User validation in ontology alignment

Zlatan Dragisic[1], Valentina Ivanova[1], Patrick Lambrix[1],
Daniel Faria[2], Ernesto Jiménez-Ruiz[3], and Catia Pesquita[4]

[1] Linköping University and the Swedish e-Science Research Centre, Sweden
[2] Gulbenkian Science Institute, Portugal
[3] University of Oxford, UK
[4] LaSIGE, Faculdade de Ciências, Universidade de Lisboa, Portugal

Abstract. User validation is one of the challenges facing the ontology alignment community, as there are limits to the quality of automated alignment algorithms. In this paper we present a broad study on user validation of ontology alignments that encompasses three distinct but interrelated aspects: the profile of the user, the services of the alignment system, and its user interface. We discuss key issues pertaining to the alignment validation process under each of these aspects, and provide an overview of how current systems address them. Finally, we use experiments from the Interactive Matching track of the Ontology Alignment Evaluation Initiative (OAEI) 2015 to assess the impact of errors in alignment validation, and how systems cope with them as function of their services.

1 Introduction

The growth of the ontology alignment area in the past years has led to the development of many ontology alignment systems. In most cases, these systems apply fully automated approaches where an alignment is generated for a given pair of input ontologies without any human intervention. However, after many editions of the Ontology Alignment Evaluation Initiative (OAEI), it is becoming clear to the community that there are limits to the performance (in terms of precision and recall of the alignments) of automated systems, as adopting more advanced alignment techniques has brought diminishing returns [40,21]. This is likely due to the complexity and intricacy of the ontology alignment process, with each task having its particularities, dictated by both the domain and the design of the ontologies. Thus, automatic generation of mappings should be viewed only as a first step towards a final alignment, with validation by one or more users being essential to ensure alignment quality [12].

Having users validate an alignment enables the detection and removal of erroneous mappings, and potentially the addition of alternative mappings, or altogether new ones, not detected by the alignment system. Additionally, if user validation is done during the alignment process, it enables the adjustment of system settings, the selection of the most suitable alignment algorithms, and the incorporation of user knowledge in them [40]. While users can make mistakes, experiments have shown that user validation is still beneficial up to an error rate of 20% [26], although the exact error threshold will depend on the alignment system and how it makes use of the user input.

The relevance of user involvement in ontology alignment is evidenced by the fact that nearly half of the challenges facing the community identified in [46] are directly

related to it. These include *explanation of matching results* to users, fostering *user involvement* in the matching process, and *social and collaborative matching*. Moreover, the lack of evaluation of the quality and effectiveness of user interventions was identified as one of the general issues after six years of experience in the OAEI [12], leading to the introduction of the Interactive Matching track in the OAEI 2013 campaign [40] where user validation was simulated using an oracle. This track was extended in 2015 [4] to also take into account erroneous user feedback to the systems as well as additional use cases.

There have been earlier studies addressing user involvement in ontology alignment and identifying and evaluating the requirements and techniques involved therein [14,17,21,28]. More recently, the requirements for fostering user support for large-scale ontology alignment were identified and current systems were evaluated [24]. However, these studies focused mostly on the user interface of alignment systems. While that is a critical aspect for user involvement, there are other important aspects which have been largely unaddressed, such as how systems cope with erroneous user input or how they maximize the value of limited input.

In this paper we present a broader study of user validation in ontology alignment. In Section 2, we identify the key issues regarding user validation of ontology alignments by reviewing existing systems and literature related to ontology alignment, as well as drawing from our experience in the field. These issues pertain to three categories: the user profile, the alignment systems' services and their user interfaces. In Section 3, we first assess how current systems deal with the identified issues in a qualitative evaluation (Subsection 3.1), then use the experiments from the Interactive Matching track of the OAEI 2015 campaign to show how some of these issues impact alignment quality (Subsection 3.2). While the experiments from the OAEI Interactive track considered the erroneous input as a function solely of user knowledge, here we discuss them in light of different aspects of user expertise.

2 Issues regarding user alignment validation

Alignment validation requires users to first become familiar with the ontologies and their formal representations, and to grasp the view of the ontology modelers, before being able to understand and decide on the mappings provided by an alignment system or creating mappings by themselves [15]. Thus, it is a cognitively demanding task that involves a high memory load and complex decision making, and is inherently error-prone because of different levels of expertise, differences in interpretation or perception, and human biases [17].

There are three categories of issues that affect alignment validation: the profile of the user, i.e., his domain and technical expertise, and his expertise with the alignment system (Subsection 2.1); the system services, concerning how systems formulate user interactions and how they capitalize on user input (Subection 2.2); and the user interfaces, including the impact of visualization and interaction strategies on the alignment validation process (Subsection 2.3).

2.1 User profile

The **domain expertise of the user** concerns his knowledge about the domain of the aligned ontologies, and therefore his ability to assess the correctness of a mapping conceptually (e.g., whether two ontology classes mapped as equivalent actually represent the same concept in the domain). Thus, domain expertise is critical for alignment quality, and the lack thereof is likely to be the main source of erroneous input from a user, particularly in specialized domains with complex terminology such as the life sciences.

The **technical expertise of the user** pertains to his knowledge about ontologies themselves, and his experience in knowledge engineering and modeling, and therefore his ability to assess the correctness of a mapping formally (i.e., whether a mapping is logically sound given the constraints of the two ontologies). While domain knowledge is critical for alignment validation, domain experts are often not familiar with knowledge engineering concepts and formal representations [5], and may have difficulty grasping the consequences of a mapping in the context of the ontologies, or even in perceiving subtle differences in modeling that make that mapping incorrect.

While alignment system users will usually fall under the categories of domain expert or knowledge engineer, it should be noted that domain and technical expertise are not disjoint. Indeed, the development of tools like Protégé has allowed domain experts to delve into knowledge engineering [20]. Nevertheless, the differences between these two user types are important for the design of every knowledge-based system, and should be addressed both when designing the system and when building support for it. For instance, in order to assist users with limited technical expertise, alignment systems should provide information about the structure of the ontologies and the entailments of a mapping in a manner that is intuitive to understand. Likewise, in order to assist users with limited domain expertise, systems should provide detailed contextual and conceptual information about the mapping. Indeed, a recent study showed that, given enough contextual help, the quality of the validation of non-domain experts can approximate that of domain experts [36] – although this is likely to depend on the domain in question.

The final aspect of user expertise is **expertise with the alignment system**, which concerns the user's familiarity with the functionality of the system and its visual representations. Novice users can face comprehension difficulties and make erroneous decisions, not for lack of domain or technical expertise, but because they cannot fully acquire the information made available about a mapping or its entailments. It is up to the alignment system to be as intuitive as possible in both functionality and visual representations so that novice users can focus on the alignment process and are not limited by their lack of expertise with the system [35]. In this context, it is important to consider that different visual representations are suited for conveying different types of information, as we will detail in Subsection 2.3. Systems should also provide support to expert users in the form of shortcuts or customizations, so that they can speed up their work.

Users can be expected to make mistakes in alignment validation [5,22], be that due to lack of domain expertise, technical expertise, or expertise with the alignment system. However, the possibility of user errors is often disregarded in existing alignment systems. On the one hand, it is true that users are generally expected to make less errors than automated systems, and experiments have shown that up to an error rate of 20%,

user input is still beneficial [26]. On the other hand, there are risks to taking user input for granted, particularly when that input is given during the alignment process, and inferences are drawn from it, leading to the potential propagation of errors. An example of this is given in [26], where user validated relations during an alignment repair step are fixed, meaning that they cannot be removed during subsequent steps, and other potentially correct relations may have to be removed instead.

User errors can be prevented to some extent by warning the user when contradicting validations are made [23] or by preemptively removing mappings that lead to logical conflicts. In a multi-user setting (e.g. [7,43]), errors may be diluted through a voting strategy, where the mapping confidence is proportional to the consensus on the mapping, by accepting the decision made by a majority of the users [43], or by adopting a more skeptical approach where full agreement between the users is required[7]. However, given the limited availability of users for alignment validation, systems cannot rely on having multiple users to prevent user errors.

One way of assessing the impact of the user profile on the alignment quality is by simulating the user input by an oracle with different error rates [33], which is the strategy we have adopted in our evaluation (Subsection 3.2).

2.2 System services

Alignment validation is an extensive task, particularly when large ontologies are involved, as alignments can include several thousand mappings. Since users capable of performing alignment validation are a scarce and valuable resource, alignment systems cannot expect them to be able to validate a whole alignment. Rather, they must limit their demand for user intervention and exploit that intervention to maximize its value, wherein lies one of the main challenges of alignment validation [26,38].

With regard to demand for user intervention, several strategies have been implemented by alignment systems for limiting the number of mapping suggestions to be validated by the user (**suggestions selection** - {Sys.e}). The simplest and most common of these consists of employing threshold values for different alignment algorithms. Other, more sophisticated filtering approaches include filtering with respect to principles (e.g., consistency, locality, and conservativity) [25] or quality checks [3], selecting only "problematic" mappings where different alignment algorithms disagree [8], and using a similarity propagation graph to select the most informative questions to ask the user [44].

One common strategy that both reduces demand for user intervention and exploits that intervention is to automatically reject alternative mappings for a concept when the user validates one of that concept's mappings [26,31].

With regard to exploiting user interventions, systems can adopt different strategies depending on the **stage of involvement** of the user in the alignment process: *before* ({Sys.a}), *after* ({Sys.b}), *during* ({Sys.c}), or *iterative* ({Sys.d}).

When the validation happens *before* the matching process, the user provides an initial partial alignment which is then used by the system to guide the matching process. The partial alignment can be used in the preprocessing phase to reduce the search space [30], as input for the alignment algorithms [11,30], or to select and configure the algorithms to use [29,40,42,49].

When the validation is performed *after* the automatic alignment process, the input of the user cannot be exploited for aligning the ontologies. However, many systems still filter out mapping suggestions which are in conflict with user validations before proceeding to a final reasoning and diagnosis phase [23,26,31,37].

When the validation is done *during* the alignment process, input from the user can be extrapolated through the use of **feedback propagation** ({Sys.f}) techniques to fully exploit it. When the validation is *iterative*, the user is asked for feedback on several iterations of the alignment process, where in each iteration the alignment from the previous iteration is improved [29].

Feedback propagation techniques usually consist of propagating mapping confidence from validated mappings to those in their neighborhood, be that neighborhood defined from the structure of the ontologies [31,37,44] or from the pattern of similarity scores from the various alignment algorithms [8,30]. They usually require that the validation is done during the alignment process or is iterative, but one form of feedback propagation that systems can implement regardless of when the validation takes place in the alignment process, is conflict detection ({Sys.g}) [18,26]. This consists of testing user validated relations against the ontologies, report on the violation of logical constraints (e.g., unsatisfiable classes), and possibly ask for revalidations of certain relations to resolve the conflict.

Demand for user involvement in the matching process can be evaluated by measuring the number of questions (mapping suggestions) the system asks the user, and comparing it to the actual size of the alignment produced by the system. The effectiveness with which systems exploit user involvement can be evaluated by measuring their improvement in performance (in terms of precision and recall) over the fully automated process, and relating it with the number of questions asked.

2.3 User interface

A graphical user interface (UI) is an indispensable part of every interactive system, as the visual system is humans' most powerful perception channel. Alignment validation is a cognitively demanding task that involves a high memory load – ontologies are complex knowledge-bases, and validating each mapping requires considering the structure and constraints of two ontologies while also keeping in mind other mappings and their logical consequences – and thus is all but impossible without visual support.

Given the complexity of ontologies and alignments, a critical aspect of visualizing them is not overwhelming the user. Humans apprehend things by using their working memory, which is limited in capacity (it can typically hold 3 ± 1 items) and thus can be easily overwhelmed when too much information is presented [48]. However, this limitation can be expanded by grouping similar things, a process called "chunking", which can be exploited by visualization designers to facilitate cognition and reduce memory load [39]. For instance, encoding properties of entities and mappings with different graphical primitives facilitates their identification and enables their chunking.

Another critical aspect of ontology alignment visualization is providing the user with sufficient information to be able to decide on the validity of each mapping, which includes lexical and structural information in the ontologies, and potentially other related mappings. This naturally competes with the need not to overwhelm the user with

information, and a balance between the two must be struck. As we discussed in Subsection 2.1, different user types are likely to have different information requirements, and alignment systems must cater to all.

The **Visual Information Seeking Mantra**, {UI.a}, [45] defines seven low-level tasks to be supported by information visualization interfaces in order to enable enhanced data exploration and retrieval: overview, zoom, filter, details-on-demand, relate, history, and extract. The former six of these were further refined for the purpose of ontology visualization [27], and all are relevant in the context of striking a balance between providing information and avoiding memory overload.

Providing enhanced information while addressing the working memory limits is also the goal of the field of **visual analytics**, {UI.b}, which combines data mining and interactive visualization techniques to aid analytic reasoning and obtain insights into (large) data sets. The application of visual analytics to ontology alignments facilitates their exploration and can provide quick answers to questions of interest from the users [2,6,8,32,34].

Another technique at the disposal of alignment systems is that of providing **alternative views** {UI.c} [6,17,29,34]. Different views may be more suitable for performing different tasks – for instance, graphs are better for information perception, whereas indented lists are better for searching [19] – and by providing alternate views, systems need not condense all relevant information into a single view, and thus avoid overwhelming the user. Also relevant in this context are maintaining the user focus in one area of the ontology [37], and preserving the user's mental map (e.g., by ensuring that the layout of the ontology remains constant).

Two strategies that facilitate chunking are grouping mappings together by different criteria to help identify patterns {UI.d}, and distinguishing between different types of mappings and their provenance {UI.e} – particularly between validated and candidate mappings [17]. Color-coding is a common and effective technique for implementing both strategies.

With regard to facilitating the decision making process, showing context and definitions of terms {UI.f} is essential, and providing recommendations and/or ranking ({UI.g}) facilitates the process by allowing the user to focus on a specific set of mappings. Also important is the **explanation of mapping suggestions** by presenting the provenance of, or justification for, a mapping ({UI.h}). Likewise, the user should be provided with feedback about the consequences of his decision {UI.i} about a mapping with regard to the alignment and ontologies, possibly through a trial execution [17].

Justifications have been identified as one of the future challenges of ontology alignment, given that many alignment systems merely present confidence values for mappings as a form of justification [38]. They require particular attention to the user type: domain experts will require detailed contextual information and a clear explanation of how a mapping suggestion was inferred, whereas for knowledge engineers summarized provenance information might suffice.

Three distinct justification approaches have been identified [13]: proof presentation, strategic flow, and argumentation. In the proof presentation approach, the explanation for why a mapping suggestion was created is given in the form of a proof, which can be a formal proof, a natural language explanation (e.g., [17,47]), or a visu-

alization (e.g., [29]). In the strategic flow approach the explanation is in the form of a decision flow which describes the provenance of the acquired mapping suggestion (e.g., [10,15]). Finally, in the argumentation approach, the system gives arguments for or against certain mapping suggestions, which can be used to achieving consensus in multi-user environments.

In addition to providing visual information to support the decision process, alignment systems need to provide functionalities for the user to interact with the alignment in order to validate it. The most basic level of interaction is to allow the user to either accept or reject mapping suggestions {UI.j}. Additionally, the functionality of adding a mapping manually or refining a mapping suggestion {UI.k} is also important, since the system may not have captured a mapping that is required according to the user, or may not have correctly identified the mapping relationship [1,6,15,16,29].

An important functionality is searching and filtering {UI.l}, which contributes to minimize the user's cognitive load [1,6,17,34]. It is relevant to enable searching/filtering both of the ontologies (e.g., to analyze the structural context of a mapping suggestion, or look for a concept to map manually) [17,34] and of the mapping suggestions themselves [6,17,29].

Given the extension of the validation process, allowing the user to add metadata in the form of annotations {UI.m} [17,29], and accommodating interruptions or sessions {UI.n} are important functionalities. However, while many systems enable interruptions through saving and loading the ontologies and alignment, this often does not preserve the provenance information.

Finally, allowing the creation of temporary mappings {UI.o} in order to test decisions is a relevant functionality for supporting the decision process [34], as is enabling trial execution to help the user understand the consequences of his decisions {UI.i} [17].

3 Evaluation

In this section, we assess how state of the art ontology alignment systems take into account the three aspects discussed above: user profile, system services, and user interface. We start by surveying systems and evaluating them qualitatively with regard to key features of their interfaces and services for processing user input in Subsection 3.1. Then, we assess the impact of erroneous user input to the system through a series of experiments from the Interactive Matching track of OAEI 2015 in Subsection 3.2. In these experiments, we simulate user input with varying error rates, which can reflect lack of user of expertise, but also limitations of the system's user interface. How these errors impact the alignment is dependent on the system services.

3.1 Qualitative evaluation

We performed a qualitative evaluation of state of the art systems that incorporate user validation in the alignment process and have a mature user interface: AgreementMaker [6,8,9], AlViz [34], AML [18,41], CogZ/Prompt [16,17,37], COMA [1], LogMap [26], RepOSE [23], and SAMBO [29,31]. The results of this evaluation are summarized in Table 1.

Table 1: Issues regarding user interaction in ontology alignments addressed by state-of-the-art systems.

Issue				Agreement Maker	AIViz	AML	CogZ Prompt	COMA	LogMap	SAMBO	RepOSE
System services	Stage of involvement	{Sys.a}-before,{Sys.b}-after, {Sys.c}-during, {Sys.d}-iterative		{Sys.b+d}	{Sys.b}	{Sys.b}	{Sys.c}	{Sys.a,b+d}	{Sys.b}	{Sys.a,b+d}	{Sys.b+d}
	Suggestions Selection	{Sys.e} threshold/advanced filtering		✓	✓-	✓	-	✓-	✓	✓-	✓-
	Feedback Propagation	{Sys.f} recomputation		✓	-	-	✓	✓	-	✓	✓
		{Sys.g} conflict detection /blocking/revalidation		✓-(*)	-	✓-	✓- -	-	✓- -	✓- -	✓-
User Interface	Alignment Presentation	{UI.a} 7 visual info-seeking tasks		✓	✓	✓-	✓	✓- -	-	✓- -	✓- -
		{UI.b} visual analytics		✓	✓-	-	-	-	-	-	-
		{UI.c} alternative views		✓	✓	✓	✓	-	-	✓	-
		{UI.d} grouping		✓	✓	✓	✓	-	-	✓	✓
		{UI.e} validated/candidate mappings		✓-	✓-(**)	✓	✓	✓- -(**)	-	✓-	✓-
		{UI.f} metadata & context		✓	-	✓	✓	-	✓	✓- -	✓-
		{UI.g} ranking/recommendations		-	✓- -	✓- -	✓- -	-	✓-	-	✓
		Mapping Explanation	{UI.h} provenance & justification	✓- -	✓- -	✓-	✓-	✓- -	✓- -	-	✓- -
			{UI.i} impact of decisions/ consequences of actions	✓-	-	-	✓- -	-	✓-	-	✓- -
	Alignment Interaction	{UI.j} accept/reject mapping		✓	✓-	✓	✓	✓-	✓	✓	✓
		{UI.k} create/refine mapping		✓	✓	✓	✓	✓	-	✓	✓-
		{UI.l} search		-	✓	✓	✓	✓	-	✓	-
		{UI.m} user annotation		-	-	-	✓	-	-	✓	-
		{UI.n} session		✓-	✓-	✓-	✓-	✓	✓	✓	✓-
		{UI.o} create temporary mapping		-	✓	✓-	✓	-	-	-	-

In the table ✓marks that all of the listed items are supported by the system while - marks that the issue is not covered by the system. Combinations such as ✓- and ✓- - mark that one or two of the listed items are not supported. (*) in a multi-user environment; (**) candidate and validate mappings cannot be distinguished in the user interface;

Regarding system services, the majority of the systems ask for validations after running the matching algorithms (AgreementMaker, AlViz, AML, LogMap, and RepOSE), CogZ/Prompt involves the user during the alignment process, while SAMBO and COMA allow validations both before and after the alignment process. AgreementMaker, COMA, SAMBO, and RepOSE also allow multiple iterations of the alignment process, and allow for user involvement in multiple validation sessions.

The majority of the systems use some form of thresholds to select mapping suggestions to present to the user for validation. AML and AgreementMaker use a more refined strategy for identifying "problem" mappings to present to the user, which relies on the variance of the similarity scores of their various alignment algorithms. LogMap presents mapping suggestions that cause the violation of alignment principles such as consistency, locality, and conservativity.

With respect to feedback propagation, most systems implement at least a conflict detection mechanism, such as checking if the validated mapping contradicts previously validated mappings or results in an incoherent or inconsistent integrated ontology (AML, CogZ/Prompt, LogMap, SAMBO, RepOSE). AlViz does not implement such mechanisms and accepts user's feedback without any additional steps. AgreementMaker employs a blocking propagation strategy where the user can control to how many similar instances the validation is propagated. Revalidation is supported by AML and RepOSE as a part of the conflict resolution phase. AgreementMaker, CogZ/Prompt, COMA, RepOSE and SAMBO employ some form of recomputation, where the user's input is used to guide the matching process. For example, AgreementMaker propagates the user's decision to similar mappings thus increasing/decreasing the similarity value.

Regarding the representation of the ontologies and the alignment, systems typically represent ontologies as trees or graphs. Graphs are usually used as an additional representation (AlViz, CogZ) and rarely as a main representation (AML, RepOSE). Mappings are typically represented as links between corresponding nodes, or sometimes as a list/table of pairs (AML, SAMBO, CogZ, COMA, LogMap). The list/table view is used to support different interactions by systems. About half of the systems support more than one view of the alignments and ontologies, often a tree and a graph view which are more suitable for different alignment tasks [19]. Most of the systems employ strategies to group the mappings together: SAMBO presents all mappings for a particular concept together, CogZ, AML, LogMap, and RepOSE show the local neighborhoods of a mapping up to different number of levels. AgreementMaker and AlViz combine the different views with clustering algorithms and interaction techniques to support the comparison of the similarity values calculated by the different matchers (AgreementMaker) or clustering nodes of the ontologies according to a selected relationship (AlViz).

Most systems also provide detailed information for mappings individually, such as the context of the mapping and its state (e.g., whether it is accepted). However few systems provide interface support for features regarding explaining the mappings, such as why the system has suggested the mapping or how the current validation would affect other candidate or validated mappings. Most systems provide only a similarity value or employ color coding as a form of explanation for the mapping, which is insufficient for users to make informed decisions (one exception is CogZ which shows a short natural language explanation for the mapping). Thus our evaluation survey confirms findings

from [24] that explanations for mappings suggestions are not well supported by the user interfaces of alignment systems, and continues to be a challenge for the alignment community [46]. Ranking and recommendation functionalities are also rarely provided by systems.

Interactions for accepting, rejecting and creating mappings manually are supported by most of the systems but the different systems do not always present this information to the user – rejected mappings for instance are rarely shown. AlViz and COMA do not distinguish between validated and candidate mappings, thus the user cannot keep track of already visited mappings. Creating temporary mappings is supported by AlViz, AML, and CogZ. Interactions to support the 7 information visualization seeking tasks are provided to a different extent by the different systems with overview usually supported and filter, history and relate rarely supported. Search is often supported but a previous survey of some of these systems found serious limitations [24]. Two systems (CogZ and SAMBO) allow the user to annotate mappings during the validation process. Sessions are directly (COMA, LogMap, SAMBO) or indirectly (by saving and loading files) supported by all systems.

3.2 Experiments in the OAEI campaign

The Interactive Matching track of OAEI was extended in 2015 to take into account erroneous user validations and assess varying error rates and their impact of the performance of alignment systems [4]. Systems were evaluated according to their performance in terms of precision and recall versus the reference alignments, as well as in terms of number of interactions required and time between interactions.

Although the original purpose of introducing error rates in the Interactive Matching track was to simulate users with different expertise levels, the results can be interpreted and discussed in a broader light with regard to how system services are affected by and cope with errors, irrespective of their cause.

3.2.1 Setup and systems

The OAEI SEALS client[5] allows interactive systems to pose questions regarding the correctness of a mapping to an oracle, which will simulate a user by checking the reference alignment from the respective OAEI task, and answering with a predefined error rate. In this experiment the error rates considered were 0% (perfect oracle), 10%, 20% and 30%. Systems were evaluated on three datasets from the OAEI: Conference (16 small ontologies), Anatomy (2 medium-sized ontologies) and LargeBio (3 large ontologies). For the sake of brevity, we will present only results for the Anatomy track, as the other results are similar (they can be found in [4]).

The systems that participated in the OAEI 2015 Interactive track were AML, JarviSOM, LogMap and ServOMBI. We note that not all of these systems have user interfaces, but they implement an interface to communicate with the oracle, so we can automatically evaluate the impact of the user input to the resulting alignment. We could not

[5] The SEALS client is the infrastructure used in the OAEI to automate the evaluation of ontology matching systemshttp://oaei.ontologymatching.org/2016/seals-eval.html

evaluate other systems with this experimental setup, as it requires compliance with the OAEI's SEALS client.

Apart from JarvisOM, which involves the user during the computation of the alignment, the systems all make use of user interactions exclusively in post-alignment steps. Both LogMap and AML request feedback on selected mapping suggestions and filter mapping suggestions based on the user validations. The former interacts with the user to decide on mapping suggestions which are not clear-cut cases, whereas the latter employs a query limit and other strategies to minimize user interactions. ServOMBI asks the user to validate all of its mapping suggestions and uses the validations and a stable marriage algorithm to decide on the final alignment. JarvisOM is based on an active learning strategy known as query-by-committee: at every iteration JarvisOM asks the user for pairs of entities that have the highest disagreement between committee members and lower average euclidean distance, and at the last iteration, the classifiers committee is used to generate the alignment.

3.2.2. Results and discussion

The evaluation results for the Anatomy track are shown in Table 2. As expected, the performance of all systems improves when they have access to an all-knowing oracle (Or^0 in the table) in comparison with their non-interactive performance (N/A in the table). Also as expected, when we increase the oracle's error rate, we observe that the performance of all systems deteriorates. However, it takes an error rate of 30% for the user interaction not to be beneficial to most systems, which corroborates the observations in [26]. The way in which the systems exploit user interactions, how they benefit from them, and how they are affected by errors are very different.

AML is the only system that improves more in terms of recall than in terms of precision with user interactions, because it exploits them in part to test mappings with lower similarity scores than it accepts in non-interactive mode. This is why it is the system that asks the most negative questions from the oracle, proportionally. As a result, when the error rate increases, AML's precision drops below the non-interactive precision (at 20%), but its recall remains higher than the non-interactive recall. AML is also the only system that is affected linearly by the errors, as evidenced by the fact that its performance as measured against the oracle (i.e., assuming the oracle errors are instead correct) remains constant at all error rates. This means that, unlike the other three systems, AML does not extrapolate from the user feedback about a mapping to decide on the classification of multiple mapping candidates. While extrapolation (be it through active learning, feedback propagation, or other techniques) is an effective strategy for reducing user demand, it also implies that the system will be more heavily impacted by user errors.

JarvisOM is the system that most depends on user interactions, as evidenced by the very poor quality of its non-interactive alignment. Thus, it is the system that most improves with user interactions, and the only one that improves substantially in both precision and recall. It is also the one that makes the least requests from the oracle – only 7-8 requests per alignment – as it uses these requests in an active learning approach rather than to validate a final alignment. This means it is the system that extrapolates the most from the user feedback, which as expected, makes it the one that is most affected by user errors – its F-measure drops by 26% between 0 and 30% errors. However, it

Table 2: Interactive Anatomy alignment evaluation

Oracle	System	P/F/R	P/F/R Or	TReq	DReq	TP	TN	FP	FN	Size
N/A	AML	.96/.94/.93	-	-	-	-	-	-	-	1477
	JarvisOM	.36/.17/.11	-	-	-	-	-	-	-	458
	LogMap	.92/.88/.85	-	-	-	-	-	-	-	1397
	ServOMBI	.96/.75/.62	-	-	-	-	-	-	-	971
Or^0	AML	.97/.96/.95	.97/.96/.95	312	312	73	239	0	0	1491
	JarvisOM	.86/.75/.67	.86/.75/.67	7	7	4	3	0	0	1173
	LogMap	.98/.91/.85	.98/.91/.85	590	590	287	303	0	0	1306
	ServOMBI	1/.76/.62	1/.76/.62	2136	1128	955	173	0	0	935
Or^{10}	AML	.96/.95/.95	.97/.96/.95	317.3	317.3	66.3	218	23	10	1502
	JarvisOM	.76/.68/.67	.76/.68/.67	7	7	3.3	3	0.3	0.3	1475
	LogMap	.96/.89/.83	.96/.89/.83	609	609	261.3	288.3	33.7	25.7	1302
	ServOMBI	1/.71/.55	1/.74/.59	2198.7	1128	857.3	156.3	16.7	97.7	843
Or^{20}	AML	.94/.94/.94	.97/.96/.95	321.7	321.7	66.3	186.7	52.3	16.3	1525
	JarvisOM	.53/.60/.71	.53/.60/.71	8	8	4.7	1	1.3	1	2055
	LogMap	.95/.88/.82	.95/.88/.81	630	630	233	274	69	54	1321
	ServOMBI	.99/.66/.49	1/.71/.55	2257	1128	767.3	131.3	41.7	187.7	758
Or^{30}	AML	.93/.93/.94	.97/.96/.95	306	306	54	168.7	61.3	22	1526
	JarvisOM	.51/.49/.53	.51/.49/.53	7.3	7.3	4	1.7	1	0.7	1509
	LogMap	.94/.87/.82	.92/.86/.80	663	663	200.7	270.7	105.3	86.3	1334
	ServOMBI	.99/.60/.43	1/.68/.52	2329.7	1128.3	663.3	129	44.3	291.7	659

Systems were evaluated with user interactions simulated by an oracle with different error rates (Or^x corresponds to an error rate of x%) and without user interactions (N/A). The "P/F/R" column shows the Precision, F-measure and Recall obtained in the task; the "P/F/R Or" column shows the same parameters with respect to oracle, i.e., as if the errors made by the oracle were instead correct; "TReq" and "DReq" correspond respectively to the total number of requests and the number of distinct requests made by the system to the oracle; "TP", "TN", "FP" and "FN" are respectively the number of True Positive, True Negative, False Positive and False Negative answers given by the oracle; and "Size" indicates the number of mappings in the alignment produced by the system. All values in interactive settings with non-zero error rate are averages over 3 runs, to dilute the variance of the oracle errors.

depends so heavily on user interaction, that even at 30% errors, its results are still better than the non-interactive ones. JarvisOM is also the system where the impact of the errors most deviates from linearity, precisely because it extrapolates from so few mappings. Another curious consequence of this is that its alignment size fluctuates considerably, increasing to almost double between 0 and 20% errors, but then decreasing again at 30% errors. It should be noted that JarvisOM behaves very differently in the Conference track [4], showing a linear impact of the errors, as in that case less inferences are drawn from its 7-8 oracle requests because they represent $\sim$50% of the Conference alignments (whereas in Anatomy they represent 0.5%).

LogMap improves only with regard to precision with user interactions, which is curious considering it is the most balanced system regarding positive versus negative oracle answers. This means that, in this particular task, the positive questions LogMap

asks the oracle all correspond to mappings it would also accept in its non-interactive setting, whereas the negative questions allow it to exclude some mappings that it would also (erroneously) accept. Due to the balance between its questions, when presented with user errors, LogMap is affected with regard to both precision and recall in approximately equal measure. However, since its precision increased substantially with user interactions, it remains higher than the non-interactive precision at all error rates, unlike the recall. Another interesting observation about LogMap is that the number of requests it makes increases slightly but steadily with the error rate, whereas other systems show stable rates. This increase is tied to the fact that user errors can lead to more complex decision trees when interaction is used in filtering steps and inferences are drawn from the user feedback. For instance, during alignment repair, if the user indicates that a mapping that would be removed by the system to solve a conflict is correct, the system may have to ask the user about one or more alternative mappings to solve that conflict, thus increasing the number of requests. In this context, the present query-based evaluation does not accurately reflect an interface-based alignment validation, where the user could be shown all the mappings that cause a conflict simultaneously.

ServOMBI is the system that improves the least with user interaction, showing an increase of only 1% F-measure, and like LogMap improves only with regard to precision. It is also the system that makes the most oracle requests, as it asks the oracle about every mapping candidate it finds, and the only system that makes redundant questions (its total number of requests is almost double that of the distinct ones). Interestingly, it is also the only system that produces alignments that do not contain all the mappings identified as positive by the user, as some are apparently discarded by its stable marriage algorithm. Because it makes so many oracle requests, ServOMBI is strongly affected by user errors, so much so that at only 10% errors, user interaction is no longer beneficial in terms of F-measure. In fact, since 85% of the questions ServOMBI asks the oracle are positive, the system would have a better performance (72% F-measure) by simply accepting all its mapping candidates than it does at 10% errors. Because of its strong bias towards positive questions, ServOMBI feels the impact of the errors mostly in terms of recall and alignment size, whereas precision is hardly affected. However, given the number of false positive questions returned by the oracle at 30% errors, we would expect a drop in precision as well, but it remains almost constant as the errors increase. This attests to the ability of this system's stable marriage algorithm to filter out user errors. Interestingly, the number of total oracle requests made by ServOMBI increased with the error rate, even though the number of distinct requests remains constant – as it should, considering the system already asks the user about all mapping candidates it identifies. This means that ServOMBI is making more redundant questions.

4 Conclusions

Despite the advances in automated ontology alignment techniques, user validation remains critical to ensure alignment quality, due to the complexity and diversity of ontologies and their domains. In this broad study of user validation in ontology alignment, we encompassed three distinct but interrelated aspects: the profile of the user; the ontology alignment systems services; and their user interfaces. We assessed the services and

user interfaces of state of the art systems in a qualitative evaluation, and investigated the impact of errors in alignment validation through a series of experiments that revealed how systems cope with it, depending on their services.

The profile of the user is a key factor to take into account in alignment validation, as systems will not be able to rely exclusively on domain experts for validation, and even domain experts require extensive support for deciding on the validity of mappings – particularly if they have little technical expertise regarding ontologies and knowledge engineering. Thus, it is up to alignment systems' user interfaces to provide rich contextual information on each mapping. However, they have to balance that need with the need not to overwhelm the users with too much information, as humans have limited working memory. To that end, systems must ensure that their user interfaces convey information in an intuitive manner, and that while all required information is ready on-click, it is not all shown simultaneously. A strategy that many systems have implemented to achieve this is to provide different views of the alignment and/or each mapping.

In order to support user decisions, alignment systems' user interfaces should provide detailed explanations about mappings, and allow users to interact with the alignment in multiple ways, so as to make clear the consequences of accepting or rejecting a mapping. Allowing users to manually annotate mappings, and enabling validation over multiple sessions are also important features, due to the complexity and extensiveness of the validation task. However, these are all aspects where most current alignment systems have room for improvement.

Given the limited availability of users for alignment validation, systems should be able to prioritize the mapping suggestions they present to the users, by focusing on mappings about which they are unsure and/or those which cause conflicts. Systems can further exploit user input by extrapolating on it through feedback propagation techniques. However, as our experiments have shown, extrapolating will increase the impact of user errors, so systems should consider the profile of the user when deciding whether or not to employ feedback propagation. One possible strategy for that would be to ask the user how confident he is about each mapping, and only extrapolating on his decision when his confidence is high.

Our study should serve as a starting point towards establishing guidelines and best practices for good user interface design in the context of ontology alignment, which our evaluation of state of the art systems has shown to be necessary. Furthermore, we expect our study to help guide the development of alignment systems with regard to exploiting user interactions and coping with user errors.

For future work, we will aim to extend our evaluation by making usability assays with real users having varying degrees of expertise. We will also refine our experimental setup to better mirror the manual validation process, namely by considering the scenario where the user chooses between different conflicting mappings, rather than evaluating them independently, and by having the user provide a confidence value rather than a binary classification.

Acknowledgments. This work has been supported by SeRC, CUGS, the EU projects VALCRI (FP7-IP-608142) and Optique (FP7-ICT-318338), the EPSRC projects ED3 and DBOnto, and the Fundação para a Ciência e Tecnologia through the funding of the LaSIGE research unit (UID/CEC/00408/2013) and project PTDC/EEI-ESS/4633/2014.

References

1. D Aumüller, H H Do, S Maßmann, and E Rahm. Schema and ontology matching with COMA++. In *SIGMOD*, pages 906–908, 2005.
2. J Aurisano, A Nanavaty, and I Cruz. Visual analytics for ontology matching using multi-linked views. In *VOILA*, pages 25–36, 2015.
3. E Beisswanger and U Hahn. Towards valid and reusable reference alignments-ten basic quality checks for ontology alignments and their application to three different reference data sets. *J Biomedical Semantics*, 3(S-1):S4, 2012.
4. M Cheatham et al. Results of the ontology alignment evaluation initiative 2015. In *OM*, pages 60–115, 2015.
5. C Conroy, R Brennan, D O'Sullivan, and D Lewis. User Evaluation Study of a Tagging Approach to Semantic Mapping. In *ESWC*, pages 623–637, 2009.
6. I Cruz, F Antonelli, and C Stroe. Agreementmaker: efficient matching for large real-world schemas and ontologies. *Proc VLDB Endowment*, 2(2):1586–1589, 2009.
7. I Cruz, F Loprete, M Palmonari, C Stroe, and A Taheri. Quality-based model for effective and robust multi-user pay-as-you-go ontology matching. *Semantic Web J*, 2015.
8. I Cruz, C Stroe, and M Palmonari. Interactive user feedback in ontology matching using signature vectors. In *ICDE*, pages 1321–1324, 2012.
9. I Cruz, W Sunna, N Makar, and S Bathala. A visual tool for ontology alignment to enable geospatial interoperability. *J Visual Languages & Computing*, 18(3):230–254, 2007.
10. R Dhamankar, Y Lee, A Doan, A Halevy, and P Domingos. iMAP: discovering complex semantic matches between database schemas. In *SIGMOD*, pages 383–394, 2004.
11. S Duan, A Fokoue, and K Srinivas. One size does not fit all: Customizing ontology alignment using user feedback. In *ISWC*, pages 177–192. 2010.
12. J Euzenat, C Meilicke, P Shvaiko, H Stuckenschmidt, and C Trojahn. Ontology alignment evaluation initiative: six years of experience. *J Data Semantics*, XV:158–192, 2011.
13. J Euzenat and P Shvaiko. User Involvement. In *Ontology Matching*, pages 353–375. 2013.
14. S Falconer and N Noy. Interactive techniques to support ontology matching. In Z Bellahsene, A Bonifati, and E Rahm, editors, *Schema Matching and Mapping*, pages 29–51. 2011.
15. S Falconer, N Noy, and M-A Storey. Towards Understanding the Needs of Cognitive Support for Ontology Mapping. In *OM*, 2006.
16. S Falconer, N Noy, and M-A Storey. Ontology mapping - a user survey. In *OM*, pages 49–60, 2007.
17. S Falconer and M-A Storey. A Cognitive Support Framework for Ontology Mapping. In *ISWC/ASWC*, pages 114–127. 2007.
18. D Faria, C Martins, A Nanavaty, D Oliveira, B Sowkarthiga, A Taheri, C Pesquita, F M Couto, and I F Cruz. AML results for OAEI 2015. In *OM*, 2015.
19. B Fu, N Noy, and M-A Storey. Eye tracking the user experience-an evaluation of ontology visualization techniques. *Semantic Web J*, 2014.
20. J H Gennari, M A Musen, R W Fergerson, W E Grosso, M Crubzy, H Eriksson, N F Noy, and S W Tu. The evolution of Protégé: an environment for knowledge-based systems development. *Int J Human-Computer Studies*, 58(1):89–123, 2003.
21. M Granitzer, V Sabol, K W Onn, D Luckose, and K Tochtermann. Ontology Alignment—A Survey with Focus on Visually Supported Semi-Automatic Techniques. *Future Internet*, pages 238–258, 2010.
22. V Ivanova, J L Bergman, U Hammerling, and P Lambrix. Debugging taxonomies and their alignments: the ToxOntology-MeSH use case. In *WoDOOM*, pages 25–36, 2012.
23. V Ivanova and P Lambrix. A unified approach for aligning taxonomies and debugging taxonomies and their alignments. In *ESWC*, pages 1–15. 2013.

24. V Ivanova, P Lambrix, and J Åberg. Requirements for and evaluation of user support for large-scale ontology alignment. In *ESWC*, pages 3–20. 2015.
25. E Jiménez-Ruiz, B Cuenca Grau, I Horrocks, and R Berlanga. Logic-based assessment of the compatibility of umls ontology sources. *J Biomedical Semantics*, 2(S-1):S2, 2011.
26. E Jiménez-Ruiz, B Cuenca Grau, Y Zhou, and I Horrocks. Large-scale Interactive Ontology Matching: Algorithms and Implementation. In *ECAI*, pages 444–449, 2012.
27. A Katifori, C Halatsis, G Lepouras, C Vassilakis, and E G Giannopoulou. Ontology visualization methods - a survey. *ACM Computing Surveys*, 39(4):10, 2007.
28. P Lambrix and A Edberg. Evaluation of ontology merging tools in bioinformatics. In *Pacific Symposium on Biocomputing*, pages 589–600, 2003.
29. P Lambrix and R Kaliyaperumal. A Session-Based Approach for Aligning Large Ontologies. In *ESWC*, pages 46–60. 2013.
30. P Lambrix and Q Liu. Using partial reference alignments to align ontologies. In *ESWC*, pages 188–202, 2009.
31. P Lambrix and H Tan. SAMBO - a system for aligning and merging biomedical ontologies. *J Web Semantics*, 4(3):196–206, 2006.
32. P Lambrix and H Tan. A tool for evaluating ontology alignment strategies. *J Data Semantics*, VIII:182–202, 2007.
33. P Lambrix, F Wei-Kleiner, Z Dragisic, and V Ivanova. Repairing missing is-a structure in ontologies is an abductive reasoning problem. In *WoDOOM*, pages 33–44, 2013.
34. M Lanzenberger, J Sampson, M Rester, Y Naudet, and T Latour. Visual ontology alignment for knowledge sharing and reuse. *J Knowledge Management*, 12(6):102–120, 2008.
35. J Nielsen. *Usability Engineering*. 1993.
36. N Noy, J Mortensen, P Alexander, and M Musen. Mechanical turk as an ontology engineer? In *ACM Web Science*, pages 262–271, 2013.
37. N Noy and M Musen. Algorithm and Tool for Automated Ontology Merging and Alignment. In *AAAI*, pages 450–455, 2000.
38. L Otero-Cerdeira, F J Rodríguez-Martínez, and A Gómez-Rodríguez. Ontology matching: A literature review. *Expert Systems with Applications*, 42(2):949–971, 2015.
39. R E Patterson, L M Blaha, G G Grinstein, K K Liggett, D E Kaveney, K C Sheldon, P R Havig, and J A Moore. A human cognition framework for information visualization. *Computers & Graphics*, 42:42 – 58, 2014.
40. H Paulheim, S Hertling, and D Ritze. Towards Evaluating Interactive Ontology Matching Tools. In *ESWC*, pages 31–45. 2013.
41. C Pesquita, D Faria, E Santos, J Neefs, and F M. Couto. Towards Visualizing the Alignment of Large Biomedical Ontologies. In *DILS*, pages 104–111, 2014.
42. D Ritze and H Paulheim. Towards an automatic parameterization of ontology matching tools based on example mappings. In *OM*, pages 37–48, 2011.
43. C Sarasua, E Simperl, and N Noy. Crowdmap: Crowdsourcing ontology alignment with microtasks. In *ISWC*, pages 525–541, 2012.
44. F Shi, J Li, J Tang, G Xie, and H Li. Actively Learning Ontology Matching via User Interaction. In *ISWC*, pages 585–600. 2009.
45. B Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *IEEE Symposium on Visual Languages*, pages 336–343, 1996.
46. P Shvaiko and J Euzenat. Ontology Matching: State of the Art and Future Challenges. *Knowledge and Data Engineering*, 25(1):158–176, 2013.
47. P Shvaiko, F Giunchiglia, P Da Silva, and D McGuinness. Web explanations for semantic heterogeneity discovery. In *ESWC*, pages 303–317. 2005.
48. E Smith and S Kosslyn. *Cognitive Psychology: Mind and Brain*. 2013.
49. H Tan and P Lambrix. A method for recommending ontology alignment strategies. In *ISWC/ASWC*, pages 494–507. 2007.