UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

Helping scientists integrate and interact with biomedical data

Ana Rita Pereira Guerreiro

Mestrado em Bioinformática e Biologia Computacional

Dissertação orientada por:
Prof.ª Doutora Cátia Luísa Santana Calisto Pesquita
Prof. Doutor Daniel Pedro de Jesus Faria

2021

Acknowledgements

First, I would like to thank my supervisors, Professor Cátia Pesquita and Professor Daniel Faria, for guiding me throughout this year and for always being available to help me. Their mentoring, support, dedication, and patience were crucial for this dissertation and I am truly grateful to them.

Secondly, I want to thank my colleagues from LiSeDa who have always been available to help me, give me feedback, and discuss ideas. I would like to give special thanks to Rita Sousa who was always available to help and give me wise advice.

To all my friends, especially Bárbara Almeida and Carolina Gonçalves, for always being by my side during this phase and helping me move forward.

Last but not least, I want to thank my family, especially my parents, for always believing in me, inspiring me, and giving me all the support I needed to achieve my goals.

This work was supported by Fundação para a Ciência e a Tecnologia through the LASIGE Research Unit (UIDB/00408/2020 and UIDP/00408/2020). It was also partially supported by the KATY project which has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 101017453.

Abstract

For the past decades, the amount and complexity of biomedical data available have increased and far exceeded the human capacity to process it. To support this, knowledge graphs and ontologies have been increasingly used, allowing semantic integration of heterogeneous data within and across domains. However, the independent development of biomedical ontologies has created heterogeneity problems, with the design of ontologies with overlapping domains or significant differences.
Automated ontology alignment techniques have been developed to tackle the semantic heterogeneity problem, by establishing meaningful correspondences between entities of two ontologies. However, their performance is limited, and the alignments they produce can contain erroneous, incoherent, or missing mappings. Therefore, manual validation of automated ontology alignments remains essential to ensure their quality.
Given the complexity of the ontology matching process, is important to provide visualization and a user interface with the necessary features to support the exploration, validation, and edition of alignments. However, these aspects are often overlooked, as few alignment systems feature user interfaces enabling alignment visualization, fewer allow editing alignments, and fewer provide the functionalities needed to make the task seamless for users.
This dissertation developed VOWLMap — an extension for the standalone web application, WebVOWL — for visualizing, editing, and validating biomedical ontology alignments. This work extended the Visual Notation for OWL Ontologies (VOWL), which defines a visual representation for most language constructs of OWL, to support graphical representations of alignments and restructured WebVOWL to load and visualize alignments. VOWLMap employs modularization techniques to facilitate the visualization of large alignments, while maintaining the context of each mapping, and offers a dynamic visualization that supports interaction mechanisms, including direct interaction with and editing of graph representations. A user study was conducted to evaluate the usability and performance of VOWLMap, having obtained positive feedback with an excellent score in a standard usability questionnaire.

Keywords: Biomedical Ontologies, Ontology Alignment, Alignment Visualization, VOWL

Resumo Alargado

Nas últimas duas décadas, devido à expansão da tecnologia, a quantidade e complexidade de dados biomédicos disponíveis aumentaram, e ultrapassam em larga escala a capacidade humana de os processar. Para auxiliar no processamento destes dados, ontologias e grafos de conhecimento têm vindo a ser cada vez mais utilizados, permitindo a integração semântica de dados heterogéneos intra e inter-domínios. Contudo, o desenvolvimento e criação indiscriminada de ontologias biomédicas culminou em problemas de heterogeneidade, com a criação de ontologias dentro do mesmo domínio com diferenças significativas, comprometendo a interoperabilidade entre as mesmas.

Várias técnicas de alinhamento de ontologias automatizadas foram desenvolvidas para lidar com o problema da heterogeneidade semântica, que vieram permitir o estabelecimento de correspondências relevantes entre entidades de duas ontologias. No entanto, o desempenho dos sistemas que aplicam estas técnicas é limitado, e muitas vezes os alinhamentos produzidos podem conter correspondências erradas, incoerentes ou em falta. Desta forma, a validação de um alinhamento por utilizadores continua a ser uma parte essencial do processo de produção de alinhamentos com elevada qualidade.

Dada a complexidade das ontologias e do processo de alinhamento, a capacidade de fornecer uma interface gráfica com suporte para a visualização de um alinhamento e com os recursos necessários para apoiar a exploração, navegação e inspeção do alinhamento, é fundamental para o processo de validação de um alinhamento. Apesar da sua importância, a validação de alinhamentos por utilizadores é um aspeto tido pouco em consideração, uma vez poucos sistemas de alinhamentos apresentam uma interface com a visualização do alinhamento, muito poucos permitem que os utilizadores editem o alinhamento e menos ainda fornecem as funcionalidades necessárias para tornar a tarefa menos complicada para o utilizador, como permitir a interação com a visualização ou fornecer informações contextuais sobre cada correspondência.

O principal objetivo desta dissertação foi desenvolver uma ferramenta que permitisse ultrapassar as limitações encontradas no estado da arte para a visualização, validação e edição manual de alinhamentos de ontologias biomédicas. Desta forma, no âmbito desta dissertação, foi desenvolvida uma ferramenta web, VOWLMap, para a visualização, edição e validação de alinhamento de ontologias biomédicas. O trabalho desenvolvido implementou a *Visual Notation for OWLOntologies* (VOWL), que define uma representação visual para a maioria dos construtores da linguagem OWL, e estendeu esta notação para suportar representações gráficas de alinhamentos. Ao aplicar uma notação visual, o trabalho desenvolvido serve o propósito de facilitar a compreensão da visualização do alinhamento a utilizadores que estão menos familiarizados com os conceitos adjacentes ao processo de validação. Isto é particularmente importante no domínio biomédico, onde os utilizadores, apesar de serem especialistas no domínio, possuem poucos conhecimentos sobre ontologias e o seu formalismo.

Esta ferramenta é constituída por duas interfaces, que permitem a realização e adequação de diferentes tarefas no decorrer do processo de validação. A primeira interface é constituída por uma listagem de todos os *mappings* presentes no alinhamento, onde é possível validar cada mapping ou até adicionar novos *mappings*. Ao clicar num *mapping* em específico na lista, o utilizador é transportado para a segunda interface. Esta interface é constituída por uma visualização em grafo gerada para o respetivo *mapping*. Esta visualização é dinâmica e os utilizadores podem interagir com o grafo através de diversos mecanismos, como reposicionamento dos nós ou alteração das características da visualização. Nesta visualização, os utilizadores podem validar o *mapping* visualizado, ou *mappings* presentes na vizinhança, e a cor de cada mapping muda consoante o *status* atribuído. Uma funcionalidade adicional, não fornecida pelas visualizações dos sistemas de alinhamento existentes, é a possibilidade de editar o alinhamento através da edição direta na visualização em grafo. No VOWLMap, os utilizadores podem remover *mappings* da visualização, adicionar novos *mappings* entre entidades presentes no grafo, ou até refinar um determinado *mapping* através da mudança da entidade da ontologia *source* ou *target*. A visualização é automaticamente atualizada com as alterações feitas e o alinhamento final exportado contém tanto a informação sobre a validação de cada mapping, como as alterações que possam ter sido feitas ao alinhamento durante o processo de validação.

Para lidar com o tamanho das ontologias biomédicas e dos seus alinhamentos, o VOWLMap aplica técnicas de modularização de ontologias, de modo a facilitar a visualização de alinhamentos grandes, mantendo, ao mesmo tempo, o contexto individual de cada correspondência. Para tal, em cada visualização, é apresentada ao utilizador apenas a vizinhança do *mapping* que está a ser visualizado, sendo possível alterar a vizinhança desde 0 até a um máximo de 3 *edges* de distância.

Foi realizado um estudo observacional com utilizadores para avaliar a usabilidade e desempenho do VOWLMap para o contexto de validação de alinhamentos. Neste estudo foi pedido aos utilizadores que validassem dois alinhamentos de domínios diferentes utilizando o VOWLMap. Os resultados obtidos permitiram perceber que as tarefas de validação propostas foram realizadas com sucesso e, a maioria das novas funcionalidades implementadas para este propósito foram consideradas muito úteis pelos utilizadores. Os resultados de um questionário standard de usabilidade (SUS), indicaram uma excelente avaliação pelos utilizadores, tendo sido obtido um score de 85. Para além disso, foi possível perceber que a ferramenta funciona em todo o seu potencial na maioria dos browsers e sistemas operativos em que foi testada.

Quando comparados os sistemas de visualização de alinhamento e a ferramenta desenvolvida, foi possível verificar uma melhoria significativa, uma vez que o VOWLMap é o único dos sistemas a satisfazer a esmagadora maioria dos requerimentos necessários para visualizar e validar alinhamentos. Uma possível continuação deste trabalho seria estender a ferramenta, de modo a que, para além de suportar a visualização de *mappings* entre classes das duas ontologias, fosse possível também visualizar *mappings* entre propriedades e, consequentemente, proceder à validação de alinhamentos com este tipo de correspondências. Para além do estudo realizado, seria interessante no futuro realizar estudos comparativos com utilizadores, de modo a que fosse possível comparar a performance e a usabilidade do trabalho desenvolvido com as visualizações e interfaces dos sistemas atuais de alinhamento.

Palavras Chave: Ontologias Biomédicas, Alinhamento de Ontologias, Visualização de Alinhamentos, VOWL

Contents

List of Figures

List of Tables

Acronyms

AM AgreementMaker.

AML AgreementMakerLight.

API Application Programming Interface.

DAG Directed Acyclic Graph.

DTD Document Type Definition.

FOAF Friend of a Friend.

GO Gene Ontology.

IRI Internationalized Resource Identifier.

JSON JavaScript Object Notation.

NCIT National Cancer Institute Thesaurus.

OWL Web Ontology Language.

RDF Resource Description Framework.

SKOS Simple Knowledge Organization System.

SUS System Usability Scale.

URL Uniform Resource Locator.

VOWL Visual Notation for OWL Ontologies.

W3C World Wide Web Consortium.

XML Extensible Markup Language.

Chapter 1

Introduction

For the past two decades, due to technological advances, the amount and complexity of biomedical data available has increased, far exceeding the human capacity to process it. Therefore, it was necessary to develop systems capable of managing, integrating, and analysing all of this data in an efficient way [Hoehndorf et al., 2015].

A common strategy to help solve the heterogeneity of biomedical data involves using knowledge graphs and ontologies to describe data under a common vocabulary [Hoehndorf et al., 2015]. Ontologies provide a structured model of a domain to define, relate and share representations of biomedical knowledge [Rubin et al., 2007]. Thus, biomedical ontologies allow the semantic integration of heterogeneous data within and across domains, and the interoperability between different databases.

There are many specialists working on the development of ontologies in the biomedical domain, from biologists to clinical researchers and physicians [Rubin et al., 2007]. However, the independent development of biomedical ontologies by different groups has created heterogeneity problems, leading to the design of ontologies on the same domain with overlapping domains or significant differences, such as logical incompatibilities between them [Faria et al., 2018]. Moreover, the differences in the modeling of this domain created semantic heterogeneity between biomedical ontologies, such as completely different concepts to express the same reality [Euzenat and Shvaiko, 2007]. Therefore, to ensure interoperability between these ontologies and to support sharing and reusing of the knowledge they contain, ontology matching techniques can be applied to establish semantic relations between the entities of two ontologies [Shvaiko and Euzenat, 2013].

Various ontology matching techniques have been developed to tackle the semantic heterogeneity problem [Granitzer et al., 2010]. In most cases, matching systems apply automated algorithms to generate an alignment without any human intervention. However, due to the complexity of the matching process, the performance of these automated systems is limited, and the alignments produced by them can contain erroneous, incoherent, or missing mappings [Li et al., 2019]. For this reason, human intervention for validation of the alignments produced by these systems is indispensable to guarantee alignment quality.

User validation of an alignment allows detecting and removing erroneous mappings, and adding alternative or new ones not detected by alignment systems. Additionally, when users validate an alignment during the alignment process, their input can be leveraged by the matching systems, allowing the adjustment of system settings, the selection of the most suitable alignment algorithms, and the incorporation of user knowledge [Li et al., 2019].

Given the complexity of ontologies and the ontology matching process, an important feature of ontology matching systems is the ability to provide visualization and a user interface with the necessary features to support the exploration, validation, and edition of alignments. These features are particularly relevant in the biomedical domain, where users are often domain experts but not experts in ontologies, which can cause difficulties in the interpretation of a mapping in the context of the ontologies and their formalism (e.g. evaluating if a mapping is logically sound, given the constraints of the two ontologies) [Li et al., 2019]. However, it can be quite difficult to consider the structure and constraints of two ontologies while keeping in mind other mappings and their logical consequences without visual support.

With the increasing size and number of ontologies and alignments [Hoehndorf et al., 2015], it has become clear that comprehensive and more interactive visualizations are key features in user involvement in alignment validation, as they can provide a better understanding of the alignment and support the decision-making process [Li et al., 2019]. Nevertheless, few alignment systems provide a user interface that supports alignment visualization, editing and navigation strategies, and even fewer provide the functionalities needed to make the task seamless for the user, such as interaction with the visualization or contextual information about the mappings.

1.1 Objectives

This dissertation aims to improve user interaction for validation of biomedical knowledge graph alignments, to overcome the existing limitations of the state of the art. To this end, this work focuses on four main objectives:

1. Support visualizations of the context of mappings, i.e., the support for visualization of the entities of the two ontologies and the correspondences between them.

2. Find the right balance between informativeness and cognitive overload providing contextual information to help the decision process (such as lexical and structural information in the ontologies), but avoid overwhelming users with too much information.

3. Enable alignment editing, by providing functions to allow manual interaction and validation of the alignment.

4. Perform a user evaluation to assess the performance and usability of the proposed approaches.

1.2 Contributions

The main contributions of this dissertation can be enumerated as follows:

1. Development of a webtool that supports the visualization, edition and validation of biomedical ontologies alignments — VOWLMap.

2. Extension of the VOWL notation with additional elements for the representation of a mapping context in visualizations that employ this notation.

3. Extension of the Alignment RDF format to support the validation process.

4. User evaluation of VOWLMap, to assess the usability and performance of the developed approaches.

5. Poster in International Semantic Web Conference 2021 Posters and Demos Track titled ”VOWLMap: graph-based ontology alignment visualization and editing”.

6. Accepted article in International Semantic Web Conference 2021 workshop on Visualization and Interaction for Ontologies and Linked Data (VOILA) titled ”VOWLMap: graph-based ontology alignment visualization and editing”.

1.3 Document Structure

The rest of this document is structured in five chapters, as follows:

- **Chapter 2** (Background) defines and explains the basic concepts needed to understand the work.

- **Chapter 3** (Related Work) surveys the relevant work developed in the scope of this dissertation to this date.

- **Chapter 4** (VOWLMap) presents the proposed methodology and describes the developed tool, including its user interface and main features.

- **Chapter 5** (Evaluation) presents the evaluation of the developed tool, including the methodology employed, results and discussion.

- **Chapter 6** (Discussion) discusses the obtained results and compares VOWLMap with the state of the art covered in Chapter 2.

- **Chapter 7** (Conclusion) discusses the main conclusions of this work, and indicates some directions for future work.

Chapter 2

Background

In order to understand the challenges and requirements of ontology alignment visualization in the biomedical domain, it is necessary to formalize ontologies and the ontology matching process in the context of this domain, and introduce the concepts of interactive ontology alignment and user validation.

2.1 Biomedical Ontologies

The term *ontology* is used in different communities with different meanings. There are several differences in the description of this term between the philosophical sense, which has a well-established tradition, and the computational sense, which has emerged in the knowledge engineering community [Guarino et al., 2009].

In the context of computer science, one of the first definitions emerged in the early 1990s, by Gruber, who defined an *ontology* as "an explicit specification of a conceptualization" [Gruber, 1993]. In other words, we can define an ontology as a structured way of defining a common vocabulary for domain knowledge that allows it to be shared and reused throughout systems. Ontologies are generally structured as Directed Acyclic Graph (DAG), where the nodes are the classes or individuals, and the edges are the semantic relations between them (Figure 2.1).

Various languages have been developed to encode ontologies. The Web Ontology Language (OWL) was developed by the World Wide Web Consortium (W3C) as a language for publishing and sharing ontologies on the World Wide Web. OWL builds on Resource Description Framework (RDF) and uses RDF's XML-based syntax, and the basic representation features of OWL are based on the knowledge representation languages called Description Logic [Horrocks and Patel-Schneider, 2011].

The core components of an OWL ontology are: classes describing sets of individuals with similar characteristics (e.g. the class *Protein* refers to the set of all proteins), and can have a natural language definition and logical definitions; individuals (or instances) the basic elements of the domain and the specific members of a class, representing the application of the class with data (e.g. *yourself* is an instance of *Person*); datatypes, that describe sets of data values; and properties describing relationships between

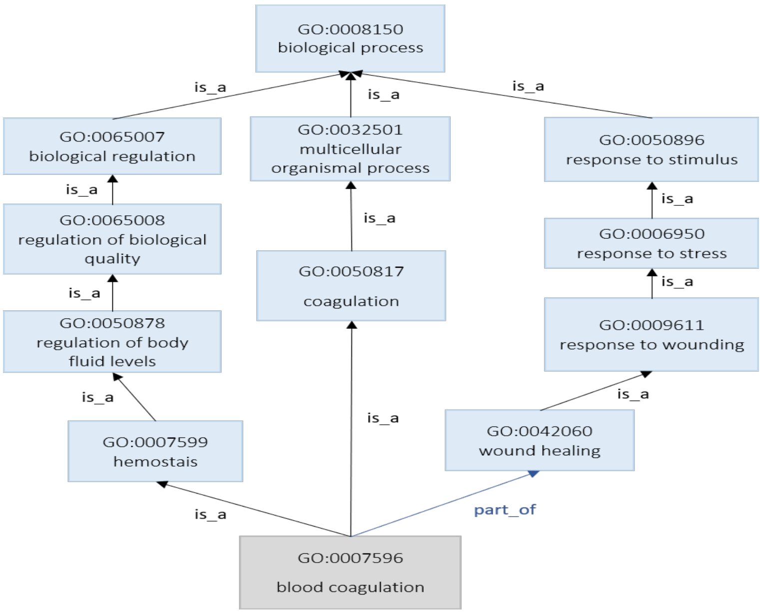


Figure 2.1: Example of a Directed Acyclic Graph (DAG) representing a subset of the Gene Ontology (GO) centered on the term GO:0007596 ”blood coagulation”. The rectangles represent the GO classes and the edges represent the relations between them (blue represents *part_of* relations and black represents *is_a* relations).

pairs of individuals; [Horrocks, 2008; Horrocks and Patel-Schneider, 2011]. In OWL the properties are divided into three types: object properties, which relate individuals to other individuals; data properties, which relate individuals to data values; and annotation properties, which are used to add metadata to entities [Horrocks and Patel-Schneider, 2011].

In OWL, an ontology consist of a set of axioms, that include both conceptual schema, describing constraints on the structure of the domain (Terminology Box), and instance level statements, asserting facts about specific entities, such as individuals (Assertion Box) [Horrocks and Patel-Schneider, 2011]. Figure 2.2 shows an example of axioms from a portion covering human anatomy of the National Cancer Institute Thesaurus (NCIT).

2.2 Ontology Matching

Across the same domain, different users may need to exchange data or to be able to access data while using different software tools. Thus, to ensure integration and interoperability between ontologies and ontology-based biomedical systems and reduce the semantic gap between ontologies of the same domain, it is necessary to find the correspondences between this information, which is commonly known as ontology matching [Shvaiko and Euzenat, 2013].

Ontology matching is the process of finding correspondences between entities of different ontologies.

```
<!-- http://human.owl#NCI_C12220 -->

<owl:Class rdf:about="http://human.owl#NCI_C12220">
    <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Lip</rdfs:label>
    <rdfs:subClassOf rdf:resource="http://human.owl#NCI_C38617"/>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="http://human.owl#UNDEFINED_part_of"/>
            <owl:someValuesFrom rdf:resource="http://human.owl#NCI_C12419"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <oboInOwl:hasDefinition rdf:resource="http://human.owl#genid7758"/>
    <oboInOwl:hasRelatedSynonym rdf:resource="http://human.owl#genid7759"/>
</owl:Class>


<!-- http://human.owl#NCI_C12221 -->

<owl:Class rdf:about="http://human.owl#NCI_C12221">
    <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">External_Upper_Lip</rdfs:label>
    <rdfs:subClassOf rdf:resource="http://human.owl#NCI_C12223"/>
    <oboInOwl:hasRelatedSynonym rdf:resource="http://human.owl#genid7733"/>
</owl:Class>
```

Figure 2.2: Example of axioms from portion covering human anatomy of the National Cancer Institute Thesaurus (NCI) in OWL format.

Euzenat and Shvaiko [2007] define the matching process as a function *f* that returns an alignment *A'* between a pair of ontologies *o* and *o'*, an optional input alignment *A*, a set of parameters *p* (e.g. thresholds, weights), and a set of external resources *r* (e.g. common knowledge):

$$A' = f(o, o', A, p, r) \tag{2.1}$$

The alignment is the output of the matching process, i.e., a set of correspondences between entities belonging to the matched ontologies [Euzenat and Shvaiko, 2007]. More formally, given two ontologies, Euzenat et al. [2011] represent a correspondence by a quintuple: *(id,e,e',r,n)* where *id* is an identifier of the given correspondence, *e* and *e'* are mapped entities of the two ontologies, *r* is the semantic relation between them and *n* is a confidence measure for the correspondence between *e* and *e'*, i.e., the degree of certainty in the correspondence.

Due to the complexity of the biomedical domain and the peculiarities of these ontologies, biomedical ontologies provide various challenges in the field of ontology matching [Faria et al., 2018; Pesquita et al., 2014]:

- **Large Size:** these ontologies often have tens of thousands of classes, and handling such large ontologies can be computationally challenging throughout the ontology matching pipeline in matching systems.
- **Rich and Complex Vocabulary:** in the biomedical domain it is common to have multiple annotations to describe each class, such as labels and different synonyms.

- **Different Modeling Views:** different groups of scientists may model the same domain differently, and biomedical ontologies for the same domain may have serious differences in organization.
- **Particular Semantics:** biomedical ontologies typically have few properties and simple semantics.
- **Multiple inheritance** or possess more than one kind of **hierarchical relation**: partonomy (defined by *part of* relations) is as important as taxonomy (defined by subclass relations, such as *is a*), as they usually complement each other.

Manual creation of mappings between concepts is excessively time consuming and infeasible for all but very small ontologies in terms of both efficiency and effectiveness. Therefore, several automated ontology matching systems have been developed to tackle the semantic heterogeneity problem [Granitzer et al., 2010; Pour et al., 2020]. However, the performance and accuracy of the automated systems are limited due to the complexity and intricacy of the process, which is determined by both the domain and the design of the ontologies. For that reason, automatic generation of mappings should be considered as a first step towards generating a final alignment, making user validation an essential step to guarantee alignment quality [Euzenat et al., 2011].

In order for the alignments produced by different means (manually or by automated systems) to be treated uniformly, Euzenat [2004] developed an alignment format and an application programming interface (API), to provide a consensual format to express an alignment. Currently, the Alignment format is expressed in RDF and its formal description can be stated as follows [Euzenat, 2004]:

- ***Alignment* element** - describes a particular alignment and contains a specification of the alignment and a list of cells:

  xml - indicates if the alignment can be read as an XML file compliant with the DTD;

  level - informs the level of the alignment, used to characterize the type of the correspondences;

  type - describes the type of the alignment;

  onto1 - the URL of the first aligned ontology;

  onto2 - the URL of the second aligned ontology;

  map - the correspondence between entities of the ontologies;

- ***Ontology* element** - provides information regarding the aligned ontologies. This element contains three attributes:
  - **rdf:about** - contains the URI identifying the ontology;
  - **location** - contains the URL corresponding to the location where it is possible to find the ontology;
  - **formalism** - describes the language in which the ontology is expressed through its name and URI;

- ***Cell* element** - describes each correspondence of the alignment. This element provides the following attributes:
  - **rdf:resource** - the URI identifying the correspondence;
  - **entity1** - the URI the first aligned ontology entity;
  - **entity2** - the URI the second aligned ontology entity;
  - **measure** - the confidence score, that is the confidence in the assertion that the relation holds between the first and second entity. The value can range from 0 to 1.
  - **relation** - the relation holding between the two entities.
- ***Relation* element** - contains the name identifying the relation between the two entities. This relation can be expressed through symbols (e.g. $=$ for equivalent or $<$ for subsumes), or through a fully qualified classname.

```
<?xml version='1.0' encoding='utf-8'?>
<rdf:RDF xmlns='http://knowledgeweb.semanticweb.org/heterogeneity/alignment'
         xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
         xmlns:xsd='http://www.w3.org/2001/XMLSchema#'
         alignmentSource='AgreementMakerLight'>

<Alignment>
        <xml>yes</xml>
        <level>0</level>
        <type>11</type>
        <onto1>http://mouse.owl</onto1>
        <onto2>http://human.owl</onto2>
        <uri1>http://mouse.owl</uri1>
        <uri2>http://human.owl</uri2>
        <map>
                <Cell>
                        <entity1 rdf:resource="http://mouse.owl#MA_0000313"/>
                        <entity2 rdf:resource="http://human.owl#NCI_C33504"/>
                        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.6040</measure>
                        <relation>=</relation>
                </Cell>
        </map>
```

Figure 2.3: Example of theAlignment RDF Format as produced with AgreementMakerLight [Euzenat, 2004].

This format as it is implemented supports extensions, such as additional string-valued qualified attributes, both on *Alignment* and *Cell* elements. Figure 2.3 shows an example illustrating the *Alignment, Cell* and *Relation* elements described above.

2.3 Interactive Ontology Matching and User Validation

An interactive ontology matching process is an ontology matching process considering the involvement of domain experts, by providing feedback in which mapping should be accepted or rejected [Pesquita et al., 2014].

Li et al. [2019] define the process of validating an alignment as having one or more users classifying mappings present in the alignment as correct or incorrect, replacing the incorrect mappings with correct alternatives, or adding new mappings. Furthermore, when users validate an alignment during the matching process, this allows the matching system settings or matching algorithms to be adjusted and user knowledge to be incorporated into these systems. Thus, this process allows the cooperation between users and automatic matchers in a reasonable amount of time and generates higher quality ontology alignments [Li et al., 2019].

Li et al. [2019] describe three distinct but interrelated categories of issues that affect the process of alignment validation:

- **User Profile:** in order to validate an alignment, users should be familiarized with the domain of ontologies, their formal representation, and the underlying point of view, before they are able to understand and decide for each mapping or even create mappings themselves. Following these principles, the three key aspects of the user profile are (1) *domain expertise*, i.e., the depth of knowledge about the domains of the matched ontologies, which can determine the user's ability to assess the conceptual correctness of a mapping; (2) *technical expertise*, i.e., the depth of knowledge engineering, modeling, ontologies, and their formalism, which can determine the user's ability to assess the formal correctness of a mapping; and (3) *alignment system expertise*, i.e., the familiarity with the alignment system, its functionality, and visual representations.

- **System Services:** the support provided by alignment systems, especially in the form of services to reduce user workload and exploit user intervention, depends on the stage of user's involvement in the alignment process - *before*, during the *matching* stage, during the *iterative* fashion or *after* the alignment.

- **User Interface:** visual support is important to provide a better understanding of the structure and constraints of the ontology and to consider different mappings and their logical consequences when validating a particular mapping. The *alignment visualization*, i.e., the visual support that the system provides to the user, and the *alignment interaction*, such as functionalities implemented to allow the user to interact with an alignment and validate it, are the two main aspects of the user interface that are determinant to the process of validation.

User validation by itself is a cognitively demanding task that requires a high memory load and complex decision-making processes, and it can be particularly challenging in the biomedical domain. For that reason, ontology matching systems should provide comprehensive and more interactive visualizations for user involvement in alignment validation, as it offers a better understanding of the alignment and supports the decision-making process.

Chapter 3

Related Work

3.1 Visualization of Ontologies

Ontologies are no longer used exclusively by ontology experts but also by non-experts in many different domains. However, non-expert users often have difficulty understanding ontologies, namely in the biomedical domain. Ontology visualization approaches can assist casual users in developing, exploring, and interacting with ontologies by providing a new perspective and filling the lack of background knowledge necessary to comprehend ontologies [Ozturk and Açikgoz, 2020].

To develop ontology visualizations, it is necessary to effectively display all the relevant information contained in the ontology and, at the same time, allow users to easily perform the desired operations on the ontology [Katifori et al., 2007]. Thus, the core components of visualization tools are: (1) the implementation of visualization methods, and (2) the set of the user interface and visual features that enrich the visualization [Dudáš et al., 2018].

According to Dudáš et al. [2018], various visualization methods for ontologies have been proposed and many software tools implementing them have been developed. They grouped these methods ass follows:

- **Indented list:** display a list of entities, where entities lower in the hierarchy are shown indented under their parent entity. However, these visualizations allow visualizing only hierarchical relationships between entities, such as *is a* relationships, and can create confusion when multiple inheritance is involved. A typical example of an indented list visualization is the Entity Browser in Protégé (Figure 3.1).

- **Graphs:** graphs (or node-link) visualization use nodes to represent entities of the ontology, such as classes, and links connecting the nodes, to represent the relationship between these entities. These visualizations can handle both hierarchical and non-hierarchical relations. Onyx [Ozturk and Açikgoz, 2020] implements this visualization method and offers a suitable environment for the

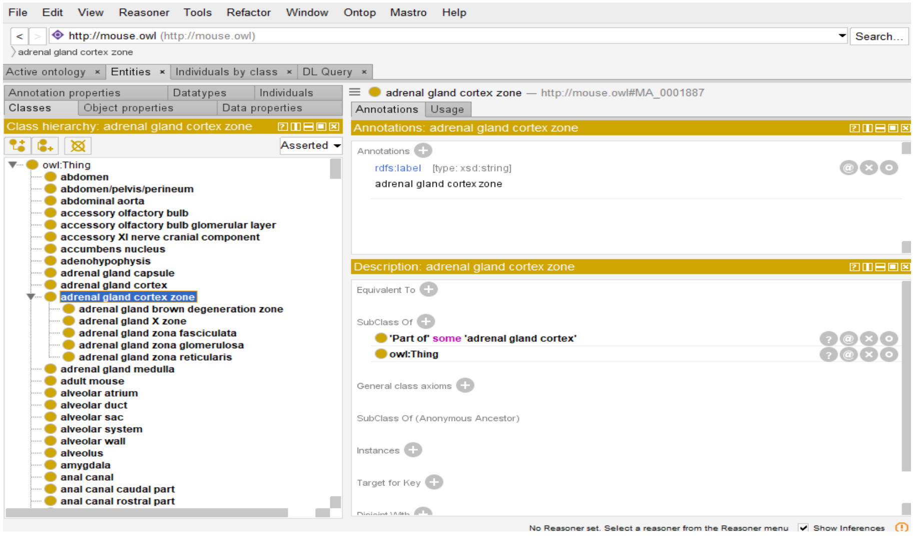


Figure 3.1: Screenshot of Entity Browser in Protégé to illustrate an indented list visualization.

.

representation of large ontologies, especially ontologies used in biomedical and health information systems (Figure 3.2).

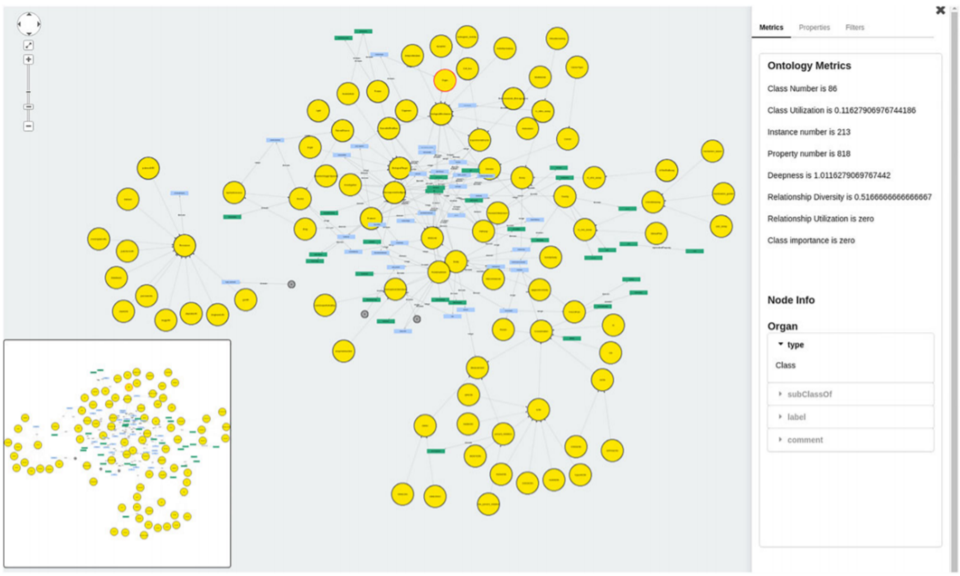


Figure 3.2: Example of Onyx user interface extracted from Ozturk and Açikgoz [2020].

.

- **Euler diagrams:** these visualizations depict entities based on hierarchical relationships as circles where a child circle is inside the parent circle. The relative position of circles can represent other types of relationships (e.g. disjointeness, when circles do not overlap). Euler diagrams can be combined with graph visualizations, with the circles being the nodes and creating links between them as it is present in OWLEasyViz [Catenazzi et al., 2009]. Figure 3.3 illustrates the user interface of this tool.

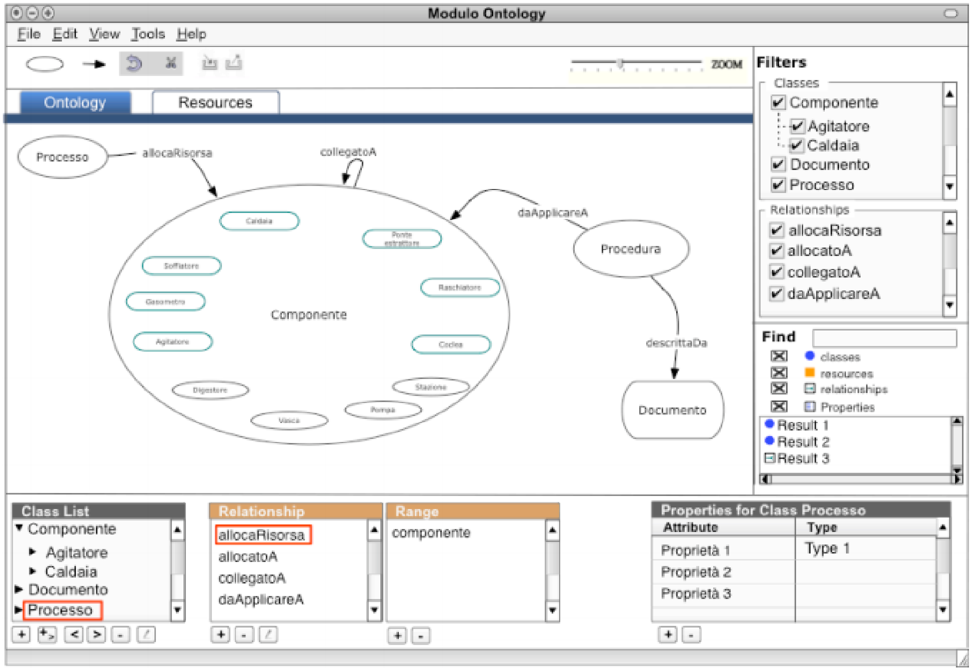


Figure 3.3: Example of OWLEasyViz user interface extracted from Catenazzi et al. [2009].

.

- **Treemaps:** these visualizations display hierarchical relationships represented by the relative position of entity rectangles - child rectangles are positioned inside their parent rectangles. Figure 3.4 contains an example of a treemap visualization realized with the Jambalaya plugin for Protégé [Storey et al., 2001].

- **2.5D visualizations:** Ontoviewer [da Silva et al., 2012] can be considered a 2.5D visualization, as it uses a node-link visualization, with nodes laid out on a 2D plane, and is enriched with a third dimension, that displays links as curves in the space above the modes (Figure 3.5).

- **3D visualizations:** one problem of 3D visualizations is that common computer screens are 2D, and the view has to be transformed from 3D to 2D, losing the advantages of depth perception. On the other hand, an advantage is that users can navigate in the 3D space and look at the graph from different angles. One example of a 3D visualization is the OntoSphere [Bosca et al., 2005] tool, that uses a node-link visualization in 3D (Figure 3.6).

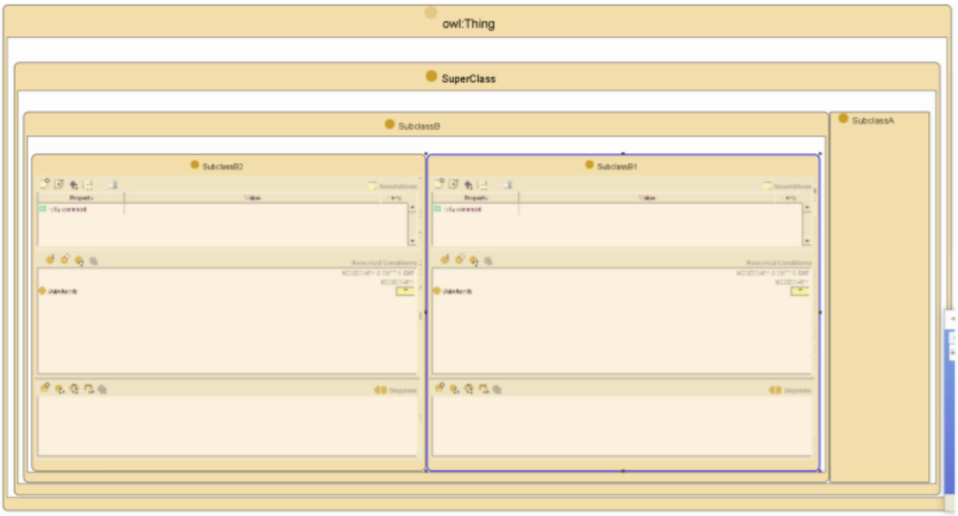


Figure 3.4: Example of Jambalaya plugin for Protégé extracted from Dudáš et al. [2018].

.

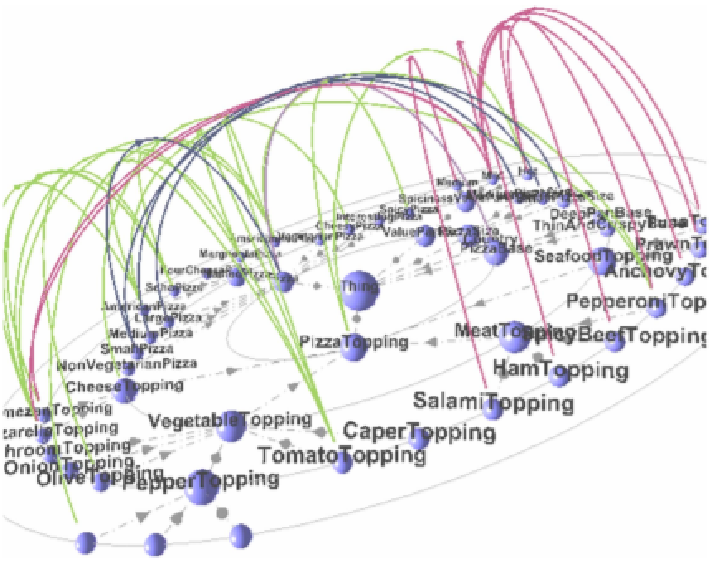


Figure 3.5: Example of Ontoviewer user interface extracted from Dudáš et al. [2018].

.

3.1.1 VOWL and WebVOWL

Lohmann et al. [2014] developed a visual language for user-oriented representation of ontologies, the Visual Notation for OWL Ontologies (VOWL), which aims to help users understand the structure of an ontology in a more intuitive way. This can be particularly helpful for users that are domain experts but do

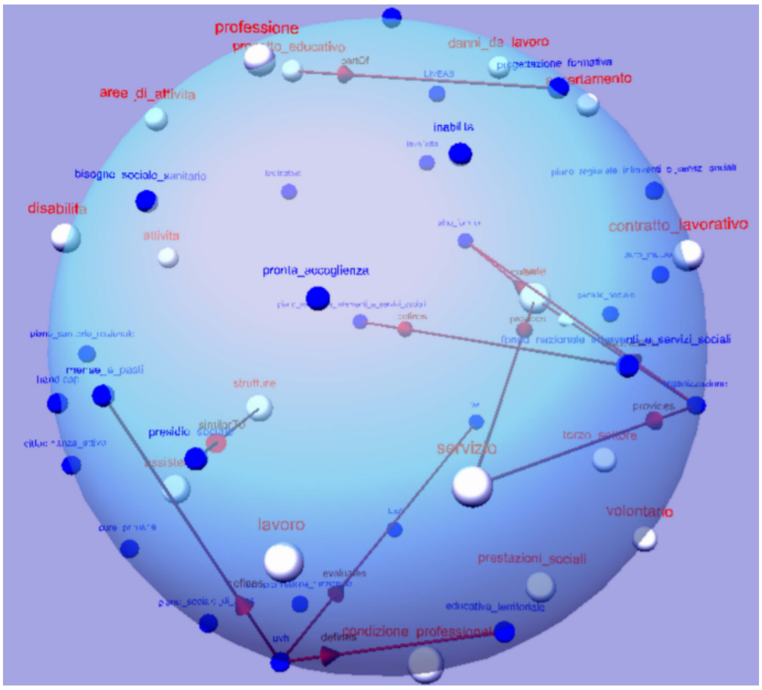


Figure 3.6: Example of OntoSphere user interface extracted from Bosca et al. [2005].

.

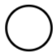
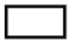
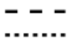
not have sufficient background in ontologies to fully understand the visualizations, as in the biomedical domain.

VOWL defines a visual representation for most of the language constructs of OWL in a force-directed graph layout to represent an ontology. This type of layout arranges the nodes so that the nodes representing the most connected classes are placed in the center of the visualization, while the least connected are on the periphery. This feature helps to emphasize a class's relative importance in the visualization, as the number of connections might be an indicator of its importance in the ontology [Lohmann et al., 2016].

The VOWL notation defines a set of graphical primitives and a color scheme that expresses certain attributes of the OWL elements (datatype or object properties, different class, and property characteristics, etc.), as indicated in Table 3.1.

In VOWL, classes are represented as circles connected by lines that represent properties with their domain and range axioms. Text is utilized for labels and cardinality constraints, and property labels and datatypes are represented as rectangles. Labels represent the text for the element given with *rdfs:label* in the language specified by the user. If available and desired, the number of instances can be implied by changing the radius of the circle from the default radius. An exception is the class representation of *owl:Thing*, which has a fixed size, as it does not carry any domain information and, despite the fact that all individuals in an ontology are instances of *owl:Thing* according to the OWL specification, this is irrelevant to the visualization.

Table 3.1: Graphical primitives of VOWL notation, organized by name, graphical representation and the respective application in the VOWL graph.

Name	Primitive	Application
Circle		*owl:Class*, *rdfs:Class*, *owl:DeprecatedClass*, *owl:Thing* and *rdfs:Resource*
Line		*rdfs:domain*, *rdfs:range* *owl:disjointWith*, *rdfs:subClassOf*, *owl:unionOf*, *owl:intersectionOf* and *owl:complementOf*
Arrowhead		*rdfs:range* (with foreground color), *rdfs:subClassOf* (with neutral color)
Rectangle		*rdfs:Datatype*, *rdfs:Literal*, property labels
Line Style		dotted line: *rdfs:subClassOf*; dashed line: *owl:disjointWith*, *owl:unionOf*, *owl:intersectionOf*, *owl:ComplementOf*; dashed border: *owl:Thing*, *rdfs:Resource*, *rdfs:Literal* and set operators.
Text	text, number, symbol	*rdfs:label*, type of property (" Subclass of" for *rdfs:subClassOf*, additional type information in brackets (e.g. "deprecated"), symbols (e.g. union symbol ("U") for *owl:unionOf* and cardinalities

Lines in VOWL can have an arrowhead pointing to the class or datatype that defines the range of the property represented by that line. *owl:Thing* is used as the domain or range when no domain or range axiom is defined for a given property, and *rdfs:Literal* is used in the case of an existing datatype property without a defined range.

Inverse properties are represented as double-edged arrows, each with a label, and when the pointer hovers the respective label, the direction of the associated property is highlighted. To limit the amount of edge crossings and provide a clearer visualization for users, subproperties are also indicated by interactive highlighting instead of explicit links between properties.

VOWL defines colors according to their function, e.g. to define deprecated or external elements of the ontology. To represent VOWL semantics, the color scheme describes how colors should relate to one another (e.g. if more than one color can be applied to a given element, priority rules determine which one should be implemented). However, colors are not mandatory to use VOWL visualizations, as they are also understandable when printed in black and white or viewed by colorblind individuals [Lohmann et al., 2016]. Other information can be provided as text instead of colors so that it is accessible even in the absence of colors.

Table 3.2: Color scheme of the VOWL notation, with listing of colors name, as well as a recommendation for a concrete color, including its hex code, and its application.

Abstract Color Name	**Concrete Color Recommendation**		**Application**
Canvas and Neutral	#fff	White	Background, *owl:Thing*, arrowhead of *rdfs:subClassOf* and , double border of *owl:equivalentClass*
Foreground	#000	Black	Lines, borders, arrowheads, text, numbers, symbols
General	#acf	Light Blue	*owl:Class*, *owl:ObjectProperty* and *owl:disjointWith*
Rdf	#c9c	Light Purple	*rdfs:Class*, *rdfs:Resource*, *rdfs:Property*
Deprecated	#ccc	Light Gray	*owl:DeprecatedClass* and *owl:DeprecatedProperty*
External	#36c	Dark Blue	Classes and properties from other ontologies linked to the ontology being visualized
Datatype	#fc3	Yellow	*rdfs:Datatyple* and *rdfs:Literal*
Datatype Property	#9c6	Light Green	*rdfs:Datatyple* and *rdfs:Literal*
Graphics	#69c	Medium Blue	*rdfs:Datatyple* and *rdfs:Literal*
Hightlighting	#f00	Red	Circles, rectangles, lines, borders and arrowheads
Indirect Hightlighting	#f90	Orange	Rectangles and circles

VOWL combines symbols used to express unions and intersections of classes with graphical representations reminiscent of Venn diagrams to communicate the underlying set operations more clearly.

Some OWL elements are merged in the visualization, such as equivalent classes, that are visually indicated by a circle with a double circle border, with their labels displayed in parenthesis. This merging of elements can happen because equivalent classes often share the same properties and, therefore, can be represented in the visualization by a single element.

One implementation of VOWL is WebVOWL [Lohmann et al., 2016], a standalone web application for interactive visualization of ontologies (Figure 3.7). WebVOWL defines a JSON schema into which ontologies need to be converted, making this tool independent of any OWL parser. This schema is designed considering the VOWL notation and contains classes, properties and datatypes of the ontology as well as the corresponding type information, additional characteristics (e.g. inverse, functional), annotations (e.g. ontology title), and ontology metrics (e.g. number of classes, properties). If an ontology defines individuals, they are listed within the classes of which they are members in the JSON file. The conversion of the OWL ontology to the JSON schema is performed using a Java-based converter, OWL2VOWL, which is deployed along with WebVOWL. This converter uses the OWL API [Horridge and Bechhofer, 2011] to access the ontology representation and transforms it into the required JSON format.

WebVOWL renders the graphical elements according to the VOWL specifications in a force-directed graph layout, implemented with the JavaScript library D3 [Bostock et al., 2011]. To generate the force-directed graph layout, WebVOWL uses a physics simulation consisting of three forces that are applied iteratively: (1) edges act as springs, while (2) nodes repel each other and (3) drag forces ensure that nodes settle. The forces cool down in each iteration and the algorithm stops after some time to ensure a stable visualization. Each time users change the layout of the graph, the algorithm is triggered again. This creates a dynamic animation that constantly repositions the nodes of the graph. The force-directed algorithm can be reset or paused in the menu, which helps reduce the load on the processor and allows elements to be rearranged and manually positioned without immediately updating the layout.

The user interface of WebVOWL is shown in Figure 3.7 and consists of the main view with the visualization, a collapsible sidebar with ontology information, and a bottom menu with controls, filters, and modes.

In the main view, users can explore the ontology and customize the visualization. WebVOWL implements basic interaction techniques with the visualization, such as pan the background, drag and drop, moving elements around to adjust the layout, and zoom.

The sidebar contains the available metadata about the ontology, such as a title, namespace, author(s) and version, a description text, and ontology metrics (e.g. number of classes, properties, individuals), which may be contained in the JSON file or computed at runtime. When an element of the graph is selected, the sidebar also displays details about that element, such as its name, type, IRI, annotations, and properties not shown in the visualization (e.g. disjoint classes). To minimize screen space, all of this data is grouped and displayed in an accordion widget.

The bottom menu includes a search mechanism: for text-based user input, matched entities with that input are presented as suggestions, and selecting an item highlights it in the visualization. In addition,

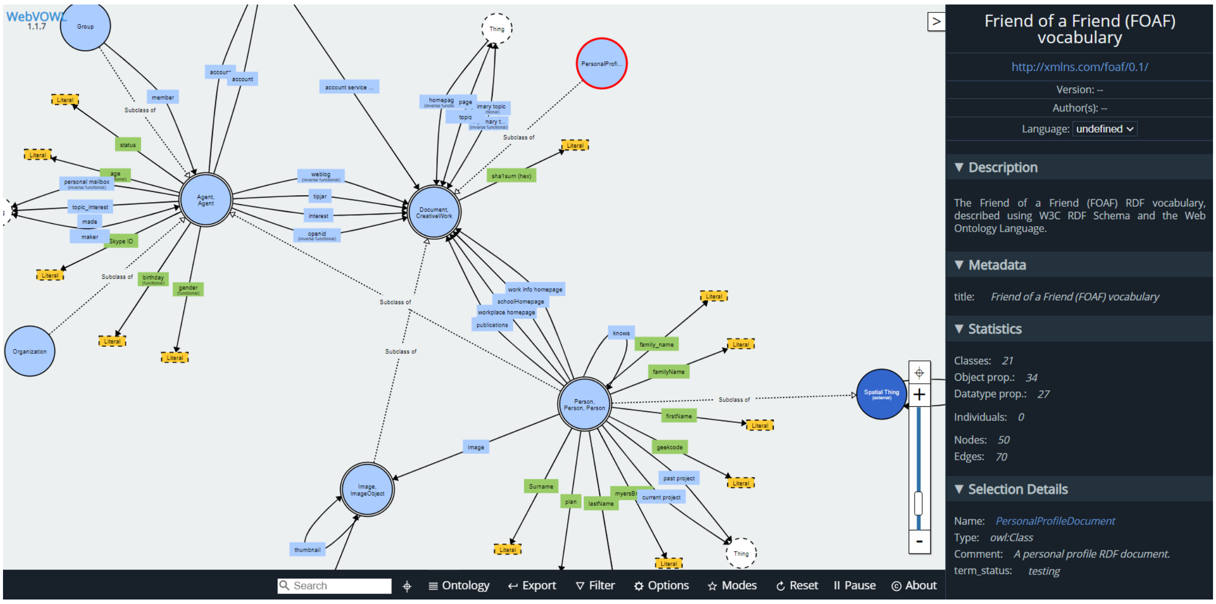


Figure 3.7: Example of WebVOWL user interface for the visualization of the Friend of a Friend (FOAF) RDF vocabulary described using the W3C RDF Schema and the Web Ontology Language. Most of elements of VOWL notation are present in this example.

.

WebVOWL has ”location and zoom” functionality that drives the user to the area of the graph where the element is present. Users can apply filters to the visualization to reduce the size of the graph. The filters available in WebVOWL filter the visualization by entity type, i.e., hide all entities of a certain type in the visualization (e.g. hide all object properties). Additionally, the filter dropdown includes a slider that controls the degree to which the graph is collapsed. This filter removes classes from the graph based on their node degree, starting with the classes with the lowest degree, and it can be used to reduce the size of the visualization, especially in large ontologies, by displaying only a subset of highly connected classes and their properties while hiding the rest. In addition, users can customize the visualization by changing the visualization options, such as class or datatype distances, label width, or applying compact notation to remove redundant information and obtain a clearer visualization.

Recently, the WebVOWL Editor [Wiens et al., 2018] has been integrated into the main WebVOWL tool, and users can test this by activating the experimental editing mode. This mode is designed to provide ontology modeling features, such as creating, editing, and deleting elements (e.g. classes, object properties) in the graph visualization. However, this mode is still experimental and there have been no updates since its implementation. In addition, WebVOWL provides the *pick-and-pin* mode, which allows to decouple selected nodes from the force-directed layout and fix them at a chosen position on the canvas. Finally, WebVOWL allows exporting the complete or filtered visualization as an SVG image or exporting the ontologies in various formats (e.g. JSON, URL).

Lohmann et al. [2016] conducted several user studies with different user groups (non-experts and experts in ontologies) to evaluate WebVOWL and VOWL. The results of the comparative evaluations with other visual notations revealed a preference for VOWL among the majority of participants, based on aspects such as clarity and distinctiveness of elements and a layout easy to use. The studies also demonstrated the applicability and usability of the use of WebVOWL.

3.2 Ontology Alignment Visualization and Validation

The user interface in alignment systems is the key component for the validation process of an alignment, as it not only provides a visualization of the alignment, but also the necessary tools to validate it.

The ability to support the visualization of ontology alignments is a very important feature in ontology matching systems, especially in the biomedical domain where many of the users are not knowledge engineering experts. The purpose of these visualizations is then to help a user understand the detailed information inside an ontology, support navigation and inspection of mappings and interactive matching [Pesquita et al., 2014]. Furthermore, the use of visualization techniques has the additional advantage of enabling the powerful visual processing abilities of humans, which allow them to efficiently explore, understand, and discover patterns in the information provided [Granitzer et al., 2010].

Given the complexity of the alignment validation process, a critical aspect of alignment visualization is to provide users with enough information to verify the correctness of each mapping, but at the same time, not overwhelm them with too much information [Li et al., 2019].

The information and functionalities provided by a user interface in matching systems depend on the characteristics of the employed visualizations. Typically, alignment visualizations are supported by two paradigms [Pesquita et al., 2014; Granitzer et al., 2010]:

- **Trees:** these representations use a standard tree widget to represent the hierarchy of ontologies. Ontologies are displayed side by side, while mappings are displayed as lines or curves connecting the tree nodes or all displayed in a list. Different alignments can be represented by using different colors. When visualizing a large number of mappings, the crossing of links can result in clutter.Tree-based representations are not suitable for providing an overview, since only a small part of the class hierarchy can be visible at once. Figure 3.8 shows an example of AgreementMaker's [Cruz et al., 2009] interface that employs this visual paradigm.

- **Graphs:** these representations enable the navigation and exploration of ontologies and provide insight into the structure of the ontology. In graph visualizations, ontologies are typically represented with different colors, and mappings are represented as links between mapped entity nodes. Typically, these visualizations are not suitable to provide an overview of the alignment, as they usually do not scale to very large datasets. Figure 3.9 shows an example of the graph visualization of a mapping, provided by the AgreementMakerLight [Faria et al., 2013] interface.

Visualizations in matching systems are usually accompanied by additional widgets that provide advanced functionality for displaying, inspecting, and manipulating the mappings generated by the matching algorithms [Granitzer et al., 2010].

Over the years, several tools for visualizing ontology alignments have been developed, those that are integrated with matching systems, deployed as part of these systems, and others that have been developed as independent tools.

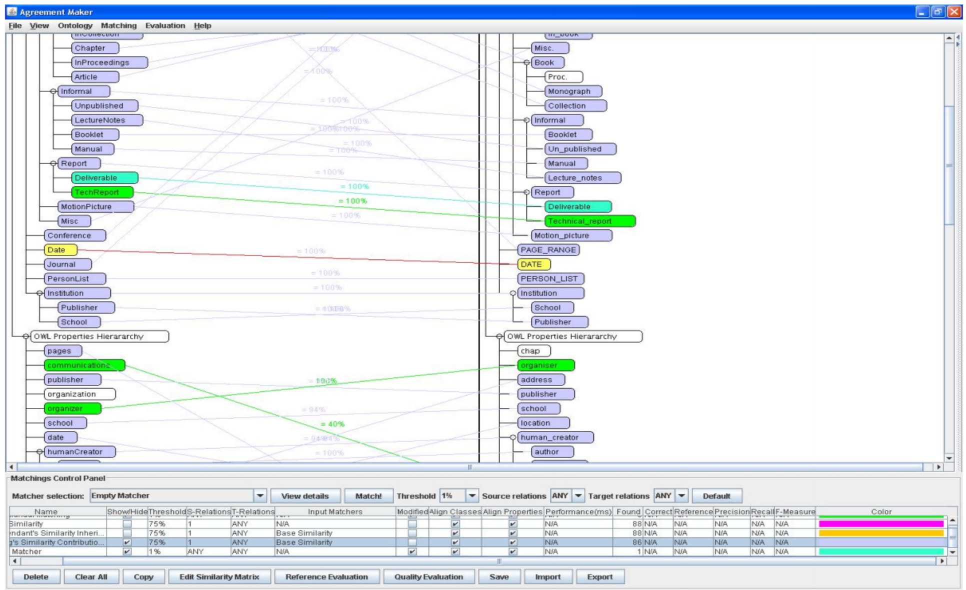


Figure 3.8: Example of AgreementMaker graphical interface from Cruz et al. [2009]. Both ontologies are visualized as trees with the mappings depicted as lines between the two mapped elements color-coded according to the matcher that calculated the similarity values (above). The control panel allows users to run or manage matching methods and their results (below). Users can interact with the visualization by reducing the number of lines depicted or edit the alignment produced by adding, deleting, and updating mappings.

The **AgreementMaker** [Cruz et al., 2009] interface represents ontologies as indented trees, displayed side-by-side in scrollable panes (Figure 3.8). Mappings are represented by a straight line indicating their similarity score and are color-colored to distinguish between accepted, rejected, candidate, and manually created mappings. The properties of the classes are accessible by clicking on the node and this information is displayed in a separate detail view. During the validation process, users can accept, reject or manually create new mappings. However, this tool is not capable of handling ontologies with tens of thousands of classes, does not allow the visualization of non-hierarchical relationships or of multiple inheritance, which are important characteristics of biomedical ontologies.

The user interface of **AgreementMakerLight** [Faria et al., 2013] provides two different views of the alignment (Figure 3.9): a graph view, where users can visualize a specific mapping and its local context, and a list view, which serves as an overview. In the graph view, both ontologies are represented

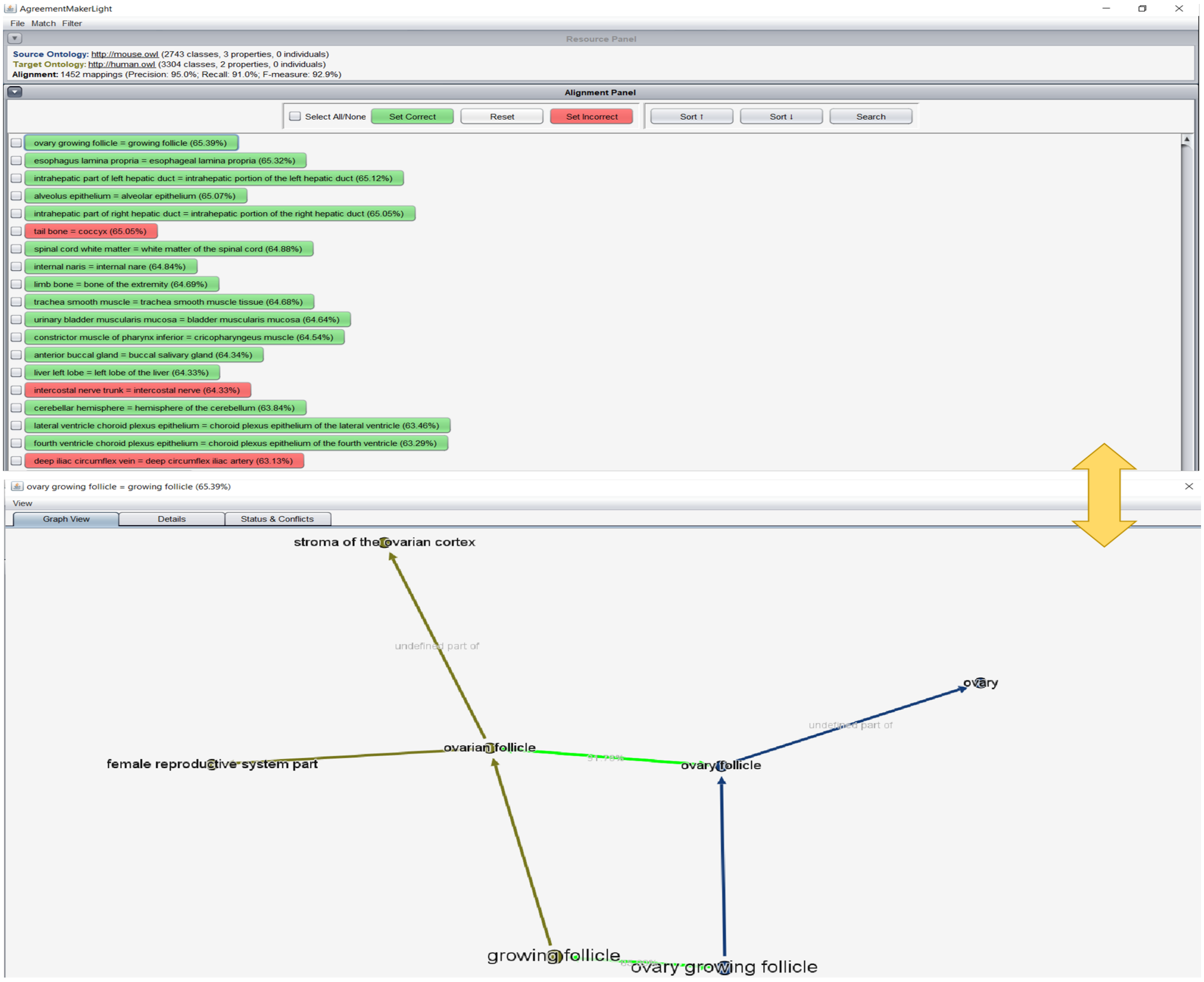


Figure 3.9: AML user interface. The list view (top) displays the full alignment, with mappings colored according to their status - green for correct mappings and red for incorrect ones - and provides users with the functionalities to accept or reject mappings and search for a specific mapping. The Graph View example shows a mapping between the entities *ovary growing follicle* and *growing follicle*, with the mapping status indicated by color.

in the same graph with different colors (dark blue for the nodes and edges of the source ontology and pale green for the nodes and edges of the target ontology), and the mappings are represented as double-edged arrow labeled with their confidence score, whose color depends on the mapping status: gray for candidate mappings, green for mappings marked as correct, and red for mappings marked as incorrect. All mappings between entities present in the selected neighborhood are displayed. Users can pan and zoom the graph, and further specify its characteristics. The neighborhood radius accepted by AML goes from 0 to a maximum of 5 edges, but by default they are shown at a distance of two. The list view provides information about the source and target ontologies, supports searching for mappings, and allows users to accept or reject mappings, remove the incorrect ones and create new mappings. However, it is not possible to get an overview of the alignment visualization, and in some cases, when the graph of the mapping has a great number of nodes and edges, the visualization can overwhelm users with too much information, risking cognitive overload, and users end up not taking advantage of the visualization.

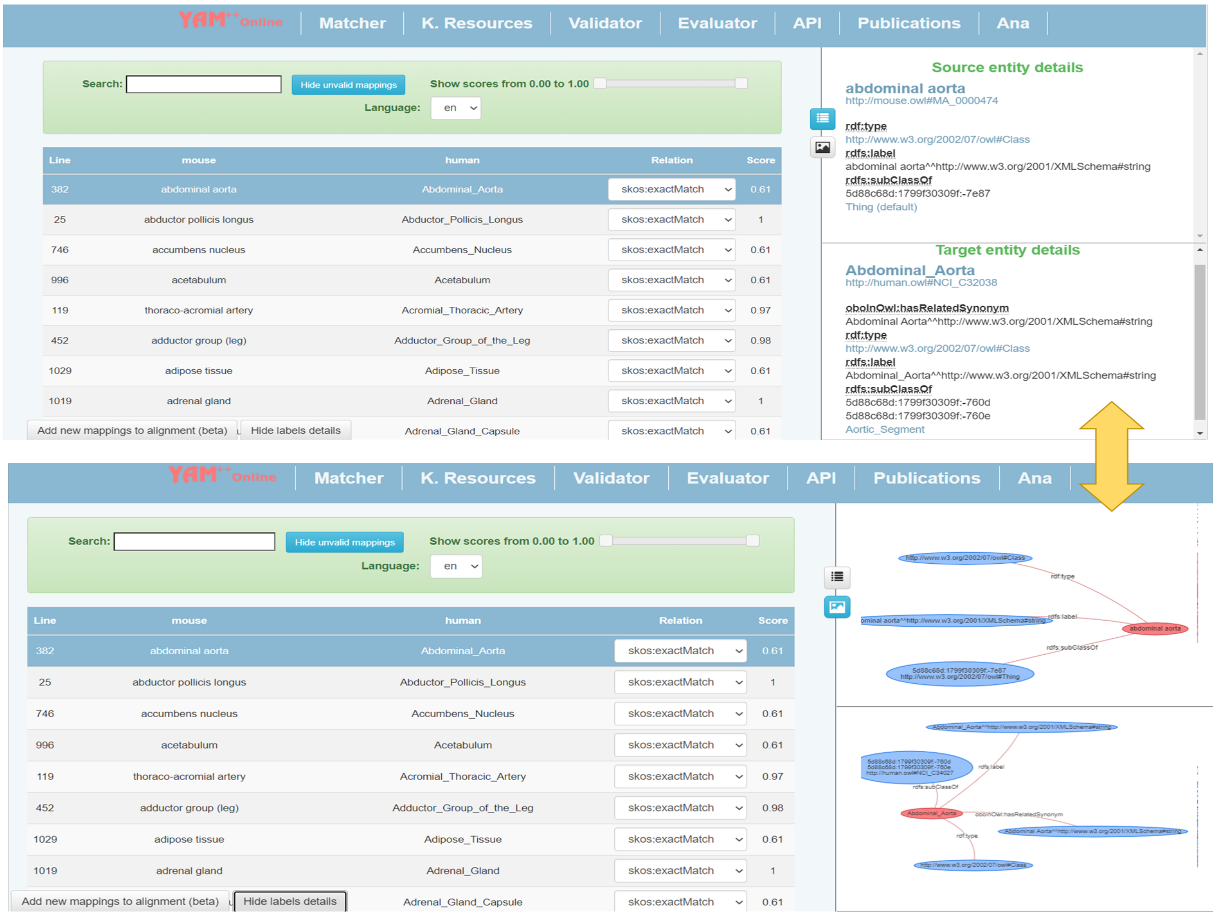


Figure 3.10: YAM++ *online* user interface. On the main page, a list of mappings (pairs of labels of matched concepts), where users can validate mappings. On the right side, a contextual description for each of the two concepts is displayed in each row, containing all alternative labels, and the labels of parents and children. Users can choose between this textual description (top) and a graphical visualization for the mapped concepts that reaches a distance of 2 (bottom).

As **YAM++ *online*** [Bellahsene et al., 2017] can accept as input an alignment not necessarily generated by this tool, it can be considered as a standalone web tool for visualizing ontology alignments. The user interface of this tool consists of two panels (Figure 3.10). The left panel consists of a list of all mappings and their respective confidence score, where it is possible to validate each mapping. In the right panel, users can alternate between a description area view, which contains information about the mapped concepts, and two graph visualizations, one for each mapped entity and its local context up to a maximum distance of 2 edges. There is no visual representation for mappings in these graph visualizations. Although it is possible to add new mappings to the alignment, users cannot remove mappings in this interface. Also, this tool has a search mechanism that allows users to search for a certain mapping by entering the label of the source or target entities. The list is filtered, presenting only the matched entities for the input they provide.

Figure 3.11: Example of LogMap information view for the mapping *integumental_system=integumentary_system*. The view is divided into sections that provide information such as lexical metadata, conflicting mappings, and ambiguous mappings. At the top of the view, users can accept or reject the mapping.

LogMap [Jiménez-Ruiz and Cuenca Grau, 2011] presents to users only the mappings that require user feedback. Each candidate mapping is presented with its confidence score. Each mapping view is divided into expandable sections that contain information about the context of the mapped entities, lexical metadata, such as synonyms and alternative labels, and ambiguous and conflicting mappings, to help users understand the consequences of their validation (Figure 3.11). Users can accept or reject the mapping based on the ambiguity and conflict criteria, but it is not possible to create new correspondences. LogMap presents mappings to users according to their impact on other mappings, i.e., mappings that conflict with other mappings are displayed first. In addition to this list presentation of mappings, the system does not employ any alignment visualization.

VocBench [Stellato et al., 2020] is an open source web platform for the collaborative development and managing of datasets complying with Semantic Web standards, such as OWL ontologies, SKOS(/XL) thesauri, Ontolex-lemon lexicons and generic RDF datasets. This tool allows the management of alignments from two different sources: loading an alignment file or using a remote alignment system. In both cases, VonBech allows to validate the alignments. The user interface consists of a table with the candidate mappings (Figure 3.12). Each row reports the source and target entities of each mapping, the proposed mapping relation with the confidence score, two buttons allowing to accept or reject the mapping, and the indication of that mapping status. The status is reported through an icon: ”✓” for accepted mappings, ”X” for rejected mappings, and ”error” that is applied when the relation between the two aligned entities cannot be asserted for some reason. It is possible to accept or reject the entire alignment or mappings with confidence score above or under a certain value. Users can access to metadata about each entity, such as alternative labels and synonyms, by clicking in the entity in the row, which can help the alignment validation process, particularly when validating biomedical ontologies. However, it is not possible to create new mappings or remove mappings in this interface, and the system does not employ any mapping visualization beyond the list presentation.

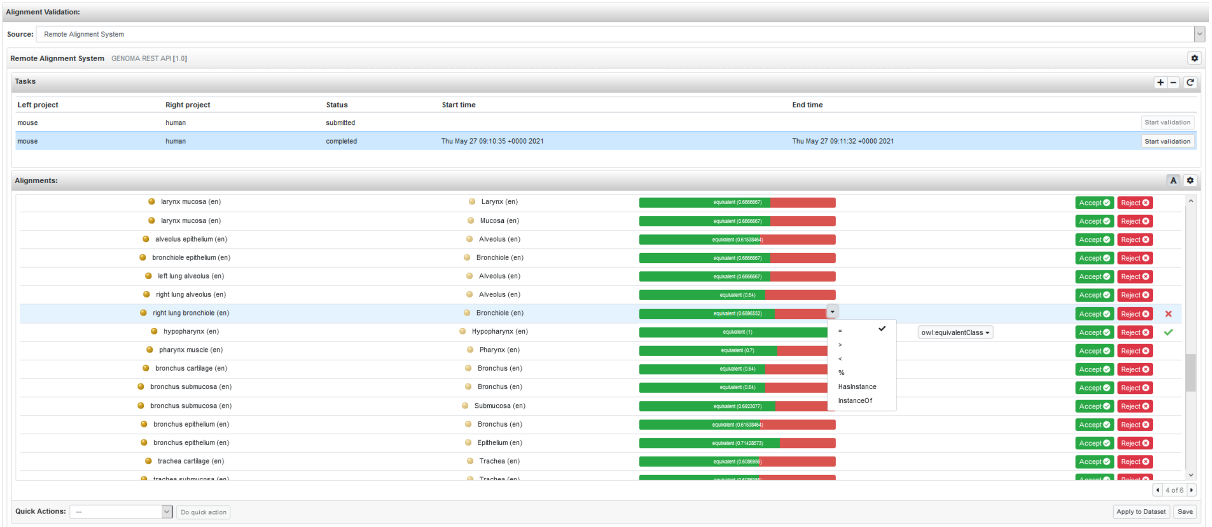


Figure 3.12: Example of the user interface of VonBech for alignment validation. Users can change the mapping relation among a set of relations available in INRIA’s Alignment format [1], and choose the mapping property accordingly. This example shows the mappings status icon for a rejected mapping between the entities *right lung bronchiole* and *Bronchiole*, and for an accepted mapping between the entities *hypopharynx* and *Hypopharynx*.

In Chapter 6, Table 6.1 compares the requirements for validation and visualization of alignments addressed by these alignment systems, and compares them with the developed work.

Chapter 4

VOWLMap

The first step in developing VOWLMap was to analyze the requirements described in Li et al. [2019] and select the requirements that this tool targets to achieve the objectives defined in Section 1.1. In the second step, existing development and visualization options were considered. Since the developed tool is independent of an alignment system, this provided the flexibility to consider a variety of technologies for this work rather than being limited to the alignment system ecosystem.

The success of javascript-based visualization for complex data afforded by D3.js [Bostock et al., 2011] was a motivation to consider a browser-based architecture since it allows easier use by domain experts, as no specific software needed to be installed.

Building on this focus, the next step was to investigate existing browser-based ontology visualization systems, where WebVOWL met the defined functional and technical requirements. The final step was then to extend WebVOWL and the VOWL notation to meet the defined requirements and achieve the main goal of this dissertation, which is to develop a tool that improves user interactions when manually validating alignments of biomedical ontologies. We create a release open source for the work developed in this chapter [Guerreiro et al., 2021] and it is available on GitHub [1].

4.1 Requirements Analysis

Considering the fundamental aspects for alignment validation described in Section 2.3, VOWLMap targets use cases where users are domain experts but may be technical and alignment system novices and only become involved after the alignment process, interacting with the alignment but not with the alignment system, since the input alignments have been previously generated by ontology matching systems.

Considering the target use cases and following Li et al. [2019], it was decided that the developed work should focus on the following functional requirements derived from the literature review and empirical experiences:

[1] https://github.com/liseda-lab/VOWLMap

- **(R1)** Loading ontologies and corresponding alignment.
- **(R2)** Provision of alternative alignment views to support alignment validation.
- **(R3)** Support for visual information seeking tasks [Shneiderman, 2000]:

  (R3.1) *overview* (provide an overview of the entire collection).

  (R3.2) *zoom* (zoom in or out on items of interest).

  (R3.3) *filter* (remove uninteresting elements from the visualization).

  (R3.4) *details-on-demand* (select an item and obtain details).

  (R3.5) *relate* (view relationships between elements).

  (R3.6) *history* (store a history of actions, to support undo, for instances).

  (R3.7) *extract* (allow extraction of subsets of elements).
- **(R4)** Indication of mapping status, that is distinguish between validated and candidate mappings.
- **(R5)** Visualization of metadata, such as definitions and synonyms, particularly important when visualizing biomedical ontologies.
- **(R6)** Visualization of a mapping context, i.e., displaying the neighborhood of the entities involved in the mapping, including neighboring mappings.
- **(R7)** Accept and reject mappings, i.e., mark a mapping as correct or incorrect.
- **(R8)** Create and refine mappings, i.e., manually add a completely new mapping or refine an existing mapping by altering the source or target entity to a more suitable one.
- **(R9)** Search, i.e., the ability to search for ontology entities and mappings by their labels.
- **(R10)** Session support, given the extension of the validation process, accommodating interruptions is essential.
- **(R11)** Export into different alignment formats.

WebVOWL already supports some of the functional requirements we identified (all R3 except 3.6, R5, R9 partially, and R10), but not those directly related to alignment visualization, validation, and editing (R1, R2, R4, R6, R7, R8, R11).

4.2 Extension of VOWL

The graphical primitives and color scheme defined by the VOWL notation do not allow the representation of alignments and a mapping context in the VOWL graph. Therefore, the proposed methodology foresees the extension of the VOWL notation to improve the VOWL graph in such a way that it is possible to (1) represent the two ontologies with different colors, (2) represent mappings between the two ontologies, and (3) have a representation of each mapping status.

The original notation represents the ontology classes and some properties, such as object properties and disjoints, with the *general color*, for which it is recommended to be light blue. Additionally, external elements to the ontology, i.e., classes and properties from other ontologies that are linked in the ontologies being visualized, are represented with the *external* color, which is a darker version of the *general* color (recommended to be dark blue). In order to represent the target and source ontologies of the alignment with different colors in the same graph, the elements of the target ontology were considered *external* to the source ontology. To ensure this, when generating the JSON file in the data processing step, the attribute "external" is assigned to all target elements. Thus, in the VOWL graph, the source ontology elements are represented with the *general* color (light blue) and the target ontology elements are represented with the *external* color (dark blue).

In the VOWL graph, the graphical primitives for representing properties and their labels were used to represent mappings between the two ontologies. Each mapping is treated as a property between two mapped classes, represented by a solid line with arrowheads at both ends. Unlike other properties, the labels for mappings are not provided by the *rdfs:label* elements, but by the confidence score annotation, present in the JSON file for each mapping. Thus, each score is represented by a rectangle whose borders are colored according to the status of the mapping.

Table 4.1: Extension of the VOWL notation color scheme, listing the new colors names, along with its hex code and application. New colors have been added to the colors shown in table 3.2 to represent a mapping *status*.

Abstract Color Name	Concrete Color Recommendation		Application
Unreviewed Mapping	#8c8c8c	Medium Gray	Lines, borders of rectangles, arrowheads and checkbox
Correct Mapping	#070	Dark Green	
Incorrect Mapping	#b30202	Dark Red	
Unsure Mapping	#f2a114	Medium Yellow	

It is possible to assign one of four defined status to each mapping - *correct, incorrect, unsure*, and *unreviewed*, which is the default status. Four different colors have been added to the notation to represent each status value: dark green for *correct*, dark red for *incorrect*, medium yellow for *unsure* and medium grey for *unreviewed*. The line, arrowheads, and border of the score rectangle are colored according to respective mapping status. The extended version of the VOWL notation is resumed in Table 4.1.

The color scheme of VOWL already includes some variations of the colors mentioned above, which could potentially lead to misinterpretations between: (1) *incorrect* mappings in dark red and highlighting in VOWL red; (2) *unsure* mappings in medium yellow and datatype in VOWL yellow; (3) *unreviewed* mappings in medium gray and deprecated elements in VOWL light gray; and to a lesser extent (4) *correct* mappings in dark green and data properties in VOWL light green. However, the fact that mappings have specific graphical representations in this extension regardless of their color and a boxed label with the confidence score, should enable a clear distinction between mappings and all these cases. Note also that under this extension, external elements to the ontology, represented by dark blue in VOWL, no longer have a representation in either the source or the target ontology.

4.3 Alignment RDF Format Extension

The Alignment RDF format described in Section 2.2 does not contemplate the validation process, such as indicating the status of a mapping or distinguishing between candidate and validated mappings. In order to export an alignment resulting from a validation process that includes all the changes made by users during this process, the present work extended the alignment RDF format described by Euzenat [2004]. The formal description of this extension is available in Appendix A. [2].

The RDF alignment extension captures the validation information in a new attribute, *revision*, that can optionally be placed within the *Cell* element, as illustrated in Figure 4.1. This new attribute contains the *status* attribute, that indicates the status for that correspondence, which can assume four different values: *correct*, *incorrect*, *unsure* e *unreviewed*. This extension also anticipates the possibility of adding the author and a timestamp of each revision, with an *author* and *timestamp* attributes contained inside the new *revision* attribute, even though this is not yet implemented in VOWLMap when an alignment is exported.

4.4 Data Processing

As mentioned in Section 3.1.1, WebVOWL defines a JSON schema into which ontologies need to be converted. After converting the two ontologies into two JSON files using OWL2VOWL, these two files need to be merged with the alignment to obtain a final JSON file that can be successfully loaded into the VOWLMap and contains all the information needed for the alignment visualization.

To facilitate this process, this work developed a Python-based tool that receives the two ontologies JSON files and the RDF alignment file and merges them into a single JSON file. The resulting file contains the same information about the source and target ontologies that is present in the input files,

[2] This extension is formally described in https://github.com/liseda-lab/AlignmentValidation

```xml
<?xml version='1.0' encoding='utf-8'?>
<rdf:RDF xmlns='http://knowledgeweb.semanticweb.org/heterogeneity/alignment'
       xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
       xmlns:xsd='http://www.w3.org/2001/XMLSchema#'>

<Alignment>
       <xml>yes</xml>
       <level>0</level>
       <type>??</type>
       <onto1>http://mouse.owl</onto1>
       <onto2>http://human.owl</onto2>
       <uri1>http://mouse.owl</uri1>
       <uri2>http://human.owl</uri2>
       <map>
              <Cell>
                     <entity1 rdf:resource="http://mouse.owl#MA_0002031"/>
                     <entity2 rdf:resource="http://human.owl#NCI_C12774"/>
                     <measure
rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.8836</measure>
                     <relation>=</relation>
                     <revision>
                            <status>unreviewed</status>
                     </revision>
              </Cell>
       </map>
```

Figure 4.1: Excerpt of RDF file contemplating the Alignment extension, with the new *revision* and *status* tags.

with an exception for any elements of the target ontology that are assigned the attribute "external" to allow the two ontologies to be represented by different colors. The mappings present in the alignment file are added as properties and the respective score and status are added as property annotations. If there is no indication of the respective mapping status, the default value *unreviewed* is assigned.

To tackle the challenge of loading large alignments, such as those from the biomedical domain, when the number of nodes present in the alignment is higher than 5000, the developed tool only loads elements of the ontologies into the JSON file that are at a maximum distance of three edges of each mapping. This approach allows reducing the size of the JSON file, by removing unnecessary information, since the VOWLMap graph visualization only extends the neighborhood to a maximum of three edges of the mapping. On the one hand, this facilitates memory management and helps optimized the time of uploading, important challenges when visualizing biomedical ontology alignments, on the other hand, it limits the creation of new mappings during the validation process, since it is only possible to create new mappings in the neighborhood of the already existing mappings. Although the tool allows to generate a graph visualization for new mappings, their neighborhood may be incomplete, since not all information is contained in the JSON file.

4.5 User Interface

The user interface of VOWLMap extends the one provided by WebVOWL with several new features to support visualization, manual validation, and editing of an alignment. The VOWLMap user interface is divided into two views: an alignment panel (Figure 4.2) and a graph visualization (Figure 4.3).

The sidebar and bottom menu are located across both views. The collapsible sidebar contains all the information described in Section 3.1.1 for WebVOWL, such as a description of the alignment, metadata, and alignment metrics (computed by summing the metrics of the two ontologies), as well as some additional information about the alignment, such as the IRI of the source and target ontologies and the number of mappings. Additionally, the ability to enter or edit the title has been added. The fact that this information is grouped and displayed in an accordion widget means that the system does not need to condense all the relevant information in the visualization, saving screen space and avoiding overwhelming the user.

The *Selection Details* works in a similar way to WebVOWL, providing users with easier access to information about a selected element of the graph. When a mapping is selected, information about that mapping is displayed, such as the main label of the source and target entities along with their synonyms or definitions, if available in the ontologies. Next to each entity label, a new feature has been created: an automatic link to Wikipedia for that term. Since these links are automatically generated from the source and target labels, there may not be a Wikipedia page available for some terms. However, in the cases that the page exists, this feature can help users validate a mapping, by providing extra information. Lastly, a *dropdown* was included in this section, that allows users to change the mapping status in the sidebar, in both views. When in the alignment panel, if users change the status in the *Selection Details* area, the checkbox in the list is updated and vice-versa. In the same way, if users change the status in the *Selection Details* area in the graph visualization, the graph is updated and the color of the mapping changes according to its status. This last feature will be further described in Section 4.5.2.

The bottom menu allocates different functionalities, some directly related to the graph visualization and others concerning the load and export processes and search mechanisms. The *Alignment* menu element permits users to select the previously generated JSON file, that contains the alignment, from their computer and uploads it in VOWLMap.

The *Filter* element allows users to apply filters to the graph and remove a set of elements from the visualization to reduce the size of the graph. The available filters in VOWLMap are the same offered by WebVOWL: the *datatype properties* (removes all datatype properties along with the datatypes they are pointing to), *object properties* (removes all object properties), *solitary subclasses* (removes subclasses that are only connected to their superclass and do not have any link to other classes), *class disjointness* (removes disjoint relations and is activated by default, since this kind of relations tends to clutter the graph without being of large interest to the user), and *set operators* (removes set operators, such as classes representing unions, intersections or complement operations). The *degree of collapsing* filter that removed classes from the graph based on their node degree was removed from the filter menu and a new menu element, *Neighborhood*, was added. This menu element allows users to select the distance of the neighborhood of the entities involved in the mapping. By default, the mapping graph displays the neighborhood at a distance of one, but users can change the distance from zero up to a maximum of three edges.

The *Options* element contains the same elements present in WebVOWL, except the *compact notation* and *color external* options, that are by default activated to always obtain a graph with two ontologies in different colors.

In the *Modes* element, the pick-and-pin mode was maintained and the ontology editing mode was altered so that it could satisfy the alignment editing functions. Thus, some graph editing functionalities were refined to match the defined requirements, and functionalities that violated the alignment structure were removed. The ontology editing mode per se is no longer available to users to activate and is now incorporated in the graph visualization. The alignment editing functions will be described in more detail in Section 4.5.2.

The *Reset* and *Pause* button continues to allow reset and pause of the algorithm and the interactive visualization so that users can manually position the nodes in the graph.

The *Search* mechanism offered by WebVOWL was extended in VOWLMap to support the search in the alignment context. Thus, whether in the alignment panel or the graph visualization, users can search for a specific mapping in the search bar, by entering the label of the source entity, the equal sign, and the label of the target entity (in a similar way to how mappings are presented in the list). When users start typing, suggestions will appear, and by selecting one of the suggestions or by entering the full name, a graph visualization for the searched mapping is generated. In the graph visualization, WebVOWL supported search mechanism was preserved, so that users can search for entities present in the graph visualization. The ”location and zoom” functionality, that drives the user to the region of the graph where that element is, was also maintained.

In order to support the *session* requirement, VOLWMap saves the alterations made to the original alignment (e.g. a new mapping, a mapping validation) in a cached version, as long as VOWLMap is open in the browser, even if users open different alignments. When loading the original alignment file, if available, the cached version is always loaded by default, instead of the information contained in the JSON file. When this occurs, a *Reload alignment* button appears in the right upper side of the alignment panel, that allows users to discard the alterations made and load the original alignment.

Finally, in addition to all the supported formats available in WebVOWL to export the results, the options to export the alignment in JSON and RDF format were implemented.

4.5.1 Alignment Panel

When the JSON file is loaded, the alignment panel is opened. This panel is composed of a list that displays the full alignment, divided into three columns (Figure 4.2): a first clickable column with a string consisting of the source's main label, an equal sign, and the target's main label (e.g. *cardia = stomach cardiac region*), that drives users to the graph visualization of that mapping when clicked; a second one with the respective value of the confidence score; and the third one with a clickable status checkbox, where users can validate the mappings, by changing the status to one of the four options available - *correct, incorrect, unsure* and *unreviewed*. Each status value is associated with the respective color defined in the extension of the VOWL notation and with a symbol that reflects the information of each option: ”✓” for *correct*, “X” for *incorrect*, and “?” for *unsure*. This feature is particularly helpful to make the interface more comprehensible when viewed by colorblind people. For the default status, *unreviewed*, no symbol or color was assigned to the checkbox. The list can be sorted by score or by status.

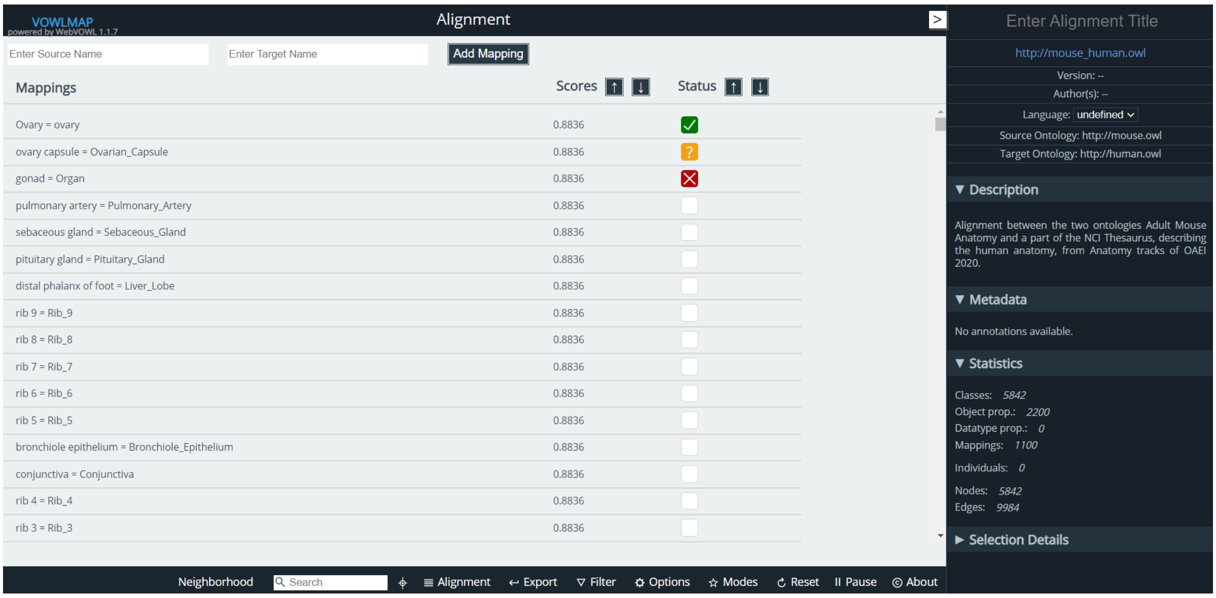


Figure 4.2: Example of alignment panel view of VOWLMap. In this example it is possible to view the 4 values for the status checkbox, associated with the respective color and symbol. Users can sort the list of mappings by the score value or by the status value.

Additionally, the alignment panel allows users to create new mappings in the alignment by entering the label of the respective source and target entities in the two input forms positioned above the list. This mechanism can be useful especially when there is a need to add new mappings between entities not present in the graph visualization, as will be explained below. When a new mapping is added by this mechanism, the graph visualization for that new mapping is automatically generated.

4.5.2 Graph Visualization

By selecting one specific mapping of the list, a graph visualization for that mapping is generated. VOWLMap uses a graph to represent the mapped classes and their neighborhood (Figure 4.3). The elements of the two ontologies are represented with different colors: light blue for the source ontology and dark blue for the target ontology. The mappings between the two mapped classes are represented as double-edged arrows, labeled with their respective confidence scores. All mappings between ontology entities present in the selected neighborhood are shown in the graph.

The color of each mapping representation, i.e., the double-edged arrow and the border of the score rectangle, changes according to its status. As aforementioned, it is possible to change the status of a selected mapping in the graph in the sidebar. The visualization is updated every time the status is modified so that the mapping color could match the status value: dark green to *correct*, dark red to *incorrect*, medium yellow to *unsure* and medium grey to *unreviewed*. This feature helps users in the decision process, by allowing them to consider other mappings and their status in the neighborhood and their logical consequences.

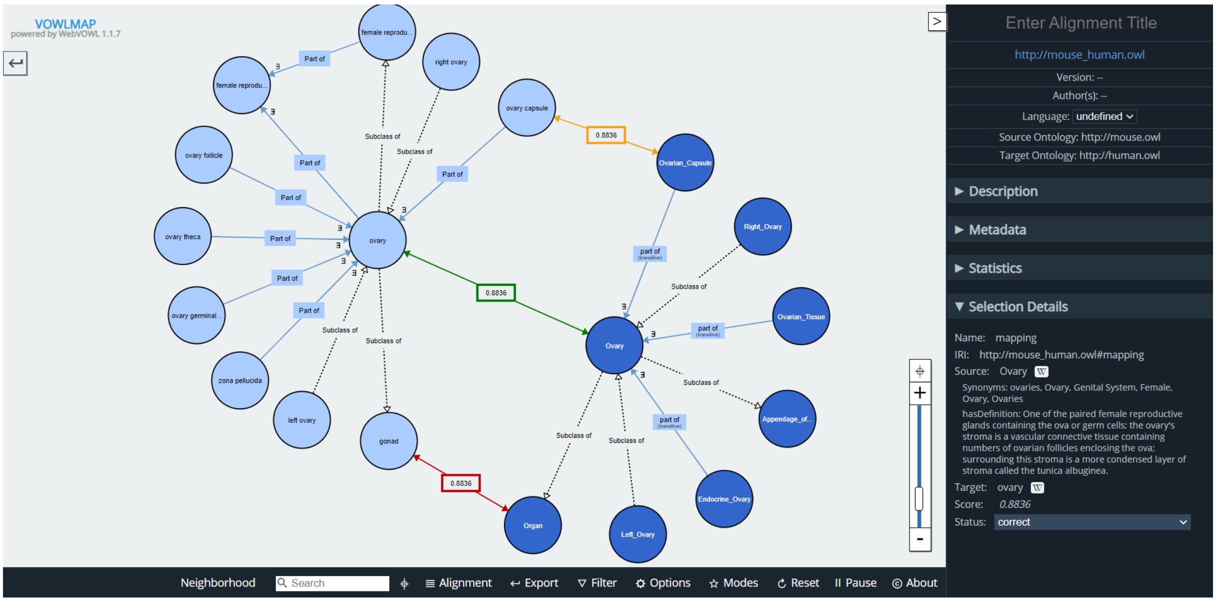


Figure 4.3: Graph Visualization of VOWLMap for the mapping between *ovary* and *Ovary*, and the neighborhood at a distance of 1. All mappings in the neighborhood, such as *gonad* - *Organ* and *ovary capsule* - *Ovarian_Capsule* are shown in the visualization colored according to their status.

It is possible to interact with the visualization as mentioned before, by zooming in or out, pan the background, move elements around, apply filters or change the visualization options to adapt the force-directed layout and extend or reduce the neighborhood.

In addition to VOWLMap allowing the validation of a mapping while visualizing the graph, it also allows editing the alignment through interactions with the graph. As aforementioned, the editing mode was restructured and incorporated in the visualization, so that the editions made to the alignment could preserve the alignment and ontologies structure and elements. For that reason, contrary to WebVOWL, where users were able to remove or add any element in the graph, including classes or object properties, in VOWLMap the editing functions on the graph were restricted to mapping properties only — users can only add mappings, remove or refine mappings.

When users hover the pointer over a class, an arrow appears on the border of the circle. By dragging this arrow from a class of one of the ontologies to another class at the opposite ontology, users can create a new mapping to the alignment and the visualization (Figure 4.4). New mappings are automatically considered *correct* and, for that reason, are represented in the graph with the green color, and assigned the maximum confidence score. In the visualization, users can only create new mappings between entities present in the graph. However, if users desire to create a mapping between entities not displayed in the visualization, they can make use of the add functionality present in the alignment panel. The new mappings created in the visualization are automatically added to the list in the alignment panel, and when exporting the final results, the new mappings are added to the exported file independently of its format.

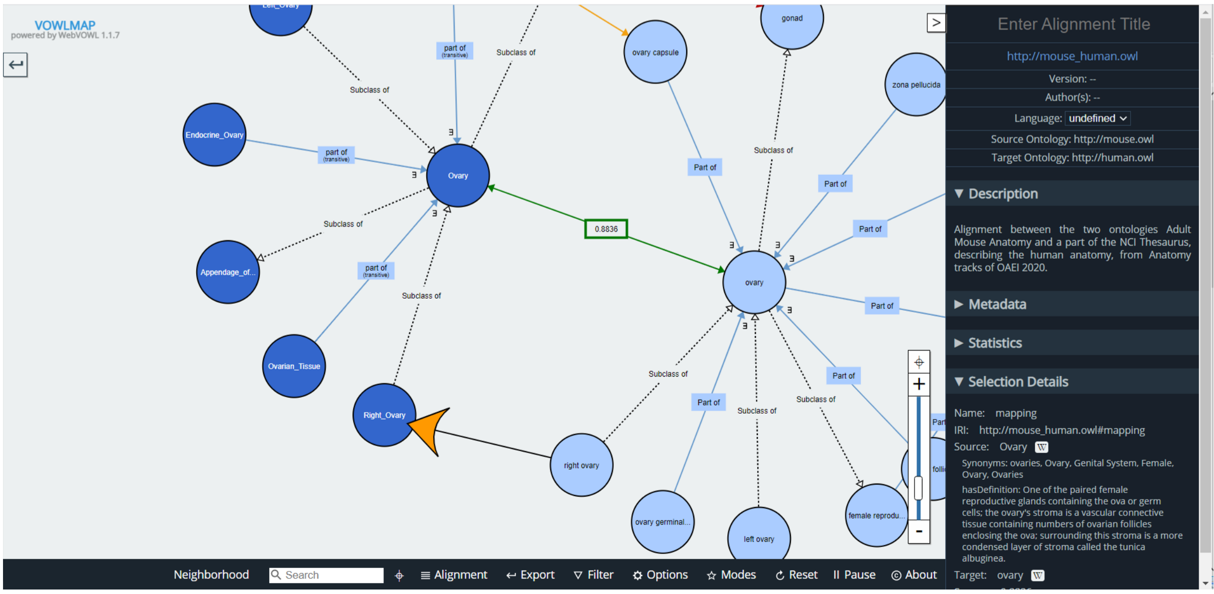


Figure 4.4: Example of creation of a new mapping between the entities *right ovary* and *Right_Ovary*.

If users hover the pointer over the score rectangle of a mapping, a clickable button appears which allows users to remove that mapping from the visualization and the alignment (Figure 4.5). When users click on the remove button, a warning appears, informing users that the selected mapping will be permanently removed from the alignment and that information will be lost, offering users the opportunity to retreat from or to confirm their decision.

Furthermore, VOWLMap allows users to manually refine mappings. When users hover the pointer over the score rectangle of a mapping, two arrows at each end of the line appear (Figure 4.5). Users can change the source or target entity to a more appropriate one by dragging any of the arrows to a new source or target entity present in the graph. This feature is especially useful when a mappings should be adjusted to a superclass or subclass of one of the entities. Once again, the list of the alignment panel and the exported file are updated with the alterations made to the target or source entity (the original mapping is replaced with the new one in the alignment).

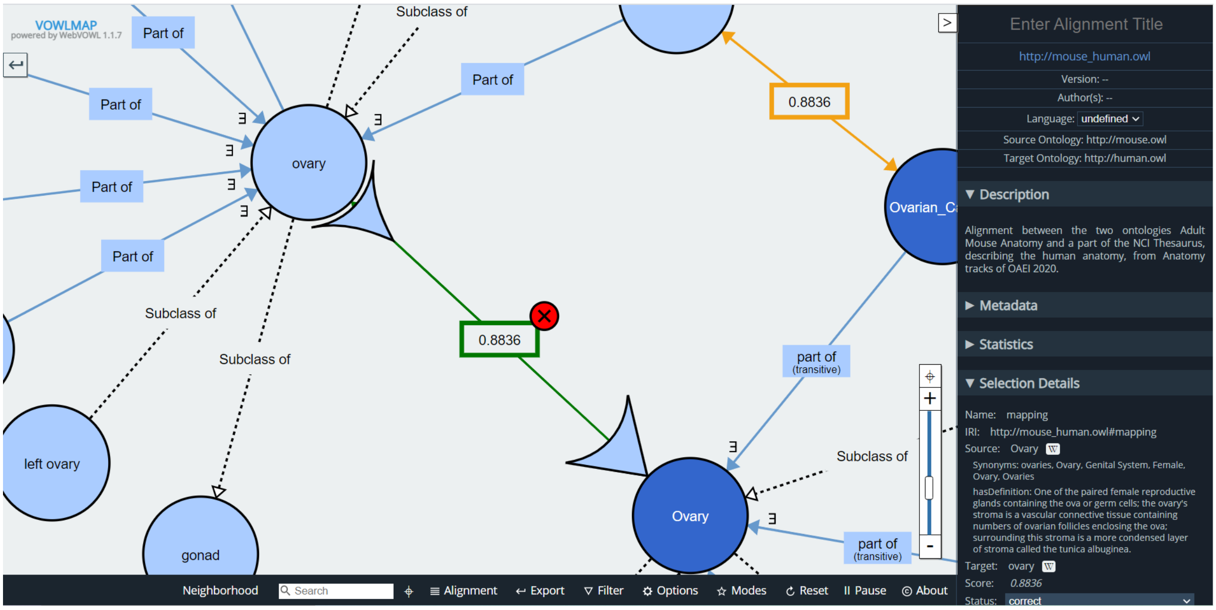


Figure 4.5: Example of graph editing functions of VOWLMap. When the pointer hovers the score rectangle the removing button and the two arrows at each end appear, to allow removing and refining of the mapping, respectively.

Chapter 5

Evaluation

This chapter describes the user study conducted to evaluate VOWLMap. The first section of this chapter introduces the general methodology implemented to perform the evaluation and the second section presents and discusses the results obtained from this evaluation.

5.1 Evaluation Methodology

According to Pesquita et al. [2018], there are six essential aspects to describe a user study in a Semantic Web context. Taking this into consideration, the next sections describe these aspects for the proposed methodology of the evaluation of VOWLMap.

5.1.1 Purpose

The main purpose of VOWLMap is, as aforementioned, to visualize, validate and edit alignments. The tasks supported by this interactive tool that are under evaluation in this user study can fit in the following categories proposed by Pesquita et al. [2018]: *creation*, since it provides means to create and edit the content of alignments (e.g. remove, create or refine mappings); and *management*, because the tool includes validation features, that allow the manipulation of the existing content.

5.1.2 Users

For this evaluation, 4 users were recruited from a pool of graduated students, with different backgrounds (life sciences, health sciences, computer science and engineering, and bioinformatics) and levels of expertise in alignment validation (Figure 5.1). All participants had prior knowledge of at least one of the domains of the aligned ontologies. It was possible to test the tool in 3 major browsers (Google Chrome, Mozilla Firefox and Safari) and 3 operating systems (Microsoft Windows, Linux and macOS). Participants of the user study cover some of the characteristics of the *target users* aforementioned for VOWLMap.

5.1.3 Tasks

The users were asked to validate two alignments from different domains with VOWLMap. Ontologies and their reference alignment from the Conference and Anatomy tracks of OAEI 2020 [1] were used as datasets for this evaluation, given that these domains are not too complex for a non-domain expert to understand.

The two alignments contained 20 mappings, 10 correct selected from the reference alignment, and 10 incorrect selected from the erroneous mappings found by the AML matching system [Faria et al., 2013]. This selection ensures that the shown incorrect mappings reflect some of the errors that automated tools make, including mappings between entities with similar labels. Users were not aware of the proportion between positive and negative mappings in the datasets.

In the first task, users were asked to validate an alignment of the conference organization domain, between two ontologies *Conference* and *ekaw*.

In the second task, users were asked to validate an alignment between the two ontologies Adult Mouse Anatomy and a part of the NCI Thesaurus, describing the human anatomy.

5.1.4 Setup

The study was an observational, task-oriented, usability study. It was performed remotely, due to the constraints imposed by the COVID-19 pandemic, with users employing VOWLMap on their machines. The meetings between me and each user were conducted in the *Zoom* platform, with recorded audio and video. Before starting recording, participants were asked to turn their camera off, change their name to "Participant" and share their computer screen.

For this study, an online platform to present instructions and collect answers was used [2]. As part of the form, instructions to download and locally run VOWLMap and to perform the requested tasks were given. Additionally, a small instructional video was also available, explaining how to use VOWLMap and some background information in how to validate alignments [3].

5.1.5 Procedure

The meeting for each user was recorded, and accompanied by a team researcher that assisted participants during the process and answered any question related to VOWLMap.

After clicking on the form link provided, participants were informed that the study was entirely voluntary and that no personal information would be collected and asked to confirm their consent to record the interview in audio and video.

In the following step, users were asked 7 multiple-choice questions, designed to gather data to characterize the user profile (e.g. background knowledge on the domains of the ontologies aligned, background

[1] http://oaei.ontologymatching.org/2020/

[2] The form is available at https://bit.ly/36HreUP

[3] Tutorial video available at: https://youtu.be/aCFtHtuN5Gk

knowledge in ontology alignment), and the machine they were operating on (e.g. browser used to run VOWLMap, operating system, display resolution).

In the next section, the participants were asked to watch a tutorial video, explaining and exploring the various features of VOWLMap. It was possible to control that participants watched the full video. After that, participants were given instructions to download and locally run VOWLMap on their machines. Note that the process of converting ontologies and alignments into JSON was not required, since the JSON files were made available to the participants.

In the following section, users were asked to validate two alignments from different domains. No specific instructions were given regarding the type or extent of validation required. During the performance of tasks, we answered questions and doubts asked by the participants regarding the functioning of VOWLMap and encouraged users to explore all VOWLMap functionalities. After each task, users were asked to upload the RDF file of their final alignment in the questionnaire platform and the amount of time that each task took.

In the last section, users rated the features of VOWLMAP in a Likert scale, ranging from “Not useful” to “Very useful” and answered the System Usability Scale (SUS) questionnaire [Brooke, 1996]. This scale allows the user to evaluate the degree of agreement or disagreement towards a certain statement on a 5 point scale, where 1 corresponds to “Strongly disagree” and 5 corresponds to “Strongly agree”. At the end of the questionnaire, an open-ended question allowed users to provide final further suggestions or feedback about VOWLMap. Additionally, all suggestions made by users during the entire process were taken into consideration.

5.1.6 Analysis

The following metrics were calculated to analyse the results:

1. **amount of time each task took**, by calculating the difference between the start and end times reported by users.
2. **frequency of correct, incorrect, and unsure mappings** in each alignment.
3. **the true positives, false positives, true negatives, and false negatives**, according to the reference alignment for each pair of ontologies.
4. **frequency of new and refined mappings**, and which ones were correct according to the reference alignment.
5. **results of features evaluation and SUS questionnaire**.

Additionally, a qualitative analysis was made during each meeting and afterward, by analysing the recorded video, to assess the interaction of users with the tool and note in how many mappings each feature was used. The observations and relevant suggestions made during the meetings are also reported in Section 4.2.

5.2 Results and Discussion

5.2.1 Users Profile

The four users recruited to participate in the study had diverse backgrounds. Figure 5.1 illustrates the results of user background and their specific knowledge related to the tasks. The majority indicated previous basic understanding of knowledge graphs and alignments as well as a background in life sciences. Regarding the machine they were operating on, User 1 used Mozilla Firefox on Linux, User 2 used Google Chrome on Microsoft Windows, User 3 used Sarafi on macOS, and User 4 used Mozilla Firefox on Microsoft Windows. The results have shown that the developed tool works perfectly in all of the operating systems and major browsers, with the exception of Safari, where it was possible to determine that some of the graph editing functions did not always function properly.

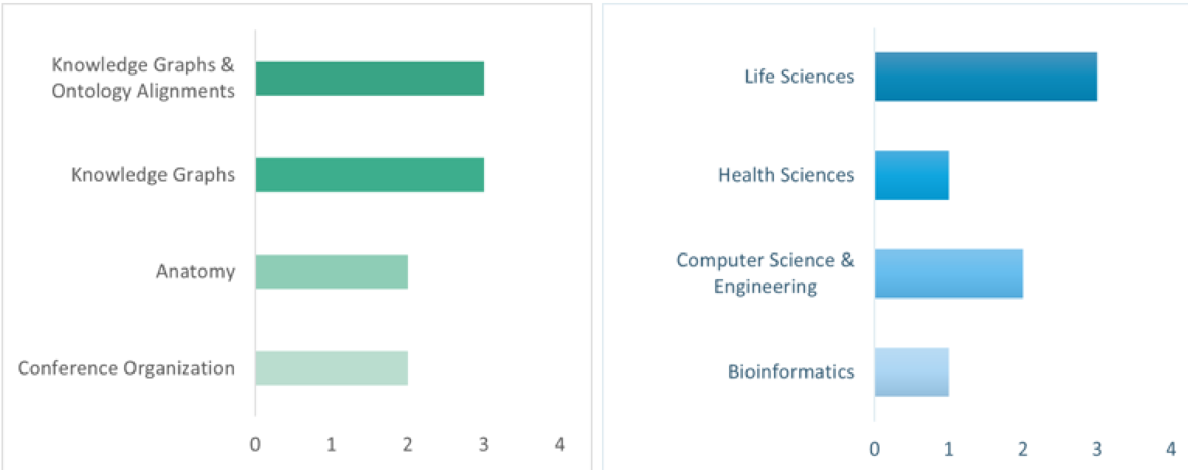


Figure 5.1: Background profile of users (left chart) and user level of expertise in the alignment validation, knowledge graphs, ontologies and the domains of the ontologies used to produce the alignments (right chart).

5.2.2 Task Success and Time Evaluation

Task success was evaluated by the frequency of mappings that were reviewed (i.e., marked as *correct*, *incorrect* or *unsure*), as well as by the frequency of true positives, false positives, true negatives and false negatives. Furthermore, the mappings created or refined by users during each task were evaluated as *correct* or *incorrect*, according to each reference alignment.

In both tasks users were able to mark most mappings as *correct* or *incorrect*, with a low number of mappings marked as *unsure* (maximum 2 out of 20). User 2 was the only one not to classify any mapping as unsure, and also the one with the highest overall accuracy, with 95% in both tasks. The proportion of *correct* and *incorrect* mappings was very similar for all users, indicating that the validation was not

Table 5.1: Evaluation of task success and time. The count of Correct, Incorrect, Unsure, New and Refined mappings is present for each user by task, as well as the count of False Positives (FP), False Negatives (FN), True Positives (TP), True Negatives (TN), the new and refined mappings that are correct according to the reference alignment, and finally, the time took in each task.

		Cor.	**Inc.**	**Uns.**	**New**	**Ref.**	**FP**	**FN**	**TP**	**TN**	**New Cor.**	**Ref. Cor.**	**Time (mm:ss)**
Task 1	User 1	7	12	1	6	0	0	3	7	9	1	-	45:54
	User 2	11	9	0	1	0	1	0	10	9	1	-	30:50
	User 3	9	9	2	0	0	1	0	8	9	-	-	14:37
	user 4	9	10	1	0	0	1	1	8	9	-	-	10:32
Task 2	User 1	12	8	0	13	0	2	0	10	8	8	-	23:51
	User 2	11	9	0	2	0	1	0	10	9	2	-	43:53
	User 3	10	10	1	2	3	1	0	10	5	2	3	19:30
	User 4	9	9	2	0	0	0	0	9	9	-	-	08:35

biased towards users considering most mappings as either *correct* or *incorrect*. All users had a fairly high accuracy classifying the mappings, ranging from 80% to 95% across the two tasks. The average accuracy was greater in Anatomy (91.3%) than Conference (86.3%) which mirrors the fact that automated alignment systems have worse results in the latter, suggesting it is a more difficult task. Interestingly, users identified erroneous mappings better in Conference than Anatomy, whereas the reverse was true for correct mappings.

Most users were able to refine and create correct mappings, which suggests that users benefit from editing features in the graph visualization for the validation process. User 3 refined three incorrect mappings to correct mappings, according to the reference alignment for the Anatomy track, which explains the lower number of true negatives in the second task, since these three incorrect mappings did not enter in this count (Table 5.1). Regarding the creation or refinement of mappings, User 1 was the most prolific but had an overall precision of only 47.4%. Users 2 and 3 created or refined only a few mappings, but with 100% precision, whereas User 4 neither created nor refined mappings. As users were asked to not remove any of the original 20 mappings but rather mark them as incorrect, the use of the removing function is not reflected by Table 5.1. However, during both tasks, this feature was used by three users, to remove mappings they previously added (Table 5.4).

Even though users validate mappings in a different order in both tasks, this factor does not seem to have an impact on the time taken by users in each validation (Figures 5.2 and 5.3). Users did take more time in average when validating the alignment from the biomedical domain, which can be explained by the greater use of the editing features in this task that are time consuming (Figure 5.4). Moreover, it was frequently to have users validating more than one mapping in the same visualization (mapping being visualized and the nearby mappings), and, in that case the amount of time was equally divided by

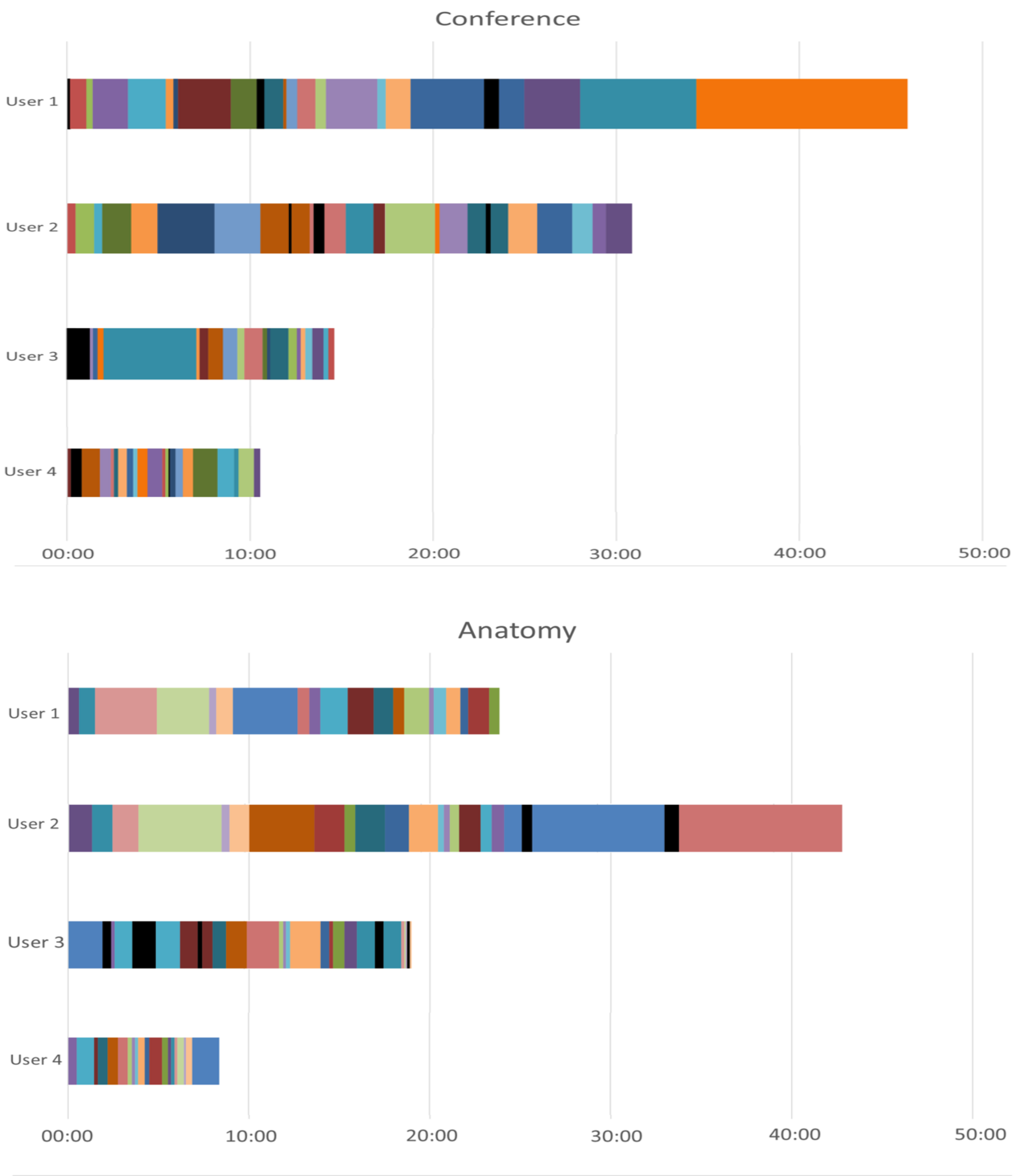


Figure 5.2: Timelines for Task 1 (Conference) and Task 2 (Anatomy). Each color represents a different mapping, with black representing interruptions for requesting help or clarifications about VOWLMap.

the mappings validated in one single visualization. The simultaneous validation of mappings was used in abundance and considered very useful by users, especially when users extended the neighborhood (Figure 5.4).

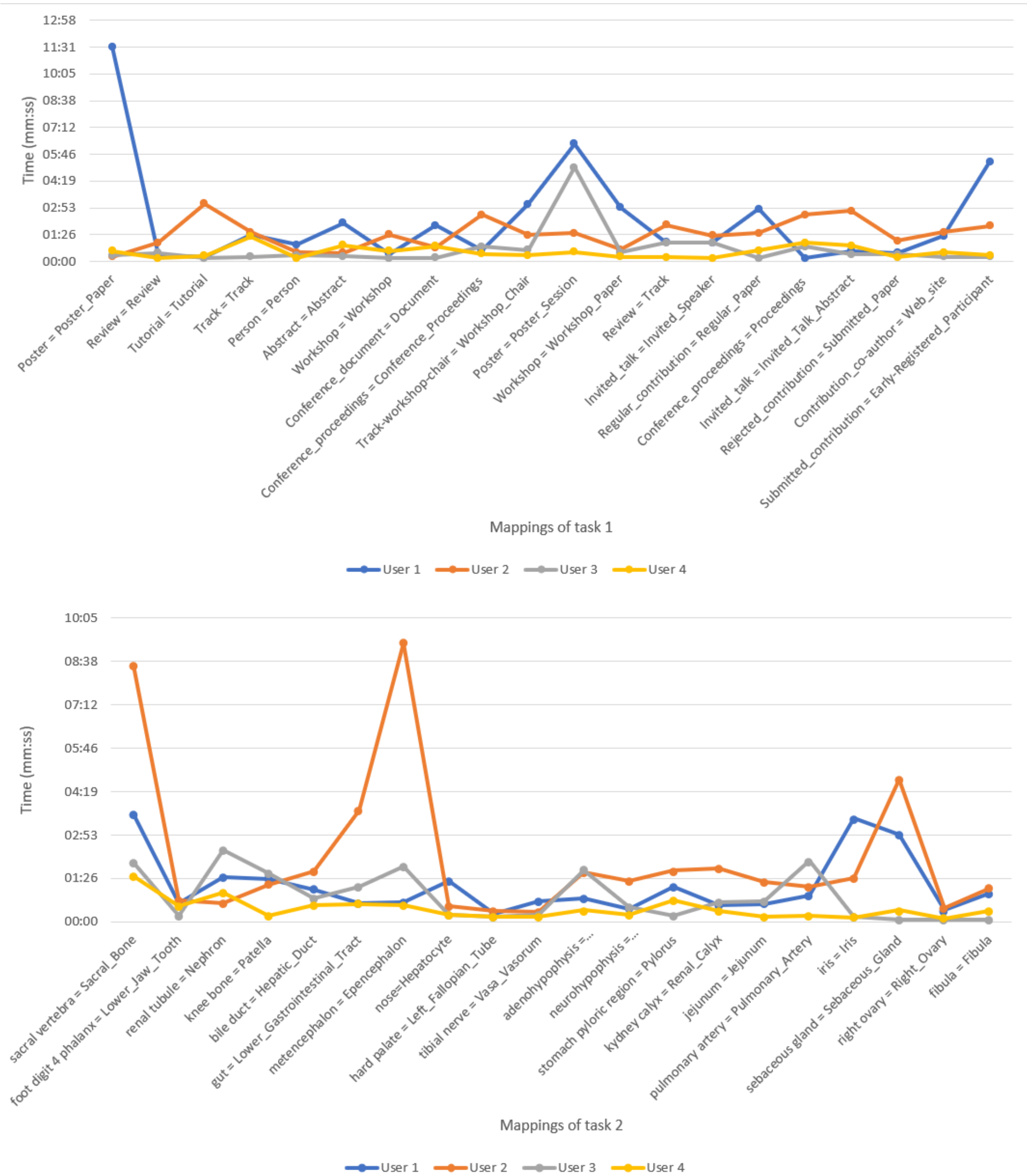


Figure 5.3: Evaluation of the time taken by users for each mapping in Task 1 (above) and in Task 2 (below). The order of validation was not considered in this representation.

All users requested help or clarifications about VOWLMap, but time spent on this was comparatively short (2.5% - 9.8%). Familiarity with VOWLMap seemingly had no bearing on the speed or accuracy with which users classified mappings, as initial mappings had neither higher time nor lower accuracy on average. On the contrary, Users 3 and 4 seem to have left mappings they found more challenging to classify for last, taking a lot more time in these (Figure 5.2). Interruptions for help or clarification were also not concentrated at the start of the task, although Users 1 and 2 did ask for clarifications at the start of the evaluation. Overall, these facts speak well to the intuitiveness of VOWLMap's visualizations and functionalities.

5.2.3 Feature Evaluation

The participants were asked to rate the usefulness of a number of VOWLMap features. In general, the results showed that users were able to make use of the various features under evaluation to support both tasks (Figure 5.4). Table 5.2 presents the results of features evaluation for each user, as well as the overall mean for each feature.

Overall, all the features received elevated scores, with all features except *Wikipedia Links* presenting a mean rate above 4, showing that users considered most features highly useful for the purpose of visualization and validation of ontology alignments. *Graph Visualization* and *Graph Interactivity* were consistently the highest rated features, with all of the users assigning the maximum score to these features, as well as the most used by users in both tasks (Figure 5.4).

Two users find the validation in the graph more useful than the validation in the list, but they also attribute an elevated score to the *List Validation*. User 4 was the only to consider the *List Validation* more useful than the *Graph Validation*, which is confirmed by the preference for the use of the *List Validation* in the second task. Nevertheless, even when users chose to validate mappings in the list, the *Graph Visualization* and *Graph Interactivity* were very used (Table 5.4). Moreover, the *Graph Interactivity* was the most used feature by all of the users in both tasks, which reveals that, regardless of the way a mapping is validated, the interactive visualization is an advantage in the validation process.

Table 5.2: Features Evaluation by each user.

	List Visualization	List Validation	List Editing	Graph Visualization	Graph Validation	Graph Editing	Graph Interactivity	Wikipedia Links
User 1	5	5	5	5	5	5	5	4
User 2	4	4	5	5	5	5	5	4
User 3	5	4	5	5	5	5	5	4
User 4	5	5	4	5	4	3	5	4
Mean	4.73	4.72	4.73	5	4.73	4.40	5	4

The editing features were considered very useful for all the users, with the exception of User 4. The lowest score attributed can be justified by the lack of exploring of these features, since the user did not feel the need to use any of the two editing functions during both tasks (Table 5.4). The only feature that was scored 3 by any user was *Graph Editing*, by User 4, who notably was the only user that attempted neither additions nor refinements of mappings, making the least use of this functionality among the four users.This functionality was more used in Anatomy than in Conference by the other three users, as they created and refined more mappings in the former than the latter. Even though *List Editing* was considered very useful for most of the users, this feature was not very used by users in both tasks (Figure 5.4).

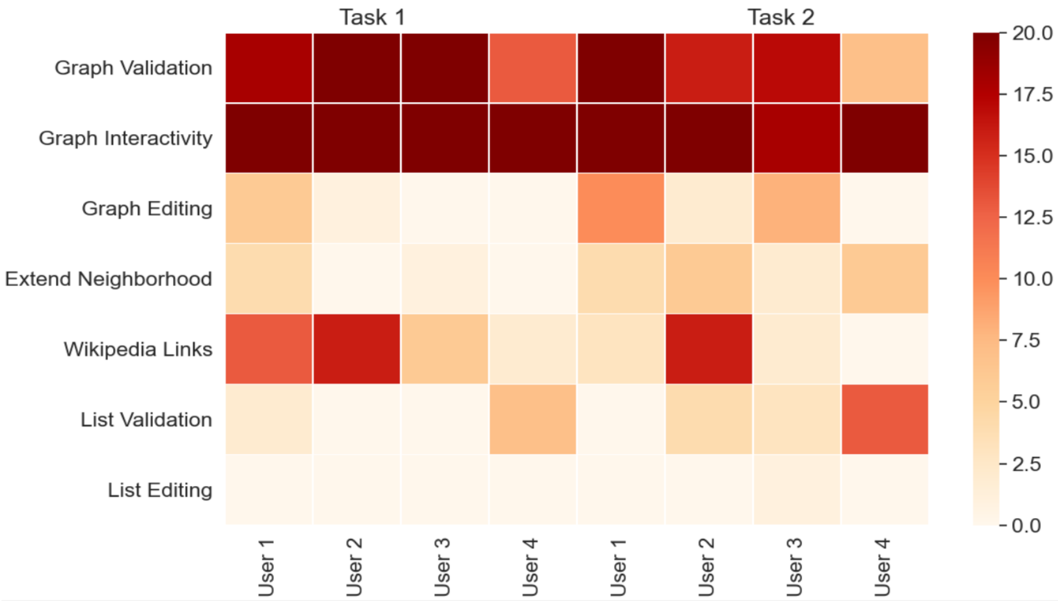


Figure 5.4: Frequency of mappings where users use each feature for each task. The scale ranges from 0 mappings (light color) to 20 mappings (dark color).

Wikipedia Links were consistently rated with the score 4 by all users. The slightly lower score can be explained by the fact that, as the links are automatically generated by the source and target labels, sometimes there are no Wikipedia pages available for the terms or in other cases users are redirected to the same page for both source and target when their labels are very similar or equal. In these cases, Wikipedia links are not useful and do not provide extra information to help users in the decision process.

Considering the results presented in Figure 5.4, it is possible to conclude that users used more features, in particularly extending the neighborhood and the editing features, when validating an alignment from the biomedical domain (task 2). Based on that, it is possible to reach the conclusion that the features offered by VOWLMap are more suitable to validate an alignment between ontologies from the biomedical domain.

5.2.4 Usability Evaluation

Users were also asked to respond to a SUS questionnaire. To calculate the final value, first was subtracted to 5 the values answered in even question and, then, the odd numbers were subtracted by one. In the second phase, all the values obtained in the previous phase were summed up and then multiplied by 2.5. The responses of each user to the questionnaire, as well as the final score obtained for each one are summarized in Table 5.3. According to Brooke [1996], 68 is considered the average, and the maximum rate (excellent) is attributed to scores above 80.3. Considering the overall final SUS score obtained (85), the usability of VOWLMap corresponds to excellent.

Table 5.3: Results of the System Usability Scale (SUS) questionnaire for VOWLMap.

Question	**User 1**	**User 2**	**User 3**	**User 4**
I think that I would like to use this system frequently.	4	5	4	5
I found the system unnecessarily complex.	2	1	2	2
I thought the system was easy to use.	4	5	4	4
I think that I would need the support of a technical person to be able to use this system.	2	1	1	1
I found the various functions in this system were well integrated.	4	5	5	4
I thought there was too much inconsistency in this system.	2	1	4	1
I would imagine that most people would learn to use this system very quickly.	3	5	5	5
I found the system very cumbersome to use.	2	1	1	1
I felt very confident using the system.	5	5	4	4
I needed to learn a lot of things before I could get going with this system.	4	1	1	2
SUS score	**70**	**100**	**82.5**	**87.5**

5.2.5 Qualitative Evaluation

In addition to the task-oriented and usability questionnaire, users were also asked to give feedback in an open question. One user suggested that Wikipedia links should be available for all classes, and not only for the source and target classes of mappings.

Table 5.4: Features used by each user in the validation process. The results show in how many mappings, from both tasks, each feature was used.

	Features Used	Count of Mappings
User 1	Graph validation	38
	Graph Interactivity	40
	Graph Editing	16
	Extend Neighborhood	16
	Wikipedia links	8
	List Validation	2
	List Editing	0
User 2	Graph validation	36
	Graph Interactivity	40
	Graph Editing	3
	Extend Neighborhood	32
	Wikipedia links	6
	List Validation	4
	List Editing	0
User 3	Graph validation	37
	Graph Interactivity	38
	Graph Editing	8
	Extend Neighborhood	8
	Wikipedia links	3
	List Validation	3
	List Editing	1
User 4	Graph validation	20
	Graph Interactivity	40
	Graph Editing	0
	Extend Neighborhood	2
	Wikipedia links	6
	List Validation	20
	List Editing	0

Chapter 6

Discussion

Developing a tool for the visualization and validation of biomedical ontology alignments is not a trivial task, as most biomedical experts are generally nor ontology experts, and it is necessary to consider the characteristics of these ontologies, as well to balance the need of effectively displaying all the information with the need of providing the functionalities required to perform the validation and make this task seamless for the user. In general, the developed work focused on addressing the challenges in visualizing biomedical ontology alignments and overcoming the existing limitations of existing systems for alignment visualization.

A considerable challenge faced when developing VOWLMap was the large size visualization of biomedical ontologies and alignments between them. To tackle this issue, VOWLMap employs modularization techniques to facilitate the visualization of large alignments while maintaining the context of each individual mapping. This is achieved by only loading the information that will be displayed in the visualization, discarding all entities that are not at a maximum distance of 3 from each mapping and reducing the size of the file. Besides that, the fact that the graph visualization is generated for each mapping, allows to compartmentalize the information that is related, avoiding overwhelming the user with a visualization of the entire alignment.

Moreover, VOWLMap supports diverse interaction mechanisms with the alignment, where users can interact with the graph layout, disposing nodes and edges as better suits them, and directly edit the graph representations. This type of dynamic visualization is not employed by any visualization of the described systems. Furthermore, this interactive and dynamic visualization allied with the extension of an already existing visual notation, is an essential advantage compared to other systems, especially in the biomedical domain, where users are often less familiar with ontology alignments and highly benefit from the visualization. The extension of the VOWL notation includes graphical primitives to represent ontology elements and mappings that are independent of the color scheme. As a result, VOWLMap is a more inclusive system, as it is the only one of the systems described that has a visual interface capable of being used by colorblind people.

Most of the available interfaces are coupled with automated ontology alignment systems, which theoretically can support more sophisticated types of interaction that may happen during the alignment process, but in practice can make them less versatile by tying visualizations to specific implementation technologies. By being developed with a browser-based architecture, VOWLMap enables an easier use by domain experts, since no specific software needs to be installed.

When comparing the functional requirements supported by VOWLMap and the systems described in Section 3.2, it is possible to recognize an improvement in the support for visualization, manual validation and editing of alignments. Table 6.1 summarizes the previous defined requirements and how they are met by the systems described in Section 3.2 and VOWLMap.

Table 6.1: Requirements addressed by the alignment systems with visualization and VOWLMap. In the table ✓marks that all of the requirement is supported by the tool, the ✓✗marks that the requirement is not entirely supported and the ✗marks that the requirement is not supported at all.

	AM	AML	YAM++	LogMap	VocBench	VOWLMap
(R1) Load Alignments	✓	✓	✓	✓	✓	✓
(R2) Alternative Views	✗	✓	✗	✗	✗	✓
(R3.1) Overview	✓	✓	✓	✗	✓	✓
(R3.2) Zoom	✓	✓	✓	✗	✗	✓
(R3.3) Filter	✓	✗	✗	✗	✗	✓
(R3.4) Details-on-demand	✓	✗	✗	✗	✗	✓
(R3.5) Relate	✓	✓	✓	✗	✗	✓
(R3.6) History	✓	✗	✗	✗	✗	✗
(R3.7) Extract	✓	✗	✗	✗	✗	✓
(R4) Mapping Status	✓	✓	✓	✗	✓	✓
(R5) Metadata	✓	✓	✗	✓	✓	✓
(R6) Context	✓✗	✓	✗	✓	✗	✓
(R7) Accept/Reject mapping	✓	✓	✓	✓	✓	✓
(R8) Create/Refine mapping	✓✗	✓✗	✓✗	✗	✗	✓
(R9) Search	✗	✓	✓	✗	✗	✓
R(10) Session	✓✗	✓✗	✓✗	✓	✓✗	✓
(R11) Export Alignments	✓	✓	✓	✓	✓	✓

In opposition to most of the systems covered, VOWLMap offers six of the seven visual information-seeking tasks [Shneiderman, 2000]. These features are particularly important to achieve the desired balance between providing information and avoiding memory overload, as well as support the exploration and retrieval of information. In addition to the metadata and context contained in both ontologies, VOWLMap provides an external link to Wikipedia, that can be determinant in the decision-making process, providing additional information and saving users time. None of the described systems provide a link to an external source of information.

Although most of the systems allow to manually create new mappings, VOWLMap offers a refinement functionality, not supported by any of these systems, by allowing to change the source or the target of a certain mapping directly in the visualization.

In addition, the search functionality offered by VOWLMap extends the ones offered by AML and YAM++, allowing not only the search for candidate mappings, but also the search for specific ontology entities. This last feature is particularly important to allow users to analyse the structural context of a candidate mapping or to look for ontology entities to map in the visualization. Besides that, YAM++ and AML only allow the search in the list view, while in VOWLMap users can access the search mechanism in both list and graph visualization, which allows user to easily switch from a visualization of one mapping to another, without having to leave the graph visualization.

With the exception of LogMap, most of the alignment systems with visualization only support the session requirement indirectly, by allowing users to save and load alignments. VOWLMap is the only of the addressed systems that directly accommodates sessions, i.e., interruptions in the validation process, allowing users to validate mappings in one or more sessions, without losing any information of the previous validated mappings and without having to save and reload the alignment into the tool.

Overall, the developed tool satisfies the combination of all the requirements not addressed by the existing systems, overcoming their limitations and providing new functionalities. Besides that, VOWLMap offers a new intuitive and interactive visualization, while fulfilling the main objectives defined for the present work. The results obtained in the user evaluation support the usability of the tool and its performance, showing that users not only benefit from the visualization and features employed when validating an alignment, but also they likely succeed in this task.

Chapter 7

Conclusion

Visualization of ontology alignments is the key feature to support user validation of an alignment. Most of the current ontology matching systems provide visualization strategies that do not consider the characteristics of biomedical ontologies, and as a result, they are not entirely suitable for the visualization, exploration and validation of alignments in the context of the biomedical domain. The present work focused on overcoming the limitations in user interaction and visualization of biomedical ontology alignments, existing in current approaches.

This dissertation proposed, developed and evaluated a browser-based tool for biomedical ontology alignment visualization, validation and editing: VOLWMap. Overall, VOWLMap fulfilled the goals for which it was designed, as it complies with all the requirements for user validation of ontology alignments, laid out in Li et al. [2019] (albeit partially, in the case of the visual information seeking tasks), which sets it above established ontology alignment systems.

Unlike most alignment visualization and validation tools, VOWLMap is independent of an alignment system, which offers a simpler use by the target users, and employs a visual notation (VOWL), that allows a more intuitive and comprehensive representation of alignments and can be easily understandable by users less familiar with ontologies and their alignments. Moreover, this notation was extended in this work, standardizing the representation of a mapping context and alignments when using this notation. Furthermore, the Alignment format was also extended to allow saving the information obtained in the validation process.

The visualization of large ontologies, such as ontologies from the biomedical domain, cause serious scalability problems. One of the main challenges of this work was to offer a suitable environment for the representation of large ontologies, by employing modularization techniques to present the visualization of an alignment as smaller and interconnected graph visualizations.

VOWLMap was evaluated in a small user study, revealing that it is intuitive and easy to use, as no learning curve was observed with respect to the time or accuracy of the validation tasks. Moreover, users made use of most of VOWLMap's features, and generally considered them useful in the feedback they provided. Although user tests were small-scale, as it was an observation study, it was possible to evaluate how users interacted with the tool and explored the available features. The success of both tasks for all users revealed that VOWLMap is capable of offering visual support and functionalities to allow users to interact and successfully validate an alignment, without having necessarily domain or knowledge engineering and ontology alignments expertise.

7.1 Limitations and Future Work

Despite the amount of features already provided by VOWLMap, there is also space for improvements. The main limitation of this work concerns the creation of new mappings during the validation process. The file that is uploaded to VOWLMap contains only the neighborhood of the original alignment, and, for that reason, the tool only allows the creation of new mappings between entities present in the neighborhood of already existing mappings. Furthermore, since not all entities are present in the JSON file, when the tool generates the graph visualization for a new mapping, the neighborhood may not be complete. This limitation was a necessary trade-off to optimize the uploading and visualization of large alignments.

Another limitation of VOWLMap is that it only supports the visualization and validation of mappings between class entities of ontologies. This occurs because WebVOWL itself does not support a representation between two properties or between individuals and an implementation of this kind of representation would imply more profound and extensive alterations in the platform. In the future, it would be interesting to further extend the VOWL notation to support a graphical representation of mappings between properties, and provide in VOWLMap the necessary features allow the edition and validation of such mappings.

Furthermore, the user study provided feedback on what could be improved in VOWLMap. One limitation that was observed during the user study was that the browser Safari did not support some of the editing functions. Future work may include the refinement of the tool so that the totality of the features could be used in this browser, as well as integrate some of the qualitative assessments made by the users to further improve VOWLMap.

Another direction for future work is to add features to provide overall statistics to assist in the validation process monitoring, such as displaying the mapping coverage for the aligned ontologies, the number of mappings reviewed by the current or previous user, and the number of changes made to that point. Additionally, it would be interesting to support validation of inter-annotator agreement, by allowing experts to exchange their intermediate results or by storing the revision along with the changes made by each author.

Moreover, the performed user study presented some limitations, as the users were all students and only covered some of the characteristics of the target user study, which limits the assessment of the suitability of the tool for these users. For that reason, there is still a need to perform a broader evaluation, with a bigger number of users that are not students and that fulfill the characteristics of the target users.

Finally, it would be interesting to perform a comparative evaluation with the available systems for ontology alignments visualization and validation, with in-person observations, to compare and assess if users find VOWLMap better suited for the purpose of validating biomedical ontologies alignments and to delve deeper into how users interact with the tool.

References

Bellahsene, Z., Emonet, V., Ngo, D., and Todorov, K. (2017). Yam++ online: A web platform for ontology and thesaurus matching and mapping validation. In Blomqvist, E., Hose, K., Paulheim, H., Ławrynowicz, A., Ciravegna, F., and Hartig, O., editors, *The Semantic Web: ESWC 2017 Satellite Events*, pages 137–142, Cham. Springer International Publishing. 23

Bosca, A., Bonino, D., and Pellegrino, P. (2005). Ontosphere: more than a 3d ontology visualization tool. In *Swap*. Citeseer. 13, 15

Bostock, M., Ogievetsky, V., and Heer, J. (2011). D³ data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309. 18, 27

Brooke, J. (1996). *SUS: A quick and dirty usability scale*, volume 189, pages 189–194. London: Taylor and Francis. 41, 48

Catenazzi, N., Sommaruga, L., and Mazza, R. (2009). User-friendly ontology editing and visualization tools: the owleasyviz approach. In *2009 13th International Conference Information Visualisation*, pages 283–288. IEEE. 13

Cruz, I. F., Antonelli, F. P., and Stroe, C. (2009). Agreementmaker: Efficient matching for large real-world schemas and ontologies. *Proc. VLDB Endow.*, 2(2):1586–1589. 20, 21

da Silva, I. C. S., Freitas, C. M. D. S., and Santucci, G. (2012). An integrated approach for evaluating the visualization of intensional and extensional levels of ontologies. In *BELIV '12: Proceedings of the 2012 BELIV Workshop: Beyond Time and Errors - Novel Evaluation Methods for Visualization*, BELIV '12, New York, NY, USA. Association for Computing Machinery. 13

Dudáš, M., Lohmann, S., Svátek, V., and Pavlov, D. (2018). Ontology visualization methods and tools: a survey of the state of the art. *The Knowledge Engineering Review*, 33:e10. 11, 14

Euzenat, J. (2004). An api for ontology alignment. In McIlraith, S. A., Plexousakis, D., and van Harmelen, F., editors, *The Semantic Web - ISWC 2004*, pages 698–712, Berlin, Heidelberg. Springer Berlin Heidelberg. 8, 9, 30

Euzenat, J., Meilicke, C., Stuckenschmidt, H., Shvaiko, P., and Trojahn, C. (2011). Ontology alignment evaluation initiative: Six years of experience. *J. Data Semantics*, 15:158–192. 7, 8

Euzenat, J. and Shvaiko, P. (2007). *Ontology matching*. Springer-Verlag, Heidelberg (DE). 1, 7

Faria, D., Pesquita, C., Mott, I., Martins, C., Couto, F., and Cruz, I. (2018). Tackling the challenges of matching biomedical ontologies. *Journal of Biomedical Semantics*, 9. 1, 7

Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I. F., and Couto, F. M. (2013). The agreementmakerlight ontology matching system. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*, pages 527–541, Berlin, Heidelberg. Springer Berlin Heidelberg. 20, 21, 40

Granitzer, M., Sabol, V., Onn, K. W., Lukose, D., and Tochtermann, K. (2010). Ontology alignment—a survey with focus on visually supported semi-automatic techniques. *Future Internet*, 2(3):238–258. 1, 8, 20, 21

Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220. 5

Guarino, N., Oberle, D., and Staab, S. (2009). *What Is an Ontology?*, pages 1–17. Springer Berlin Heidelberg, Berlin, Heidelberg. 5

Guerreiro, A., Pesquita, C., and Faria, D. (2021). Vowlmap. doi: 10.5281/zenodo.5807866. 27

Hoehndorf, R., Schofield, P. N., and Gkoutos, G. V. (2015). The role of ontologies in biological and biomedical research: a functional perspective. *Briefings in Bioinformatics*, 16(6):1069–1080. 1, 2

Horridge, M. and Bechhofer, S. (2011). The owl api: A java api for owl ontologies. *Semantic Web*, 2:11–21. 18

Horrocks, I. (2008). Ontologies and the semantic web. *Commun. ACM*, 51(12):58–67. 6

Horrocks, I. and Patel-Schneider, P. F. (2011). Knowledge representation and reasoning on the semantic web: Owl. *Handbook of semantic web technologies*, pages 365–398. 5, 6

Jiménez-Ruiz, E. and Cuenca Grau, B. (2011). Logmap: Logic-based and scalable ontology matching. In Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N., and Blomqvist, E., editors, *The Semantic Web – ISWC 2011*, pages 273–288, Berlin, Heidelberg. Springer Berlin Heidelberg. 24

Katifori, A., Halatsis, C., Lepouras, G., Vassilakis, C., and Giannopoulou, E. (2007). Ontology visualization methods—a survey. *ACM Comput. Surv.*, 39(4):10–es. 11

Li, H., Dragisic, Z., Faria, D., Ivanova, V., Lambrix, P., Jiménez-Ruiz, E., and Pesquita, C. (2019). User validation in ontology alignment: functional assessment and impact. *The Knowledge Engineering Review*, 34:e15. 1, 2, 9, 10, 20, 27, 55

Lohmann, S., Negru, S., Haag, F., and Ertl, T. (2014). Vowl 2: User-oriented visualization of ontologies. In Janowicz, K., Schlobach, S., Lambrix, P., and Hyvönen, E., editors, *Knowledge Engineering and Knowledge Management*, volume 8876, pages 266–281, Cham. Springer International Publishing. 14

Lohmann, S., Negru, S., Haag, F., and Ertl, T. (2016). Visualizing ontologies with vowl. *Semantic Web*, 7(4):399–419. 15, 17, 18, 19

Ozturk, O. and Açikgoz, H. G. (2020). Onyx: A new canvas-based tool for visualizing biomedical and health ontologies. *Expert Systems*, 37(1):e12380. e12380 10.1111/exsy.12380. 11, 12

Pesquita, C., Faria, D., Santos, E., Neefs, J., and Couto, F. (2014). Towards visualizing the alignment of large biomedical ontologies. In *Data Integration in the Life Sciences*. 7, 9, 20

Pesquita, C., Ivanova, V., Lohmann, S., and Lambrix, P. (2018). A framework to conduct and report on empirical user studies in semantic web contexts. In Faron Zucker, C., Ghidini, C., Napoli, A., and Toussaint, Y., editors, *Knowledge Engineering and Knowledge Management*, pages 567–583, Cham. Springer International Publishing. 39

Pour, N., Algergawy, A., Amini, R., Faria, D., Fundulaki, I., Harrow, I., Hertling, S., Jimenez-Ruiz, E., Jonquet, C., Karam, N., et al. (2020). Results of the ontology alignment evaluation initiative 2020. In *Proceedings of the 15th International Workshop on Ontology Matching (OM 2020)*, volume 2788, pages 92–138. CEUR-WS. 8

Rubin, D. L., Shah, N. H., and Noy, N. F. (2007). Biomedical ontologies: a functional perspective. *Briefings in Bioinformatics*, 9(1):75–90. 1

Shneiderman, B. (2000). The eyes have it: A task by data type taxonomy for information visualizations. *IEEE Symposium on Visual Languages, Proceedings*. 28, 53

Shvaiko, P. and Euzenat, J. (2013). Ontology matching: State of the art and future challenges. *Knowledge and Data Engineering, IEEE Transactions on*, 25:158–176. 1, 6

Stellato, A., Fiorelli, M., Turbati, A., Lorenzetti, T., van Gemert, W., Dechandon, D., Laaboudi-Spoiden, C., Gerencsér, A., Waniart, A., Costetchi, E., and Keizer, J. (2020). VocBench 3: A collaborative Semantic Web editor for ontologies, thesauri and lexicons. *Semantic Web*, 11(5):855–881. Publisher: IOS Press. 24

Storey, M.-A., Musen, M., Silva, J., Best, C., Ernst, N., Fergerson, R., and Noy, N. (2001). Jambalaya: Interactive visualization to enhance ontology authoring and knowledge acquisition in protégé. In *Workshop on interactive tools for knowledge capture*, volume 73. 13

Wiens, V., Lohmann, S., and Auer, S. (2018). Webvowl editor: Device-independent visual ontology modeling. In *Proceedings of the ISWC 2018 Posters & Demonstrations, Industry and Blue Sky Ideas Tracks*, volume 2180 of *CEUR Workshop Proceedings*. CEUR-WS.org. 19

Appendix A

Alignment RDF Format Extension

Formal description

revision element

The *revision* element is an attribute of the *Cell* element, and contains the information resulting from the revision or validation of a specific correspondence between entities of the ontologies. It contains one attribute:

- *status* element

  (value: String; default: unreviewed) contains a string identifying the result from the revision or validation of the relation between the first and the second entity. This may be given by four different values: unreviewed (default), correct, incorrect and unsure.

```
<Cell>
        <entity1 rdf:resource="http://mouse.owl#MA_0001705"/>
        <entity2 rdf:resource="http://human.owl#NCI_C33487"/>
        <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.94</measure>
        <relation>=</relation>
        <revision>
                <status>unreviewed</status>
        </revision>
</Cell>
```

- *author* element

  (value: String;) contains a string identifying the author of that revision or validation.

```
<Cell>
        ...
        <revision>
                <status>unreviewed</status>
                <author>John</author>
        </revision>
</Cell>
```

- *timestamp* element

  (value: String;) contains a string identifying the time that that revision or validation occurred.

```
<Cell>
        ...
        <revision>
                <status>unreviewed</status>
                <author>John</author>
                <timestamp>2021-09-04*13:23:55</timestamp>
        </revision>
</Cell>
```

Figure A.1: Formal description of the extension of the Alignment RDF Format.