Is Ontology Alignment like Analogy? – Knowledge Integration with LISA

Elie Raad, Joerg Evermann

Is Ontology Alignment like Analogy? — Knowledge Integration with LISA

Elie Raad
Faculty of Business Administration
Memorial University of Newfoundland
St. John's, Canada
elie.raad@mun.ca

Joerg Evermann
Faculty of Business Administration
Memorial University of Newfoundland
St. John's, Canada
jevermann@mun.ca

ABSTRACT

Ontologies are formal descriptions of a domain. With the growth of the semantic web, an increasing number of related ontologies with overlapping domain coverage are available. Their integration requires ontology alignment, a determination of which concepts in a source ontology are like concepts in a target ontology. This paper presents a novel approach to this problem by applying analogical reasoning, an area of cognitive science that has seen much recent work, to the ontology alignment problem. We investigate the performance of the LISA cognitive analogy algorithm and present results that show its performance relative to other algorithms.

General Terms

Algorithms

Categories and Subject Descriptors

H.1.2 [**Models and Principles**]: User/Machine Systems—*human information processing, software psychology*; I.2.0 [**Artificial Intelligence**]: General—*cognitive simulation*; I.2.4 [**Artificial Intelligence**]: Knowledge representation formalisms and methods

Keywords

Ontology, Alignment, Analogy, LISA, Cognition

1. INTRODUCTION

Ontologies are explicit specifications of a conceptualization, describe knowledge about a domain [13], and are a key technology of the semantic web vision. They are typically described using the OWL Web Ontology Language and are intended to facilitate semantic interoperability between parties [14]. However, the distributed nature of the semantic web makes it possible and even likely that multiple parties independently develop ontologies with partially overlapping sets of concepts. While these ontologies may be comparable in their goals, they can vary greatly in the way they define and represent concepts. This variation is due to a variety of factors, ranging from the culture and linguistic particularities to the chosen level of abstraction, the expressiveness of the representation language, and the purpose of the ontology.

In the context of the semantic web, ontology alignment plays a crucial role to integrate different but overlapping ontologies. Ontology alignment attempts to establish correspondences between concepts of different ontologies that are similar in meaning but have different names or structure. We argue that a successful ontology alignment system should derive the same correspondences that a human user would derive[1]. While we explicitly state this argument here, its general acceptance is implicit in the fact that ontology alignment systems are evaluated against reference alignments that are created by humans [4].

Existing research in the area of ontology alignment has focused on estimating lexical similarity between concepts (syntactic or semantic similarity), mapping structural correspondences between ontologies (tree-based or graph-based representations), and the role of human users (for assigning matchers' weights, fine-tuning parameters, confirming or rejecting alignment results, etc.) [1, 15, 22, 18]. While these approaches are based on well justified logical, linguistic, and statistical matchers, they do not explicitly take into account how humans derive correspondences between concepts.

Therefore, this work is an initial, exploratory study on how to develop ontology alignment systems that are explicitly based on human alignment processes. More specifically, we explore whether the cognitive process of analogy making or analogical reasoning can serve as the basis for ontology alignment systems.

The motivation of this work was drawn from the observation that two research domains, the semantic web and cognitive science, have many common elements: in particular the field of entity and/or concept mapping. Previously, this has been studied separately in the two research domains without efforts to bridge the two and transfer findings from cognitive science (analogical reasoning) to semantic web challenges (ontology alignment). Thus, the aim of this work is also to highlight this relationship and bring the work in the cognitive sciences to the attention of semantic web researchers.

Cognitive principles have also recently been applied to database schema matching, a problem similar to ontology alignment. Work in that area has empirically investigated how humans make schema matching decisions [3, 5] and initial studies show that cognitive principles of similarity can in principle be applied [20, 6], but do not yet provide any evaluation of their effectiveness.

While there is a great amount of research on analogical reasoning and numerous computational models have been proposed, this initial exploratory study uses the LISA model [16], which has been shown to produce similar results to human subjects in many exper-

[1]The issue of different humans arriving at different correspondences for the same problem [5, 21] is discussed in detail in Section 6.

SAC'14 March 24-28, 2014, Gyeongju, Korea.

imental studies.

The remainder of the paper is structured as follows. Section 2 introduces the cognitive research on analogical reasoning systems. Section 3 presents LISA, the analogical reasoning systems adopted in this research. Section 4 details the framework for implementing the ontology alignment approach using LISA. Section 5 describes our experimental method and results, including a comparison with existing ontology alignment approaches. Section 6 discusses the implications of our results, limitations of the study, and suggests several directions for future research.

2. ANALOGICAL REASONING

Analogical reasoning is a cognitive process that maps concepts from one domain (the base) to concepts of another domain (the target) and is based on three main principles [9, 11]: one-to-one mapping, parallel connectivity, and systematicity:

- First, humans are shown to construct and to strongly prefer ontologies that place each concept of the base domain in correspondence with at most one object of the target domain.
- Second, analogies become more useful when placing objects of the base domain in correspondence with objects in the target domain by virtue of the objects' structure rather than superficial similarities. This systematicity principle reflects people's perception on interconnected systems of relations. It drives analogies to prefer sets of relations that include common higher-order relations and to map knowledge such that a system of relations that holds among the base objects also holds among the target objects. For example, in the analogy "an electric battery is like a reservoir" there is no resemblance between the objects' surface attributes (e.g., shape, size, color). However, the analogy builds on the fact that both objects store potential energy, i.e. structural similarity.
- Finally, the principle of parallel connectivity states that if two relations are mapped to each other, their objects must also be placed in correspondence. For example, if the predicate *stores (reservoir, energy)* matches the predicate *stores' (electric battery, energy')* then *reservoir* must match *electric battery* and *energy* must match *energy'*.

Reasoning by analogy has received significant attention in cognitive systems research [8, 17] and a number of computational models of analogy-making systems have been proposed, implemented, and experimentally evaluated [11]. We present three approaches that have undergone experimental evaluations and that show they are reasonable approximations for human analogical reasoning, albeit on simple, artificial problems.

SME [7] is a computational model of the structure-mapping theory of analogy [9]. The domain knowledge in SME is represented using a graph structure where all entities of the domain, as well as their relations, are represented as nodes of the graph. The alignment process is driven by common relations and functions, while attributes are neglected. SME uses a local-to-global alignment process to reflect the cognitive processes during analogy making and begins by matching identical predicates and their associated elements. The resulting local matches are then integrated into structurally consistent clusters. Finally, a global mapping is established by maintaining consistent clusters and merging them into a small number of sets that are maximal in size. SME has been experimentally shown to generate analogies that are good matches to analogies made by human subjects [19].

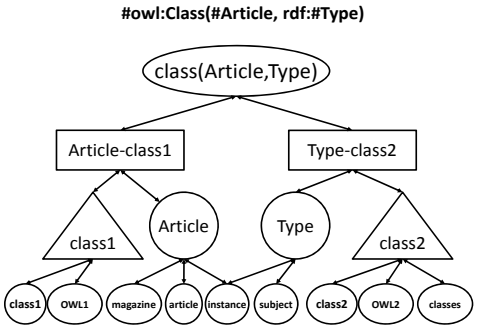

Figure 1: Example to illustrate the LISA architecture.

LISA [16] is a hybrid connectionist and symbolic architecture. The LISA architecture separates the long-term memory (LTM) and the working memory (WM) component, closely reflecting actual cognitive constraints. LISA represents domain knowledge in a hierarchy of propositions, sub-propositions, predicates and objects, and semantic units; the latter are shared between the two analogs. Mirroring synchronous and asynchronous activation in human memory, the driver analog resides in the active memory and controls the sequence of activating ("firing") the propositions. The activation propagates to the semantic units and, because these are shared, to the recipient analog. The resulting patterns of activation determine the correspondence of elements between the two analogs.

SIAM [12] is a computational model of human similarity judgment. SIAM models the elements of two domains as a network of interconnected nodes that represents all possible feature-to-feature, object-to-object, and role-to-role correspondences. SIAM first computes similarities between the feature nodes. Then, SIAM identifies objects pairs that are consistent with the feature correspondences. Based on the object-object similarities, SIAM calculates the role-role similarities while maintaining consistency. This process is repeated until the resulting similarity measures have converged.

In this work, we investigate the question whether existing computational models of cognition, specifically those of analogical reasoning, can be used as the basis for ontology alignment. We are particularly interested in exploring the utility of applying analogical reasoning models to the problem of ontology alignment and any of the above computational models could be adapted and used. While SME and LISA have had the greatest success in accounting for the range of phenomena in analogical thinking and learning [10], in this first paper we focus on LISA.

3. LEARNING AND INFERENCE WITH SCHEMAS AND ANALOGIES (LISA)

In this paper, we only present a brief overview of the architecture and principles of LISA. More details are provided in the appendix and full documentation is found in [16].

LISA represents knowledge using propositions (P), sub-propositions (SP), predicates and objects, and semantic units. In Figure 1, propositions are shown as ovals (e.g., *class(Article, Type)*), sub-propositions as rectangles (e.g., *article-class1*), predicates as triangles (e.g., *class1*), objects as large circles (e.g., *Article*), and semantic units as small circles (e.g., *magazine*). While we realize that the publication domain in this example may be overused and not very interesting, we use it because the OAEI test cases (Section 5) are based on this domain.

The semantic units are shared by the base and target analogs

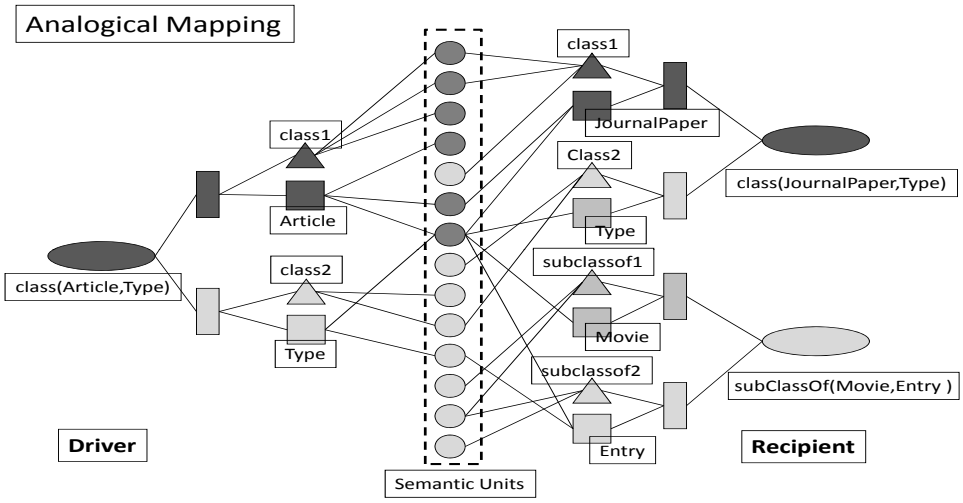

Figure 2: Connections between a driver and a recipient analog via shared semantic units.

while the remaining elements are local to each analog. Propositions are connected to their sub-propositions, which serve to bind roles (represented by predicates) to their fillers (represented by objects). Objects and predicates are functionally equivalent within LISA's computational models, but are separated for conceptual clarity. The semantic units in turn are connected to objects and predicates. Not shown in this simple example is the fact that propositions can be used in place of objects to fill roles of sub-propositions, leading to a deeper hierarchical structure. One analog plays the role of driver, the other that of receiver. Figure 2 shows an example of the analogy-making process of LISA between a driver and a recipient analog. The two propositions are *#owl:Class (#Article, rdf:#Type)* in the driver and *#owl:Class (#JournalPaper, rdf:#Type)* in the recipient; note the shared semantic units.

Propositions in the driver are activated ("firing") and this activation propagates through the two analogs connected by the shared semantic units. When a proposition is fired, it enters the active memory and activates its corresponding role-filler bindings (SPs) and its sub-units in turn. In the example, when the driver activates the proposition *#owl:Class* (#Article, rdf:#Type), its sub-propositions SP1 (*Article* and *class1*) and SP2 (*Type* and *class2*) are fired in sequence (i.e. asynchronously). When SP1 is fired, the semantic units connected to its object *#Article* (*magazine*, *article*, *instance*, not labelled in Figure 2) and predicate *class1* (*class1*, *OWL1*, not labelled in Figure 2) become active at the same time, i.e. synchronously. Similarly, the semantic units connected to SP2 (*Type* and *class2*) are activated upon activation of SP2. Details of this activation propagation algorithm are shown in the appendix.

At the end of the process, a specific activation pattern of semantic units is associated with each fired SP. The analogy result is determined by identifying those recipient units whose connections to the shared semantic units best fits the activation pattern of the driver's units. In Figure 2, the activation pattern of semantic unit results in establishing a mapping between *#owl:Class (#Article, rdf:#Type)* and *#owl:Class (#JournalPaper, rdf:#Type)*. In LISA, mappings are always between units of the same type (objects to objects, predicates to predicates). Hence, the mapping in the example is established between the objects *#Article* and *#JournalPaper*, *#Type* and *#Type*, and between the predicates *#owl:Class* and *#owl:Class*.

4. APPLYING LISA TO ONTOLOGY ALIGNMENT

Our framework for using LISA for ontology alignment consists of three components: 1) RDF parser, 2) LISA encoder, and 3) Ontology matcher. Our system accepts as input a source and a target ontology expressed in OWL. To apply LISA, the ontologies must be represented ("encoded") in the form of LISA proposition, sub-propositions, objects, predicates, and semantic units as described in Section 3. This encoding admits many design decisions. In this section, we describe the choices we have made and refer the reader to our discussion of future work where we describe plans for exploring this design decision space.

4.1 RDF Parser

To represent an OWL ontology as a set of simple propositions, in a form similar to LISA propositions, we first represent the OWL ontology in RDF triples format[2]. Next, the extracted RDF data is represented in the form of N-triples[3], a particular surface grammar for representing RDF triples, to be more easily parsed and transformed to LISA input format. Table 1 shows an example of two OWL statements that are transformed to the N-Triple format.

4.2 LISA Encoder

The LISA encoder transforms the N-Triple representations into LISA input files. An RDF statement is a ternary relation between a subject, a predicate, and an object that is similar to a proposition in LISA (although LISA propositions can be of higher arity than three). This suggests an intuitive encoding of RDF subjects as LISA objects, RDF predicates as LISA predicates, and RDF objects as LISA objects. Sub-propositions, while important in the analogy-making process, are not explicitly represented in LISA input.

In our encoding, RDF subjects and RDF objects are represented as LISA objects if the RDF triple has a *rdf#type* predicate and an *owl#class* object. In other words, we represent OWL classes as LISA objects. We also encode RDF subjects and objects as LISA objects if the RDF triple has a *rdf#type* predicate and an object of type *owl#ObjectProperty* or *owl#DatatypeProperty*. In other

[2]http://www.w3.org/TR/owl2-mapping-to-rdf/
[3]http://www.w3.org/2001/sw/RDFCore/ntriples/

Table 1: Example showing the transformation from OWL to N-Triples to LISA format.

	Ontology - Class Example	Ontology - ObjectProperty Example
OWL	<owl:Class rdf:ID="Article"> <rdfs:label xml:lang="en">Article</rdfs:label> <rdfs:comment xml:lang="en">An article from a journal or magazine.</rdfs:comment>	<owl:ObjectProperty rdf:ID="event"> <rdfs:domain rdf:resource="#Proceedings"/> <rdfs:range rdf:resource="#Conference" /> <rdfs:label xml:lang="en">event</rdfs:label> <rdfs:comment xml:lang="en">The event concerned with the proceedings.</rdfs:comment> <owl#ObjectProperty>
N-Triples	<onto.rdf#Article> <rdf#type> <owl#Class> . <onto.rdf#Article> <rdf-schema#label> "Article" . <onto.rdf#Article> <rdf-schema#comment> "An article from a journal or magazine" .	<onto.rdf#event> <rdf#type> <owl#ObjectProperty> . <onto.rdf#event> <rdf-schema#domain> <#Proceedings> . <onto.rdf#event> <rdf-schema#range> <#Conference> . <onto.rdf#event> <rdf-schema#label> "event" . <onto.rdf#event> <rdf-schema#comment> "The event concerned with the proceedings." .
	AnalogExample - Class and ObjectProperty	
LISA	Defpreds // Definition of the predicates and their semantic units (in square brackets). class [class1 OWL1] [class2 OWL2 classes] ; objProperty [objProperty1 rdfproperty1] [objProperty2 rdfproperty2] ; end ; Defobjs // Definition of the objects and their semantic units (following each object). Article magazine article instance ; Type instance subject ; Prop_#event event concerned proceedings #proceedings #conference ; end ; DefProps // Definition of the propositions, P1 and P2 in the following. P1 class(Article, Type) ; P2 objProperty (Prop_#event Type) ; end ;	

words, we also represent OWL properties as LISA objects. Intuitively, when OWL classes and OWL properties are represented as LISA objects, we can then make statements in LISA about these objects, i.e. LISA propositions, which represent the statements in the OWL ontology about the classes and properties. Table 1 presents two examples, a class and an object property, that are transformed to LISA's input format. The "Defpreds" section of the LISA input defines the predicates and their semantic units, corresponding to the triangles and the small circles in Figure 1. The "Defobjs" section of the LISA input defines the objects and their semantic units, corresponding to the large and small circles shown in Figure 1. Finally, the "Defprops" section of the LISA input defines the propositions of an analog, corresponding to the ellipses in Figure 1. Further details about the LISA input format can be found in [16].

If the RDF triples are not of the above forms, i.e. the triples do not represent an OWL class or OWL property, their subjects and objects are encoded as LISA semantic units. More specifically, the semantic units are generated from RDF triple statements with the following predicates: *rdf-schema#label*, *rdf-schema#comment*, *rdf-schema#subClassOf*. In these cases, we parse the RDF subject and RDF object, i.e. the label or comment, and each parsed element is represented as a LISA semantic unit. Further, the anonymous classes in RDF triple statements may contain descriptions, constraints, and other semantic elements related to the *owl#class* objects (which are represented as LISA objects). Table 1 shows some cases of different predicates such as *rdf-schema#label*, *rdf-schema#comment rdf-schema#range*, and *rdf-schema#domain* that their objects are parsed and encoded as semantic units.

4.3 Ontology Matcher

The encoded base and target analogs are then sent to the LISA algorithm presented in Section 3. We use the public version of LISA[4]. The generated alignments are influenced by the order and grouping with which the propositions are fired by the driver analog. To represent human cognitive constraints, only a small number of propositions can be activated together by the driver analog (approximately two to three propositions). In this first exploratory study, we fire propositions individually and in the order in which they are defined in the ontology (and thus appear in the LISA input specification). The exploration of different configurations of LISA is left to future work (Section 6).

In adhering to the main cognitive principles of analogical reasoning, LISA typically emits one-to-one mappings between objects and predicates of the two analogs, and each mapping has a confidence score. In the rare case when LISA emits a many-to-one mapping, we have used only mapping with the highest confidence score. No further post-processing of LISA results is performed.

5. EXPERIMENTAL EVALUATION

We conducted various experiments to test and evaluate LISA's performance on ontology alignment tasks. We begin by describing our experimental benchmark ontology alignment test set and the evaluation metrics. This is followed by our results and a comparison with existing approaches.

5.1 The OAEI 2010 Benchmark Dataset

To evaluate LISA's performance, we used the benchmark tests

[4]http://internal.psychology.illinois.edu/~jehummel/models.php

available from the Ontology Alignment Evaluation Initiative[5] (OAEI). OAEI is a consensus reference test set for evaluating ontology alignment systems. Since 2004, many ontology alignment systems participated in the annual evaluation contests and their performance has been published. We conducted our experiments using the OAEI benchmark series of tests in its 2010 version [2]. Detailed descriptions of all tests are found on the OAEI web page.

The reference (and source/base) ontology for the alignment tests has 33 named classes, 24 object properties, 40 data properties, 56 named individuals and 20 anonymous individuals in the academic publications domain. The 1xx set of tests is a simple test where a reference ontology is compared with itself or with its restriction to OWL-Lite. The 2xx set of tests are systematic tests where a reference ontology is compared with various modified versions of itself. The variations are obtained by 1) replacing entities by random strings, synonyms, names with different conventions or names in another language, 2) discarding or modifying comments, instances, properties, 3) suppressing, expanding, or flattening subsumption hierarchies, 4) expanding, replacing, or flattening classes. These two sets of tests are artificial in the sense that the reference alignments are not "created" by humans in the sense that human cognitive analogy processes were at work, but are the result of the specific transformations of the source ontology.

In contrast, the 3xx set of tests includes four real target ontologies, which have been manually aligned with the reference ontology. Thus, the reference alignments for these tests represent the result of human cognitive processes. The target ontologies in the 3xx set of tests comprise four real ontologies: 1) *BibTeX/MIT* (15 named classes, 40 data properties), 2) *BibTeX/UMBC* (16 named classes, 6 object properties, 25 data properties), 3) *Karlsruhe* (56 named classes, 72 object properties), and 4) *INRIA* (41 named classes, 40 object properties, 11 data properties). One reason for using the 2010 version of the tests is that published results are available for these non-synthetic tests with real ontologies, which have been omitted in later OAEI test suites.

To evaluate the performance of LISA, the alignments for OWL properties and classes are compared to the set of reference alignments using the precision and recall. For space reasons, we report only the composite F-score metric.

5.2 Results

Table 2 provides the F-scores and shows that LISA performs well on the 1xx and 2xx sets of tests. The F-scores show that LISA's performance was better than most systems that participated in the OAEI 2010 contest[6].

The 3xx set of tests is the most relevant to measure the effectiveness of the alignment approaches, as it is the most realistic and, more importantly, the reference alignments are created by humans. The varying F-scores indicate that the slight decrease in recall is compensated by high precision. Over all the 3xx set of tests, LISA performed reasonably well and ranked 6th out of the 12 competing approaches, as illustrated in Figure 3.

A closer look at the tests shows that LISA's performance varies greatly between tests. First, elements such as labels, comments, instances in the ontologies are of considerable importance for LISA to perform well (e.g., test 101, 221, 228). This is not surprising as these elements are parsed to form the semantic units that connect the driver and recipient analog in LISA. Second, LISA did not perform well when the hierarchy of an ontology is flattened (e.g., test 222, 237, 239, 246). In contrast, when the hierarchy is expanded (test 223), LISA scored well. This suggests that the structural information represented by a class hierarchy is important, as would be expected from the description of LISA earlier. Third, LISA achieved better results when the alignment did not include the alignment of properties, i.e. properties were suppressed (e.g., test 228, 233, 236). Finally, as we expected from our ontology encoding and importance of labels, the performance of LISA in Table 3 on "scrambled" tests (201–202 and 248–266), where concept names or labels are replaced by random strings, was relatively poor.

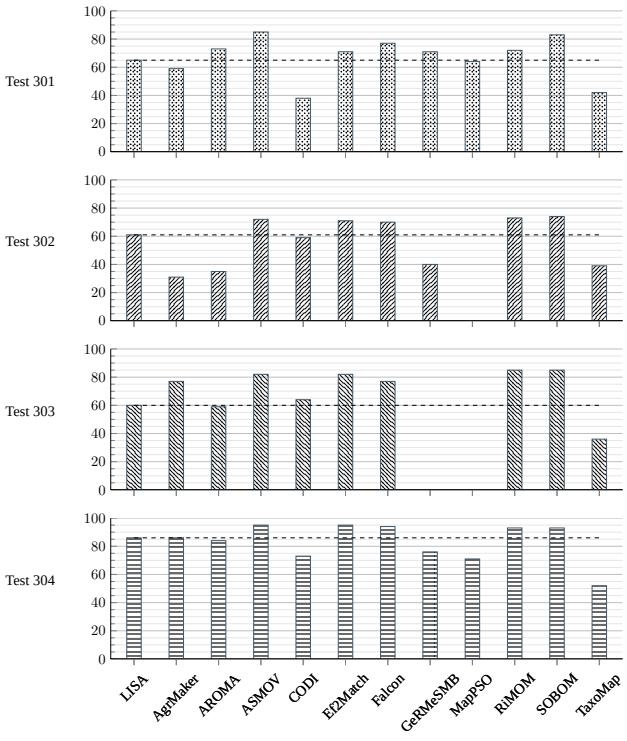

Figure 3: F-scores for the OAEI 3xx tests, comparing LISA with systems that participated in the 2010 OAEI contest.

Figure 4 shows the results of each of the test sets for LISA and for all the systems that participated in the 2010 OAEI contest [2]. When we compute the average F-scores across all tests for each of the systems, LISA is again ranked 6th out of the 12 systems, as illustrated in Figure 4.

6. DISCUSSION AND CONCLUSION

This paper has presented an application of LISA, a computational cognitive model of analogical reasoning, to the task of ontology alignment. To our knowledge, this is the first work to use human cognitive analogy-making in the ontology alignment context. This initial, exploratory application was intended to answer the question whether ontology alignments are like analogies. Our results are frequently better than the existing state-of-the-art. However, our approach does not yet dominate other approaches on all tests, especially those with real ontologies for which human reference alignments are available (the OAEI 3xx tests).

In light of this, one could answer our research question in different ways. Pessimistically, the performance of LISA might suggest that ontology alignment is not like making analogies. If it was, LISA should perform better, especially on the OAEI 3xx set of

[5] http://oaei.ontologymatching.org/

[6] OAEI results obtained from http://oaei.ontologymatching.org/2010/results/benchmarks/index.html

Table 2: Comparing LISA F-scores with systems that participated in the OAEI 2010 contest.

Test #	LISA	AgrMaker	AROMA	ASMOV	CODI	Ef2Match	Falcon	GeRMeSMB	MapPSO	RiMOM	SOBOM	TaxoMap
101	95	98	98	100	100	100	100	100	100	100	100	50
221	94	97	98	100	97	100	100	100	100	100	100	50
222	35	98	98	100	100	100	100	99	100	100	100	45
223	92	94	93	100	100	100	100	96	98	98	99	44
224	100	98	97	100	100	100	99	100	100	100	100	50
225	90	98	98	100	99	100	100	100	100	100	100	50
228	100	100	100	100	100	100	100	100	100	100	100	100
232	93	97	96	100	96	100	99	100	100	100	100	50
233	100	100	100	100	93	100	100	98	100	100	100	100
236	100	100	100	100	100	100	100	100	100	100	100	100
237	76	98	96	100	99	100	99	100	99	100	100	45
238	91	93	92	100	99	100	99	96	97	98	98	44
239	65	98	98	98	98	98	100	98	98	98	98	93
240	92	91	83	98	95	98	100	85	92	94	98	88
241	100	100	98	100	93	100	100	98	100	100	100	100
246	51	98	97	98	98	98	100	98	98	98	94	93
247	75	88	80	98	98	98	100	91	89	94	98	88

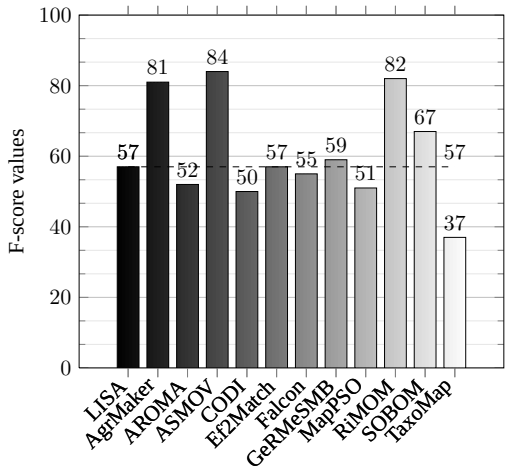

Figure 4: Mean F-scores across all tests, comparing for LISA with systems that participated in the 2010 OAEI contest.

tests. However, given that other cognitive models of analogy remain to be explored, we are not yet ready to come to this general conclusion. Perhaps LISA is not the correct model for analogical reasoning; after all, it has only been experimentally tested on small problems. It may be the case that the size of the ontologies in the alignment problems is such that humans use a different strategy than they would on small analogy problems. Hence, the reference alignments generated by humans may be generated by a different process than analogical reasoning. Thus, while LISA may accurately represent analogical reasoning, the task of ontology alignment is not based on analogies and thus the mismatch and relatively low performance on our evaluation. Finally, given the fact that LISA (and our ontology encoding) is highly configurable, perhaps our initial, naive application is inappropriate for the ontology alignment problem. This would suggest that significant further improvements can be achieved with this approach in the future.

In Section 1 we indicated that there is much variability in the way humans make alignment decisions [5, 21], and many times such decisions depend on the context of ontology integration [5]. This however, is an area of research that has not yet received much attention. As we are not aware of how the OAEI 3xx reference alignments are developed, there may be validity problems surrounding this issue in the sense that the reference alignments may be idiosyncratic to some group of people [4]. This in turn may be the reason why not only LISA but most other approaches fail to recover the reference alignment: The concept of analogy or similarity held by researchers and encoded in their alignment algorithms does not coincide with the concept of analogy or similarity held by the creator(s) of the reference alignments. Later rounds of the OAEI contest introduced other tests for which human-based reference alignments are available. The performance of LISA on these recent tests remains to be explored.

Our future work will focus on three different avenues for improvement. First, LISA is configurable, for example in terms of the firing order of propositions, the grouping of propositions, etc. Moreover, LISA offers the ability to fire one or more propositions at the same time, as well as to reverse the driver and recipient analogs. Thus, systematic experimental comparisons are required to determine the optimal configuration of LISA for different alignment problems.

Second, the encoding of ontologies in LISA admits many design decisions. For example, we have chosen the intuitive route via an RDF representation of OWL ontologies and a more or less direct mapping to LISA representation elements. Other encodings are possible. For example, many of the OWL constructs that are now encoded as semantic units may be represented as LISA objects in their own right. This would strengthen the structure of the problem representation, yet at the same time reduce the number of semantic units by which activation progresses from driver to recipient. The effect that such a change in encoding might have remains to be explored in systematic experiments.

Third, while LISA has been demonstrably successful in explaining human analogies in the psychology lab, other computational models, such as SME, exist and remain to be explored, each with

Table 3: Scrambled tests' F-scores, comparing LISA with systems that participated in the OAEI 2010 contest.

Test #	LISA	AgrMaker	AROMA	ASMOV	CODI	Ef2Match	Falcon	GeRMeSMB	MapPSO	RiMOM	SOBOM	TaxoMap
201	95	91	94	100	12	76	97	94	42	100	95	50
202	22	88	0	88	0	7	0	38	5	80	63	1
248	20	71	0	86	0	1	0	36	5	63	48	1
249	13	87	1	88	1	7	0	34	5	78	63	1
250	60	56	0	62	0	11	0	5	3	73	20	0
251	15	78	0	86	0	7	0	43	5	68	46	1
252	20	77	0	86	0	7	0	36	2	68	50	1
253	10	71	1	87	1	1	0	41	6	61	46	1
254	53	56	0	42	0	0	0	5	3	56	0	0
257	27	56	0	49	0	11	0	5	6	70	25	0
258	6	78	1	86	1	7	0	27	6	65	46	1
259	5	77	1	86	1	7	0	28	4	63	50	1
260	50	60	0	56	0	12	0	0	11	66	5	0
261	45	53	0	56	0	11	0	5	3	54	9	0
262	6	56	0	42	0	0	0	5	3	52	0	0
265	6	60	0	46	0	12	0	0	3	63	5.5	0
266	0	53	0	44	0	11	0	5	6	51	4	0

their own configuration space and encoding decisions.

Besides improving the performance of cognitive methods for ontology alignment using these approaches, work is also required to investigate the effect of reasoning support. As OWL ontologies admit inferences over the subsumption hierarchy, it is possible to also use inferred subsumption relationships in the ontology alignment algorithm, in addition to those explicitly axiomatized in the ontology. Furthermore, we intend to examine the scalability of LISA to larger ontologies than the OAEI benchmark test cases. A formal complexity analysis is complicated by the fact that LISA is a connectionist architecture with a convergence-based iterative algorithm, so that upper bounds to convergence must be developed, instead of time-complexity results.

In conclusion, our research has not only opened up possibilities for future work in the area, but has also contributed to a theoretical foundation of ontology alignment in the psychology of analogical reasoning. This can provide some guidance for the field of ontology alignment in that results from the ongoing work in cognition may be transferred to improve ontology alignment performance.

7. REFERENCES

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

[2] J. Euzenat, A. Ferrara, C. Meilicke, J. Pane, F. Scharffe, P. Shvaiko, H. Stuckenschmidt, O. Sváb-Zamazal, V. Svátek, and C. T. Dos Santos. Results of the ontology alignment evaluation initiative 2010. In *OM*, 2010.

[3] J. Evermann. An exploratory study of database integration processes. *IEEE Transactions on Knowledge and Data Engineering*, 20(1):99–115, 2008.

[4] J. Evermann. Theories of meaning in schema matching: A review. *Journal of Database Management*, 19(3):55–83, 2008.

[5] J. Evermann. Contextual factors in database integration — a delphi study. In *Proceed. 29th International Conference on Conceptual Modeling (ER 2010)*, 2010.

[6] J. Evermann. Applying cognitive principles of similarity to data integration–the case of siam. *Proc. Americas Conf. on Info. Sys.*, 2012.

[7] B. Falkenhainer, K. D. Forbus, and D. Gentner. The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, 41(1):1 – 63, 1989.

[8] R. M. French. The computational modeling of analogy-making. *Trends in Cognitive Sciences*, 6(5):200 – 205, 2002.

[9] D. Gentner. Structure-mapping: A theoretical framework for analogy. *Cognitive science*, 7(2):155–170, 1983.

[10] D. Gentner. Psychology in cognitive science: 1978–2038. *Topics in Cognitive Science*, 2(3):328–344, 2010.

[11] D. Gentner and K. D. Forbus. Computational models of analogy. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2(3):266–276, 2011.

[12] R. L. Goldstone and D. L. Medin. Time course of comparison. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 20(1):29, 1994.

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

[14] P. Hitzler and F. van Harmelen. A reasonable semantic web. *Semantic Web*, 1(1):39–44, 2010.

[15] D. Hong-Hai and R. Erhard. Matching large schemas: Approaches and evaluation. *Information Systems*, 32(6):857 – 885, 2007.

[16] J. E. Hummel and K. J. Holyoak. Distributed representations of structure: A theory of analogical access and mapping. *Psychological Review*, 104(3):427, 1997.

[17] D. C. Krawczyk, K. J. Holyoak, and J. E. Hummel. Structural constraints and object similarity in analogical mapping and inference. *Thinking & Reasoning*, 10(1):85–104, 2004.

[18] P. Lambrix and H. Tan. SAMBO-A system for aligning and merging biomedical ontologies. *Web Semant.*, 4(3):196–206, 2006.

[19] A. Lovett, E. Tomai, K. D. Forbus, and J. M. Usher. Solving geometric analogy problems through two-stage analogical mapping. *Cognitive Science*, 33(7):1192–1231, 2009.

[20] R. Lukyanenko and J. Evermann. A survey of cognitive theories to support data integration. *Proc. Americas Conf. Info. Sys.*, 2011.

[21] A. Tordai, J. van Ossenbruggen, G. Schreiber, and B. Wielinga. Let's agree to disagree: on the evaluation of vocabulary alignment. In *Proc. Int. Conf. Knowledge Capture*, K-CAP '11, pages 65–72. ACM, 2011.

[22] H. Wei, Q. Yuzhong, and C. Gong. Matching large ontologies: A divide-and-conquer approach. *Data & Knowledge Engineering*, 67(1):140 – 160, 2008.

APPENDIX

The LISA algorithm is described in Table 4. Knowledge is represented using propositions ("P units"), sub-propositions ("SP units"), predicate and object units, and semantic units as described in Section 3. Each unit in a driver analog can have a mapping relationship and mapping weight with units of the same type in the receiver analog. These mapping relationships are dynamically created. Each unit has an activation value and can receive and transmit "input" to update its activation and that of connected units. The follow paragraphs describe central steps in the algorithm. For formal details, the reader is referred to Appendix A in [16].

Step 2a: Propositions (P units) operate in either parent, child or neutral mode. In parent mode, P units pass activation input to their sub-propositions (SP units). In child mode, P pass activation input to the SP they are part of. In neutral mode, P units pass activation input both upwards and downwards. Initially, P units in the driver analog are in neutral mode, except those actively selected into working memory (step 1), which are in parent mode. P units in the receiver analog update their modes on the basis of their inputs from SP units above and below and inputs from P units in the driver (either in parent or child mode) via mapping connections.

$$m_i = \begin{cases} \text{Parent} & \text{if} SP^{\uparrow} - SP^{\downarrow} + P^{parent} - P^{child} > \theta \\ \text{Child} & \text{if} SP^{\uparrow} - SP^{\downarrow} + P^{parent} - P^{child} < -\theta \\ \text{Neutral} & \text{otherwise} \end{cases}$$

Step 2b: Each SP unit consists of an excitor and an inhibitor unit. Each inhibitor causes the corresponding excitor's activation to oscillate over time. In combination with strong SP-to-SP inhibition, the inhibitors cause separate SPs to fire at different times.

In the driver analog, SP excitors receive excitatory input from P units above (p), inhibitory input from other driver analog SPs ($\sum_j e_j$) with a sensitivity s that decays from a maximum value immediate after an SP fires, and inhibitory input from their own inhibitors (I). The net input is

$$n = 1 + p - \frac{s \sum_j e_j}{1 + \text{NSP}} - I$$

where NSP is the number of SPs in the driver with activation > 0.2. An SP inhibitor receives input only from the corresponding excitor.

P units in parent mode receive input from the excitors of the SP units below with net input $n = \sum_j e_j$, where e_j is the activation of the excitors of SP unit j. P units in child mode, predicate or object units receive input from the excitor and the inhibitor of the SP above with net input $n = \sum_j (e_j - I_j)$, where I_j is the activation of the inhibitor of SP unit j.

Step 2c: The global inhibitor becomes active whenever all SP excitors in the driver analog have an activation below a certain threshold. When active, it inhibits all units in the recipient analog to inactivity.

Table 4: LISA Algorithm from [16] (P = Proposition, SP = Subproposition)

Repeat until convergence of mapping weights:

1. Select a set of P units of the driver analog ("phase set P_S")
2. Repeatedly update the state of the network in discrete time units $t = 1 \ldots 220 \times |\text{SP}|$. For each step do:
   (a) Update modes of all P units in recipient and driver
   (b) Update inputs of all units in P_S (P, SP, predicate and object units)
   (c) Update global inhibitor
   (d) Update inputs to all semantic units
   (e) Update input to units in recipient
   (f) Update activation of all units
   (g) Create mapping connections
3. Update mapping weights

Step 2d: Semantic units receive weighted input from predicate and object units in both the driver and receiver analog with the net input computed as $\sum_j a_j w_{ij}$ where a_j serves to normalize activation to between 0 and 1. Input from the receiver analog occurs only after each SP in the driver has fired at least once.

Step 2e: Units in the recipient analog receive four inputs: within-proposition excitatory input P, within-class inhibitory input C (i.e. SP-to-SP, object-to-object, etc.), out-of-proposition inhibitory input O (i.e. P units inhibit SP units of other P units, and SP units inhibit predicate and object units of other SP units), and both excitatory and inhibitory input via the cross-analog mapping relationships M. The net input is computed as

$$n = P + M - C - \pi O - \pi \Gamma$$

Specific formulas for each of the these components for different kinds of units are given in [16].

Step 2f: At the end of each iteration and after computation of all excitatory and inhibitory input, all units update their activation according to $\Delta a = \gamma n(1 - a) - \delta a$ where γ is a growth rate and δ is a decay rate; n is the net input (excitatory minus inhibitory).

Step 2g: When the activation of a unit in the recipient analog first exceeds 0.5, mapping connections are created between it and units of the same type in the driver analog. These are important for step 7 in the subsequent iteration.

Step 3: Each mapping weight w_{ij} is updated, taking into account competing mapping hypotheses and ensuring a 1:1 mapping constraint.