International Conference on Applied Internet and Information Technologies, 2016
DOI:10.20544/AIIT2016.28

Approach for Synonyms Detection in Conceptual Data Model

Zoltan Kazi, Ljubica Kazi, and Biljana Radulovic

University of Novi Sad, Technical faculty "Mihajlo Pupin" Zrenjanin,
Djure Djakovica bb, 23000 Zrenjanin, Serbia
zoltan.kazi@gmail.com

Abstract. Detection of synonyms in data modeling is considered as a significant problem, especially within the semantic evaluation of a conceptual data model. This paper presents an approach for synonyms detection in a system for conceptual data model semantic evaluation. It is based on automated reasoning in ontology mapping with conceptual data model with tool that formalizes ontology and conceptual data model and merges them with a set of reasoning rules. Reasoning was done with Prolog system. These rules are created for ontology–to-conceptual data model mapping, as well for synonyms extraction. Examples of testing reasoning rules are also shown in the paper.

Keywords: synonyms detection, conceptual data model, ontology, reasoning.

1. Introduction

Research in information system design evaluation has recently received considerable attention in information technology community [1]. In the field of models in information system development [2] introduces a general metrics framework related to syntax, semantic and pragmatic aspect of a model quality evaluation. A comparative analysis and categorization of many systems analysis and design methods has been presented in [3].

Data quality research [4] is related to development of methodologies, frameworks and tools for measurement and improvement of data models and data in databases. Results in this field propose frameworks that define set of quality characteristics, metrics that could measure the level of quality characteristics achievement in particular case and the set of activities to perform in aim to perform measurement and metrics data processing.

This paper presents the developed system for synonyms detection within the evaluation of conceptual data models, based on ontology mapping. In the synonyms detection, methods of the composite matching, combined with structural analysis were used.

2. Related Work

Problems of schema matching [5] could appear in applications such as schema integration, data warehouse, semantic query processing etc. The matching solutions based on comparison of names, constraints and structure could be applied to various schema types, relational model, object oriented models and conceptual data model (CDM). Shvaiko and Euzenat [6] presented techniques of schemas matching based on terminological (linguistics-based), semantic (ontology-based) and structure (elements relations-based), including a confidence measure for the level of correspondence between the matching entities.

Similarly, ontology mapping, as “the process of linking corresponding terms from different ontologies”, could be used in query answering or for navigation on the Semantic Web [7]. Authors, in paper [8], defined a matching at lexical, semantic (related to synonyms) and structural level. Kalfoglou and Schorlemmer conducted survey [9] on methods and frameworks for comparison and merging of ontologies with different, similar or equal vocabulary, as well as possibly different ontology languages and corresponding ontology tools.

3. The Proposed Approach for Synonyms Detection in CDM

The approach for synonyms detection in CDM proposes automated reasoning in mapping ontology elements with appropriate elements of conceptual data model. The basic idea is comparing elements of conceptual data model with elements of previously created ontology, which represent knowledge, i.e. semantics of the business domain. If corresponding elements of conceptual data model could be matched or mapped with elements from ontology, then conceptual data model could be considered as “semantically correct” regarding the related ontology to the appropriate extent (percentage, i.e. semantic mark).

Underlying assumptions for the proposed approach are:

– Created ontology describes the business domain of interest,
– Created ontology could be transformed to a form suitable for automated reasoning,
– Created CDM could be transformed to a form suitable for automated reasoning,
– Ontology elements and conceptual data model elements could be compared.

Possible situations in the comparison of conceptual data model elements to ontology elements are:

– CDM element is “equal” to an ontology element (“matching” of ontology element to conceptual data model element) – both elements have the same form, i.e. they are equal words,
– CDM element is “similar“ to any ontology element - CDM element is considered a “synonym” to an ontology element (“mapping” of ontology element to conceptual data model element) – both elements have similar structural characteristics (neighboring) with other elements in a way that could be considered similar, or “synonyms” and could be processed as mapping elements. Conceptual data model element is not considered as “synonym” to an ontology element,
– CDM element could not be related to any ontology element.

Automated reasoning is based on set of rules that are used for decision making and extracting elements within the comparison of ontology elements with conceptual data model elements. Automated reasoning enables extracting of matching elements, mapping elements ("synonyms"), as well as uncovered elements (not matching and not mapping elements). Uncovered elements could be:

– Elements from ontology that are not matching and not mapping with conceptual data model elements;
– Elements from conceptual data model that are not matching and not mapping with ontology elements.

The proposed approach is developed into a system presented at the component diagram at Fig. 1. The proposed system integrates using tools for ontology creation and conceptual data model creation with automated reasoning engine. The proposed system also consists of an integration tool that enables transformation of ontology and CDM into a form suitable for automated reasoning, integration of the transformed input with reasoning rules and starting the automated reasoning engine.

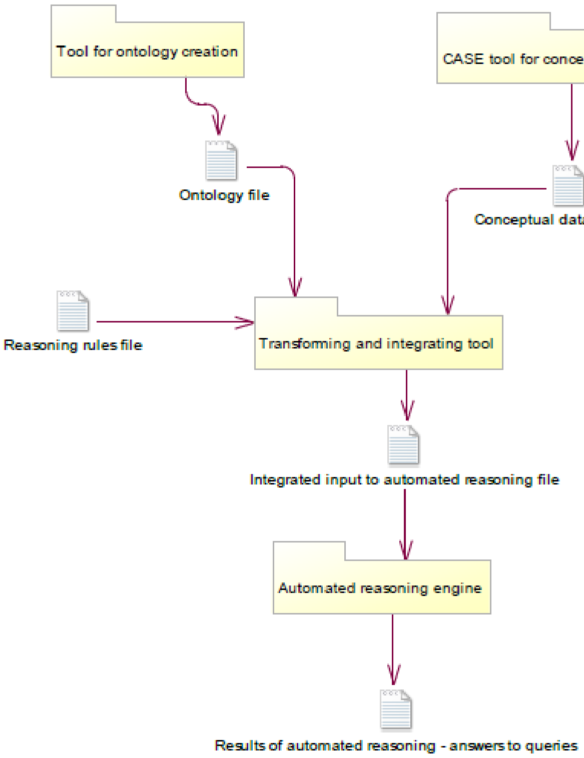

Fig. 1. Conceptual data model evaluation schema

3.1. Ontology Presentation

"Ontologies have been proposed as an important and natural means of representing real world knowledge for the development of database designs" [10]. In broader definition, ontologies are categorized as types of conceptual models [11], but more commonly used term in practice relates conceptual data models as separate concepts [12]. Practical implementation of ontologies is based on widely accepted standards such as World Wide Web Consortium standard formats such as OWL – Ontology Web Language and RDF - Resource Document Framework. Structure of ontology in OWL format consists of a collection of OWL elements [13], which could be transformed into a RDF expression. This expression is a collection of triplets: RDF (S, P, O), where: S is a subject, P is a predicate, and O is an object.

Main purpose of ontology is to capture and share knowledge in a specific domain of interest. A main characteristic of ontology is hierarchy of concepts and objects (i.e. instances of concepts established by using different semantic links). Ontology is used to describe words that represent various concepts, or can be used as taxonomy that shows how particular areas of knowledge are related. Basic ontology concepts are: classes, subclasses, properties, sub-properties, domains and ranges. Objects relations are well defined with object properties characteristics and data properties with data ranges belong to objects that are connected in specific domain. Structure of ontology consists of a collection of OWL/RDF elements, transformed into RDF expression that is accepted by World Wide Web Consortium. In the RDF form, ontology represents a collection of triplets, consisting of subject, predicate and object RDF(S, P, O), where S is subject, P is predicate and O is object.

3.2. Data Model Formalization

A data model enables representation of a real world through a set of data entities and their connections [14] that are represented in various forms: diagram (schema) with data dictionary as well as formal languages representation, such as predicate logic calculus [15]. In papers [16], [17] that are based on [18], formal presentation of a conceptual data model is extended to S = (E, A, R, C, P), where:

– E is a finite set of entities,
– A is a finite set of attributes,
– R is a finite set of relationships,
– C is a finite set of restrictions concerning attributes domains, relationships constraints, integrity rules for entities, attributes and relationships,
– P is a finite set of association rules for entities, attributes, relationships and restrictions.

3.3. Reasoning Rules

Rule 1 – extracts object properties from the ontology that are covered by relationships in the data model, but one of the entities are covered by the appropriate OWL class, while the second has a different name except object properties that are already discovered:

ontorelsinent(XC1,YOP,XC2,XE1,YR,XE2):-
rdf(XC1,type,class), rdf(XO1,classassertion,XC1),
rdf(XO1,type,namedindividual), rdf(XC2,type,class),
rdf(XO2,classassertion,XC2),rdf(XO2,type,namedindividual),
rdf(YOP,type,objectproperty), rdf(XO1,YOP,XO2),
ent(XE1),ent(XE2), rel(YR), p(XE1,YR), p(YR,XE2),
(XE1=XC1;XE2=XC2),YR=YOP,
not ontorel(XC1,YOP,XC2,XE1,YR,XE2),
not ontorelsinrel(XC1,YOP,XC2,XE1,YR,XE2). (1)

Rule 2- extracts object properties from ontology, that are not covered by relationships in the data model, but one of the entities is covered by the appropriate OWL class, while the second has a different name except object properties that are already discovered:

ontorelsinentrel(XC1,YOP,XC2,XE1,YR,XE2):-
rdf(XC1,type,class),rdf(XO1,classassertion,XC1),
rdf(XO1,type,namedindividual),rdf(XC2,type,class),
rdf(XO2,classassertion,XC2),rdf(XO2,type,namedindividual),
rdf(YOP,type,objectproperty),rdf(XO1,YOP,XO2),
ent(XE1),ent(XE2),rel(YR),p(XE1,YR),p(YR,XE2), (XE1=XC1;XE2=XC2),not ontorel(XC1,YOP,XC2,XE1,YR,XE2),
not ontorelsinrel(XC1,YOP,XC2,XE1,YR,XE2),
not ontorelsinent(XC1,YOP,XC2,XE1,YR,XE2),
not ontorelsinent2(XC1,YOP,XC2,XE1,YR,XE2). (2)

Rule 3 – extracts data properties from ontology that are not covered by attributes with equal names in the data model, but attribute data type is equal to the ontology data property range:

ontodataatribtypesin(X,Y,X1,Y1):-rdf(X,type,dataproperty),
rdf(X,range,Y),atr(X1),res(Y1),p(X1,Y1),Y=Y1,not X=X1. (3)

In previously presented rules, the special symbols represent:

Variables: X, X1, X2, XC1, XC2, XE, XE1, XE2, XC1, XC2, XE1, XE2, XO1, XO2, Y, Y1, YP, YR, YOP, and YER;

Constant values: type, class, object property, data property, range, named individual, and class assertion;

Predicates: rdf, ent, atr, rel, res, p, ontoclassent, ontoclassnoent, ontodataatrib, ontodatanoatrib, ontodataatribtype, ontocard, and ontorel.

4. The Synonyms Detection Example

Fig. 2 presents simple example of an ontology schema with basic domain ontology elements: classes, objects as class instances and relations for objects. Each object has data property with a range that defines specific datatype. These data properties are not visible on schema but exist in the ontology dictionary. The domain is related to the international scientific conference organization. The part of domain related to an author's (or participant's) submission of a paper to a conference is presented as an ontology graph. Certain details such as data properties and data types are not presented in the ontology graph.

Ontology schema on Fig. 2 shows similarities and differences. Intentional differences, for the purpose of synonyms detection, are made in the case of:

Entity type: "author" (in the ontology) vs. "participant" (in the CDM diagram).

Relationships: "attend" (in the ontology) vs. "participate" (in the CDM diagram).

Attributes: "phone" (in the ontology) vs. "phone_num" (in the CDM diagram), "email_address" (in the ontology) vs. "email" (in the CDM diagram).

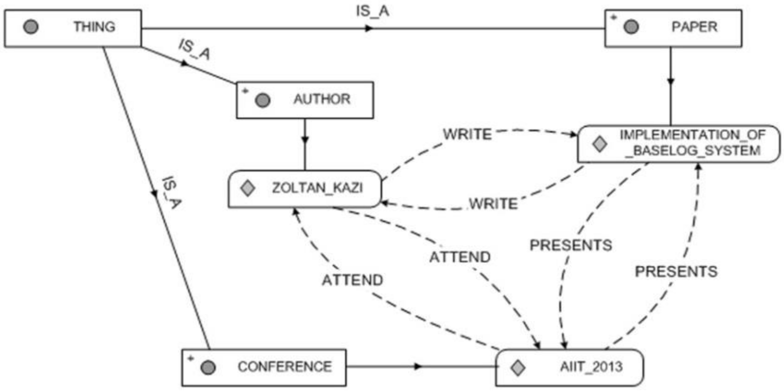

Fig. 2. Domain ontology graph

To enable synonyms detection with the previously presented reasoning rules (1)-(3) automated reasoning is performed within Prolog system. Application of rules for synonyms detection is demonstrated with previously presented examples of intentional differences of the ontology compared to the CDM model.

4.1. Testing Rules

Previously presented examples demonstrate using reasoning rules for synonyms detection upon formalized ontology and data model. Prolog detects and extracts synonyms among all elements of ontology and conceptual data model. In previous examples, it has been demonstrated that appropriate synonyms extraction rules successfully detect synonyms in entities, relationships and attributes.

Example 1 - Prolog query (upon Rule 1, (1)) and answer that extracts synonyms - ontology class "author" with data model entity type "participant".

?- ontorelsinent (XC1,YOP,XC2,XE1,YR,XE2).
XC1 = author,
YOP = write,
XC2 = paper,
XE1 = participant,
YR = write,
XE2 = paper.

Example 2 - Prolog query (upon 2, (2)) and answer that extracts synonyms - ontology object property "attend" with data model relationship "participate".

?- ontorelsinentrel(XC1,YOP,XC2,XE1,YR,XE2).
XC1 = author,
YOP = attend,
XC2 = conference,
XE1 = write,
YR = participate,
XE2 = conference.

Example 3 - Prolog query (upon Rule 3, (3)) and answer that extracts synonyms - ontology data property "phone" with data model attribute "phone_num". It also extracts synonyms - ontology data property "email_address" with data model attribute "email".

?- ontodataatribtypesin(X,Y,X1,Y1).
X = phone,
Y = string,
X1 = phone_num,
Y1 = string,
X = email_address,
Y = string,
X1 = email,
Y1 = string.

5. Conclusion

This approach enables processing of matching elements from both conceptual data model and ontology, similar elements as synonyms and uncovered elements. It is particularly useful in synonyms detection, which is helpful in overall semantic evaluation of conceptual data model. Advantage of this system is also in externally stored reasoning rules that could be enhanced, changed or added, according to the needs of particular type of data model, enabling evaluation of different data model types and modeling technology, i.e. file formats.

The proposed approach is particularly applicable in situations where single ontology is created as a basis for evaluation of a group of conceptual models with the same semantics. The particular focus in this research was on synonyms detection. Automated reasoning rules for synonyms extraction were presented, as well as an example of their usage with a simple part of domain related to international conference organization.

6. References

1. Wang, S., & Wang, H. Towards Innovative Design Research in Information Systems, Journal of Computer Information Systems, Vol. 51, No. 1, pp. 11-18, (2010).
2. Van Belle, J.P. A Framework for the Evaluation of Business Models and its Empirical Validation, The Electronic Journal Information Systems Evaluation 9(1): 31-44, (2006).
3. Siau, K., & Rossi, M. Evaluation techniques for systems analysis and design modelling methods – a review and comparative analysis, Info Systems J 21, pp. 249–268, (2011).
4. Batini, C., & Scannapieco, M. "Data Quality", Springer, (2006).
5. Rahm, E., & Bernstein, P. A Survey of Approaches to Automatic Schema Matching. The VLDB Journal, vol. 10, pp. 334–350, (2001).
6. Shvaiko, P., & Euzenat, J. A Survey of Schema-Based Matching Approaches, Journal on Data Semantics, vol. IV, (2005).
7. Fernandez, M., Gomez-Perez, A., & Juristo, N. Methontology. From Ontological Art Towards Ontological Engineering. AAAI Technical Report SS-97-06, (1997).
8. Trojahn, C., Moraes, M., Quaresma, P., & Vieira, R.A. Cooperative Approach For Composite Ontology Mapping. Journal of Data Semantics, Lecture Notes in Computer Science 4900, Springer, (2008).
9. Kalfoglou, Y., & Schorlemmer, M. Ontology Mapping: The State of the Art, The Knowledge Engineering Review Journal, vol. 1, pp. 1-31, (2003).
10. Sugumaran, V., & Storey, V.C. Ontologies for Conceptual Modeling: Their Creation, Use, and Management. Data & Knowledge Engineering, vol. 3, pp. 251–271, (2002).
11. Spaccapietra, S., March, S.T., & Kambayashi, Y. Conceptual Modeling. In: Proceedings of 21st International Conference on Conceptual Modeling, Tampere, Finland, (2002).
12. Weber, R. Conceptual Modeling and Ontology Possibilities and Pitfalls. In: Proceedings of the 21st International Conference on Conceptual Modeling, Spaccapietra S, March ST, Kambayashi Y, editors. Lecture Notes in Computer Science, vol. 2503, 1–2, Springer-VerlagBerlin, Germany, (2002).
13. Benslimane, S., Malki, M., & Bouchiha, Dj. Deriving Conceptual Schema From Domain Ontology: A Web Application Reverse Engineering Approach,. The International Arab Journal of Information Technology, vol. 2, pp. 167-176, (2010).
14. Chen, P. The Entity-Relationship Data Model – Toward a Unified View of Data, ACM Transactions on Database Systems, vol. 1, no 1, pp. 9-36. (1976).
15. Elmasri, R., & Navathe, S.B. Fundamentals of Database Systems, Addison Wesley, (2007).
16. Kazi, Lj., Kaz,i Z., Radulovic, B., & Letic, D. Using Automated Reasoning System for Data Model Evaluation, In Proc. International Symposium on Intelligent Systems and Informatics SISY 2010, Subotica, Serbia, ISBN 978-1-4244-7394-6; (8): 522-527, (2010).
17. Kazi, Z., Radulovic, B., Kazi, Lj., & Bhat, M. Ontology-based System for Conceptual Data Model Evaluation, The International Arab Journal of Information Technology, vol. 13, no. 5, (2016).
18. Emer, M.C., Vergilio, S.R., & Jino, M. Testing Relational Database Schemas with Alternative Instance Analysis, 20th International conference on software engineering and knowledge engineering SEKE, San Francisco, (2008).