IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 8, 2021, accepted January 22, 2021, date of publication February 4, 2021, date of current version July 1, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3057081

Ontology Matching: State of the Art, Future Challenges, and Thinking Based on Utilized Information

XIULEI LIU[1,2], QIANG TONG[1], XUHONG LIU[3], AND ZHIHUI QIN[1,2]

[1]Laboratory of Data Science and Information Studies, Beijing Information Science and Technology University, Beijing 100101, China
[2]Beijing Advanced Innovation Center for Materials Genome Engineering, Beijing Information Science and Technology University, Beijing 100101, China
[3]Beijing Key Laboratory of Internet Culture and Digital Dissemination Research, Beijing Information Science and Technology University, Beijing 100101, China

Corresponding author: Xiulei Liu (xiuleiliu@hotmail.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB1400402, in part by the Qin Xin Talents Cultivation Program, Beijing Information Science and Technology University, in 2020, in part by the Beijing Education Commission for General Project of Science and Technology Plan under Grant KM202111232003, and in part by the Beijing Natural Science Foundation under Grant 4204100.

ABSTRACT Information used in existing ontology matching solutions are usually grouped into four categories: lexical information, structural information, semantic information, and external information, respectively. By summarizing and analyzing the approaches for utilizing the same kind of information, this paper finds that lexical information is mainly analyzed based on text and dictionary similarity. Similarly, structural information and semantic information are mainly analyzed based on graph structure and reasoner, respectively. The approaches for aggregating information analysis results are discussed. Challenges in the analysis of various types of information for existing ontology matching solutions are also described, and insights into directions for future research are provided.

INDEX TERMS Information classification, information analysis, ontology matching, semantic web.

I. INTRODUCTION

With the development in recent years in semantic Web, many research institutions around the world have advanced ontology theories and techniques, yielding widespread use of ontology in many applications [1], [2]. By using formal representations of concepts and interrelations in a field, ontology enables knowledge to be reused, shared, and interoperated across data sources and disciplines in various applications [2]. However, there is lack of an ontology that is unique, widely acknowledged, and covers all fields in the real world. Most ontologies that represent the model of the sharing concept in similar, or the same, fields are constructed and maintained by knowledge engineers with different expertise in terminology. The heterogeneity of these ontologies hinders the application systems from sharing, reusing, and interoperating knowledge [3]. Therefore, offering an approach to overcome ontology heterogeneity is a major challenge for ontology-based applications.

Ontology matching is one solution to the ontology heterogeneity problem. The purpose of ontology matching is to establish a relation between entities in heterogeneous ontologies that depict similar, or the same, fields. This relation is known as correspondence. The application systems can therefore share, reuse, and interoperate knowledge in a broader field by using these correspondences when transmitting heterogeneous information. Ontology matching is widely used in ontology engineering, biological medicine, P2P information sharing, Web service composition, and the semantic Internet of Things [4]. Ontology matching can be described as the function shown in Equation 1 and represented in Fig. 1.

$$A^I = f\left(O^l, O^2, A, p, r\right) \tag{1}$$

where O^1 and O^2 denote the pair of ontologies to be matched, A denotes the set of known correspondences between ontologies, p denotes the set of input parameters, r denotes the external resources consulted during the matching process, and A^1 denotes a pair of correspondence generated during the matching process. The correspondences in A^1 and A can be represented by quadruple, *<e, e', n, R>*, where e and *e'* denote the same type of entities from different ontologies, *n* denotes the reliability of the established correspondence, which is typically represented by a real number in the range

The associate editor coordinating the review of this manuscript and approving it for publication was Mohamed Elhoseny.

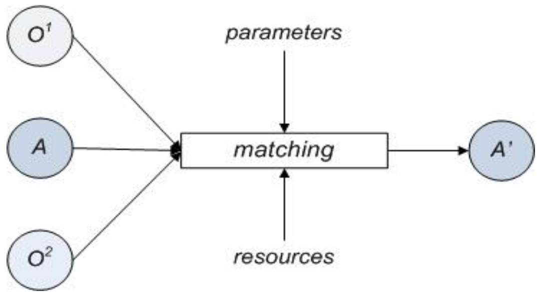

FIGURE 1. Ontology matching process [5].

[0, 1] and is known as the confidence degree, and R denotes the semantic relation between two entities and generally has four types: include($\sqsupseteq$), beInclude($\sqsubseteq$), disjoint($\perp$), and equivalent($\equiv$). In this paper, we do not differentiate ontology matching, ontology alignment, and ontology mapping, but details are available in the literature [8]. Ontology matching involves many topics, such as setting system parameters, evaluation of matching, user interaction [12], and ontology matching debugging [13]. This paper focuses on correspondence generation, which is a major task in ontology matching.

In the last decade, ontology matching yielded many ontology matching schemes. The authors of [6] compared early ontology matching schemes in terms of theoretical framework, experience report, coordinator, translator, and tool using. In [7], matching schemes were compared in terms of heterogeneous phenomena that can be solved by the matching schemes. In [8], matching schemes were compared according to heterogeneous ontologies and whether they are local or general ontologies. In [9], ontology matching tools were compared from the perspective of the user. Progress in ontology matching and the major obstacles (e.g., data evaluation and method, debugging techniques, and large-scale ontology matching) were discussed in [10]. Authors of [11] described the advantages and disadvantages of competing schemes according to results of Ontology Alignment Evaluation Initiative (OAEI) tests. However, they seldom analyze the solutions in utilized information.

Section II provides a summary and classification of information used by various ontology matching schemes: lexical, structural, semantic, and external. In Section III, IV, and V, approaches of different ontology matching schemes for processing identical information are analyzed, compared, and summarized; challenges to analysis of information in ontology matching schemes are described; and insights into these challenges are provided. Section VI analyzes methods of various matching schemes for aggregating information analysis. Section VII concludes the paper.

II. INFORMATION USED IN ONTOLOGY MATCHING

In the context of ontology-based applications, ontology matching has become a widely discussed topic in research communities. Information used in ontology matching schemes in recent years is summarized below:

Lexical Information: This information mainly comes from natural language information (e.g., words and short sentences) in ontology non-description logic axioms (e.g., entity label axiom and comment axiom). Through the ambiguity information of words, the lexical information of entities can be marked, the problem of word sense combination can be explored, the lexical information can be enriched, and the lexical relationship between entities can be inferred based on the entity labels.

Structural Information: This information is a representation of ontology semantic information, such as part-of, is-a, sub-class and sub-property. Through the graph-based similarity propagation algorithm, the mapping unit of the information is obtained, and the adjacent elements of the concept in the ontology structure and their semantic relations are fully excavated.

Semantic Information: This information usually comes from the explanation of the ontology description language, such as model theoretic semantics (used in this paper) and Resource Description framework (RDF) compatible semantics for explaining the Web Ontology Language Description Logic (OWL-DL) ontology description language. Instead of being shown explicitly, these pieces of information are hidden in the ontology constructors and axioms. Different semantic techniques can be used to read them, such as the tableau algorithm, which is commonly used in the reasoner, and the structure subsumption algorithm, which is commonly used in non-standard reasoning problems.

External Information: Various sources outside ontology, mostly used to offset insufficient background knowledge during ontology matching. These commonly used external information include WordNet, Linked-Open data, CyC-ontology, Wikipedia, and DBpedia.

III. ANALYSIS OF LEXICAL INFORMATION AND EXTERNAL INFORMATION (WordNet)

While defining ontology labels and comments, knowledge engineers mostly describe entities using words of natural language (e.g., Book), word combinations (e.g., booktitle), abbreviations (e.g., Misc), and short sentences. Therefore, lexical information analysis is essential to ontology matching.

A. ANALYSIS OF PRESENT SITUATION

Most existing solutions analyze ontology lexical information using the text similarity method and the dictionary-based similarity method, such as LiLy [4], Ontology Mapping by Particle Swarm Optimisation (MapPSO) [14], UFOme, Integrated Learning In Alignment of Data and Schema (ILIADS) [15], and Automated Semantic Mapping of Ontologies with Validation (ASMOV) [16].

The text similarity method can be classified into the string-based method and the descriptive document-based method. It is primarily used as an approach to the randomness in the input by knowledge engineers, such as books and book, as well as comes and come. The string-based method [17] is original and frequently used, including the edition distance-based method [18], [19], the prefix/suffix similarity method, and the Jaro-Winler fraction method. The descriptive document-based method [4], [45] views an entity's relevant texts as the descriptive document, and determines whether entities are interrelated by processing the descriptive documents of different entities and performing similarity computations. The descriptive document-based method is related to lexical information and also involves the structural information of the entity [4], [19].

The dictionary similarity-based method [17], [18] first uses natural language processing techniques (e.g., stem extraction, stop word, and participle elimination) to process entity labels and comments by viewing entity labels and comments as natural language texts. Next, language resources are used to mine the similarity between entities to improve the quality of matching. Language resources that are available include dictionaries, thesauruses, and terminology. Some terminology's description of the peering relation and the hierarchical relation (e.g., synonymic and antonymic) can be used to improve the ability in processing heterogonous phenomena (i.e., homographic and synonymic) in ontology. The most popular external resource is WordNet [10], [25].

To achieve improved performance, many matching schemes compute the similarity of lexical information by combining text similarity with dictionary-based similarity [14]–[16], [18], [42], [43].

In [18], edition distance-based similarity (denoted by simDistance) and WordNet-based word sense similarity (denoted by simWordNet) are first computed. Next, weighted sums of the two similarities simLexical = α^*simDistance + β^*simWordNet are computed to obtain lexical similarity between entities. Cupid in [17] also employed this method to combine string similarity with WordNet-based word sense similarity.

ASMOV computes lexical similarity between entities by introducing thresholds as well as combining text similarity and word sense similarity, as shown in the following equation:

$$s^L\left(e,e'\right) = \left\{\begin{array}{ll} 1.0 & if\ l=l' \\ 0.99 & if\ l' \in synl\ (l) \\ 0.0 & if\ l' \in ant(l) \\ Lin\left(l,l'\right), & if\ l \in w \cap l' \in w \cap l' \notin syn(l) \\ \frac{tok(l) \cap tok(l')}{max(|tok(l)|,|tok(l')|)}, & otherwise \end{array}\right\} \tag{2}$$

where l and l' denote the labels of entities e and e', W denotes WordNet, syn(l) and ant(l) denote the set of synonyms and antonyms of l, Lin(l, l') denotes the information theory-based text similarity proposed by [46], and tok(l) denotes the set of words corresponding to the entity label. For example, the set of words corresponding to 'bookTitle' is {{book}, {title}}.

In [19], RI Mission of Mercy (RIMOM) used two methods to compute text similarity between entities: edition distance-based similarity (denoted by sim_Name (e1, e2)) and vector distance-based similarity (denoted by sim_Vec (e1, e2)), where e1 and e2 come from entities of different ontologies. Next, the following equation is used to combine these two similarities and form the final lexical similarity between entities:

$$sim\left(e_1,e_2\right) = \frac{w_{name}\sigma\left(sim\_Name\left(e_1,e_2\right)\right) + w_{Vec}\left(sim\_Vec\left(e_1,e_2\right)\right)}{w_{name}+w_{\mathrm{Vec}}} \tag{3}$$

where $sim(e_1, e_2)$ denotes lexical similarity between e_1 and e_2, σ denotes the sigmoid function and $\sigma(x) = 1/(1+exp(5(x-\alpha))(\alpha\ is\ experimentally\ set\ to\ 0.5)$, and w_{name} and w_{VEC} represent the weights based on entity label similarity between entities and the weights based on structure similarity between entities, respectively.

In [15], ILIADS first used the Jaro-Winkler fraction method to compute text similarity between entities and then computed senses similarity using WordNet. Finally, the maximum of text similarity and lexical similarity was defined as the lexical similarity.

RiMOM, Lily, ASMOV, Cupid, Optimal Linear Arrangement (OLA), ILIADS and the method in [18] adopt similar approaches when using lexical information. First, the methods compute text similarity based on the string method or the document description method. Then, the methods compute lexical similarity using external WordNet. Finally, the methods analyze lexical information by computing the weighted sum of different similarities. The differences between methods lie in their approaches for computing text similarity or senses similarity, and in the use of their respective empirical formulas for computing the sum of these similarities. For example, ASMOV combines these similarities in a discrete manner. When the proportion of the structural information reaches a threshold (e.g., γ) in the ontology, RiMOM only considers the similarity of vector distance relevant to structural information. But when the proportion of lexical information reaches a threshold in the ontology, RiMOM only considers the similarity of edition distance relevant to lexical information, whereas only the highest similarity is taken into account in ILIADS.

Unlike the aforementioned approaches, S-Match [22]–[24] defines entity labels by combining the word's proper senses and semantic elements (e.g., *conjunction* $\sqcap$ *and disjunction* $\sqcup$) to represent the entity's lexical information. WordNet is used as the lexical relation between the reasoning entities in the knowledge base (e.g., $\sqsubseteq$, $\sqsupseteq$, $\perp$, $\equiv$) instead of computing the similarity between entities.

B. CHALLENGES

Section III.A showed that the problems described below are neglected in the similarity method adopted by many existing schemes and the S-Match method for computing lexical relation [4], [7]:

Senses Combination: Concept labels and comments primarily consist of many words. However, the above methods do not consider relation between two concept words. Consider the concept pair to be matched <Book1, Book2 Title>. Since the sense similarity of <*Book1, Book2* > is 1, the final similarity is very large. In practice, Book and Book Title is different and may cause erroneous correspondences. Another example is the concept of "*monograph or collection*" whose senses information represents "*monograph* ⊔ *collection*".

Senses Fuzziness: Senses are not correlated but express the same concepts. Consider the following two properties <*Book, publishedBy, IEEE*> and <*Book, hasPublisher, IEEE*>. In this example, "*published*" and "*publisher*" are not isomeric because their senses are not the same, but they express the same role of publisher.

Difficulty in Detecting the Right Senses: It is still a challenge to derive the correct senses of words from the ontology context in the natural language processing domain because there is no effective approach. Consider the matching of the following concept pair <*Report1, Report2*>. Using s detecting techniques may provide the interpretation of Report1: *a written document describing the findings of some individual or group*, and the interpretation of Report2: *a short account of the news. It is unreliable to compute sense similarity based on the wrong senses.*

C. FUTURE WORK

Insights into the challenges described in Section III.B are discussed below:

To address problems of senses fuzziness and the difficulty in detecting senses, natural language processing techniques should be used to derive proper word senses and external source-based sense-extending techniques should also be employed. Extending the word's proper sense represents the possible interpretations of this word in context. This overcomes the flaw that only proper senses of the word are detected and is helpful in the detection of potential correspondence. For example, proper senses of the word "published" are *publish, bring out, put out, issue, release: prepare and issue for public distribution or sale*, and proper senses of the word "publisher" are *publisher, publishing house, publishing firm, publishing company: a firm in the publishing business*. The *pertainym* relation in WordNet can be used to detect the extension of the proper senses of "*published*". The extension of "*published*" may contain the proper senses of "*publisher*".

According to the entity label method defined in S-Match, label the conjunction or disjunction of a group of words provides an approach to senses combination and improves the probability of detecting potential correspondence. But, concepts and logic elements, in property ranges and property domains, are neglected while processing properties from different ontologies in S-Match. Consider <*Book*⊔*Report***title***String*>. Like "**title**", important information is available from "*Book*", "⊔", and "*Report*". But these data are not analyzed in S-Match. It is only viewed as a factor in the computation of similarity by most similarity-based methods. Hence, the definition of entity label in S-Match should be extended to introduce property domains, concepts in property ranges, and logic elements.

IV. STRUCTURAL INFORMATION ANALYSIS

Structural information analysis is essential in ontology matching because two similar ontologies are structurally similar.

A. ANALYSIS OF PRESENT SITUATION

In ontology matching approaches, ontology is represented by a graph to help facilitate the analysis of structural information [44]. Node similarity in the graph is computed to represent similarity between entity structures. Methods for representing the ontology graph include the RDF(S) graph [18], senses sub-graph, layered entity graph [15], [43], and directed/undirected graph. Schemes for computing similarity between nodes in the graph include similarity flooding [26], [42], graph matching [20], circuit model [4], Bayesian decision theory [27], Markov network [28], association rule paradigm [29], and so on.

The main idea of Cupid [2] is to perform structure matching on schema of the tree, whereas the similarity between two entities depend on their linguistic similarity, data type similarity, and neighboring node similarity. OLA describes ontology as a directed labeled graph, where the node corresponds to the ontology entity and the edge corresponds to the ontology relation. Concept similarity in OLA depends on similarity with the neighbor of the corresponding node in the labeled graph.

In [18], an RDF graph is constructed to analyze the ontology structural information. Next, parental node similarity and child node similarity are computed and added together in a weighted manner to form the structure similarity between entities. Ontology Integration based OWL DL Closure(OIODC) [31] constructs an RDFS graph for the ontology and then iteratively applies the ontology inference rule [27], [30]. If an assertion satisfies one of these rules, a new assertion will be generated and added to the original assertion until all assertions dissatisfy the conditions for triggering the inference rules. At this point, iterations should be terminated to obtain the ontology closure graph. Structure similarity between entities will be compared according to the ontology closure graph.

Structure similarity in ASMOV consists of relation similarity, internal/external similarity of the property, internal/external similarity of the concept, and the individual's internal similarity. Internal similarity of the concept is defined as restrictions in the property relevant to this concept, such as the restriction value and the restriction cardinality in the property. Assuming that the concepts c and c' are from different ontologies, $P(c)$, is the set of associated ontology properties defined for the concept c. $P(c')$ is similarly defined. ASMOV processes each pair of properties from $P(c)$ and $P(c')$ separately. Assuming that $p_m \in (c)$ and $p_n \in P(c')$, the similarity between them consists of two components: $S^{card}(p_m, p_n)$ and $s^{value}(p_m, p_n)$. If the restriction cardinality maximum and minimum of pm in c are identical to the restriction cardinality maximum and minimum of pn in *c'*, then $s^{card}(p_m, p_n)$ equals 1, otherwise 0. If the restriction value maximum and minimum of pm in c are identical to the restriction value maximum and minimum of p_n in *c'*, then $s^{value}(p_m, p_n)$ equals 1, otherwise 0. The specific sense of the potentials are not analyzed when ASMOV considers the property potentials. Even if two properties have different maximums and minimums for potentials in a concept, they may have the same senses. For example, when knowledge engineers construct the same concept of "*Book*", some think it should be $\geq 0 hasAuthor$, while others think it should be $\geq 1 hasAuthor$, but ASMOV views them as different. Therefore, it is essential to analyze senses information in the constructors and axioms.

For most of the aforementioned schemes (e.g., Cupid, OLA, ASMOV, and OIODC), these rules will fail when there is minimal text information in the ontology. The reason is that when the entities lack text information to assess the similarity between them, the idea will be inapplicable to the judgment of their neighbors' similarity. The idea is extended by Lily and RiMOM [19], [26] to enable similarity to be disseminated in the graph while ensuring the final similarity converges in order to be able to derive more correspondences.

According to RiMOM, if two entities from two different ontologies are similar, the structure similarity between their associated entities will increase. Therefore, it constructs an RDF for the ontology using the similarity flooding strategy to compute the similarity between these nodes in the graph. Lily allows similarity to be further disseminated in the graph because the more similar the entity's neighbors, the more similar the entities themselves, while ensuring that the similarity will converge finally. Lily is based on the circuit model (e.g., annealing function, electrical resistivity, and current distribution) to semantically extract the ontology concept sub-graph and the property sub-graph. The similarity dissemination algorithm is used to process structure information in ontology. In [26], the initial similarity between the ontology entities is computed (e.g., the similarity based on the label method). Then, the similarity dissemination graph is constructed using the structural information of the ontologies to be matched. Nodes in the graph are the pair of entities from different ontologies. Finally, the relation between different pairs of entities is used to re-compute the initial similarity between the two paired entities in order to disseminate the similarity.

IMatch [28] uses a first-line matcher (e.g., Edna) to compute potential correspondence and then employs the Markov network-based second-line matching to analyze ontology structural information. DSSim computes ontology matching by combining the lexical matcher with the graph representing the ontology structure. Evidence theory is used to improve matching accuracy. In [32], CSR processes structural information via classification-based learning schemes. Instead of directly using structural information, S-Match and other systems employ ontology structural information indirectly through the use of ontology semantic information. Details are described in Section V.A.

B. CHALLENGES AND FUTURE WORK

Section IV.A showed that most existing methods are based on graph structure-based similarity to obtain a similarity value bounded in the range [0, 1], without analyzing the semantics of description logic axioms and constructors. Therefore, they are unable to solve heterogeneity in logic representation. For example, $A\perp B$ and $A\sqsubseteq +B$ express the same sematnics. This type of problem is typically solved via analysis of ontology semantic information as shown in Section V.

The structural information contains more effective information, so it is essential to analyze the structural information in the ontology matching process. In future work, we can try to learn from some related theories and algorithms in graph theory, because graphs can be used to represent structured objects. In addition, various similarity measures from different angles, or a combination of various similarity measures are worthy of in-depth study, so as to carry out more accurate and effective matching.

V. SEMANTIC INFORMATION ANALYSIS

After defining entity labels and comments, knowledge engineers use ontology constructors and axioms to impose further constraints on entities according to the human way of thinking. For example, consider Mother $\sqsubseteq$ Women $\sqcap\exists$ hasChild. Woman. It represents mother, and confines the concept to mothers having a female child. Therefore, the senses hidden in the axioms and constructors indicate the semantic information that knowledge engineers express while constructing the ontology. But due to differences in their way of thinking, knowledge engineers use different constructors and axioms to represent the same or similar entities. For example, both $A\sqsubseteq +B$ and $B\sqsubseteq +A$ represent $A\perp B = \varphi$. Compared to methods that compute entity similarity using structural information only, semantic information-based schemes provide a more effective approach to ontology heterogeneity.

A. ANALYSIS OF PRESENT SITUATION

Most existing semantic information analysis methods use semantic information indirectly via inferences from the reasoner. The two major ideas are described as follows:

1. Directly infer correspondence using the reasoner with inputting conclusions from the lexical analysis phase [16], [22]–[24], [34].
2. Compute entity similarity using the inference from the reasoner [15], [33].

Semantic analysis methods for ontology matching can be classified into the propositional satisfiability method and the DL reasoners according to deduction rules.

Parameters for the propositional satisfiability problem usually need the input of conjunctive normal form and then the entity relation is determined. The conjunctive normal form cannot effectively explore semantics in description logic elements (e.g., disjunction, and full existential restriction). As a result, conjunctive normal form-based schemes are primarily used in ontology matching processes that are represented by simple ontology language (e.g., S-Match [21]–[24]), and are seldom used in ontology represented by OWL-DL. DL reasoners use the tableau algorithms that can interpret disjunction and full existential restriction. Therefore, reason-based methods are capable of processing all types of syntactic elements in OWL-DL (e.g., ILIADS and Kosimap [33]).

In [15], ILIADS performs ontology matching by combining the similarity algorithm with the incremental logic reasoning algorithm. The algorithm determines entity similarity by the use frequency of senses in WordNet and labels potential correspondences. Finite ontology axioms are extracted through the heuristic algorithm and used for logic reasoning. Not all axioms from the ontology are used for reasoning, resulting in possible loss of semantics or incorrect reasoning. Like ILIADS, KOSIMap employs semantic information by using reasoners. It pre-processes heterogeneous ontologies and inputs them into the reasoner to obtain inferences. Entity similarity is computed using these interferences.

S-Match Provides Two Types of Representations: entity labels and entity concepts. Entity labels involve the senses of words in entity labels and represent the lexical information of labels. Entity concept is related to context and represents some logic relation. S-Match defines the concept of a node whose logic expression is computed as the intersection of the concepts of all labels from the root node to the node itself. CtxMatch [34] and S-Match submit expressions of conclusions obtained at the lexical analysis-based phase to the deciders, and define the inferences of the deciders as the correspondence between entities. ILIADS uses the inferences of the deciders as the foundation for computing semantic similarity.

The so-called semantic matching in ASMOV is actually based on translation of structural information. It still uses similarity to generate ontology matching results. The semantics-based method is only used to ensure that correspondences contain no semantic inconsistency.

In summary, ILIADS and KOSIMap use inferences of reasoners as factors in similarity computation. S-Match and CtxMatch infers correspondences by inputting the expression, obtained at the lexical analysis phase, into reasoners.

B. CHALLENGES

The two reasoner-based semantic information analysis methods described in Section.V.A outperform methods that only use structural information, but they only employ inferences of reasoners about formalized ontology semantics. Hence, formalized semantics contained in ontology constructors and axioms (e.g., $\sqsupseteq$, $\sqcap$, $\sqcup$, $\forall$) are not fully explored (e.g., $(\forall R \cdot C)^I = \left\{a \in \Delta^I \mid \forall b \cdot (a, b)\right\} \in R^I \rightarrow b \in C^I$). Consider examples of the following two axioms:

- Book $\sqsubseteq$ Reference, Reference $\sqsubseteq$ Publication;
- Book $\sqsubseteq$ Publication, Publication $\sqsubseteq$ Reference.

Directly applying reasoners will enable the two groups of axioms to reach the same inferences, i.e., Book $\sqsubseteq$ Publication. The two groups of axioms do not express identical semantic information. Therefore, instead of completely expressing the semantics hidden in the ontology, reasoner-based semantic analysis methods can only indicate inferences of these semantics.

C. FUTURE WORK

Insight into the challenges specified in Section V.B are given below.

To completely explore semantics implicated in ontology (see [35], [36] for definitions of formalized semantics) and directly manipulate the reasoning process, proper reasoning algorithms should be selected to analyze semantic information during the ontology matching process.

There are two types of DL reasoning algorithms [37], [42]: structure subsumption reasoning algorithms and tableau algorithms. The former algorithm is typically used for non-standard reasoning problems [35], while the latter algorithm is typically used by reasoners (e.g., Pellet and Racer). When confronting constructors or axioms in ontology, tableau algorithms add individuals according to certain rules and check whether the ontologies are consistent. Semantic information of an entity in ontology is not explicitly expressed. Structure subsumption reasoning algorithms involve two phases: (1) convert entity description to normal form and (2) compare syntactic structures between normal forms. Reducing entities in ontology to the normal forms will enable formalized semantics of the entity to be explicitly represented. The relation between entities can be obtained because comparing syntactic structures between normal forms is equivalent to comparing formalized semantics between two entities. Therefore, normal form techniques are needed to explicitly represent formalized semantics hidden in ontology. Normal form comparison techniques are needed to couple the matching process with the ontology formalized semantics reasoning.

Direct use of the DL structure subsumption reasoning algorithm will yield the following problems. Consider the three axioms:

- Monograph1$\sqsubseteq$Book1
- Monograph2$\sqsubseteq$Book2
- Book1 $\equiv$ Book2

Intuitively, Monograph1 and Monograph2 should possess equal relations. But from the DL perspective, even if Book1 $\equiv$ Book2, it cannot determine that there is any relation between them, because the sets they represent contain random sizes and positions in the sets represented by Book (Book1 or Book2); This is similar to the motion of two particles in Brownian movement. Therefore, when using the structure subsumption reasoning algorithms to analyze an entity's semantic information, one must introduce the entity's lexical information to constrain the entity's formalized semantics such that the sets represented by Monograph1 and Monograph2 can still have random sizes and positions in the sets represented by Book. However, they share the same direction and size.

In summary, in addition to manipulating the reasoning process, structure subsumption reasoning algorithms can explicitly represent formalized semantics hidden in ontology.

This can offset flaws in reasoner-based semantic information analysis.

VI. AGGREGATION OF INFORMATION ANALYSIS RESULTS

Upon the analysis of ontology information, different ontology matching schemes will provide the final correspondence by aggregating analysis results in various ways.

A. ANALYSIS OF PRESENT SITUATION

Lily, ILIADS, ASMOV, RiMOM and Falcon-AO [38] have one thing in common: they all obtain correspondence between entities by computing weighted sums of different information analysis results (e.g., similarity) and comparing them with specified thresholds. Aggregation of information analysis results from the abovementioned ontology matching methods can be illustrated by the following equation:

$$W_1 S_1 + \ldots + W_i S_i + \ldots + W_n S_n \tag{4}$$

where w_i is the specified weight and S_i is the similarity of information analysis. Matching methods derive final correspondence by comparing results from this equation with the specified threshold μ. By using this method of aggregation, information analysis results are independent. The difference between these solutions is the method used for setting weights and thresholds. For example, RiMOM sets wi via the dynamic multi-strategy method, ILIADS sets μ via the heuristic algorithm, FOAM [39], [40] specifies wi via the ordered weighted average method, and ASMOV sets wi and μ empirically. In [41], the discrete weighting method is used to aggregate lexical information and semantic information analysis results. In the following, RiMOM and Lily are taken as an example to illustrate the differences between them.

The basic idea of RiMOM is that different matching schemes use a type or many types of information and thus each matching scheme has pros and cons, but a scheme alone is insufficient to effectively solve the problem. Therefore, it is possible to achieve better results by combining different matching schemes. In RiMOM, the dynamic multi-strategy method is used for weight selection. Various techniques are gamed through the use of parameters and aggregated to provide desired results. A system that uses similar methods is also described in [40]. Lily primarily uses the semantic description-based text matcher and the similarity dissemination matcher. The semantic description-based text delivers reliable similarity when rich texts are present in ontology. When the evaluated similarity is greater than θ (the set value being 0.65), aggregated similarity is the similarity that describes the text, without taking into account the results obtained after similarity dissemination. When the similarity produced by text matching is smaller than θ, the average of the two matching methods is defined as the final similarity. Note that instead of comparing aggregated similarity with the specified threshold, Lily selects the final correspondence using the greedy method.

In [21]–[24], [34], expressions of lexical information or structural information are directly submitted to the reasoner

TABLE 1. Notation and Definition.

Notation	Definition
O^1, O^2	the pair of ontologies to be matched
syn(l)	the synonyms set of l
Lin(l, l′)	the information theory
tok(l)	the set of words corresponding to the entity label
w_{name}	the weights based on entity label similarity between entities
$sim(e_1, e_2)$	lexical similarity between e_1 and e_2

and the inference from the reasoner is taken as the correspondence between entities. This type of solution was described in Section V.A.

In addition to the two abovementioned techniques (i.e., weighted sum and reasoner-based reasoning), there are other schemes for aggregating information analysis results. In [29], AROMA uses two important conditions that allow the algorithm to access implication values and generate matching rules. Correspondence between entities is selected according to implication intensity. By analyzing ontology information through machine learning techniques, an automatic ontology matching method was proposed in GLUE and [20]. A machine learning classifier is employed to determine whether individuals of concept B in ontology O2 are individuals of concept A in ontology O1. Statistical analysis on distributions is conducted to generate a similarity matrix. Entity correspondence is obtained by compulsory relaxation method.

B. CHALLENGES

As discussed in Section VI.A, the basic idea of some ontology matching methods is to compute similarities for different types of information, compute the weighted sum of the similarities, and then to compare the results with the specified threshold to obtain the final correspondence between entities. Other schemes enter expressions of lexical information or structural information into the reasoner and define the inference of the reasoner as entity correspondence. However, these schemes fail to define formalized semantics of the information analysis results, as well as the relation between the final correspondence and ontology semantics. Therefore, it is not properly interpreted in the aggregation methods, including the interpretation of the domain the domain Δ^I and the function $(\cdot)^I$ (refer to [35], [36] for a definition of the interpretation).

C. FUTURE WORK

The challenges described in Section VI.B involve the approach to aggregation of ontology matching information analysis results and the basic ideas of the entire matching process. Therefore, by combining Sections III.C, IV.B, and V.C, we gain certain insights such that we can use DL theory to define formalized semantics (or interpretations) of

TABLE 2. A summary of ontology matching solutions.

Categories	Present situation	Challenges	Future work
Analysis of Lexical and External Information	1.LiLy [4]; 2.MapPSO [14]; 3.UFOme; 4.ILIADS [15]; 5.ASMOV [16]	1.Senses combination; 2.Senses fuzziness; 3.Difficulty in detecting the right senses	1.Derive proper word senses; 2.Extend entity label's definition
Structural Information Analysis	1.Similarity flooding [26], [42]; 2.Graph matching [20]; 3.Circuit model [4]; 4.Bayesian decision theory [27]; 5.Markov network [28]; 6.Association rule paradigm [29]	1.Analyze the semantics of description logic axioms and constructors; 2.Solve heterogeneity in logic representation.	1.Various similarity measures from different angles; 2.Combination of various similarity measures.
Semantic Information Analysis	1.Infer correspondence using the reasoner [16], [22]-[24], [34]; 2.Compute entity similarity using the inference from the reasoner [15], [33]	1.Formalized semantics contained in ontology constructors and axioms are not fully explored.	1.Manipulating the reasoning process; 2.Represent formalized semantics hidden in ontology;
Aggregation of Information Analysis Results	1.Weighted sum [38] 2.Reasoner-based reasoning [29]	1.Define formalized semantics of the information analysis results.	1.Use DL theory to define formalized semantics of information analysis results.

information analysis results. First, define formalized semantics of entity labels according to the open world described by WordNet ontology. Next, use structural subsumption reasoning algorithms to process semantic information. This process can be defined by DL theory. Formalized semantics of the correspondence (e.g., $\sqsubseteq$, $\sqsupseteq$, $\perp$, $\equiv$) inferred in this way are correlated with the ontology.

The notations used in the paper are collated in TABLE 1. The methods discussed in this manuscript are summarized in TABLE 2.

VII. CONCLUSION

This paper summarizes and compares ontology matching solutions that use the same type of information, and analyzes the challenges in different types of information. In addition to analyzing the same type of information and different types of information, ontology matching still faces the following problems. Firstly, under the condition of limited resources, we match large ontology by loading comprehensive information. Secondly, we use external resources to supplement structural and semantic information to improve system performance. Finally, we use internal and external information in the ontology to achieve information combination.

Many ontology matching methods have been proposed in the decade of ontology matching development. This paper provides a deeper understanding of the methodological differences in the field of ontology matching. Based on the results of the analysis, the current challenges and problems in the field of ontology matching are also provided, which are the foundation of the research on ontology matching. In future work, we will continue to conduct in-depth research on the ontology matching scheme, and optimize the ontology matching scheme.

REFERENCES

[1] X. Liu, A. Bouguettaya, J. Wu, and L. Zhou, "Ev-LCS: A system for the evolution of long-term composed services," *IEEE Trans. Services Comput.*, vol. 6, no. 1, pp. 102–115, 2013.

[2] J.-W. Jeong, H.-K. Hong, and D.-H. Lee, "Ontology-based automatic video annotation technique in smart TV environment," *IEEE Trans. Consum. Electron.*, vol. 57, no. 4, pp. 1830–1836, Nov. 2011.

[3] A. K. Akanbi and M. Masinde, "Towards semantic integration of heterogeneous sensor data with indigenous knowledge for drought forecasting," in *Proc. Doctoral Symp. 16th Int. Middleware Conf.*, Dec. 2015, pp. 1–5.

[4] L. Zhu, G. Hua, and W. Gao, "Mapping ontology vertices to a line using hypergraph framework," *Int. J. Cognit. Comput. Eng.*, vol. 1, pp. 1–8, Jun. 2020.

[5] D. Bonino, E. Castellina, and F. Corno, "The DOG gateway: Enabling ontology-based intelligent domotic environments," *IEEE Trans. Consum. Electron.*, vol. 54, no. 4, pp. 1656–1664, Nov. 2008.

[6] J. Gracia and E. Mena, "Semantic heterogeneity issues on the Web," *IEEE Internet Comput.*, vol. 16, no. 5, pp. 60–67, Sep. 2012.

[7] X. Su and J. A. Gulla, "Semantic enrichment for ontology mapping," in *Proc. Int. Conf. Appl. Natural Lang. Inf. Syst.*, 2004, pp. 217–228.

[8] N. Choi, I.-Y. Song, and H. Han, "A survey on ontology mapping," *ACM SIGMOD Rec.*, vol. 35, no. 3, pp. 34–41, Sep. 2006.

[9] X. Zhang and M. Li, "A survey on semantic sensor Web: Sensor ontology, mapping and query," *Int. J. u- e-Service, Sci. Technol.*, vol. 8, no. 10, pp. 325–342, Oct. 2015.

[10] P. Shvaiko and J. Euzenat, "Ontology matching: State of the art and future challenges," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 1, pp. 158–176, Jan. 2013.

[11] P. Pantel and D. Lin, "Discovering word senses from text," in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2002, pp. 613–619.

[12] G. Pirró and D. Talia, "UFOme: An ontology mapping system with strategy prediction capabilities," *Data Knowl. Eng.*, vol. 69, no. 5, pp. 444–471, May 2010.

[13] S. Duan, A. Fokoue, and K. Srinivas, "One size does not fit all: Customizing ontology alignment using user feedback," in *Proc. Int. Semantic Web Conf.*, 2010, pp. 177–192.

[14] M. Maree and M. Belkhatir, "Addressing semantic heterogeneity through multiple knowledge base assisted merging of domain-specific ontologies," *Knowl.-Based Syst.*, vol. 73, pp. 199–211, Jan. 2015.

[15] P. Maio and N. Silva, "An extensible argument-based ontology matching negotiation approach," *Sci. Comput. Program.*, vol. 95, pp. 3–25, Dec. 2014.
[16] C. E. Yap and M. H. Kim, "Instance-based ontology matching with rough set features selection," in *Proc. Int. Conf. IT Converg. Secur. (ICITCS)*, Dec. 2013, pp. 1–4.
[17] M. Paolucci, T. Kawamura, T. R. Payne, and K. P. Sycara, "Semantic matching of Web services capabilities," in *Proc. Int. Semantic Web Conf.*, May 2002, pp. 333–347.
[18] J. Li, J. Tang, Y. Li, and Q. Luo, "RiMOM: A dynamic multistrategy ontology alignment framework," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 8, pp. 1218–1232, Aug. 2009.
[19] J. Bock and J. Hettenhausen, "Discrete particle swarm optimisation for ontology alignment," *Inf. Sci.*, vol. 192, pp. 152–173, Jun. 2012.
[20] X. Xue and Y. Wang, "Optimizing ontology alignments through a memetic algorithm using both matchfmeasure and unanimous improvement ratio," *Artif. Intell.*, vol. 223, pp. 65–81, Jun. 2015.
[21] M. M. Hussain and S. K. Srivatsa, "A study of different ontology matching system," *Int. J. Comput. Appl. Technol.*, vol. 37, no. 12, pp. 10–16, Jan. 2012.
[22] C. Trojahn, P. Quaresma, and R. Vieira, "Exploiting majority acceptable arguments for ontology matching," *Int. J. Artif. Intell. Educ.*, vol. 8, no. 12, pp. 1–9, 2012.
[23] F. Giunchiglia, A. Autayeu, and J. Pane, "S-match: An open source framework for matching lightweight ontologies," *Semantic Web*, vol. 3, no. 3, pp. 307–317, 2012.
[24] K. Hussein and J. Omer, "A framework for improving the performance of ontology matching techniques in semantic Web," *Int. J. Adv. Comput. Sci. Appl.*, vol. 3, no. 1, 2012.
[25] M. Vargas-Vera and M. Nagy, "Establishing agent trust for contradictory evidence by means of fuzzy voting model: An ontology mapping case study," *Comput. Hum. Behav.*, vol. 30, pp. 745–752, Jan. 2014.
[26] N. Jian, H. Wei, C. Gong, and Y. Qu, "Falcon-AO: Aligning ontologies with falcon," in *Proc. Workshop Integrating Ontologies*, Jan. 2005, pp. 85–91.
[27] C. Esposito, M. Ficco, F. Palmieri, and A. Castiglione, "A knowledge-based platform for big data analytics based on publish/subscribe services and stream processing," *Knowl.-Based Syst.*, vol. 79, pp. 3–17, May 2015.
[28] L. Nanni, Y. M. G. Costa, A. Lumini, M. Y. Kim, and S. R. Baek, "Combining visual and acoustic features for music genre classification," *Expert Syst. Appl.*, vol. 45, pp. 108–117, Mar. 2016.
[29] E. Kaufmann and A. Bernstein, "Evaluating the usability of natural language query languages and interfaces to semantic Web knowledge bases," *J. Web Semantics*, vol. 8, no. 4, pp. 377–393, Nov. 2010.
[30] P. Ochieng and S. Kyanda, "Large-scale ontology matching: State-of-the-art analysis," *ACM Comput. Surveys*, vol. 51, no. 4, pp. 1–35, Sep. 2018.
[31] V. Spiliopoulos, G. A. Vouros, and V. Karkaletsis, "On the discovery of subsumption relations for the alignment of ontologies," *J. Web Semantics*, vol. 8, no. 1, pp. 69–88, Mar. 2010.
[32] X. Liu, B. Cheng, J. Liao, P. Barnaghi, L. Wan, and J. Wang, "OMI-DL: An ontology matching framework," *IEEE Trans. Services Comput.*, vol. 9, no. 4, pp. 580–593, Jul. 2016.
[33] M. Zand, S. Doraisamy, A. Abdul Halin, and M. R. Mustaffa, "Ontology-based semantic image segmentation using mixture models and multiple CRFs," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3233–3248, Jul. 2016.
[34] X. Hu, Z. Feng, S. Chen, K. Huang, J. Li, and M. Zhou, "Accurate identification of ontology alignments at different granularity levels," *IEEE Access*, vol. 5, pp. 105–120, 2017.
[35] T. T. A. Nguyen and S. Conrad, "Ontology matching using multiple similarity measures," in *Proc. 7th Int. Joint Conf. Knowl. Discovery, Knowl. Eng. Knowl. Manage.*, 2015, pp. 603–611.
[36] A. Vennesland, "Matcher composition for identification of subsumption relations in ontology matching," in *Proc. Int. Conf. Web Intell.*, Aug. 2017, pp. 154–161.
[37] X. S. Xue, and J. S. Wang, "An ontology matching technology usingdually individual annotation," *J. Fuzhou Univ. (Natural Sci. Ed.)*, vol. 44, no. 1, pp. 64–70, 2016.
[38] L. Otero-Cerdeira, F. J. Rodríguez-Martínez, and A. Gómez-Rodríguez, "Ontology matching: A literature review," *Expert Syst. Appl.*, vol. 42, no. 2, pp. 949–971, Feb. 2015.
[39] X. Xue and J. Liu, "Collaborative ontology matching based on compact interactive evolutionary algorithm," *Knowl.-Based Syst.*, vol. 137, pp. 94–103, Dec. 2017.
[40] M. Al Boni, D. T. Anderson, and R. L. King, "Hybrid measure of agreement and expertise for ontology matching in lieu of a reference ontology," *Int. J. Intell. Syst.*, vol. 31, no. 5, pp. 502–525, May 2016.
[41] T. Kiren and M. Shoaib, "A novel ontology matching approach using key concepts," *Aslib J. Inf. Manage.*, vol. 68, no. 1, pp. 99–111, Jan. 2016.
[42] K. Sun, Y. Zhu, and J. Song, "Progress and challenges on entity alignment of geographic knowledge bases," *ISPRS Int. J. Geo-Inf.*, vol. 8, no. 2, p. 77, Feb. 2019.
[43] S. Karampatakis, C. Bratsas, O. Zamazal, P. Filippidis, and I. Antoniou, "Alignment: A hybrid, interactive and collaborative ontology and entity matching service," *Information*, vol. 9, no. 11, p. 281, Nov. 2018.
[44] X. Liu, X. Hou, J. Yu, Y. Gao, Y. Zhang, and Y. Zhang, "Accessing alignments of ontologies via IoT based on SKOS data model," *China Commun.*, vol. 14, no. 6, pp. 137–155, 2017.
[45] M. T. Dhouib, F. Z. Catherine, and A. G. B. Tettamanzi, "An ontology alignment approach combining word embedding and the radius measure," in *Proc. Int. Conf. Semantic Syst.* Cham, Switzerland: Springer, 2019, pp. 191–197.
[46] A. A. Amer and H. I. Abdalla, "A set theory based similarity measure for text clustering and classification," *J. Big Data*, vol. 7, no. 1, pp. 1–43, Sep. 2020.

XIULEI LIU received the Ph.D. degree in computer science from the Beijing University of Posts and Telecommunications, in March 2013. He was a Visiting Ph.D. Student with the CCSR, University of Surrey, from October 2008 to October 2010. Since January 2017, he has been an Associate Professor with the Computer School, Beijing Information Science and Technology University, China. His research interests include semantic sensor, semantic web, knowledge graph, and semantic information retrieval.

QIANG TONG received the Ph.D. degree from the Department of Computer Science and Technology, Tsinghua University, in 2012. Since August 2018, he has been a Lecturer with the Computer School, Beijing Information Science and Technology University, China. His research interests include image recognition, computer vision, and machine learning.

XUHONG LIU received the Ph.D. degree in computer science from the Beijing Institute of Technology, in 2005. She is currently an Associate Professor with the Beijing Information Science and Technology University, China. Her research interests include document information processing, machine learning, and natural language processing.

ZHIHUI QIN is currently pursuing the master's degree with the Computer School, Beijing Information Science and Technology University, China. Her main research interests include machine learning and knowledge graph.