Asian Journal of Information Technology 15 (4): 676-688, 2016
ISSN: 1682-3915

Technical Review on Ontology Mapping Techniques

Kaladevi Ramar and Geetha Gurunathan
Jerusalem College of Engineering, Anna University, 600100 Chennai, India

Abstract: Due to an increased awareness of potential ontology applications in industry, public administration and academia, a growing number of ontologies are created by different organizations and individuals. Furthermore, ontology users or developers not only use their own ontologies but also want to integrate or adapt other ontologies or even apply multiple ontologies to solve a problem. Ontology mapping is the solution for finding semantic correspondences between similar elements of different ontologies. This correspondence is required in fields like data integration, information transformation and information retrieval. Even though, a lot of mapping techniques are developed, the objectives are not fully achieved. This review study aims to discuss various categories of ontology mapping techniques, the mapping strategies, challenges and further research requirements.

Key words: Interoperability, integration, ontology mapping, semantic, heterogeneity

INTRODUCTION

Increase in web and distributed computing leads access to a large number of independently created and managed data sources of broad variety due to the design and development autonomy of systems. For the effective utilization of the heterogeneous data, various processing techniques are required to resolve the information heterogeneity. And, also to enrich data integration, in order to provide a uniformed access for various agencies, industries and research institutions for better decision making. Information heterogeneity comprises of syntactic, structural and semantic heterogeneities (Stuckenschmidt and Harmelen, 2005; Van Rijsbergen, 1979). Advanced solutions for syntactic and structural heterogeneity are available (Wiederhold, 1992).

There is a need for more sophisticated techniques to solve the semantic heterogeneity. Various approaches viz. synonym sets, term networks, concept lattices, features and constraints have been suggested as solutions for solving semantic heterogeneity (Stuckenschmidt and Harmelen, 2005). But, the required contextual information is implicit that was not handled by the above solutions. To overcome this issue, the use of an explicit context model is needed.

Ontologies are the main component of semantic web to solve the semantic heterogeneity problem. Ontology is a formal, explicit specification of a shared conceptualization (Gruber, 1993). Ontology mapping provides semantic correspondence between ontologies which in turn provides solution to the semantic heterogeneity problem. Ontology mapping is the determination of semantic correspondences between similar elements in different ontologies (Noy, 2004; Doan and Halevy, 2005). Semantic correspondence refers to different relationships, i.e., the equivalence (=), the broader (⊇), the narrower (⊆), etc. and elements could be classes, properties, instances and relations between the instances of two ontologies. So, ontology mapping is an essential technique to establish interoperability between systems or services using ontologies. These mapping can be used for various tasks like ontology evolution, ontology integration, data integration, data warehouses, peer-to peer information sharing, web service composition, search and query answering.

Previously various solutions have been proposed for achieving semantic interoperability using ontology mapping. The techniques could be fully automatic (Li *et al*., 2009) or semi-automatic (Noy and Musen, 2000) and centralized (Niles and Pease, 2001) or decentralized. Also, the solution varies based on the techniques used. Typical decentralized techniques include machine learning techniques (Doan *et al*., 2003), graph based techniques (Melnik *et al*., 2002), heuristic and rule based techniques (Ehrig and Staab, 2002), string based techniques (Do and Rahm, 2002), reasoning and theorem proving techniques (Giunchiglia *et al*., 2007) and Probabilistic techniques (Mitra *et al*., 2004).

Several surveys (Shvaiko and Euzenat, 2013; Amrouch and Mostefai, 2012; Zhu, 2012) are made in ontology mapping to analyse the pros and cons of available techniques and the emergence of this field. The existing evaluations show that the field of ontology mapping has made a measurable improvement. However

Corresponding Author: Kaladevi Ramar, Jerusalem College of Engineering, Anna University, 600100 Chennai, India

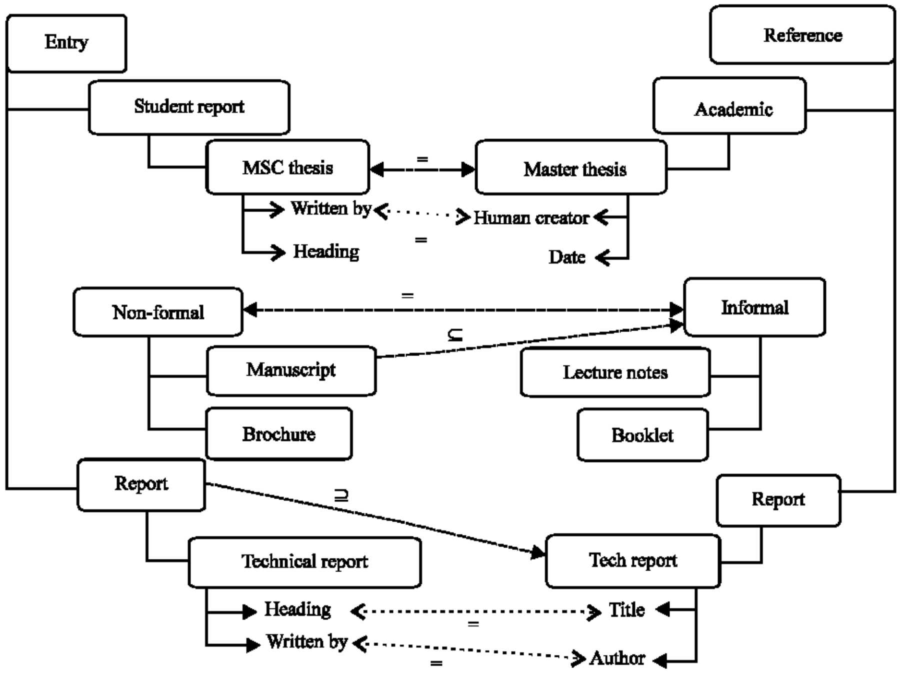

Fig. 1: Ontology mapping scenario

in the year 2014, Ontology Alignment Evaluation Initiative (OAEI) campaign has suffered with significant decreasing participating systems (14 vs. 23 in 2013 OAEI) and in the year 2015, the number is increased to 22. It shows there is an ups and downs of progress in this research area. To make progression of this area and to enhance the existing mapping systems, the state of the art systems need to be analysed systematically. In order to achieve optimum results based on quality and efficiency and also to integrate the mapped ontologies with minimum or no loss, actions have to be taken. In this review, we are trying to reconcile the lack in progress by identifying the challenges such as performance measures of matching techniques, aggregation of matching techniques, testing with real time test cases, different semantic heterogeneity handling issues, exploitation of background knowledge, reusing of mapped ontologies. We are suggesting possible research direction based on the above challenges.

Various ontology mapping techniques: Ontology mapping is used to "Establish the correspondences among ontologies and to determine the set of overlapping concepts, concepts that are similar in meaning but have different names or structure" (Noy and Musen, 2000). Ontology mapping is concerned with reusing existing ontologies, expanding and combining them by some means and enable a larger pool of information and knowledge in similar domains to be integrated to support new communication and use.

The cardinality of the computed alignment varies from simple mapping (1:1 mapping) to complex mappings (1:n, n:1 and m:n). The correspondence between ontology is defined by 4 tuple, i.e, <‘id’, ’e_1’, ‘e_2’, ‘r’>, ‘id’ is an identifier for the given correspondence; ‘e_1’ and ‘e_2’ are entities, e.g., classes and properties of the first and the second ontology respectively, ‘r’ is a relation, e.g., equivalence (=), more general (⊇), less general (⊆), overlap (n), disjointness (∩) holding between ‘e_1’ and ‘e_2’. If the relation is equivalence then correspondence holds a confidence measure which is a number typically ranged in between [0, .., 1].

A simple mapping scenario is shown in Fig. 1. For instance, a researcher may want to merge his bibliographic information with similar research groups. The O_1 and O_2 are two bibliographic ontologies with distinct representation. Classes and sub classes are shown in rounded rectangle. Attributes are shown with their corresponding classes. The classes (‘MSC Thesis’, ‘Master Thesis’), (‘Non-Formal’, ‘Informal’) of O_1 and O_2, respectively are semantically similar and are mapped using equivalent (=) relation. The class ‘Report‘ of O_1 is more general than ‘TechReport’ of O_2. Similarly the class ‘Manuscript’ of O_1 is less general than ‘Informal’ of O_2. Likewise, many real world cases also demonstrate the need for ontology mapping to support schema/data integration.

Ontology mapping techniques are based on centralized (external resources) or decentralized (direct) architectures. The centralized approach uses external background knowledge from different sources. The external knowledge can be WordNet, upper ontology, domain specific corpora, Linked Open Data (LOD), Wikipedia, etc. Upper ontology is called reference ontology. Normally, ontology to reference mapping or ontology to ontology mapping using reference ontology has been done in centralized approach. For example, DOLCE (Gangemi and Guarino, 2003) is an upper ontology developed in the WonderWeb project. It aims at providing a common reference framework for ontologies to facilitate information sharing among them. The DOLCE captures ontological categories based on natural language and human common sense in its representation. SUMO (Suggested Upper Merged Ontology) was created by the IEEE standard upper ontology working group with the goal of developing a standard upper ontology to promote data interoperability, information search and retrieval, automated inference and natural language processing.

The decentralized approach creates mappings by exploiting different kinds of information between ontologies. For example, structural information (e.g., subclass and superclass relationships, domain and range of properties, instance of classes and graph structure of ontologies) can give insight into ontologies, lexical information (e.g., names, definitions and distance between strings) can find and rank mapping results, additional information (e.g., WordNet) provides semantics for the entities in ontologies, instance information is used to find the similarity of classes and it is useful for learning based approaches to provide necessary training sets. The main types of ontology mapping approaches are structure based methods terminological based methods, instance based methods, semantic reasoning methods, hybrid methods and ontology mapping using background knowledge. Each of these categories can be subjected to separate analysis but they are not entirely independent of each other. The following section reviews each category of technique with example.

MATERIALS AND METHODS

Structure based methods: Structure based techniques is based on the underlying structure of input ontologies such as ISA hierarchy, sibling concepts, relation between graph nodes. In ISA hierarchy, the hypothesis is if the direct super-concepts and/or the direct sub-concepts of two concepts are similar then the two compared concepts may also be similar. In metrological hierarchies, it connect the entities through part-whole relations, i.e., an entity lower in the hierarchy is part of any entity higher in the hierarchy to which it relates. Taxonomic hierarchies connect the entities through a parent-child relationship. If sibling concepts of two concepts are similar then two compared concepts may also be similar. The hypothesis is that if the relations and related classes are similar, the two compared concepts may be also similar. In structured context, input ontologies are considered as labelled graphs. The similarity comparison between a pair of nodes from the two ontologies is based on the analysis of their positions within the graphs. The intuition behind this is that if two nodes from two ontologies are similar, their neighbours must also be similar. Three structure based techniques namely, Similarity Flooding, Anchor-PROMPT and Anchor-Flood are discussed here.

Similarity Flooding (SF) is an automatic generic ontology matching based on the idea of fixpoint computation to determine corresponding nodes in the graphs (Melnik *et al*., 2002). Schemas are represented as directed labelled graphs, grounding on the open information model specification. The algorithm manipulates them in an iterative fixed-point computation to produce an alignment between the nodes of the input graphs. The technique starts from string-based comparison, such as common prefix suffix tests of the vertices labels, to obtain an initial alignment which is refined within the fixed-point computation. The principle of the similarity flooding algorithm is that the similarity between two nodes depends on the similarity between their adjacent nodes. From iteration to iteration the spreading depth and the similarity measure increases till the fixed-point is reached. The result of this step is a refined alignment which is further filtered to finalize the matching process. Though the Similarity Flooding algorithm can be applied to 1-n mapping, it obtains this feature by decreasing the threshold of similarity. This method produces not a real 1-n mapping and works only for direct labelled graphs. The algorithm is based on the assumption that adjacency contributes to similarity propagation.

Anchor-PROMPT is an extension of PROMPT (Noy and Musen, 2000) algorithm which is an ontology mapping and merging system (Noy and Musen, 2001). It takes ontologies as directed labelled graph, concepts as nodes and relations as arcs. This algorithm is based on the assumption that if two pairs of terms are similar and there are paths connecting them, then the elements in these paths are often similar as well. Anchor-PROMPT has two limitations. One is that it is time consuming. Another is Anchor-PROMPT does not perform well when one ontology has a deep hierarchy with many inter-linked classes and the other ontology has a shallow hierarchy with few levels.

Anchor-Flood is an ontology schema matching algorithm that takes the essence of the locality of reference by considering neighbouring concepts and relations to align the entities of ontologies. It starts off a seed point called an anchor. Then small blocks of concepts and related properties are dynamically collected as neighbourhood information from the anchor points. Local alignment process will align the small blocks using lexical and structural information. Aligned pairs are considered the new anchors. This process is repeated until all of the anchors are processed. The strength of Anchor-Flood is that it reduces the number of comparisons between entities by only matching small blocks not whole ontologies which increases efficiency.

Terminological based methods: Terminological methods compare strings. These methods can be applied to the name, the label or the comments of the ontology entities to find its similarity. Before comparison the strings needs to be normalized. Numerous normalization procedures are available such as elimination of multiple blank spaces in between words, punctuation elimination, case standardization, digit suppression and stop-word elimination. They are typically based on the assumption that the more similar the strings, the more likely they are to denote the same concepts. Various string matching techniques are available from distance based functions to token-based functions. Examples of distance based functions which are extensively used in matching systems are prefix, suffix, n-gram and edit distances. Examples of token based functions are Jaccard similarity, TF/IDF, Cosine similarity and Jensen-Shannon distance. The COMA, OLA and I-Sub ontology mapping techniques are discussed for this category.

COMA is an automatic ontology matching tool based on the composition of several matchers (Do and Rahm, 2002; Aumueller *et al*., 2005). The objective of this method is to identify semantic correspondences between metadata structures or models such as database schemas, XML message formats and ontologies. It provides a user interface to easily upload input and obtain results. The results can be evaluated with human edited matching results (also called golden standard). The COMA can construct new matchers and match strategies from its flexible combination of matchers. Most of the matchers are string-based, such as affix, n-gram and edit distance. Schemas are internally encoded as directed acyclic graphs where elements are the paths. This aims at capturing contexts in which the elements occur. It presumes interaction with users who approve obtained matches and mismatches to gradually refine and improve the accuracy of a match. The COMA++ is built on top of COMA by elaborating in more detail the alignment reuses operations and provides a more efficient implementation of the COMA algorithms and a graphical user interface. For domain specific entities its performance is lowered.

OLA (OWL Lite Aligner) is designed to use each component of ontology, i.e., classes, properties, names, constraints, taxonomy and even instances (Euzenat and Valtchev, 2004). The algorithm utilizes all possible terminological, structural and extensional characteristics of ontologies. Similarity between labels can be produced by terminological methods (e.g., string distance, linguistic evaluation). Similarity between data values and data types can be provided by specialized external similarity measures such as Euclidean distance, symmetric difference distance. The OLA uses a family of distance based algorithms which converts definitions of distances based on all the input structures into a set of equations. Then distances are linearly aggregated. As a system, OLA considers the alignment as a solution to a clearly stated optimization problem. This solution is not the global optimum so the algorithm has to be launched several times.

I-Sub compares the similarity of strings by considering their similarity along with their differences (Stoilos *et al*., 2005). The function of similarity is motivated by the substring string metric. This substring metric computes the biggest common substring between two strings. This computation process is further extended by removing the common substring and by repeating again for the next biggest substring until no one can be identified. The difference function is based on the length of the unmatched strings have resulted from the initial similarity matching step. Moreover, difference function is less important on the computation of the overall similarity. The difference function is derived from Hamacher product (Hamacher *et al*., 1978) which is a parametric triangular norm.

Instance based methods: Instance based ontology mapping techniques determine the similarity between concepts of different ontologies by examining the extensional information of concepts that is the instance data. The basic idea of instance-based mapping is that the more significant the overlap of common instances of two concepts is the more related these concepts are. The most frequently used measures to find similarity between classes are Jaccard similarity, K-statistic, cosine similarity based on TF/IDF. Machine learning methods are also used to create mappings between ontologies. The efficiency of the method is based on instances available in ontologies and it works better if more instances are available. Machine learning has separated into two phases. The first phase is the learning or training phase.

In the second phase, the learnt matcher is used for matching new ontologies. There are several well-known machine learning methods used in ontology matching such as naive bayesian learning, neural networks, decision trees and support vector machines. Also, clustering methods finds similar concepts by grouping clusters of similar instance. The GLUE and LSD methods are discussed for this category.

The LSD (Learning Source Description) is a semi-automatic multistrategy learning method used to create semantic mappings (Doan *et al*., 2001). It employs multiple learner modules which contains base learners and the meta-learner. Each module exploits a different type of information in the input schemas or data. The base learners used are: the name learner: it matches an XML element using its tag name, the content learner: it matches an XML element using its data value and works well on textual elements, Naive Bayes learner: it exploits the data value of the instance and the XML learner: it uses the hierarchical structure of input instances. Multi-strategy learning has two phases: training and matching. In the training phase, a small set of data sources has been manually mapped to the mediated schema and is utilized to train the base learners and the meta learner. In the matching phase, the trained learners predict mappings for new sources and match the schema of the new input source to the mediated schema. The LSD also examines domain integrity constraints, user feedback and nested structures in XML data for improving matching accuracy. The LSD proposes semantic mappings with a high degree of accuracy by using the multi-strategy learning approach.

The GLUE is a semi-automatic technique which uses instances for ontology mapping (Doan *et al*., 2003). The basic architecture of GLUE consists of three main modules: distribution estimator, similarity estimator and relaxation labeller. The distribution estimator takes as input two taxonomies O_1 and O_2, together with their data instances. Then, it applies machine learning techniques to compute for every pair of concepts and their joint probability distributions using Jaccard similarity measure. The distribution estimator uses multiple learners, i.e., base learners and a meta-learner. Next, GLUE feeds the outcome of learners into the similarity estimator which applies a user-supplied similarity function to compute a similarity value for each pair of concepts. The output from this module is a similarity matrix between the concepts in the two taxonomies. The relaxation labeller takes the similarity matrix, together with domain-specific constraints and heuristic knowledge and searches for the mapping configuration that best satisfies the domain constraints and the common knowledge, taking into account the observed similarities. The major problem of GLUE is that it requires a large number of instances associated with the nodes in taxonomies and these are not available in most ontology mapping cases. In general learning based methods are time consuming and accuracy of mapping is based on the size of the instance data.

Semantic reasoning methods: Semantic based algorithms handle the input based on its semantic interpretation, e.g., model-theoretic semantics. The intuition is that if two entities are the same, then they share the same interpretations. Thus, they are well grounded deductive methods. Examples are propositional satisfiability and description logics reasoning techniques. Since, semantics is usually given in a structure and hence, there are no element-level semantic techniques. Semantic relations are in the set, i.e., disjointness, equivalence, more specific and less specific. The CTXMATCH and S-match are discussed for this category.

The CTXMATCH is an algorithm for discovering semantic mappings across Hierarchical Classifications (HCs) using logical deduction (Bouquet *et al*., 2003). CTXMATCH takes two inputs H and H1 in HCs and for each pair of concepts $k \in H$, $k1 \in H1$ (a node with relevant knowledge including meaning in HC), returns their semantic relation ($\supseteq$, $\subseteq$, $\equiv$, * and $\perp$). For example, k is more general than k1 ($k \supseteq k1$), k is less general than k1 ($k \subseteq k1$), k is equivalent to k1 ($k = k1$), k is compatible with k1 ($k{*}k1$) and k is incompatible with k1 ($k \perp k1$). The contribution of the CTXMTCH is that mappings can be assigned a clearly defined model theoretic semantics and that structural, lexical and domain knowledge are considered. CTXMATCH is performed well when less data are available.

The S-Match is a schema and ontology mapping system that uses reasoning and theorem proving methods to find mappings based on CTXMATCH (Giunchiglia and Shvaiko, 2004). It uses a combination of matchers using lexical information and external resources. Then, it uses a SAT solver to find semantic relations such as equivalence (=), more general ($\supseteq$), less general ($\subseteq$), mismatch ($\neq$), union ($\cup$) and overlapping ($\cap$). In the S-match platform, the system takes input schemas in a standard internal XML format, does the pre-processing and returns enriched trees containing concepts at labels and concepts at nodes as output. These enriched trees are stored in an internal database namely PTrees where they can be browsed, edited and manipulated. The pre-processing q module has access to the set of oracles which provides the necessary a priori lexical and domain knowledge. Currently, the only oracle that S-match has is WordNet. The matching process is coordinated by match manager using three extensible libraries, the weak semantic matcher, the oracle and the SAT solver. The weak semantic matchers perform string manipulations (i.e., prefix, suffix, edit distance, soundex, etc.) and try to find the semantic relations implicitly encoded in similar words. Oracle is a strong semantic matcher which extracts semantic relations existing between concept labels which hold needed lexical and domain knowledge. The SAT solvers are a structure level strong semantic matcher to decide propositional satisfiability. The S-match runs the slowest because it needs to translate a mapping problem into a validity problem which is very time consuming and depends on the structure of the input schema.

Hybrid methods: Ontology mapping methods which combines different matchers (e.g., lexical, structural) into a single algorithm to solve ontology mapping problem are called hybrid methods. Systems which participated in OAEI campaign more than two times in last 6 year (2010-2015) are discussed here.

The RiMOM is a general ontology mapping system based on Bayesian decision theory (Li *et al*., 2009). It utilizes normalization and NLP techniques and integrates multiple strategies for ontology mapping. Also, RiMOM uses risk minimization to search for optimal mappings from the results of multiple strategies. The difficulty with this approach is that the input ontologies either hold identical labels or structure similarity for further processing. In order to utilize the availability of enormous instance, RIMOM is also concentrated on instance based matching (Zhang and Li, 2015). This semiautomatic method finds the predicate alignment between two ontologies then finds one or more suitable matchers. Finally, the matcher results are aggregated.

The CODI (Combinatorial Optimization for Data Integration) is a probabilistic logical alignment framework which aligns individuals, concepts and properties of two heterogeneous ontologies (Huber *et al*., 2011). It leveraged both lexical similarity measures and schema information and implemented an aggregation method of different similarity measures. New feature of CODI is the recognition of ontology pairs belonging to different versions of the same ontology. In instance matching CODI does not compute lexical similarities for all existing pairs of instances but utilizes object-property assertions for reducing the necessary comparisons.

ServOMap is a large scale ontology matching system (Ba and Diallo, 2013). This system relies on Information Retrieval (IR) techniques and a dynamic description of entities of different KOS (Knowledge Organization Systems) for computing the similarity between them. Its operation has three stages, i.e., initialization, candidate retrieval and post processing. Similarity, computing is performed using lexical similarity matching such as edit distance, QGram and I-Sub. Extended similarity matching is carried out by WordNet and machine learning based contextual similarity computing is used to find structural level similarities. Its performances rely heavily on the terminological richness of the input ontologies.

MapSSS is an ontology alignment system with syntactic, structural and semantic metrics (Cheatham and Hitzler, 2013). The syntactic metric is a simple lexical comparison. The structural metric is a graph-based method relies on the direct neighbours of an entity and the edges that connect the entity to those neighbours. It uses a semantic metric based on Google queries. When considering two labels, A from the first ontology and B from the second, this metric queries Google for the phrase A definition. It then searches the snippets on the first page of results for B. If B is found, the metric returns true, otherwise it returns false. If this metric returns true in both directions (i.e., googleMetric (A,B) and (B,A) are both true) then the mapping is added to the alignment. The main problem with MapSSS is due to the Google API query limit. Further, research will be needed to improve the utility of the google based semantic similarity metric.

MaasMatch is a mapping system with the initial focus of fully utilizing the information located in the concept names, labels and descriptions in order to produce a mapping between two ontologies (Schadd and Roos, 2012). This was achieved through the utilization of syntactic similarities and virtual documents which can also be used as a disambiguation method for the improvement of lexical similarities. The system produced unsatisfactory mappings when the naming and annotation features of an ontology are not present or distorted. To rectify this, the system now utilizes an internal structural similarity and a instance similarity while also a similarity flooding procedure after the aggregation step in order to discover additional mappings.

LogMap is participated in OAEI from the year 2011 onwards. It implemented highly optimized data structures for lexically and structurally indexing the input ontologies (Jimenez-Ruiz and Grau, 2011). The structures are used to compute an initial set of anchor mappings and to assign a confidence value to each mapping. The main aspect of LogMap is its iterative process that, starting from the initial anchors, alternates mapping repair and mapping discovery steps. To detect and repair unsatisfiable classes during the matching process, LogMap implements a sound and highly scalable ontology reasoner as well as a greedy diagnosis algorithm. It has three variants such as LogMapLt applies only string matching techniques, LogMapC uses a more aggressive repair algorithm, LogMapBio uses a BioPortal (Noy *et al*., 2009) as a dynamic provider of mediating ontologies instead of using a small number of preselected ontologies (Jimenez-Ruiz *et al*., 2015).

YAM++ is a self-configuration, flexible and extensible ontology matching system (Ngo and Bellahsene, 2012). Initially input ontologies are parsed by an ontology parser component. Indexing of information in ontologies is done by the annotation indexing and the structure indexing components. Entities with higher similar descriptions are filtered by candidate filtering component. From these candidate mappings, the terminological matcher component generates a set of mappings by comparing the annotations of entities. The instance-based matcher component adds new mappings through shared instances among ontologies. The results of the terminological matcher and the instance based matcher are aggregated into an element level matching result. Element level matching results are enhanced by structural matcher component using structural information of entities. The mapping results obtained from the three matches are then combined and selected by the combination and selection component to have a unique set of mappings. Finally, the semantic verification component refines the mappings to eliminate the inconsistent entities.

The AROMA uses association rule model and interestingness measure in the context of ontology matching (David, 2007, 2011). It uses the asymmetrical aspect of association rules to discover subsumption relations between hierarchies of ontologies. The AROMA first represents entity as terms and data then discovers the relations between hierarchies. To improve recall it needs to combine a structural matcher which will consider axioms such as transitivity and property cardinalities.

Mapping based on background knowledge: Most of the ontology matching approaches are restricted to the use of the information available in the ontologies (class, subclass, property, instance, axioms, etc.) being matched. Other aspects of the techniques go beyond this and use external background knowledge in the matching. The background knowledge is derived in different ways and from various kinds of knowledge sources. When using background knowledge two aspects are to be considered. One is relating the ontologies being matched to the background knowledge and second using the knowledge provided by that background source.

Semantic web is used as background knowledge for matching ontologies that explores the process of using large number of background ontologies in the matching (Sabou *et al*., 2006). In the experimental setup semantic web is used as a source of background ontologies detected using semantic search engine Swoogle. Matching between input ontologies is performed through reference ontology. Interoperability between different ontologies is realized through matching to reference ontology in the domain. Instead of matching each entity of one input ontology to other, the input entities are all matched to comprehensive domain reference ontology, yielding the mutual matches through this reference.

A fully automatic algorithm to discover and use missing background knowledge during the matching process is proposed by Giunchiglia *et al*. (2006). The discovery background knowledge is attained iteratively based on heuristic. Potential missing background knowledge is discovered using this process. A pair of matching concepts is considered a candidate match if its elements are not matched and the majority of their subconcepts in the hierarchy below are matched. Detected missing knowledge is added to the background knowledge. Once missing background knowledge is found it can be reused in the future.

The BLOOMS method using LOD as background knowledge is based on the idea of bootstrapping information already present on the LOD cloud (Jain *et al*., 2010). The BLOOMS accept two ontologies as inputs which are assumed to contain schema information. It then proceeds with the following steps. Construct a forest for each class name using information from Wikipedia. Then forests are compared to get a decision that is to which class names are to be aligned. With the help of the alignment API and reasoner post processing have done. A set of algorithms used to exploit upper ontologies as semantic bridges in the ontology matching process and presents a systematic analysis of the relationships among features of matched ontologies (Mascardi *et al*., 2010). The algorithm are based on textual, lexical and semantic web approaches and exploits the ontology features, i.e., number of simple and composite concepts, stems, concepts at the top level, suffixes and prefixes and ontology depth. Active learning framework which tries to find the most informative candidate matches to query and propagate the user correction according to the ontology structure to improve the matching accuracy is proposed (Shi *et al*., 2009). The existing active learning framework is improved by correct graph propagation algorithm, user feedback and by using upper ontologies as semantic bridges (Menendez-Mora *et al*., 2013). Semantic bridges are contributes to the overall matching process and corrects mistake matches. Its main limitation is the lack of reference alignment for further computation of the performance metrics.

Table 1: Illustrates various matching techniques, domain knowledge, aggregation methods, user interaction used by the systems under consideration

System	Input	Output	Terminological	Structural	Instance based	Semantic	Domain knowledge	Aggregation	User interaction
RiMOM (Zhang and Li, 2015)	OWL	1:1 alignments	Edit distance, vector distance, WordNet	Similarity propagation	Vector distance	-	-	Sigmoid	Automatic
CODI (Huber *et al.*, 2011)	OWL	1:1 alignments	Edit distance, vector distance, Jaro Winkler, Simth Waterman Goto, overlap coefficient and Jaccard similarity measures	Similarity flooding	Lexical matching restricted using object property assertion	-	-	Machine learning based weights	Automatic
ServOMap (Ba and Diallo, 2013)	OWL	1:1 alignments	I-Sub, QGram, edit distance, WordNet	Machine learning based (decision tree algorithm)	-	-	-	Empirically chosen parameters	Automatic
MapSSS (Cheatham and Hitzler, 2013)	`OWL	1:1 alignments -	Edit distance, vector distance	Flooding	-	Querying bt Google API	-	Weighted harmonic means	Automatic
MaasMatch (Schadd and Roos, 2012)	OWL	1:1 alignments	Token based (3-grams, Jaccard measure), vector distance, WordNet	Name path similarity	Information retrieval techniques	-	-	Weight based	Automatic
LogMap (Jimenez-Ruiz *et al.*, 2015)	OWL	N:m alignments	WordNet or UMLS-lexicon	Structural heuristics or an off-the-shelf DL reasoner	-	-	-	-	Automatic
YAM++ (Ngo and Bellahsene, 2012)	OWL	N:m (1:m) alignments	Element level: decision tree, SVM, NaiveBayes	Propagation algorithm	Sentence similarity, instance description	Global constraint optimization method	-	Dynamic weighted aggregation	Allowed
AROMA (David, 2011)	OWL	N:m alignments	Name matcher	-	Dataset extraction	Association rule model, special interestingness measures	-	Hybrid technique	Allowed

Corpus is used as background knowledge for matching ontologies (Madhavan *et al.*, 2005). It uses an approach that leverages a corpus of schemas and mappings in a particular domain to improve the robustness of schema matching algorithms. Corpus is exploited in two ways. First, increase the evidence about each element being matched by including evidence from similar elements in the corpus. Second, learn statistics about elements, their relationships and use them to infer constraints that are used to prune candidate mappings. Ontology is used as background knowledge to map source and target ontologies (Aleksovski *et al.*, 2006). Anchoring matches connect a source or target concept to one or more concepts in the background knowledge ontology. The source concept or the target concept can be anchored to multiple anchors and the background knowledge could reveal relationships on anchors that represent different properties.

So far, ontology mapping systems are analysed in structure based, terminological based, instance based, semantic reasoning based, hybrid based and background knowledge based categories. The analysis is on their mapping techniques, performance and features.

An overview of the systems which have repeatedly participated in OAEI 2010-2015 campaign (Euzenat *et al.*, 2010, 2011; Aguirre *et al.*, 2012; Dragisic *et al.*, 2014; Shvaiko and Euzenat, 2013) is presented in Table 1. The table columns presents the input format, the output column describes the alignment between entities, type of matching techniques used, i.e., terminological (string based), structural, extensional (instance based) and semantics, exploitation of domain knowledge, aggregation methods and requirement of user interaction.

From Table 1, almost all the systems except AROMA use terminological and structural methods for mapping. Domain knowledge is not exploited by these recent systems. Semantic techniques are not utilized by most of the systems. Instances are considered by some systems for enhancing mapping results. Except LogMap, other systems are using different aggregation methods for combining their individual matchers. Most of the systems are performs mapping automatically and some systems require manual intervention for editing results.

Challenges in ontology mapping: Correctness and completeness of mapping are used as an evaluation measure. Usually, the mapping results are evaluated by the performance metrics precision, recall, F-measure, fallout and overall. For the large ontologies, it is very difficult to form reference mappings that are generated manually and also the evaluation of generated mappings.

The analytical summary of ontology mapping techniques from Table 1 shows most of the techniques are relied on lexical and structural matching methods. But, domain ontologies represent the same concept with different terminologies. For example in weather domain 'minimum temperature' is represented as using different terminologies like 'lo', 'low', 'min', 'minimum', etc. Similarly in medical domain 'Hersen tumor' is same as 'brain tumor'. For such cases lexical matching techniques couldn't produce correct mapping results and structure matching techniques cannot yield good results for ontologies with more structure deviations. The ontologies which are written in a multi-lingual setting are difficult to handle by lexical methods. Similarly, mapping between different domain ontologies are not correctly handled by lexical and structural methods because of the contextual difference and structural variations. For these kinds of ontologies other types of techniques need to be suggested. Likewise, there is a need to extend the existing techniques to produce more accurate and efficient mappings.

The actual purpose of semantic web and especially ontology mapping is to semantically match entities of ontologies. Complete semantic based techniques are not developed so far because various heterogeneities occur within a domain among similar domains and between different domains. For example, in weather domain one system represents minimum temperature as 'min' with unit °C and other system use 'low' with unit °F. Likewise, schools and colleges are coming under the same domain education but the evaluation system used is 'mark' in school and 'grade' in colleges, the instructor is called 'teacher' in school and 'professor' in college. In chemical domain, one compound name is used to prepare a medicine. But, in medical domain the same chemical compound is available with different brand names. Handlings of such issues are a challenging task.

There is no single efficient technique to create mapping between ontologies. So, mapping systems use multiple individual matchers for mapping (Noy and Musen, 2000; Do and Rahm, 2002; Giunchiglia *et al*., 2007). The ways to combine the individual matchers and analysing the combined strategies are very difficult because of the empirical or heuristics based assumptions.

In order to reach interoperability over heterogeneous ontologies, two problems must be dealt with: metadata heterogeneity and instance heterogeneity (Bouquet *et al*., 2004; Kim and Seo, 1991). Metadata heterogeneity concerns the intended meaning of described information. There are two kinds of conflicts in metadata heterogeneity: structure conflict and name conflict. Structure conflict means that ontologies defined for the same domain may have different taxonomies. It can be type conflict, schema isomorphism conflict, generalization conflict and aggregation conflict (Kashyap and Sheth, 1996). Few systems handled type conflict, generalization and aggregation conflicts but they have not mentioned how the mapping results can be incorporated in the merging scenario. Name conflict means that concepts with the same intended meaning may use different names (synonyms) and the same name may be used to define different concepts (homonyms). Homonym is usually occurs between different domains and not resolved by mapping algorithms developed for single domain. Most of the systems addressed only naming conflicts especially synonyms. In real time applications the issue is how the other heterogeneities can be resolved and it is not feasible to use different mapping systems for different heterogeneities.

Instance heterogeneity concerns the different representations of instances (data). Information described by the same ontology can be represented in different ways. These differences are named as data scaling conflict, data representation conflict, data precision conflict (Kashyap and Sheth, 1996). For example, a date can be represented as 'year/month/date' format and also can be represented as 'month, date, year' format; person name can be represented as 'firstname lastname' or 'lastname, firstname', etc. Most of the instance conflict comes from time, date, year, percentage, money, person name, bibliographic data expression and measurement units.

Many of the available ontology mapping techniques are evaluated on benchmark datasets. But, it should be evaluated for case studies on the industrial size problems to find out the efficiency of the techniques for real world scenarios.

For ontology mapping centralized techniques are not much exploited because of the manipulation and availability of reference ontologies for a domain. To generate vocabularies of different context of the entities, i.e., place, time, format, culture, different perceptions of specialists and its relations have to be considered.

Another issue is about on the reusing of mapped results. Many of the systems produce mapping results and very few systems are doing merging. The ultimate purpose of mapping is that it can be reused effectively. For example, ontology integration, query answering, semantic search, etc. Most of the existing systems produce equivalence relations between entities. When performing integration, it is very difficult to find out the relation among other entities. If the relations between entities are semantic, it will be useful for integration.

RESULTS AND DISCUSSION

Research directions: Existing systems are evaluated using the standard metrics like precision, recall, F-measure, overall, etc. For large sized ontologies, it is very difficult and laborious to create expert mapping and the evaluation over actual mappings. There is a need of other kind of metrics to evaluate the performance of ontologies. One such suggestion is based on empirical measure that is how effectively the queries are answered or inference are framed or search can be done by the mapped ontology. We also conclude that there is a need to investigate mappings on a theoretical basis.

Second issue is on combining individual matchers. Two most popular approaches for combining the individual matchers are the hybrid and composite approach (Do and Rahm, 2002). Hybrid method combines different matchers together into a single algorithm. Composite method combines results of multiple matchers. For aggregating individual matchers, many techniques have been proposed. The Max technique returns the maximal similarity of individual matchers. The Weighted technique determines a weighted sum of similarity of individual matchers by assigning relative weights to individual matchers. The average technique is a case of the weighted strategy and returns the average similarity over all individual matchers. The SIGMOID technique combines multiple results using a sigmoid function, a smoothed threshold function. The HADAPT (Harmony based adaptive aggregation) technique is based on the harmony of different similarities as weight to aggregate various similarities. Harmony is defined as the similarity score of two truly mapped elements should be larger than that of all other pairs of elements that share the same row/column with the two elements in the similarity matrix which implies that the two elements of this pair mutually prefer each other. Some techniques which is less frequently used are ANZ (AVG÷number of nonzero similarities), MNZ (AVG×number of nonzero similarities). Currently, the systems that adopt the weighted technique or the SIGMOID technique need to manually set aggregation weights based on experience for different similarities or tentatively set centre position and steepness factor in the sigmoid function. However, manually predefined parameters are difficult to adapt to different mapping scenarios. The ways combine various individual matchers based on weight or some other methods and how to select appropriate aggregation methods based on number and type of individual matchers used and parameters that can truly reflect the reliability of different similarities deserves further research.

Except some ontology mapping systems all other systems have addressed naming conflicts especially synonyms. Other semantic conflicts (Kashyap and Sheth, 1996) are not properly addressed yet. But , there is a need of a complete system which addresses all semantic heterogeneities. Unsolved terminological conflicts by existing methods and homonyms can be resolved using their instance similarity. Most of the instance heterogeneities can be solved by simple transformation rules.

To effectively use the available ontologies and to increase the application of ontology mapping systems, it can be evaluated with real time ontologies other than bench mark ontologies. These are available through Google and Swoogle search engines. The LOD also has large number of heterogeneous ontologies belonging to each category. This kind of evaluation can improve the practical application of ontology mapping in more domains.

For ontologies with insufficient lexical overlap, poor structural correspondence and lack of instance, background knowledge is very much useful. But, the creation of complete and consistent background knowledge is a difficult task as it has to take many factors into consideration. To generate the reference ontology we have to consider vocabulary changes over times, place, people, culture and context. Once the background knowledge is completely developed mapping research will get another direction that is how to effectively use the knowledge using different strategies.

Main consideration of any data or application is its reusability. One of the main reusable methods of ontology is merging. Many of the available merging systems are not fully automatic; it needs user suggestion or interaction for select, delete or edit operation. Onion (Mitra and Wiederhold, 2002), PROMPT (Noy and Musen, 2000), FCA-merge (Stumme and Maedche, 2001), SAMBO (Lambrix and Tan, 2006) are some of the merging systems which need user interaction at some level. Complete automatic merging algorithm is not available because it is based on mapping results. The evaluation of integrated ontology is a tedious task. For mapping numbers of metrics are available and for merging the use of general metrics such as the relative coverage of the input ontologies, the compactness of the merge result as well as the degree of redundancy are used for evaluating merging systems. Apart from that, merging can be evaluated by degree of information retrieval by querying, searching and getting inference. All the merging issues are based on the correctness and completeness of the mapping results. We therefore think that there is a need to develop a more general methodology that includes an analysis of the integration task and supports the process of defining the role of ontologies with respect to the requirements. The development of such a methodology will be a major step in the work on ontology based information integration because it will help to integrate results already achieved on the technical side and will help to put these techniques to work in real life applications. The merging systems can also consider unmapped entities.

CONCLUSION

Ontology mapping is involved with many fields like machine learning, database, linguistics and also necessary for tasks such as search, integration, query answering and data translation. Based on the importance of this field, we have reviewed the state of the art techniques in ontology mapping under different categories and made analytical and empirical comparisons among the recent mapping techniques. The outcome of this analysis is that existing ontology matching systems have not much exploited domain knowledge. We have noticed the challenges such as, performance measures of matching techniques, aggregation of matching techniques, testing with real time test cases, different semantic heterogeneity handling issues, exploitation of background knowledge, reusing of mapped ontologies in the field of ontology mapping and suggested research directions for those challenges The review shows there is a significant fall of number of participating systems in OAEI 2014 campaign. Hope this analysis would help to address the research issues as well as develop more reference domain ontologies and mapping research towards reference ontology. It also shows the emergence to develop pure semantic mapping techniques. More importantly, reusing of the mapping results and techniques can automatically boom the field.

REFERENCES

Aguirre, J.L., B.C. Grau, K. Eckert, J. Euzenat and A. Ferrara *et al*., 2012. Results of the ontology alignment evaluation initiative 2012. Proceedings of the 7th International Workshop on Ontology Matching, November 11, 2012, Boston, MA.

Aleksovski, Z., M. Klein, W. ten Kate and F. van Harmelen, 2006. Matching unstructured vocabularies using a background ontology. Proceedings of the 15th International Conference on Managing Knowledge in a World of Networks, October 2-6, 2006, Czech Republic, pp: 182-197.

Amrouch, S. and S. Mostefai, 2012. Survey on the literature of ontology mapping, alignment and merging. Proceedings of the 2012 International Conference on Information Technology and E-Services, March 24-26, 2012, Sousse, pp: 1-5.

Aumueller, D., H.H. Do, S. Massmann and E. Rahm, 2005. Schema and ontology matching with COMA^{++}. Proceedings of the ACM SIGMOD International Conference on Management of Data, June 14-16, 2005, Baltimore, Maryland, USA., pp: 906-908.

Ba, M. and G. Diallo, 2013. Large-scale biomedical ontology matching with ServOMap. IRBM, 34: 56-59.

Bouquet, P., L. Serafini and S. Zanobini, 2003. Semantic Coordination: A New Approach and an Application. In: The Semantic Web-ISWC, Fensel, D., K. Sycara and J. Mylopoulos (Eds.). Springer, New York, USA., ISBN: 9783540397182, pp: 130-145.

Bouquet, P., J. Euzenat, E. Franconi, L. Serafini, G. Stamou and S. Tessaris, 2004. Specification of a common framework for characterizing alignment. Technical Report No. DIT-04-090, August 2004, University of Trento, pp: 1-31.

Cheatham, M. and P. Hitzler, 2013. String similarity metrics for ontology alignment. Proceedings of the 12th International Semantic Web Conference, October 21-25, 2013, Sydney, Australia, pp: 294-309.

David, J., 2007. Association rule ontology matching approach. Int. J. Semantic Web Inf. Syst., 3: 27-49.

David, J., 2011. AROMA results for OAEI 2011. Proceedings of the 6th International Conference on Ontology Matching, Volume 814, October 24, 2011, Germany, pp: 122-125.

Do, H.H. and E. Rahm, 2002. COMA-a system for flexible combination of schema matching approach. Proceedings of the 28th VLDB Conference, August 20-23, 2002, Hong Kong, China, pp: 610-620.

Doan, A.H., P. Domingoas and A.Y. Halevy, 2001. Reconciling schemas of disparate data sources: A machine-learning approach. ACM SIGMOD Record, 30: 509-520.

Doan, A., J. Madhavan, P. Domingos and A. Halevy, 2003. Ontology Matching: A Machine Learning Approach. In: Handbook on Ontologies in Information Systems, Staab, S. and R. Studer (Eds.)., Springer-Verlag, Berlin, pp: 397-416.

Doan, A. and A.Y. Halevy, 2005. Semantic integration research in the database community: A brief survey. AI Mag., 26: 83-94.

Dragisic, Z., K. Eckert, J. Euzenat, D. Faria and A. Ferrara *et al*., 2014. Results of the ontology alignment evaluation initiative 2014. Proceedings of the 9th International Conference on Ontology Matching, October 20, 2014, Trentino, Italy, pp: 61-104.

Ehrig, M. and S. Staab, 2002. QOM-quick ontology mapping. Proceedings of the 28th Annual International Conference on Very Large Data Bases, August 20-23, 2002, Hong Kong, pp: 610-621.

Euzenat, J. and P. Valtchev, 2004. Similarity-based ontology alignment in OWL-Lite. Proceedings of the 16th European Conference on Artificial Intelligence, August 22-27, 2004, Valencia, Spain, pp: 333-337.

Euzenat, J., A. Ferrara, C. Meilicke, A. Nikolov and J. Pane *et al*., 2010. Results of the ontology alignment evaluation initiative 2010. Proceedings of the 5th International Workshop on Ontology Matching, November 7, 2010, Shanghai, China, pp: 85-125.

Euzenat, J., A. Ferrara, W.R. van Hage, L. Hollink and C. Meilicke *et al*., 2011. Final results of the ontology alignment evaluation initiative 2011. Proceedings of the 6th International Workshop on Ontology Matching (OM), October 24, 2011, Germany.

Gangemi, A., N. Guarino, C. Masolo and A. Oltramari, 2003. Sweetening WordNet with DOLCE. AI Mag., 24: 13-24.

Giunchiglia, F. and P. Shvaiko, 2004. S-Match: An algorithm and an implementation of semantic matching. Proceedings of the European Semantic Web Symposium, May 10-12, 2004, Crete, Greece, pp: 61-75.

Giunchiglia, F., P. Shvaiko and Y. Yatskevich, 2006. Discovering Missing Background Knowledge in Ontology Matching. In: ECAI 2006, Brewka, G. (Ed.). IOS Press, New York, USA., ISBN: 9781586036423, pp: 382-386.

Giunchiglia, F., M. Yatskevich and P. Shvaiko, 2007. Semantic Matching: Algorithms and Implementation. In: Journal on Data Semantics IX, Spaccapietra, S., P. Atzeni, F. Fages, M.S. Hacid and M. Kifer *et al*. (Eds.). Springer, New York, ISBN-13: 9783540749875, pp: 1-38.

Gruber, T.R., 1993. A translation approach to portable ontology specifications. Knowledge Acquisition, 5: 199-220.

Hamacher, H., H. Leberling and H.J. Zimmermann, 1978. Sensitivity analysis in fuzzy linear programming. Fuzzy Sets Syst., 1: 269-281.

Huber, J., T. Sztyler, J. Noessner and C. Meilicke, 2011. Codi: Combinatorial optimization for data integration-results for OAEI 2011. Proceedings of the 6th International Workshop on Ontology Matching, October 24, 2011, Bonn, Germany.

Jain, P., P. Hitzler, A.P. Sheth, K. Verma and P.Z. Yeh, 2010. Ontology alignment for linked open data. Proceedings of the 9th International Semantic Web Conference, November 7-11, 2010, Shanghai, China, pp: 402-417.

Jimenez-Ruiz, E. and B.C. Grau, 2011. LogMap: Logic-based and scalable ontology matching. Proceedings of the 10th International Semantic Web Conference, October 23-27, 2011, Bonn, Germany, pp: 273-288.

Jimenez-Ruiz, E., B.C. Grau, A. Solimando and V. Cross, 2015. LogMap family results for OAEI 2015. Proceedings of the 10th International Conference on Ontology Matching, October 12, 2015, Bethlehem, PA., pp: 171-175.

Kashyap, V. and A. Sheth, 1996. Semantic and schematic similarities between database objects: A context based approach. Int. J. Very Large Data Bases, 5: 276-304.

Kim, W. and J. Seo, 1991. Classifying schematic and data heterogeneity in multidatabase systems. IEEE Comput., 24: 12-18.

Lambrix, P. and H. Tan, 2006. SAMBO-a system for aligning and merging biomedical ontologies. Web Semantics: Sci. Serv. Agents World Wide Web, 4: 196-206.

Li, J., J. Tang, Y. Li and Q. Luo, 2009. RiMOM: A dynamic multistrategy ontology alignment framework. IEEE Trans. Knowledge Data Eng., 21: 1218-1232.

Madhavan, J., P.A. Bernstein, A. Doan and A. Halevy, 2005. Corpus-based schema matching. Proceedings of the 21st International Conference on Data Engineering, April 5-8, 2005, Tokyo, Japan, pp: 57-68.

Mascardi, V., A. Locoro and P. Rosso, 2010. Automatic ontology matching via upper ontologies: A systematic evaluation. IEEE Trans. Knowledge Data Eng., 22: 609-623.

Melnik, S., H. Garcia-Molina and E. Rahm, 2002. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. Proceedings of the 8th International Conference of Data Engineering, February 26-March 1, 2002, San Jose, CA., pp: 117-128.

Melnik, S., E. Rahm and P.A. Bernstein, 2003. Rondo: A programming platform for generic model management. Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, June 9-12, 2003, San Diego, CA., pp: 193-204.

Menendez-Mora, R.E. and R. Ichise, 2013. Ontology matching by actively propagating user feedbacks through upper ontologies. Acad. J., 10: 85-92.

Mitra, P. and G. Wiederhold, 2002. Resolving terminological heterogeneity in ontologies. Proceedings of the ECAI-02 Workshop on Ontologies and Semantic Interoperability, July 22, 2002, Lyon, France.

Mitra, P., N. Noy and A.R. Jaiswal, 2004. Omen: A probabilistic ontology mapping tool. Proceedings of the 4th International Semantic Web Conference, November 7-11, 2004, Hiroshima, Japan, pp: 537-547.

Ngo, D.H. and Z. Bellahsene, 2012. YAM++: A Multi-Strategy Based Approach for Ontology Matching Task. In: Knowledge Engineering and Knowledge Management, Ten Teije, A., J. Volker, S. Handschuh, H. Stuckenschmidt and M. D'Acquin *et al.* (Eds.). Springer, New York, USA., ISBN: 9783642338762, pp: 421-425.

Niles, I. and A. Pease, 2001. Towards a standard upper ontology. Proceedings of the International Conference on Formal Ontology in Information Systems, October 17-19, 2001, Ogunquit, ME., pp: 2-9.

Noy, N.F. and M.A. Musen, 2000. PROMPT: Algorithm and tool for automated ontology merging and alignment. Proceedings of the 17th National Conference on Artificial Intelligence and 12th Conference on Innovative Applications of Artificial Intelligence, July 31-August 2, 2000, Austin, Texas, pp: 450-455.

Noy, N. and M. Musen, 2001. Anchor-PROMPT: Using non-local context for semantic matching. Proceedings of the Workshop on Ontologies and Information Sharing at the Seventeenth International Joint Conference on Artificial Intelligence, August 4-10, 2001, Seattle, USA., pp: 63-70.

Noy, N.F., 2004. Semantic integration: A survey of ontology-based approaches. ACM Sigmod Rec., 33: 65-70.

Noy, N.F., N.H. Shah, P.L. Whetzel, B. Dai and M. Dorf *et al.*, 2009. BioPortal: Ontologies and integrated data resources at the click of a mouse. Nucleic Acids Res., 37: 170-173.

Sabou, M., M. Aquin and E. Motta, 2006. Using the semantic web as background knowledge for ontology mapping. Proceedings of the 1st International Workshop on Ontology Matching, November 10, 2006, Athens, GA., USA., pp: 156-190.

Schadd, F.C. and N. Roos, 2012. Maasmatch results for OAEI 2012. Proceedings of the 7th ISWC International Workshop on Ontology Matching, Volume 946, December 2012, Germany, pp: 160-167.

Shi, F., J. Li, J. Tang, G. Xie and H. Li, 2009. Actively learning ontology matching via user interaction. Proceedings of the 8th International Semantic Web Conference, October 25-29, 2009, Chantilly, VA., USA., pp: 585-600.

Shvaiko, P. and J. Euzenat, 2013. Ontology matching: State of the art and future challenges. IEEE Trans. Knowl. Data Eng., 25: 158-176.

Stoilos, G., G. Stamou and S. Kollias, 2005. A string metric for ontology alignment. Proceedings of the 4th International Semantic Web Conference, Volume 3729, November 6-10, 2005, Galway, Ireland, pp: 624-637.

Stuckenschmidt, H. and F. Van Harmelen, 2005. Information Sharing on the Semantic Web. Springer, New York, ISBN: 9783540205944, Pages: 276.

Stumme, G. and A. Maedche, 2001. FCA-merge: Bottom-up merging of ontologies. Proceedings of the 17th International Joint Conference on Artificial Intelligence, Volume 1, August 4-10, 2001, Seattle, Washington, USA., pp: 225-230.

Van Rijsbergen, C.J., 1979. Information Retrieval. Butterworths, London, UK.

Wiederhold, G., 1992. Mediators in the architecture of future information systems. Computer, 25: 38-49.

Zhang, Y. and J. Li, 2015. RiMOM results for OAEI 2015. http://ceur-ws.org/Vol-1545/oaei15_paper13.pdf

Zhu, J., 2012. Survey on ontology mapping. Proceedings of the International Conference on Applied Physics and Industrial Engineering, June 2012, India, pp: 1857-1862.