# An Ontology-based Approach to Adaptive Data Processing

Haokun Chen[1,3], Xiaowang Zhang[1,3,*], and Zhiyong Feng[2,3]

[1] School of Computer Science and Technology, Tianjin University, Tianjin, China
[2] School of Computer Software, Tianjin University, Tianjin, China
[3] Tianjin Key Laboratory of Cognitive Computing and Application, Tianjin, China
[*] Corresponding author.

**Abstract.** In this paper, we present an ontology-based approach to generate workflows which are core in adaptive data processing by taking advantage of ontologies in characterizing implicit relations among tasks of data processing. Moreover, compared with manual configurations of current approaches, ontology reasoning can automatically infer preference orders of tasks and overcome the limitation of manual configurations. Experimental results show that our proposal is effective and efficient.

## 1 Introduction

Adaptive data processing, as an advanced automatic data processing, can select models and parameters by a system itself when processing variable data from applications [1]. The core problem of adaptive data processing is to design a "smart" mechanism in generating dynamically optimal workflows for variable requirements. There are some existsing approaches to generate workflows, which are mostly based on manual configurations, such as Apache Oozie [5]. They are often taxing and poor in reusability due to the limitation of single developer. It becomes interesting in generating workflows to support adaptive data processing.

In this paper, we present an ontology-based approach to generating workflows for adaptive data processing overcome the limitation of manual configurations. In our proposal, ontologies integrated with SWRL rules [2] are applied to characterize implicit relations among tasks of data processing and ontology reasoning can automatically infer preference orders of tasks. Experimental results show that our proposal is effective and efficient. For instance, in the New York Citibike, our approach can generate an optimal workflow (shown in Fig. 2) where optimal models and parameters can be selected.

## 2 Ontology Construction for Generating Workflows

The first step of our proposal is constructing ontologies in Protégé [4]. In the following, we introduce the four steps of ontology construction.

*Class Definition* There are three classes, namely, *ML_Process, ML_Process_Service*, and *Parameters*. Each of them has some subclasses.

- The *ML_Process* class represents the whole procedure of data processing and it contains four main subclasses, namely, *Data_Preprocess*, *Feature_Preprocess*, *Modeling*, and *Evaluation*.
- The *ML_Process_Service* class represents a set of all services and it contains four subclasses, namely, *Data_Preprocess_Service*, *Feature_Preprocess_Service*, *Modeling_Service*, and *Evaluation_Service*.
- The *Parameters* class represents a set of all parameters and it contains many subclasses of parameters w.r.t. variable models.

*Property Definition* There are two kinds of properties, namely, *object* (relations among classes) and *data* (relations between entities and datatype). There are 14 object properties as well as 7 data properties.

Note that *getModelRequirement* and *getParameters* contain many subproperties which are used to select models and parameters. Besides, three subproperties *hasC*, *hasGamma* and *hasClass_Weight* of *getParameters* are used to select parameters which are important to SVM.

*Entity Definition* According to a specific application scenario, we create some entities as normal [6], and these entities can be used with some rules to do some reasoning to get more facts about the specific application scenarios. The usage of the entity is shown at Section 3 clearly.

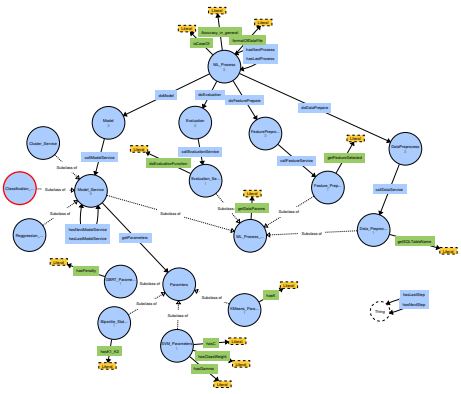

**Fig. 1.** An example of ontology for Citi-bike prediction.

*Rule Definition* We adopte SWRL [2] that is intended to be the rule language of semantic web to express the statements that can not be achieved with OWL[2]. Some SWRL rules generate the workflow according to the user requirements and the properties of the data set. Part of the rules are shown in Section 3.

*Ontology Construction* Finally, we apply Protégé [4] in constructing ontology and employ VOWL[3] to visualize the ontology schema shown in Fig. 1.

## 3 Experiments and evaluations

*New York Citi-bike parking quanlity predication* (www.citibikenyc.com/system-data): As the rents/returns of bikes at different stations in different periods are unbalanced, it is interesting to predict the number of bike parking spots to be rent from/returened to each station cluster.

**Table 1.** An example of SWRL rules for Citi-bike parking quantity predication

| | |
|---|---|
| Rule 1 | ML_Process(?a) –><br>doDataPrepare(?a,datapreparation), doFeaturePrepare(?a,featurepreparation),<br>doModel(?a,model), doEvaluation(?a,evaluation), hasNextStep(model,evaluation),<br>hasNextStep(datapreparation,featurepreparation),<br>hasNextStep(featurepreparation,model). |
| Rule 2 | accuracy_in_general(regression_service1,?a), equal(?a,true)–><br>get(regression_service1,svm_service), getParameters(svm_service,svm_parameters). |

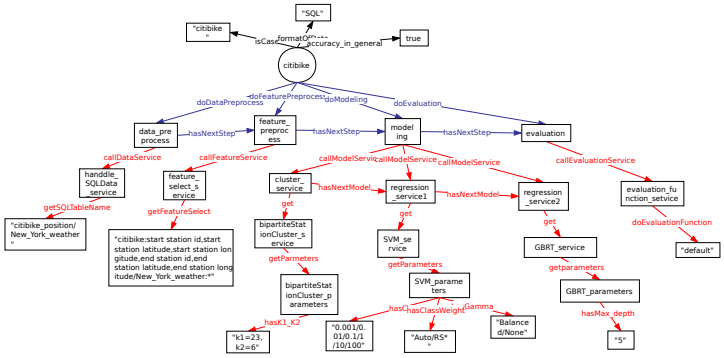


**Fig. 2.** An example of workflow for Citi-bike parking quantity predication

There are 3 steps to generate an optimal workflow as follows:

Step 1 Instantiating a Citi-bike parking quanlity predication, named *citibike*, which have three properties, namely, *isCase* (to a string "*citibike*"), *formatOfData* (to a string "*SQL*"), and *accuracy_in_general* (to a boolean value "*ture*") shown in Fig. 2 as well as SWRL rules (e.g., rules in Table 1).

Step 2 Generating a workflow by ontology reasoning via Protégé as follows: *data_preprocess* → *feature_preprocess* → *modeling* → *evaluation* shown Fig. 2.

Step 3 Executing the workflow through properties in red from *data_preprocess* to *feature_preprocess*, *modeling*, and *evaluation*.

As we can see, the results of Citi-bike parking quanlity predication via the generated workflow are the same as the results via optimally manual configurations shown in Fig. 3.

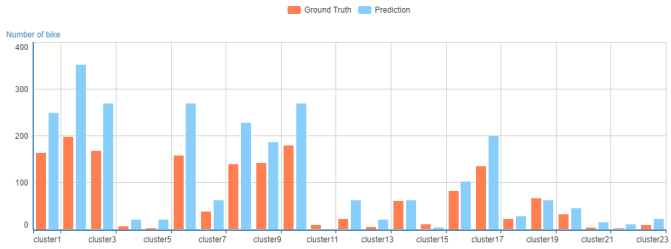


**Fig. 3.** Results of Citi-bike parking quantity predication

## 4 Conclusions

In this paper, we presente an ontology-based approach to generate workflows so that optimal models and parameters can be automatically selected in data processing. Our proposal provides a novel way for adaptive data processing via ontologies and is helpful to apply ontology techniques for data processing.

## Acknowledgments

This work is supported by the Key Technology Research and Development Program of Tianjin (16YFZCGX00210), the National Natural Science Foundation of China (61502336), the National Key R&D Program of China (2016YFB1000603), and the Seed Foundation of Tianjin University (2018XZC-0016).

## References

1. Fernández-Delgado, M., Cernadas, E., Barro, S., and Gomes Amorim, D. Do we need hundreds of classifiers to solve real world classification problems. *J. Mach. Learn. Res.*, 15(1): 3133–3181, 2014.
2. Horrocks I., F. Patel-Schneider P., Boley H., Tabet S., Grosof B., and Dean M. SWRL: A semantic web rule language combining OWL and RuleML. *W3C Member submission*, 21 May 2004.
3. Lohmann S., Negru S., Haag F., Ertl T. Visualizing ontologies with VOWL. *Semantic Web*, 7(4): 399–419, 2016.
4. Musen M.A. The Protégé project: A look back and a look forward. AI Matters. Association of Computing Machinery Specific Interest Group in Artificial Intelligence, 1(4), June 2015. DOI: 10.1145/2557001.25757003.
5. Oozie: Apache workflow scheduler for Hadoop. *The Apache Software Foundation*, September, 2010. http://oozie.apache.org/
6. Pan J.Z., Vetere G., Gomez-Perez J.M., Wu H. (Eds.) Exploiting linked data and knowledge graphs in large organisations. *Springer*, 2017.