

Reusability ontology in business processes with similarity matching

Meida Cahyo Untoro^{a,1,*}, Riyanarto Sarno^{a,2}, Nurul Fajrin Ariyani^{a,3}

^a Informatics Department, Intitut Teknik Sepuluh Nopember Surabaya, Indonesia

¹ meida16@mhs.if.its.com; ² riyanarto@if.its.ac.id; ³ nurulfajrin@if.its.com

* corresponding author

ABSTRACT

The working technology will provide information and knowledge. Information and technology can be developed in various ways, by reusing the technologies. In this study modeled the ontology of SOPs using protégé. Ontology will be matched between ontology A and B to obtain similarity and reuse ontology to create a more optimal ontology. Matching is a matching process between both ontologies to get the same value from both ontologies. Jaro-Winkler distance is used to find commonality between ontology. The result of the Jaro-Winkler distance has a value of 0 and 1, in matching will be obtained value close to 0 or 1. On matching ontology obtained two tests using 40% SPARQL query. In the test it uses Jaro-Winkler distance with a value of 0.67. This research yields matching value between ontology A and ontology B which is the same so that reuse ontology can be done for better ontology.

Keywords:

Matching
Ontology
Query
Reuse
SOP
SPARQL.

I. Introduction

Technology that works with the maximum will provide new information and knowledge. The results of the technology can be optimized again by way of reuse technology. In doing the reuse can use the ontology to group the components contained in any technology that will be reused. Ontology has the properties and interrelations of the entities with each other, and provides an abstract model of events and gives an explanation of a model. The Protégé application is used to create a model and scheme on a properly created ontology according to an existing event or phenomenon. Ontology must be made objectively, and the relationship between components and consistent according to the model and its definition. This study will discuss the reuse ontology of the Standard Operating Procedure (SOP) of a registrar model which will be searched for which components can be reused to create a new ontology. In the development of reuse ontology, we will gain new knowledge again. From the SOP analysis will be obtained some of the problems and development that must be completed and addressed to get a new ontology model to build a better ontology.

In the SOP ontology modeling will be obtained knowledge that is formalized into protégé. Ontology will provide meta data from models already created in protégé to be able to analyze which components can be reused. Reusing an effective and efficient knowledge will provide a model for the development of a new ontology [2]. Ontology can be used as a model of knowledge engineering, to solve problems residing in SOPs, and can be used as knowledge to be reused in other SOPs. In the making of ontology, it takes more time and cost compared to making a formal SOP model, in addition to ontology [3]. Ontology modeling is done in a certain language and is automatic, the integrity between ontology with abstract model so that various knowledge can be applied and combined into a new model. Ontology is able to adjust itself based on data changes in the model. The data contained in the model is a collection of attributes contained in the SOP.

In the development of ontology technology can be associated with the use of the web or better known as semantic web. At this stage provides the development of ontology. In semantic web ontology is a delineation between entities or components that provide knowledge. Thus semantic web is an application that has a certain knowledge base [4]. With semantic web, data with HTML format will be processed by the machine by formatting the data type, so the machine can do processing, gathering, and understand the relationship between components. This is because the semantic web uses Extensible Markup Language (XML), Resource Description Framework (RDF) and Ontology



Web Language (OWL). Semantic web can indeed understand human language through insert, but semantic web is not an artificial intelligent system (AI) [5]. In XML has a namespace scheme to integrate between semantic web with XML. RDF and RDF-schema are used to create statements about objects with URIs. Ontology can be reused with some of the components in the ontology model. The abundance of semantic heterogeneity on ontology, will be a problem in ontology.

In this research, the ontology model is made based on SOP that has been made and determined, where the ontology development is done. The purpose of this study is to provide information and knowledge based on the problems that occur. Based on the making of SOP with ontology model will be done matching process between ontology A and B to get more effective ontology, optimal, and better reuse process identifiers that utilize computation with protégé.

II. Related work

A. *Ontology*

Ontology is one of the formal conceptual and certain knowledge possessed by a group of people to process automatically [1][6]. Ontology is the theory of the meaning of an object, belongs to an object, and the object relation that may occur in the realm of knowledge. Ontology is very important because it can be used to explain the structure of a discipline. Ontology has a component that is used and has limits from conceptualization [7]. In other words, ontology is a concept that systematically explains everything that exists. Ontology is the definition of basic understanding and vocabulary of a domain using the combination of terms and relations [8]. In the field of ontology Artificial Intelligence (AI) has two related terms. The first ontology is a vocabulary representation that is often devoted to a particular subject or subject. Secondly as a body of knowledge to explain a particular topic of Figure 1. Ontology is explained by using concept notation, examples, relations, functions, and axioms [1]. The world wide web consortium makes an ontology language that can be used for knowledge representation. The ontology language, designed for the purpose of processing knowledge and information, is called Ontology Web Language (OWL). OWL has been formulated by World Wide Web Consortium (W3C) in making semantic web ontology [20]. The programming language used is using Extensible Markup Language (XML).

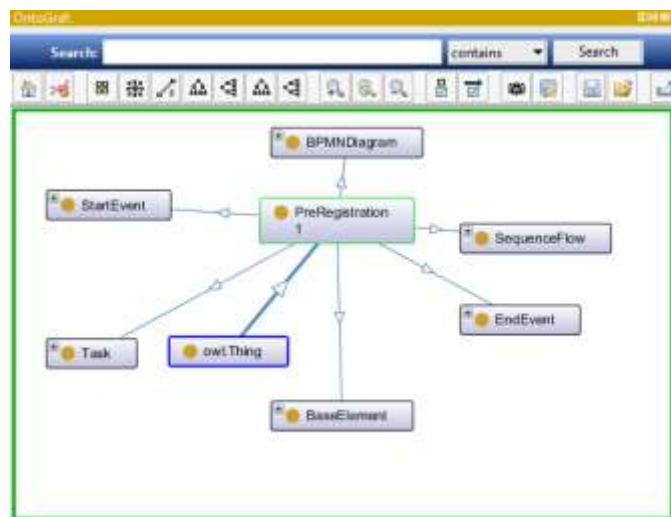


Fig. 1. Example ontology

B. *Protege*

Protégé is an ontology-based knowledge processing software. Tools can be used by knowledge specialists with the aim of designing and building an ontology, modeling the acquisition of knowledge acquisition, and entering the knowledge domain [9]. Protégé is able to visualize subclass relationships within trees, supporting the development of various drops and roots in the class hierarchy that make up "THING".

Protégé can provide an integrated knowledge base concept procession, and can alter the visual appearance of the environment by extending the system architecture to make basic knowledge modeling simple and easy Figure 2.

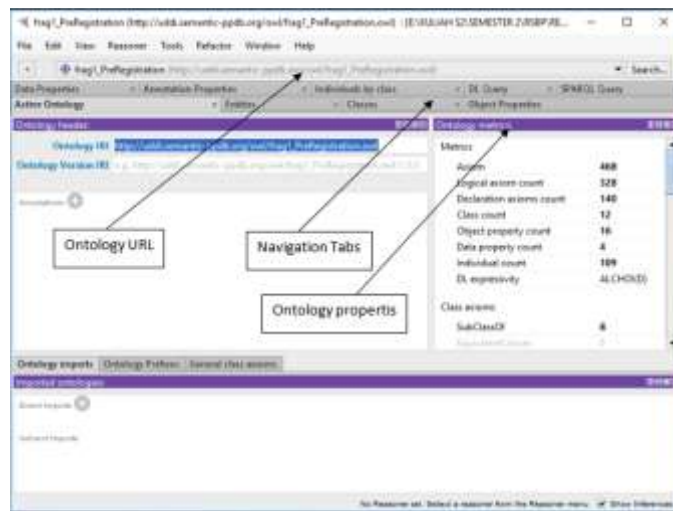


Fig. 2. Model ontology using protégé

C. Sparql

The making of Semantic Web is possible with a set of standards coordinated by the World Wide Web Consortium (W3C). The most important standards in building the Semantic Web are XML, XML Schema, RDF, OWL, and SPARQL [10][11][12]. SPARQL is a query language used to display data from RDF [13][14]. Queries used in SPARQL use URLs to execute or search for data structures RDF or OWL (hazber2015). SPARQL query language can be implemented on ontology by using RDF and OWL schemes [20]. A query that uses SPARQL can consist of "triple patterns", "or", and "and" [13]. In making the query in SPARQL the writing must be clear, in writing a clear query will get a query result in the form of a table presented by protégé in this study. In Figure 3 the following query will display the subject and object that has an ontology model with protégé. Obtained 7 subjects and 7 objects in accordance with the desired query.

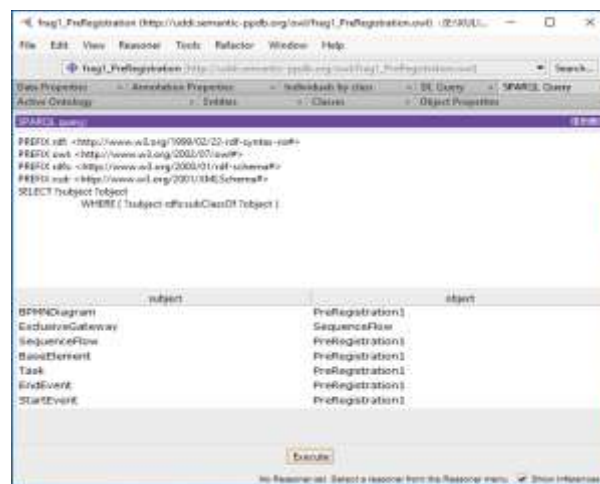


Fig. 3. Example query SPARQL

D. Matching ontology

Matching the state of the relationship between two ontologies to find the similarity between ontology and reduce the problem of heterogeneities [15]. Ontology matching can be illustrated as matching both O and O' ontology. The semantic similarity between concepts considers proper, partial, and synonymous matching [21]. For example, matching by training, and testing will be done to match the ontology with parameters as the parameters used are in the Figure 4. There are several ontology matching techniques that are used in solving the heterogeneous problems such as, terminological-

based techniques, structure-based techniques, example-based techniques, and semantic-based techniques.

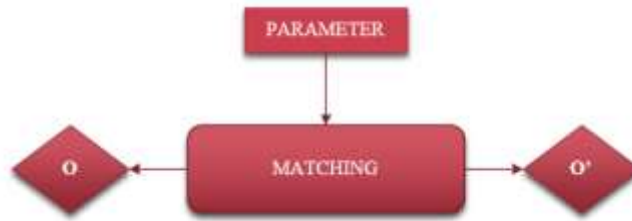


Fig. 4. Matching ontology with parameter

III. Proposed method

A. Matching ontology

The ontology determination as the basis for handling the reuse is investigated to produce a better ontology. The ontology used has some differences in the SOP used in making the ontology model with protégé will result in OWL file type. In making the ontology model must be in accordance with the SOP shown on the Figure 5, 6 and Figure 7 that have been modeled on protégé have differences which will be matched whichever component of the same ontology. Ontology matching will result in a new and reusable ontology optimal.

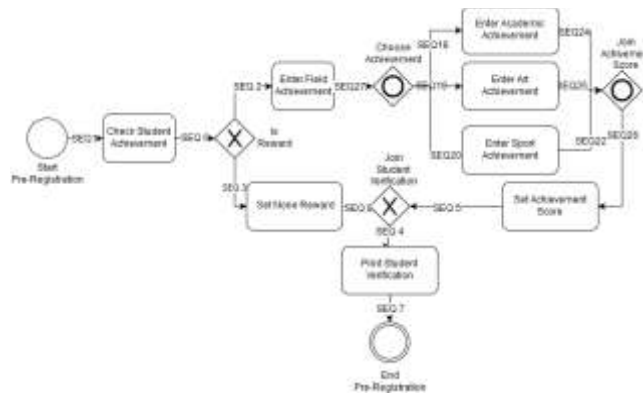


Fig. 5. SOP registration

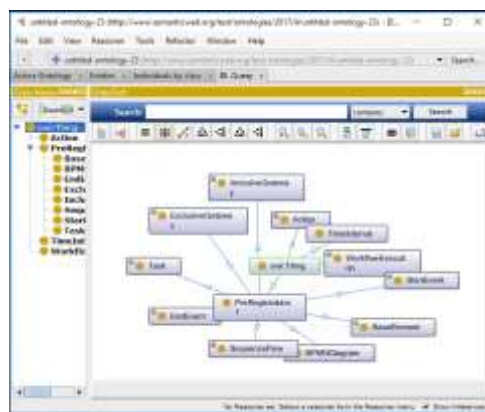


Fig. 6. Model ontology A

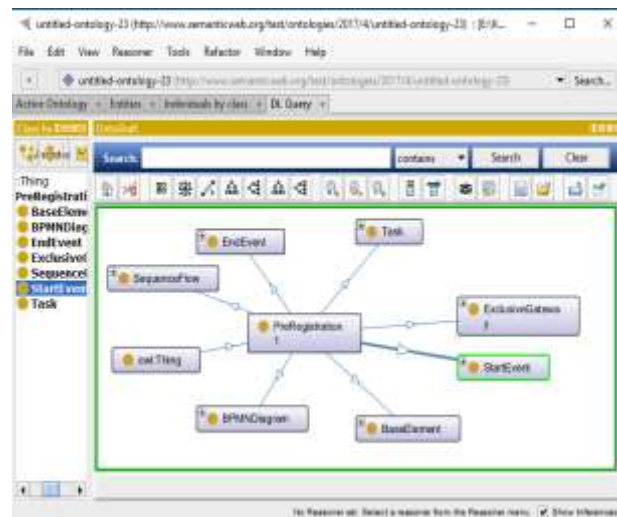


Fig. 7. Model ontology B

B. Jaro-Winkler distance

Jaro-Winkler distance is an algorithm for measuring the similarities between strings with each other. This algorithm is used in matching the same string. At Jaro-Winkler distance approaches 1 for string matching the string is almost the same, and if the Jaro-Winkler value is close to 0 then there is no similarity between strings [16][17]. Graph 1 provides information that Jaro-Winkler shows matching results between strings with a resemblance of 0.5 - 0.6 [17]. The Jaro-Winkler distance algorithm has an effective quadratic runtime complexity time complexity on string and maximum [18]. Jaro-Winkler distance will calculate the length of the string, the number of characters the same, and the number of transpositions on the string. In Jaro-Winkler distance can be seen in Equation 1. [19].

$$Jaro(S_1, S_2) = \frac{1}{3} \times \left(\frac{m}{|S_1|} + \frac{m}{|S_2|} + \frac{m - t}{m} \right) \tag{1}$$

Jaro-Winkler distance has string value 1 (| S1 |), string 2 (| S2 |), character similarity (m), and number of transposition (t). Usually S_1 is used as a reference for transposition searching. The intended transposition for the similarity of character strings to be compared. For example, in comparing the word 'StartEvent' with 'EndEvent', when viewed directly it can be said that all characters in S1 are almost identical to characters in S2, have different character sequences. Algorithm 1 Provides an explanation for finding the similarity of characters in strings residing on an ontology model built on SOP.

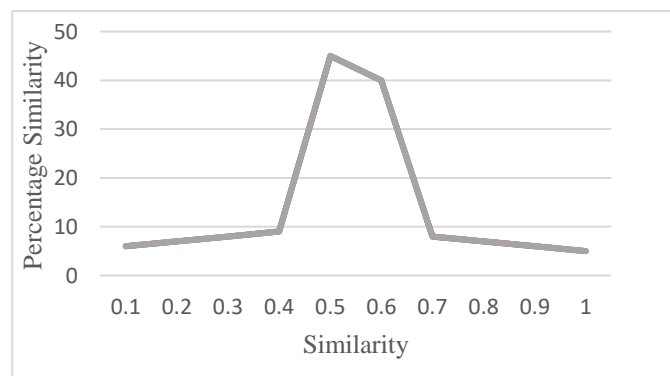


Fig. 8. Similarity matching Jaro-Winkler distance

IV. Experiment and Discussion

Reuse ontology is done to find out ontology that is currently used can be optimally re-developed from various ontology in reuse. In the first stage it aims to compare the results of ontology using SPARQL queries on protégé between ontology A and ontology B. The matching of ontology from individual classes will have to match any object properties in ontology A and B can be seen in Figure 9 and Figure 10.

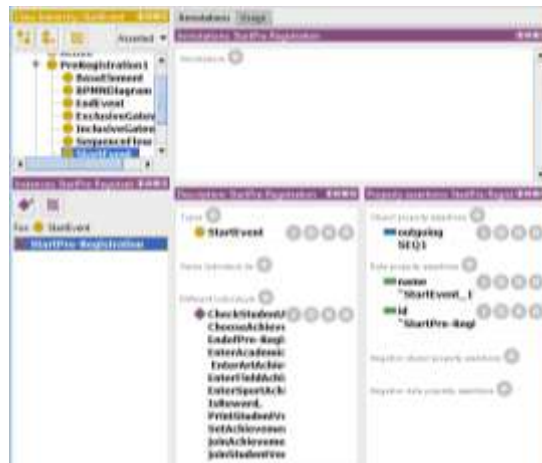


Fig. 9. Model ontology A with component

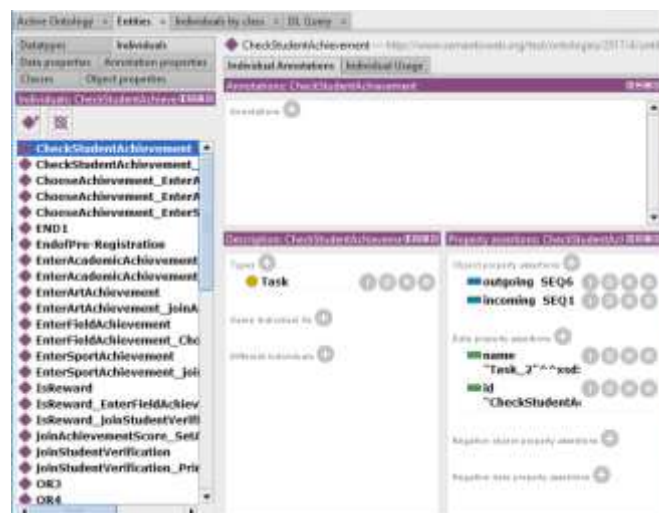


Fig. 10. Model ontology B with component

The result of matching ontology process using SPARQL query shows the similarity between ontology A and B. Matching for checking subject and object globally on ontology modeled in accordance with SOP in Figure 11 and Figure 12. In matching similarity on ontology with SPARQL has similarity of 40%.

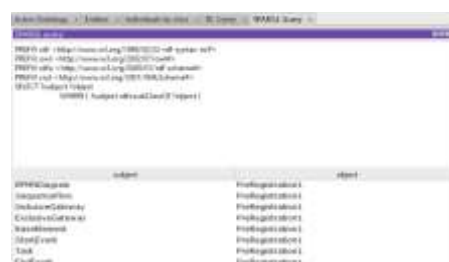
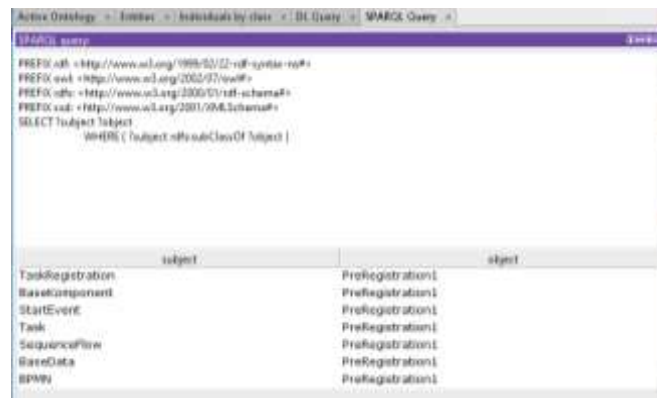


Fig. 11. Result query SPARQL ontology A



subject	object
TaskRegistration	PreRegistration()
BaseComponent	PreRegistration()
StateEvent	PreRegistration()
Task	PreRegistration()
SequenceFlow	PreRegistration()
StateData	PreRegistration()
BPMN	PreRegistration()

Fig. 12. Result query SPARQL ontology B

The next stage will be done matching ontology by using Jaro-Winkler distance in this process obtained value match in ontology. In the matching process has a value of 0.67 between ontology A and B received on Figure 13.

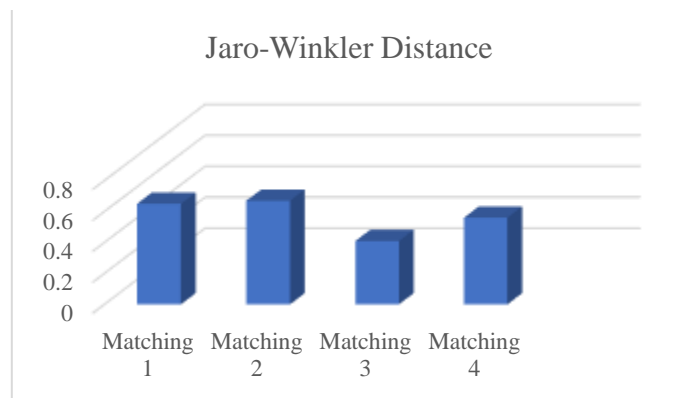


Fig. 13. Matching results Jaro-Winkler distance

After the prose matching done obtained maximum results in accordance with the data and SOP. By using SPARQL query and Jaro-Winkler distance ontology A and B have similarities between components and reuse ontology can be done to make a more optimal ontology and in accordance with the SOP.

V. Conclusion

In this research, the modeling of ontology should be based on the SOPs established first. Ontology that has been modeled will have a component of knowledge on every business process. In matching process matching between ontology A and B to get the similarity made reuse ontology. The SPARQL query gives a value of 40% of the entire component on the ontology to be reused. Jaro-Winkler distance has a value of matching process of 0.67, the matching process performed gives maximum results if the value is above 0.5 and close to 1 [17]. In the matching process will be obtained similarity between ontology can be reused to make more optimal ontology again in accordance with SOP.

Acknowledgment (*HEADING 5*)

On this occasion the author expressed gratitude profusely and highest appreciation to Prof. Ir. Drs.Ec. Riyanarto Sarno, M.Sc., Ph.D., Nurul Fajrin Ariyani, S.Kom.,Msc., Mastuti Widianingsih, S.Si.,M.Sc., and friend at Institut Teknologi Sepuluh Nopember, on the sidelines of the routine but still took the time to provide guidance, encouragement, advice and guidance from the research plan until the completion of writing paper. The author always thanks to Allah SWT who have the privilege and the pleasure of giving all large, well favors faith, health and strength in the manufacture of paper.

References

- [1] Arwan, A., Sidiq, M., Priyambadha, B., Kristianto, H., & Sarno, R. (2013). Ontology and semantic matching for diabetic food recommendations. *International Conference on Information Technology and Electrical Engineering: "Intelligent and Green Technologies for Sustainable Development"*, ICITEE 2013. Doi.org/10.1109/ICITEED.2013.6676233.
- [2] Caldarola, E. G., Rinaldi, A. M., Picariello, A., (2015). An Approach to Ontology Integration for Ontology Reuse in Knowledge Based Digital Ecosystems. *Conference: The 7th International ACM Conference on Management of computational and collective Intelligence in Digital EcoSystems (ACM MEDES'15)*. Doi.org/ 10.1145/2857218.2857219.
- [3] Fernández-López, M., Gómez-Pérez, A., & Suárez-Figueroa, M. C. (2013). Methodological guidelines for reusing general ontologies. *Data and Knowledge Engineering*. Doi.org/10.1016/j.datak.2013.03.006.
- [4] Tudorache, T., Nyulas, C., Noy, N. F., & Musen, M. A. (2013). WebProtege: A collaborative ontology editor and knowledge acquisition tool for the web. *Semantic Web*. Doi.org/10.3233/SW-2012-0057.
- [5] Jain, V., & Singh, M. (2013). Ontology Based Information Retrieval in Semantic Web: A Survey. *International Journal of Information Technology and Computer Science*. Doi.org/10.5815/ijitcs.2013.10.06.
- [6] Kunaefi, A., & Sarno, R. (2013). Ontology Mapping for Erp Business Process. *Seminar Nasional Teknologi Informasi Dan Multimedia*. Ojs.amikom.ac.id/index.php/semnasteknomedia/article/view/616.
- [7] Gandon, F. L. (2013). Ontologies in Computer Science. *Ontology Theory, Management and Design*. Doi.org/10.4018/978-1-61520-859-3.ch001.
- [8] Hayuhardhika, W., Putra, N., Sugiyanto, Sarno, R., & Sidiq, M. (2013). Weighted Ontology and weighted tree similarity algorithm for diagnosing Diabetes Mellitus. *International Conference on Computer, Control, Informatics and Its Applications: "Recent Challenges in Computer, Control and Informatics"*, IC3INA. Doi.org/10.1109/IC3INA.2013.6819185.
- [9] D. Corsar and D. Sleeman, "Reusing JessTab rules in Protégé," *AI 2005 SI*. Doi.org/10.1016/j.knosys.2005.11.010.
- [10] Hazber, M. A. G., Li, R., Gu, X., Xu, G., & Li, Y. (2015). Semantic SPARQL Query in a Relational Database Based on Ontology Construction. *2015 11th International Conference on Semantics, Knowledge and Grids (SKG)*. Doi.org/10.1109/SKG.2015.14.
- [11] Kurz, T., Schlegel, K., & Kosch, P. H. (2015). Enabling Access to Linked Media with SPARQL-MM. Doi.org/10.1145/2740908.2742914.
- [12] Meditskos, G., Dasiopoulou, S., & Kompatsiaris, I. (2016). MetaQ: A knowledge-driven framework for context-aware activity recognition combining SPARQL and OWL 2 activity patterns. *Pervasive and Mobile Computing*. Doi.org/10.1016/j.pmcj.2015.01.007.
- [13] Chiba, H., & Uchiyama, I. (2017). SPANG: a SPARQL client supporting generation and reuse of queries for distributed RDF databases. *BMC Bioinformatics*. Doi.org/10.1186/s12859-017-1531-1.
- [14] Xiang, Z., Courtot, M., Brinkman, R. R., Ruttenger, A., & He, Y. (2010). OntoFox: web-based support for ontology reuse. *BMC Research Notes*. Doi.org/10.1186/1756-0500-3-175.
- [15] Didih Rizki Chandranegara, Riyanarto Sarno. (2016). Ontology Alignment using Combined Similarity Method and Matching Method. *Informatics and Computing (ICIC), International Conference on*. DOI:10.1109/IAC.2016.7905722.
- [16] Sun, Y. (2015). A Comparative Evaluation of String Similarity Metrics for Ontology Alignment. *Journal of Information and Computational Science*. Doi.org/10.12733/jics20105420.
- [17] Zarembo, I., Teilans, A., Rausis, A., & Buls, J. (2015). Assessment of name based algorithms for land administration ontology matching. *Procedia Computer Science*. Doi.org/10.1016/j.procs.2014.12.008.
- [18] Drebler, K., & Ngomo, A. C. N. (2017). On the efficient execution of bounded Jaro-Winkler distances. *Semantic Web*. Doi.org/10.3233/SW-150209.
- [19] Maree, M., & Belkhatir, M. (2015). Addressing semantic heterogeneity through multiple knowledge base assisted merging of domain-specific ontologies. *Knowledge-Based Systems*. Doi.org/10.1016/j.knosys.2014.10.001.
- [20] www.w3.org/standards/semanticweb/ontology.
- [21] Euzenat, J., Shavaiko, P. (2010). *Ontology matching second Edition*. [Dl.acm.org/citation.cfm?id=1951780](http://dl.acm.org/citation.cfm?id=1951780)