A tool for semantic data interoperability

Allan Patrick F. Santana[1] **, Maximilian Harrisson C. Junior**[1] **, Bruno Lopes**[1]

[1]Instituto de Computação
Universidade Federal Fluminense
Niterói-RJ – Brazil

{allanpatrick,mharrisson}@id.uff.br,bruno@ic.uff.br

Abstract. *Over the years, the amount of information available to the public by digital means is growing more and more, however, this growth is done in a fragmented way by several companies. Integrating these data may lead to inconsistencies, and system errors. This integration may be difficult and expensive for companies. Data interoperability is done when data with distinct formats or from multiple sources are processed to generate unified data. This work presents a tool to make automatic alignment of data from multiple sources, integrating with the SUMO ontology so that inferences can be made with the tool*

1. Introduction

Companies and governmental agencies use various systems, but these systems usually do not interact, which may lead to data inconsistency. It is also expected that the teams in a company do not share all of their data. Data interoperability may be seen as the process of gathering data from multiple sources or with distinct formats so that it can be processed into unified and simple data for the end user or a way to read multiple groups of data in a form that the end user does not know the topology of the data. In data interoperability, establishing the semantics of the terms is a problem because different systems may use the same word or expression with different meanings or different words may denote the same things. This problem requires techniques to detect these inconsistencies in the semantics avoiding issues with how the system processes the data.

This work presents a microservice-based tool to align multiple systems and generate an integration with the SUMO Ontology [Software 2000]. In this application, we provide different services for each type of alignment (based on an architecture developed in the context of the project Interopera-PDPA / Prefeitura de Niterói-RJ); considering the terminology alignment, we employ the text distance, synonym comparison, and translation services, and for the entity alignment we have all the other services with the addition of the Deep Matcher and the Exact services. After the alignment finishes the user can utilize the tool to make inferences in the SUMO Ontology and in other formats as well like logical inferences and queries in the SQL format.

An Ontology [Studer et al. 1998] is a formal and well-defined model of knowledge. This model of knowledge has representations of the data like the relations, attributes, and properties that encompass that model of knowledge. There are many reasoners capable to make inferences on ontologies, such as FACT++ [Tsarkov and Horrocks 2006] and HERMIT [Glimm et al. 2014]. They may be used in the future for consistency checks and data inference.

This tool was developed in the context of the project Interopera-PDPA and is being used by the city of Niterói-RJ, Brazil. The source code is available at `https://github.com/frame-lab/interoperaNit`.

This paper is organized as follows. Section 2 describes related works of this project, Section 3 presents the algorithm that was developed, Section 4 shows how the algorithm is executed, Section 5 shows the conclusion and the directions this work will follow.

2. Related work

A large number of works have generated tools to improve data interoperability in many fields, such as [Blanc et al. 2004], which propose a tool to connect multiple services from different modeling tools. Also, [Yang and Zhang 2006] provides a tool to effectively generate, manage and reuse semantic interoperable building objects in design applications.

In the medical field, there is a great concern about how information can be better distributed, and studies such as [Jaleel et al. 2020] that provide a framework for the medical devices providing services like registration, subscribing, probing translation, and publishing of data. [Catley and Frize 2002] proposes a standards-compliant medical infrastructure based in XML to integrate all of their decision support tools. In [Khan et al. 2014], it is proposed an adaptative mediation engine called ARIEN, which arbitrates between the support system of hospitals to create an environment to exchange information.

Aiming to solve some interoperability problems, ontology solutions compose a powerful set of tools. Many tools were developed over the years, such as [Usadel et al. 2006] which presents an interactive ontology tool called PageMan that generate, displays, and annotates overview graphs for profiling experiments. The work of [Zeeberg et al. 2005] extended the GoMiner tool to make it work with microarrays in which it generates a map of relation with the GO ontology.

In other works, such as [Clair et al. 2019] the authors present Lipid Mini-on, an open-source tool that analyses and provides visualizations of lipid molecules data and permits the users to conduct an analysis direct from the lipid ontology. For the last example, [Carvalho et al. 2008] discusses the implementation of a probabilistic ontology tool with the problems they found and how they addressed these problems. [Poveda-Villalón et al. 2014] presents a catalog of pitfalls for ontologies and a tool for detecting pitfalls in ontologies.

This work is different from the others because it proposes a tool to make automatic alignment of databases by implementing a microservice approach in which the user chooses the best services for their databases, making the application easily expansive and simple to use. It also relies on a GUI to simplify the user experience and generates multiple outputs making the tool easily linkable with other projects.

3. Automatic alignment

The tool is structured in 4 stages: preparation, parameter matching, entity matching, and post-processing. In this work, we will focus on the last stage and how it interacts with the ontology.

In the preparation stage, the user needs to explicitly define the knowledge that is obvious in the data like unique fields, fields that need to be approximated, queries that will run at the end of the process, value separators, and the services that will be executed. In the parameter matching stage, for each service selected in the preparation, the tool searches the matches of the parameters of each database that was provided by the user, that need to pass a threshold in the service that is making the alignment.

When the parameter matching finishes, the entity matching stage begins by searching for matches among them. Two entities compose a match on the databases when they are a match for all parameters that are a match in these databases. The entity alignment is made in an exact manner by default unless the user chooses to use the approximate alignment. This alignment has the same services as the parameter matching with the addition of the Exact and Deep Matcher services.

The post-processing phase begins with the tool generating the output files aligned with the user databases, then the group branches in three paths. The first path provides a means to further enhance the aligned result by running another process of aligning the entities with the machine learning algorithm. The second path provides means of making inferences in the aligned databases by allowing the user to make queries using logic operators and in the SQL language. The third step takes the aligned database and makes another output with a map of the words used in the Suggested Upper Merged Ontology (SUMO) and uses this map to link the databases with the SUMO ontology through a parser of the SUMO ontology after that the user can then make inferences about their databases.

The Suggested Upper Merged Ontology [Software 2000] (SUMO) is currently the largest free ontology available. It is a high-level ontology that covers a wide range of content, from philosophy and mathematics to science, technology, and everyday affairs. SUMO represents its structure from simple and complex concepts, where the complex concepts are composed of constructions on simpler concepts, generating a hierarchy of concepts. All entries are mapped with the WordNet relation that can be subsumed, equivalent, or an instance. For example, “code” has an equivalent map with “ComputerProgram.”

The parser is generated by reading every concept present in the SUMO ontology and parsing them into a tree structure with the concept hierarchy. After the concepts are finished then we parser and add the formulas to the tree. Then we use the tree to link to the concepts that were found in the data that the user provided. We reserve all words from functions and parameter names, and from that, we make the inference parser.

To make the map the tool searches for all parameters found in the aligned databases and with it makes a relation of the parameter with a term in the SUMO ontology. For every term mapped in this way, an instance of the entities related to the parameter that was mapped is added to the term in the ontology. After the instances were added a relation is created between them to simulate the relation of the entity.

With the databases mapped to the ontology, the user can take all the reasoning power provided by the SUMO ontology, thus granting the user the possibility to make inferences that were previously unavailable in the databases. Some basic inferences that can be done in the ontology are listed as follows.

- Retrieve an entity - instance $\times$ entity
- Retrieve details of a term - instance $\times$ term
- Retrieve every entity with a property $\times$ hasProperty some property

4. Execution of the algorithm

This section will show an example of the full execution of the algorithm with Tables 1 and 2. For this execution, we will use the example of a city that wants to make a vaccination campaign for its inhabitants, but the tables with the people that have been vaccinated and the addresses of the people are in two distinct databases, so they will use our tool to make the alignment of the data. These tables describe 2 distinct representations of a person database. In the first table, we have the parameters Name, ID, and Street, and in the second table, we have the parameters Surname, IDS, and Salary. For our example, the city knows that the ID fields are a unique representation of their inhabitants and will use it to guide the alignment.

T1		
Name	ID	Street
Harry	95858370812	Abner Street
Jorge	85376152622	Mcalpin Street
Francisco	42597155061	Merry Road
Pedro	80221767372	New Pine Road
Paulo	24331605550	Orchid Street

Table 1. Input source 1

T2		
Surname	IDs	Salary
Smith	94051264050	$R\$1500$
Johnson	62154477602	$R\$1500$
Williams	37783561552	$R\$1500$
Brown	95858370812	$R\$1500$
Jones	42597155061	$R\$1500$

Table 2. Input source 2

The objective of this alignment will be to unify the two tables using the ID field as the focus of the alignment. To do that the user can choose an exact service to make the alignment because the ID is unique for every person, so an approximate approach will not be a good option. After the services that will be used have been chosen the next step is to separate the persons that have been vaccinated and to do that the user can choose three ways to do a SQL query, do a Boolean query or make an inference in the updated ontology. The queries can be executed following the commands of the lists 1, 2 respectively.

Listing 1. SQL query

```
SELECT *
```

```
FROM Bigbase
Where T1_ID == T2_IDs
```

Listing 2. Boolean query

```
T1_ID == T2_IDs
```

The preliminary result of the alignment will be the Bigbase table 3, it shows the full alignment of the tables because no queries or inferences are utilized in this process. For every parameter that didn't have a match a null value will be assigned and for the matched entities it will show the values of the entities that were a match.

Bigbase					
T1_Name	T1_ID	T1_Street	T2_Surname	T2_IDs	T2_Salary
Harry	95858370812	Abner Street	Brown	95858370812	$R\$1500$
Jorge	85376152622	Mcalpin Street	null	null	null
Francisco	42597155061	Merry Road	Jones	42597155061	$R\$1500$
Pedro	80221767372	New Pine Road	null	null	null
Paulo	24331605550	Orchid Street	null	null	null
null	null	null	Smith	94051264050	$R\$1500$
null	null	null	Johnson	62154477602	$R\$1500$
null	null	null	Williams	37783561552	$R\$1500$

Table 3. Result of the alignment

For every query made a new table 4 will be generated, showing the validations that were passed in this case the table will only be showing the persons Harry and Francisco because they are the only persons that were vaccinated in this context.

Query 1					
T1_Name	T1_ID	T1_Street	T2_Surname	T2_IDs	T2_Salary
Harry	95858370812	Abner Street	Brown	95858370812	$R\$1500$
Francisco	42597155061	Merry Road	Jones	42597155061	$R\$1500$

Table 4. Result of applying the query

After that, an alignment is made using WordNet, and the resulting Table (see Table 5) provides all of the mapped relations. Notice that "IDs" is not mapped so is not presented in the graph. In this example, ID is related to Idaho due to its acronym in the USA (clearly not the meaning in this example and must be treated as an error/limitation).

WordNet alignment					
word	translated_word	code	WordNet synonym	relation	file
ID	ID	09081213	Idaho	equivalent	nouns
IDs	IDs	null	null	null	null
Name	Name	06333653	Name	equivalent	noun
Salary	Salary	13279262	CurrencyMeasure	subsumed	noun
Street	Street	14485811	Sub. Asses. Attribute	subsumed	noun
Surname	Surname	06336904	Name	subsumed	noun

Table 5. Alignment of WordNet and SUMO terms

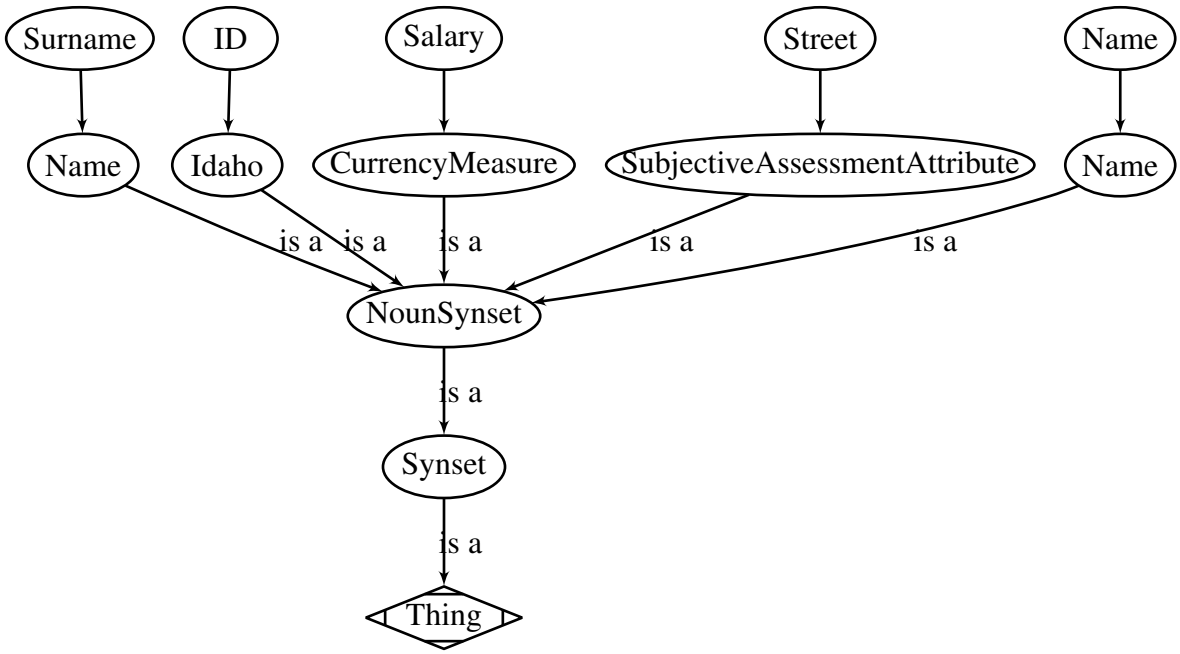

Figure 1. WordNet alignment graph

In the end, the parser reads and generates a tree for each SUMO file. A tree is composed of instances from SUMO, and they are associated with the mapped terms from the previous stage.

After all the processes were made the aligned data will be linked with the ontology, and the user will now have some options. If he is unsatisfied with his aligned data he could redo the steps with other services or he could also run the Deep Matcher service to further improve his alignment. He could export the aligned data or some of the queries made to other tools that accept the CSV format. Lastly, the user could make inferences in the linked ontology to also get the list of vaccinated persons or use the information that is already present in the ontology to make further inferences about his data, for example, using the government context present in the SUMO to make further inferences in the data.

5. Conclusion and future works

The amount of information in the world has been growing considerably, however, this growth is divided into several systems that are generally unrelated making interoperability among them a difficult task. This fact ends up generating a growing demand for interoperability. This work provides a tool to make automatic alignment of diverse data.

This work presented a tool to make the alignment of data based on a microservice architecture. The inputs are tables that are processed by the services chosen by the user. The aligned database can be used to make further improvements in the alignment with Deep Matcher or used to make inferences on the provided data, by using the link with the SUMO ontology thus providing the user with new ways to interact with their data in the already established data present in the SUMO. Also, the user can make inferences in the logic and SQL formats or export their data in the CSV format.

This tool may be used to generate an automatic alignment of data relying on microservices in which it generates an aligned database and an integration with the SUMO ontology. The implementation is available at `https://github.com/frame-lab/interoperaNit` and the focus of the next steps in the development of the tool is to include new services, the addition of new inputs formats, the possibility to use the ontology for consistency checks, show the SUMO functions that are already present in the ontology, make the ontology exportable to other tools and make the ontology retain the new data between uses.

References

Blanc, X., Gervais, M.-P., and Sriplakich, P. (2004). Model bus: Towards the interoperability of modelling tools. In *Model driven architecture*, pages 17–32. Springer.

Carvalho, R. N., Ladeira, M., Santos, L. L., Matsumoto, S., and Costa, P. C. (2008). Unbbayes-mebn: comments on implementing a probabilistic ontology tool. In *IADIS International Conference Applied Computing 2008*, pages 211–218.

Catley, C. and Frize, M. (2002). Design of a health care architecture for medical data interoperability and application integration. In *Proceedings of the Second Joint 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society] [Engineering in Medicine and Biology*, volume 3, pages 1952–1953 vol.3.

Clair, G., Reehl, S., Stratton, K. G., Monroe, M. E., Tfaily, M. M., Ansong, C., and Kyle, J. E. (2019). Lipid mini-on: mining and ontology tool for enrichment analysis of lipidomic data. *Bioinformatics*, 35(21):4507–4508.

Glimm, B., Horrocks, I., Motik, B., Stoilos, G., and Wang, Z. (2014). Hermit: An owl 2 reasoner. *J. Autom. Reason.*, 53(3):245–269.

Jaleel, A., Mahmood, T., Hassan, M. A., Bano, G., and Khurshid, S. K. (2020). Towards medical data interoperability through collaboration of healthcare devices. *IEEE Access*, 8:132302–132319.

Khan, W. A., Khattak, A. M., Hussain, M., Amin, M. B., Afzal, M., Nugent, C., and Lee, S. (2014). An adaptive semantic based mediation system for data interoperability among health information systems. *Journal of medical systems*, 38(8):1–18.

Poveda-Villalón, M., Gómez-Pérez, A., and Suárez-Figueroa, M. C. (2014). Oops!(ontology pitfall scanner!): An on-line tool for ontology evaluation. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 10(2):7–34.

Software, A. (2000). The suggested upper merged ontology (sumo).

Studer, R., Benjamins, V., and Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 25(1):161–197.

Tsarkov, D. and Horrocks, I. (2006). Fact++ description logic reasoner: System description. In Furbach, U. and Shankar, N., editors, *Automated Reasoning*, pages 292–297, Berlin, Heidelberg. Springer Berlin Heidelberg.

Usadel, B., Nagel, A., Steinhauser, D., Gibon, Y., Bläsing, O. E., Redestig, H., Sreenivasulu, N., Krall, L., Hannah, M. A., Poree, F., et al. (2006). Pageman: an interactive ontology tool to generate, display, and annotate overview graphs for profiling experiments. *BMC bioinformatics*, 7(1):1–8.

Yang, Q. and Zhang, Y. (2006). Semantic interoperability in building design: Methods and tools. *Computer-Aided Design*, 38(10):1099–1112.

Zeeberg, B. R., Qin, H., Narasimhan, S., Sunshine, M., Cao, H., Kane, D. W., Reimers, M., Stephens, R. M., Bryant, D., Burt, S. K., et al. (2005). High-throughput gominer, an'industrial-strength'integrative gene ontology tool for interpretation of multiple-microarray experiments, with application to studies of common variable immune deficiency (cvid). *BMC bioinformatics*, 6(1):1–18.