Journal of Engineering and Applied Sciences 14 (2): 399-404, 2019
ISSN: 1816-949X

Automated Ensemble Framework for Integration of Ontology Based Large Scale Semantic Knowledge Base

[1]G. Silambarasan and [2]J. Anvar Shathik
[1]CMJ University, Meghalaya, India
[2]KGISL Institute of Technology, Coimbatore, India

Abstract: Knowledge base is growing exponentially now a days using different techniques. The ontology has been used widely to integrate the knowledge base for easy retrieval of the web document contents to the user queries. Several steps have been taken in the literatures to integrate the knowledge base which contains the overlapping and complementary information. In this study, we propose a novel technique to knowledge based integration named "automatic ensemble framework for integration of ontology based sematic knowledge base". It considers the semantic heterogeneous class structures. The proposed framework provides the Solution to the NP hard problem in terms of query selection. Ensemble framework produces the multiple class structures to the knowledge base as knowledge base is large in size and structure matching model is leverages to identify the relationship based on semantic in order to integrate the complex structures of the different KBs. Integrated Knowledge base is been available to access through queries but improper information selection to query leads to complex problem which can be avoided by placing the adaptive query selection algorithm using greedy algorithms. The experimental result demonstrates that proposed model outperforms the state of art approaches in terms of effectiveness, efficiency and accuracy.

Key words: Ontology, knowledge base, semantic web, data integration, ensemble technique, efficiency

INTRODUCTION

With development of semantic web in recent years more and more data has been published in Semantic Web formats the Resource Description Framework (RDF) and the Web Ontology Language (OWL). Currently large scale Knowledge Bases (KBs) have been constructed using different technique and from different sources and it becoming large such as YAGO, ProBase, FreeBase, DBpedia, NELL and DeepDive (Hoffart *et al*., 2013; Wu *et al*., 2012). Mostly KB designed using different technique usually contains overlapping and complementary information. Moreover, as knowledge acquisition is an expensive process, reusing existing KBs is strongly desirable to reduce the cost of data management. Therefore, knowledge base integration has attracted growing interests. In the last decade, a wide variety of works have been conducted on ontology integration (Suchanek *et al*., 2011; Lacoste-Julien *et al*., 2013) which is related to the problem of knowledge base integration as an ontology can be treated as the conceptual system to underlie a particular knowledge base. To integrate KBs, both data and structure information are combined to align classes, instances and relations/properties. The alignment process has to be found based on class equivalence. Major task in the KB Integration are class structure integration and instance matching. In this research, class structure is represented as taxonomy. The class structure integration and instance matching is used for entity resolution, data integration and data cleaning (Lacoste-Julien *et al*., 2013).

In this study, we propose automatic ensemble framework for integration of ontology based sematic knowledge base. In this taxonomy integration based on ensemble mechanism is carried out initially and then aligning of the instances is carried out based on the taxonomy integration result. Instance matching is computed with each class structure which is partitioned by ensemble process, ensemble process is proposed in order to reduce the computation time by partitioning the data into different partitions. The relationship between the data is classified into more categories by employing the unsupervised classification model such as principle component analysis. It is can be used to determine the equivalence relationship and generalization relationship to integrate the two KB to generate the unified structure.

Literature review: There exist many techniques to integrate the KBs are designed and implemented efficiently. Each of these techniques follows some sort of

Corresponding Author: G. Silambarasan, CMJ University, Meghalaya, India class structure unification, among few performs nearly equivalent to the proposed framework which is described as follows.

Actively learning ontology matching via. user interaction: In this literature, we analyse the active learning framework for ontology matching which tries to find the most informative candidate matches to query of the user. The user's feedbacks are used to correct the mistake matching propagates the supervise information to help the entire matching process. Different measures are utilized to estimate the confidence of each matching candidate. A propagation algorithm is further enabled to maximize the spread of the user's guidance (Shi *et al*., 2009).

HAMSTER; Using search clicklogs for schema and taxonomy matching: In this literature, we analyse an unsupervised matching of schema information from a large number of data sources into the schema of a data warehouse. The matching process is the first step of a framework to integrate data feeds from third-party data providers into a structured-search engine's data warehouse. We utilize technique based on the search engine's click logs. Two schema elements are matched if the distributions of keyword queries that cause click-through on their instances are similar (Giaretta and Guarino, 1995).

MATERIALS AND METHODS

Proposed model: In this study, we describe the automated knowledge base integration mechanism using principle component analysis technique on class structure integration and instance matching. This process is represented as follows.

Representation of the knowledge base: A Knowledge Base (KB) is a tuple denoted by (E; L; R; P), consisting of a collection of Entities E, Literals L, Relations R holding between entities and properties P holding between entities and literals. An entity e E can be a class or an instance. E = {cUI where C and I represent a class set and an instance set.

The example of KB, there is four entities two classes, "Actor" and "Celebrity" and two instances, "Vijay" and "Sangeetha"; the date "24-6-1974" and string "Joseph Vijay" are literals; three relations "subclass", "type of" and "married to" and two properties "born" and "full name". Figure 1 describes the representation of the KB. In a knowledge base, the classes form a hierarchical structure in which different classes have "subclass/ superclass" relationships.

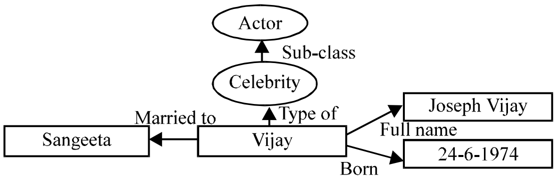

Fig. 1: Representation of KB

In order to integrate two KBs, the projection of the work is to identify the positions of entities from one KB in another one to construct a unified KB. Other words knowledge base integration is the process of identifying the position of each entity from KB1 in 2 (or vice versa) to get the unified knowledge base.

Taxonomy or class integration: Taxonomy integration is the process of identifying the position of each class node from T1 in T2 to get the unified taxonomy. We introduce class semantic relationship between classes into categories

Equivalence: The equivalence between classes refers that the two classes represent the same concept.

Generalization and specification: The concept of one class is a subclass/superclass of another one. For alignment of class, semantic relationship is has to be considered. Partitioning technique utilizes the ensemble mechanism to partition the entities, for an entity from KB1, if an equivalent relationship is found in KB2 then the position is identified.

Automatic ensemble framework for integration of ontology based knowledge bases: The proposed framework provides the solution to the NP hard problem in terms of query selection.

Query selection: Objective of taxonomy integration is to design a proper interface between the query and web data. Given two taxonomies T1 and T2 for each node in T1 we treat it as a query node and the position search space consists of all nodes of T2 (each node in T2 is called the target node). The contextual information of the target node has to be mapped to the query node.

The pruning strategy is applied according to the outcome of past queries which may further influence the future query selection. Query selection strategy is to adaptively make a sequence of decisions. Adaptive query selection algorithm using greedy algorithms has been proposed to handle instance matching.

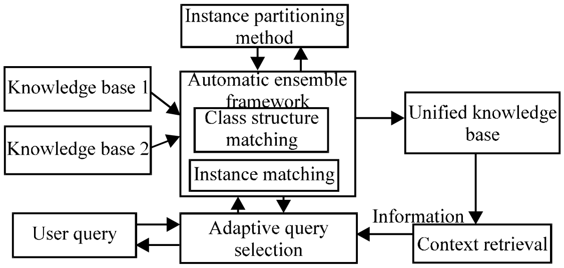

Fig. 2: Architecture diagram of the proposed model

Algorithm 1; Query selections:

```
Input: Two Taxonomy T1 and T2
Output: Query Set Q
Process
Initialize Instance pair IP
 Where Cp→0
Resultant Query Set
 For each Instance Pair IP1_IP
 Generate query Q
 Where Q = {LT1, i}
 Q = arg Max (Q)_IP
Return Q
```

The algorithm is adaptive greedy algorithm for query selection. The query set has associated with prior probability.

Class structure generation based on contextual information: The conceptual information of the each node is been derived in the two knowledge bases. It considers the semantic heterogeneous class structures. It is the process of detecting pairs of matching entities among two large, clean but overlapping collections of entities. Naive pairwise based instance matching is intractable for matching entities between two large KBs. In order to scale to large volumes of data, approximate techniques are adopted. Ensemble techniques cluster the similar entities into partitions.

The each class c from a Taxonomy T1, first reduce the instance list to size of c by computing a prior belief of generalization/specification relationships and filter the classes with prior score lower than a threshold. The detail architecture of the proposed model is described in the Fig. 2.

Ensemble framework produces the multiple class structures to the knowledge base as knowledge base is large in size and structure matching model is leverages to identify the relationship based on semantic in order to integrate the complex structures of the different KBs.

After normalizing the prior belief, we can get a probability distribution of results. Note that in our dataset, the KBs use the OWL identifier (Jean-Mary *et al*., 2009) to represent the instances, it is easy to get the instance equivalence information (this knowledge will be hidden when matching instances) for other dataset, the equivalence can be estimated using label information of instances as adopted. Initial instance matching pairs by considering the instance string representation. The similarity of a pair of candidate entities is computed using lexical similarities between entity names.

Ontology integration on semantic knowledge base: Ontology is an explicit specification of the conceptualization of a domain. Information models (such as the HL7 RIM) and standardized vocabularies (such as UMLS) can be part of ontology. Ontology provides a core component in a knowledge-based system. Ontology Integration can also carried out using metadata (Nandi and Bernstein, 2009). Metadata is the detailed description of the instance data; the format and characteristics of the populated instance data; instances and values dependent on the requirements/role of the metadata recipient (Nandi and Bernstein, 2009).

Once ontology tags are obtained for the semantic embedded information in OWL file, the system will need to compare and merge this instance to gather more domain representation for the concepts in semantic knowledge base. Ontology integration (Papadakis *et al*., 2011) is to bridge conceptual model which represented lexical word on overlapping instance of the knowledge base. Knowledge base integrated using principle component analysis (Kondreddi *et al*., 2014).

OWL is a language for defining web ontologies (Shvaiko and Euzenat, 2013) and their associated knowledge bases. The knowledge integration using ontology is associated with the discriminant between the sub class can be achieved easily. The example is represented below

Algorithm 2; OWL algorithm:

```
There are two types of animals, Male and Female
        <rdfs: Class rdf: ID = ”Male”>
         <rdfs:subClassOf rdf:resource=”#Animal”/>
        </rdfs:Class>
The subClassOf element asserts that its subject-Male-is a subclass of its
object -- the resource identified by #Animal
        <rdfs:Class rdf:ID = “Female”>
         <rdfs: subClassOf rdf: resource = “#Animal”/>
         <owl: disjointWith rdf: resource = “#Male”/>
        </rdfs:Class>
```

One animal are female too but nothing can be both male and female (in this ontology) because these two classes are disjoint (using the disjoint with tag).

RESULTS AND DISCUSSION

In this study, we describe the experimental results of the proposed framework against the existing approaches.

The experimental result demonstrates that proposed model outperforms the state of art approaches in terms of effectiveness, efficiency and accuracy. The detailed description is as follows

Dataset description: We have done extensive experiments on 2 real datasets which is as follows:

YAGO: YAGO is a semantic knowledge base in which entities, facts and events are anchored in both time and space. YAGO 2 is built automatically from Wikipedia, GeoNames and WordNet. It contains 447 million facts about 9.8 million entities. Human evaluation confirmed an accuracy of 95% of the facts in YAGO (Hoffart *et al*., 2013).

DBpedia: DBpedia is a community effort to extract structured information from Wikipedia and to make this information available on the web. DBpedia allows you to ask sophisticated queries against datasets derived from Wikipedia and to link other datasets on the web to Wikipedia data. We describe the extraction of the DBpedia datasets and how the resulting information is published on the web for human-and machine-consumption. We describe some emerging applications from the DBpedia community and show how website researchers can facilitate DBpedia content within their sites (Auer *et al*., 2007).

Evaluation: The proposed framework is evaluated against the following measures against several preprocessing steps on those data sets.

Precision: Positive predictive value is the fraction of relevant instances among the retrieved instances. Precision is the number of correct feature divided by the number of all returned feature space:

$$\text{Precision} = \frac{\text{True positive}}{\text{True positive+False positive}}$$

True positive is a number of real positive cases in the data and false negative is number of real negative cases in the data. The precision is evaluated against different dataset is depicted in the Fig. 3 and performance values is described in the Table 1 for all the dataset used in this research.

Recall: It is the fraction of relevant instances that have been retrieved over the total amount of relevant instances. The recall is the part of the relevant documents that are successfully classified into the exact classes:

Table 1: Performance comparison of methodology against measures for various dataset

Dataset/System	Precision (%)	Recall (%)	F measure (%)	Computation time (sec)
YAGO				
AEF	99.75	88.63	93.76	9
Alom	97.39	82.29	90.77	21
Hamster	96.29	85.23	92.15	41
DBpedia				
AEF	99.23	88.28	93.01	10
Alom	97.15	82.09	89.56	22
Hamster	96.19	84.98	91.85	49

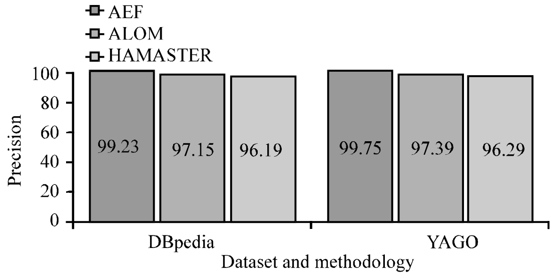

Fig. 3: Performance evaluation of the methodologies on precision against the different datasets

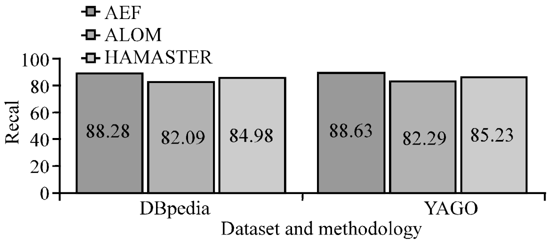

Fig. 4: Performance evaluation of the methodologies on recall against the different datasets

$$\text{Recall} = \frac{\text{True positive}}{\text{True positive+False positive}}$$

True positive is a number of real positive cases in the data and false negative is number of real negative cases in the data. The recall is evaluated against different dataset is depicted in the Fig. 4 and performance values is described in the Table 1 for all the dataset used in this research.

F-measure: It is a measure of a test's accuracy and is defined as the weighted harmonic mean of the precision and recall of the test. The performance of the methodology is described in the Fig. 5 and performance values is described in the Table 1 for all the dataset used in this research.

Computation time: It is defined as no of time taken to establish the instance matching for the different lexical

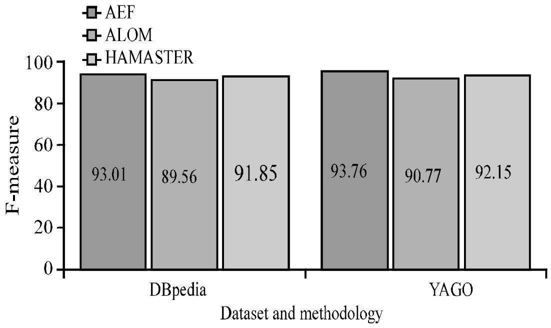

Fig. 5: Performance evaluation of the methodologies on F-measure against the different datasets

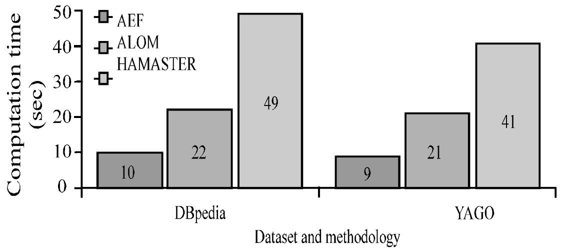

Fig. 6: Performance evaluation of the methodologies on computation time against the different datasets

words between the two heterogeneous sources. The performance evaluation chart of the computation time and its values is described in Fig. 6 and Table 1:

$$\text{Computation time} = \frac{\text{No. of time taken for single instance}}{\text{Total time taken for entire instance mapping}}$$

The evaluation of result is described in the Table 1 for DBpedia and YAGO datasets. It is observed that the proposed method is always better when compared to class structure integration and with entity mapping using ontology tags, it has provided better or comparable results.

CONCLUSION

We have designed and implemented an automatic ensemble framework for integration of ontology based sematic knowledge base. The problem of knowledge base integration has been achieved with high accuracy. The greedy based algorithm is also modelled to for query pruning. Based on the taxonomy integration result, we align the instance through an ensemble constrainst and OWL constrainst. It has capability integrated the complex structures of the representation. Finally proposed system is verified to working better through extensive results in terms of both accuracy and efficiency.

REFERENCES

Auer, S., C. Bizer, G. Kobilarov, J. Lehmann and R. Cyganiak *et al.*, 2007. Dbpedia: A nucleus for a web of open data. Proceedings of the 6th International the Semantic Web and 2nd Asian Conference on Asian Semantic Web ISWC'07/ASWC'07, November 11-15, 2007, ACM, Busan, Korea, ISBN:978-3-540-76297-3, pp: 722-735.

Giaretta, P. and N. Guarino, 1995. Ontologies and Knowledge Bases towards a Terminological Clarification. In: Towards very Large Knowledge Bases: Knowledge Building and Knowledge Sharing, Mars, N.J.I. (Ed.). IOS Press, Amsterdam, Netherlands, pp: 307-317.

Hoffart, J., F.M. Suchanek, K. Berberich and G. Weikum, 2013. YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. Artif. Intell., 194: 28-61.

Jean-Mary, Y.R., E.P. Shironoshita and M.R. Kabuka, 2009. Ontology matching with semantic verification. Web Semant., 7: 235-251.

Kondreddi, S.K., P. Triantafillou and G. Weikum, 2014. Combining information extraction and human computing for crowdsourced knowledge acquisition. Proceedings of the 2014 IEEE 30th International Conference on Data Engineering (ICDE), March 31-April 4, 2014, IEEE, Chicago, Illinois, USA., ISBN:978-1-4799-2555-1, pp: 988-999.

Lacoste-Julien, S., K. Palla, A. Davies, G. Kasneci and T. Graepel *et al.*, 2013. Sigma: Simple greedy matching for aligning large knowledge bases. Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 11-14, 2013, ACM, Chicago, Illinois, USA., ISBN:978-1-4503-2174-7, pp: 572-580.

Nandi, A. and P.A. Bernstein, 2009. Hamster: Using search clicklogs for schema and taxonomy matching. Proc. VLDB. Endowment, 2: 181-192.

Papadakis, G., E. Ioannou, C. Niederee and P. Fankhauser, 2011. Efficient entity resolution for large heterogeneous information spaces. Proceedings of the 4th ACM International Conference on Web Search and Data Mining, February 09-12, 2011, ACM, Hong Kong, China, ISBN:978-1-4503-0493-1, pp: 535-544.

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. Transac. Knowl. Data Eng., 25: 158-176.

Suchanek, F.M., S. Abiteboul and P. Senellart, 2011. Paris: Probabilistic alignment of relations, instances and schema. Proc. VLDB. Endowment, 5: 157-168.

Wu, W., H. Li, H. Wang and K.Q. Zhu, 2012. Probase: A probabilistic taxonomy for text understanding. Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, May 20-24, 2012, ACM, Scottsdale, Arizona, USA., ISBN:978-1-4503-1247-9, pp: 481-492.