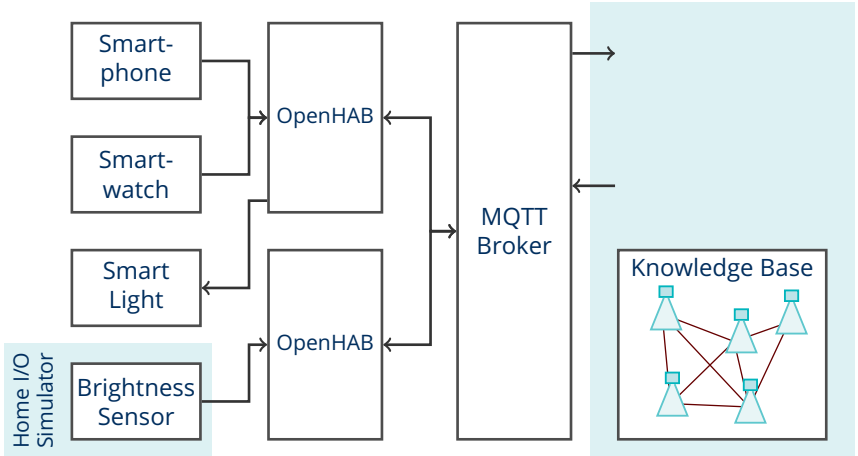




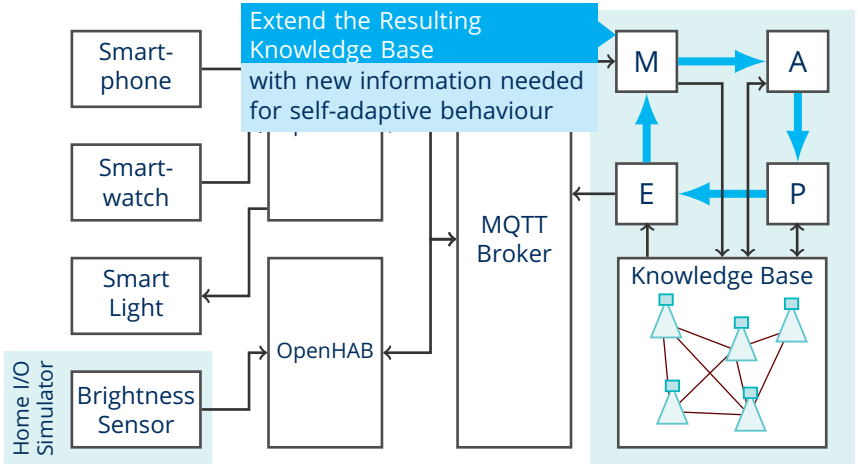
Solution: A Structured Model



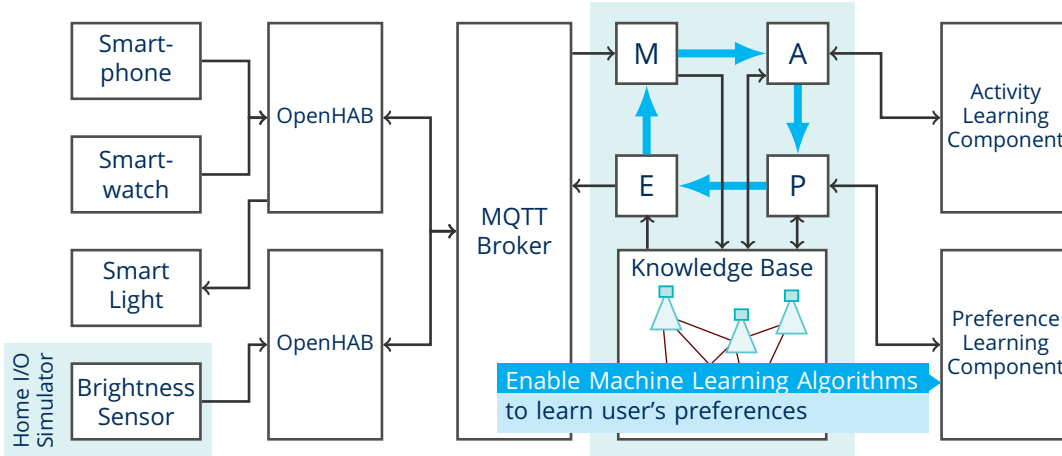
Solution: A Structured Model



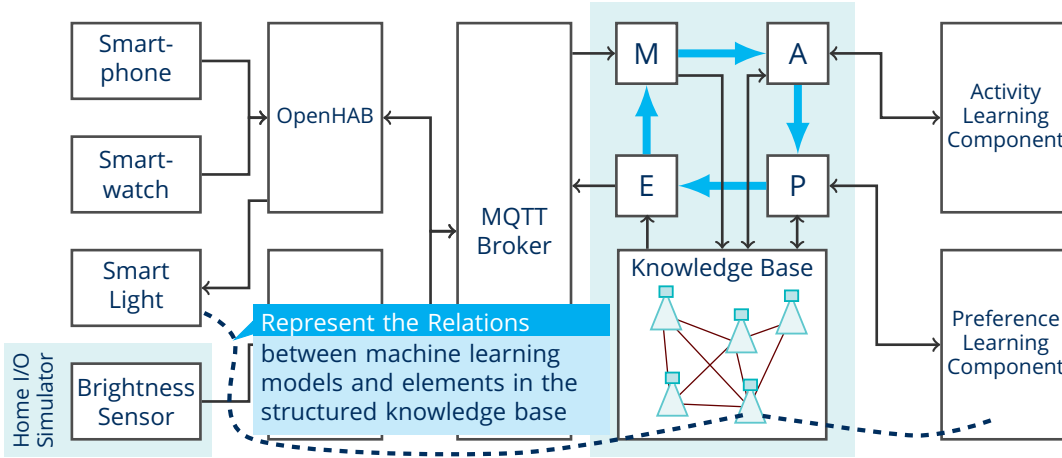
Solution: A Structured Model



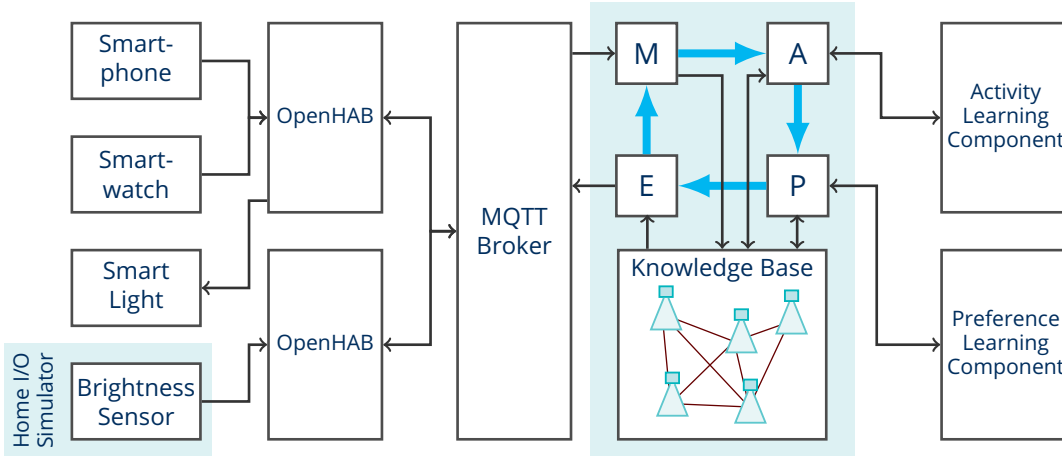
Solution: A Structured Model



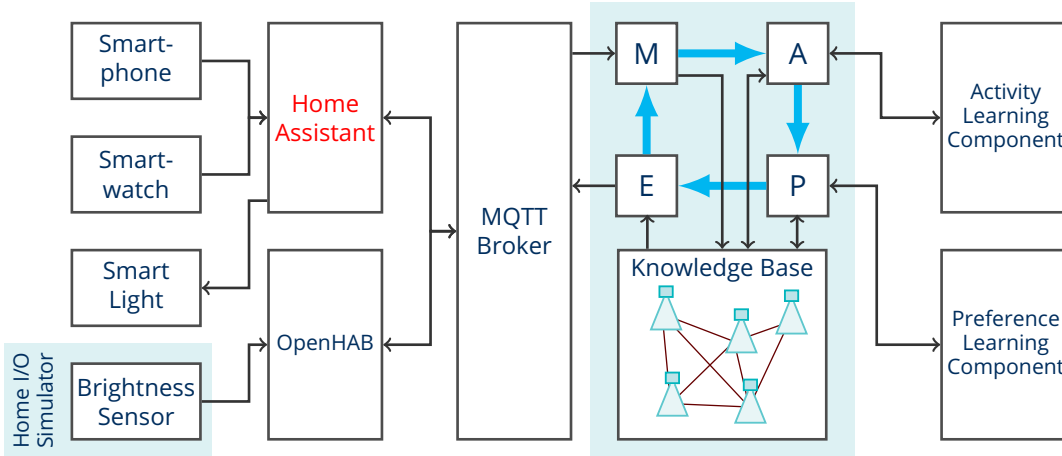
Solution: A Structured Model



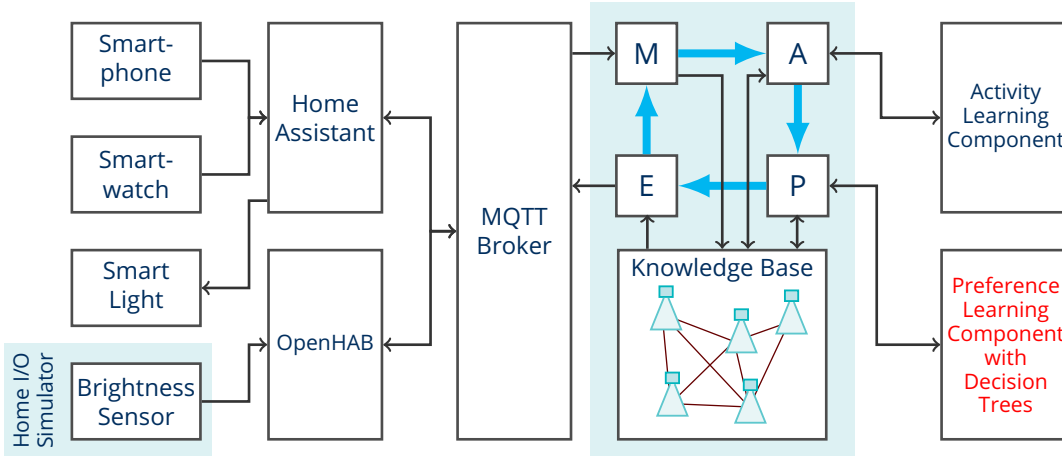
Solution: A Structured Model



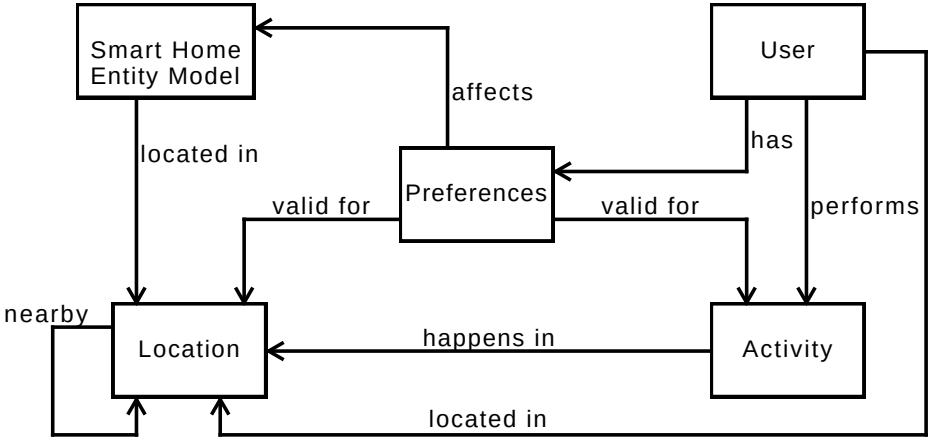
Solution: A Structured Model



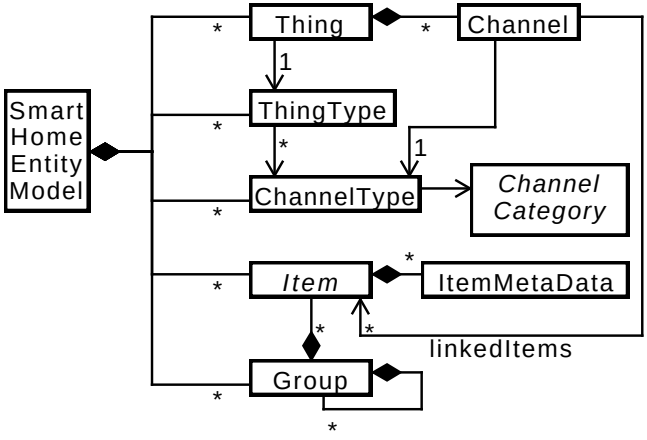
Solution: A Structured Model



The Five Aspects (to be) Modelled with RAGs



The Five Aspects (to be) Modelled with RAGs: Smart Home Entities



The Five Aspects (to be) Modelled with RAGs: Smart Home Entities

```
1  SmartHomeEntityModel ::= Thing* Group* ThingType* ChannelType* ChannelCategory*
           ItemCategory* /ActivityItem:Item/ ;
2  ThingType : DescribableModelElement ::= Parameter* ;
3  rel ThingType.ChannelType* -> ChannelType ;
4  abstract Item : LabelledModelElement ::= <_fetched_data:boolean>
           MetaData:ItemMetaData* [ItemObserver] ;
5  rel Item.Category? -> ItemCategory ;
6  rel Item.Controlling* <-> Item.ControlledBy* ;
7
8  abstract ItemWithBooleanState : Item ::= <_state:boolean> ;
9  abstract ItemWithStringState : Item ::= <_state:String> ;
10 abstract ItemWithDoubleState : Item ::= <_state:double> ;
11 ColorItem : Item ::= <_state:TupleHSB> ;
12 DateTimeItem : Item ::= <_state:Instant> ;
13 ContactItem : ItemWithBooleanState ;
```

Architecture Details (1/2)

🌲 Modelled using Relational RAGs

- Declarative specification of structure, and its relations
- Analyses on this structure
- Definition of relations 👇

```
rel MachineLearningModel.RelevantItem* <-> Item.RelevantInMachineLearningModel* ;
rel MachineLearningModel.TargetItem* <-> Item.TargetInMachineLearningModel* ;
```

Architecture Details (1/2)

🌲 Modelled using Relational RAGs

- Declarative specification of structure, and its relations
- Analyses on this structure
- Definition of relations 👇

```
rel MachineLearningModel.RelevantItem* <-> Item.RelevantInMachineLearningModel* ;
rel MachineLearningModel.TargetItem* <-> Item.TargetInMachineLearningModel* ;
```

🔌 Adapter to Middleware Platforms

- Claim: Model is general enough to map concepts of every middleware
- Currently implemented for openHAB

Architecture Details (2/2)

🔄 MAPE-K and Integration with Machine Learning

- Underlying adaptation mechanism is MAPE
- Activity recognition used in Analyse Phase to check current activity
- Preference learning used in Plan to get preferences to be applied

Architecture Details (2/2)

MAPE-K and Integration with Machine Learning

- Underlying adaptation mechanism is MAPE
- Activity recognition used in Analyse Phase to check current activity
- Preference learning used in Plan to get preferences to be applied

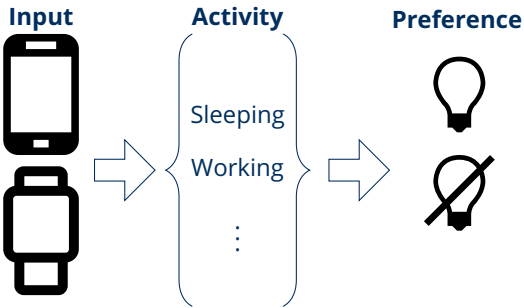
Adapter to Machine Learning

- Claim: Model is general enough to map concepts to machine learning
- Currently implemented for Neural Networks and Decision Trees

Evaluation (1/2)

💻 Setup

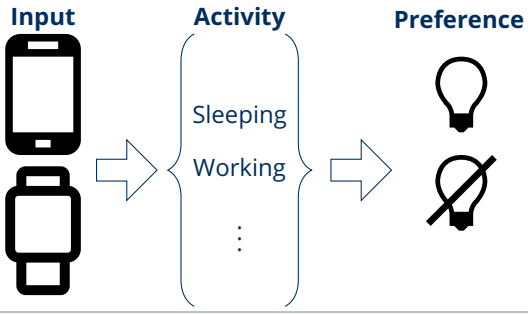
— Activity recognition based on rotation and acceleration data from phone and watch
— Resulting activity plus brightness is input for preference learning



Evaluation (1/2)

💻 Setup

— Activity recognition based on rotation and acceleration data from phone and watch
— Resulting activity plus brightness is input for preference learning



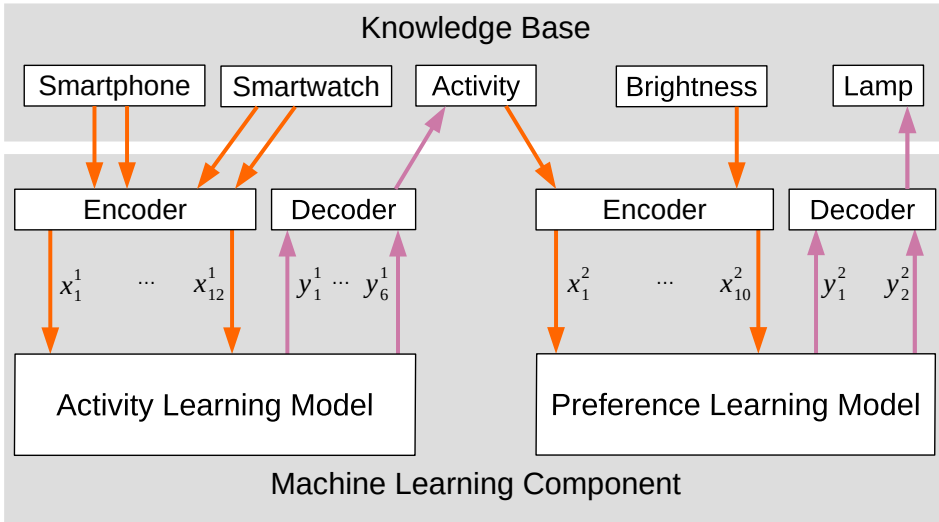
Adapters

Connecting Knowledge Base and Machine Learning Models

Encoder to transform entity states to input variables

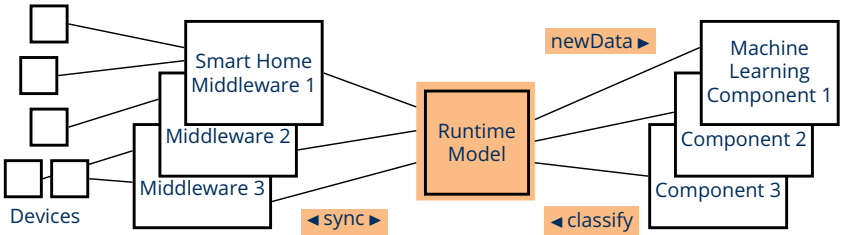
Decoder to interpret classification result

Evaluation (2/2)



Contributions

- Definition of structured model to describe smart home entities and related aspects
- Integration of model in Smart Home Middleware: Prototype using openHAB
- Integration of model with Machine Learning Algorithms: Prototype using Neural Networks
- DSL to load and serialize runtime model



Future Work

↔ Further decouple tasks of encoder and decoder

Two tasks: Feature selection and mapping between models

💥 Handle Interference of preferences of multiple users

Define order for application of preferences

📱 Challenges in user experience

Design UI explaining system behaviour, capture intend of user changes

✍ Specifications and proof of invariant

Use knowledge base to ensure consistent, predictable behaviour

TECHNISCHE UNIVERSITÄT DRESDEN

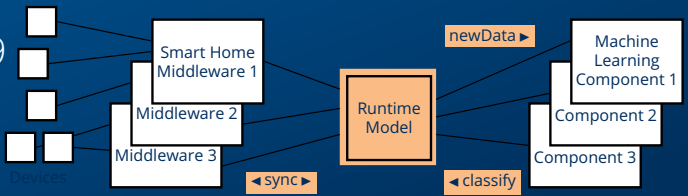
DRESDEN concept

René Schöne, Johannes Mey, Boqi Ren and Uwe Aßmann

Bridging the Gap between Smart Home Platforms and Machine Learning using Relational Reference Attribute Grammars

Models@run.time 2019

September 17, 2019



https://git-st.inf.tu-dresden.de/OpenLicht/eraser

References

Gartner, Inc. (2017).
Leading the IoT: Gartner Insights on How to Lead in a Connected World.
Technical report.

Jordan, M. I. and Mitchell, T. M. (2015).
Machine learning: Trends, perspectives, and prospects.
Science, 349(6245):255–260.