ISSN 22773061

Ontology Mapping using Fuzzy Decision Tree and Formal Concept Analysis

Manjula Shenoy K, Manipal University, India
Dr. K.C.Shet, NITK Suratkal, India
Dr. U. Dinesh Acharya, Manipal University, India

ABSTRACT

An ontology describes and defines the terms used to describe and represent an area of knowledge. Different people or organizations come up with their own ontology; having their own view of the domain. So, for systems to interoperate, it becomes necessary to map these heterogeneous ontologies.This paper discusses the state of the art methods and outlines a new approach with improved precision and recall. Also the system finds other than 1:1 relationships.

Keywords:ontology mapping, ontology matching, ontology,OWL, Semantic web, fuzzy decision tree, formal concept analysis.

Council for Innovative Research
Peer Review Research Publishing System
Journal: INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY
Vol 8, No 1

editor@cirworld.com
www.cirworld.com, member.cirworld.com

ISSN 22773061

1. INTRODUCTION

The current web WWW has billions of pages, most of which are in human readable format only. As a consequence, software agents cannot understand and process this information and much of the potential of the web has so farremained untapped. Some problems of this web are non -availability of collective information, search based on keyword, irrelevant and excessive information, Semi-structuredinformation representation etc. In response, researchers have created the vision of the Semantic Web[Berners-Lee et.al.2001], where data has structure and semantics. Ontologies describe the semantics of the data. The term ontology is borrowed from philosophy, where it refers to a systematic account of what can exist or 'be' in the world. In the fields of artificial intelligence and knowledge representation, the term refers to the construction of knowledge models that specify a set of concepts, their attributes and the relationships between them. Ontology allows explicitly specifying a domain of knowledge, which permits to access and reason about agent knowledge, incorporating semantics into data, and promoting its exchange in an explicit and understandable form. Collectively defined as "formal explicit specification of a shared conceptualization". To share information and knowledge of heterogeneous systems, one of the key issue is, the mapping of their ontologies for interoperability. Given two ontology O1 and O2, mapping one ontology onto another implies that for each entity (concept C, relation R, or instance I) in ontology O1, we try to find a corresponding entity, which has the same intended meaning, in another ontology O2. There have been several mapping approachesdeveloped so far. But many of them do find only 1:1 mapping and their precision and recall is not up to the mark for real time data. In this paper, we outline a mapping method which has improved precisionand recall and finds other type of mapping relation in addition to 1:1.

2. RELATED WORK

There have been several approaches for ontology mapping. Few of them are discussed in this section.

2.1. YAM++(YetAnotherMatcher)

It is an automatic flexible self-configuring ontology matching system for discovery of semantic correspondences between entities. The input ontologies are loaded and parsed by ontology parser. Entity information in ontology are indexed by two indexing systems namely annotation indexing and structure indexing. Then candidates having maximum similarity are filtered to reduce the search space. Next terminological matcher and instance matchers come up with mapping. The results of these mappings are aggregated and given to structural mapping system to find further mapping. Finally all these results are combined and selected through a component called combiner and selector. The final result is subjected to semantic verification to refine the found mappings.

2.2. MapSSS

This is an OWL ontology alignment algorithm designed to explore what can be accomplished using simple similarity measures. Input ontologies are treated as a directed graph with nodes corresponding to concepts, properties and individuals and an edge corresponding to relationship. The algorithm consists of syntactic, structural, and semantic metrics. These matrices are applied one after another and a positive result from one of them is treated as a match.

2.3. AROMA

It is a hybrid extensional and asymmetric matching approach designed to find relations of equivalence and subsumption. It makes use of association rule paradigm and a statistical interestingness measure.

3. OUR APPROACH

The components in overall matching system are as shown in figure 1.

Preprocessor: This component converts all labels into lowercase, removes special symbols such as _, $ etc. Also it groups annotations, concepts individuals, and properties.

Similarity Computor:

This unit analyses ontology and computes lexical, structural and instance wise similarity measures and creates table of these for every entity pair between given two ontologies.

Equivalence Filter: This filters those entities which have very high similarity in almost all measures. Also it groups pairs which are stated already similar in given ontology. The output of filter is directly fed to training set generator unit. The remaining entities are passed on to grouping and fuzzyfying system.

Grouping and fuzzyfying system: This unit aggregates all lexical,structural and instance based measures into five groups. And applies fuzzy function on it to convert it into discrete value.

Training set generator: It takes manually computed similarity pairs together with highly similar pairs and prepares it as a training set to construct decision tree.

Decision tree constructor: This unit constructs a decision tree for the training set.

ISSN 22773061

Classifier: Uses decision tree constructed and helps in learning new mapping rules between remaining entities fed from grouping and fuzzyfying unit.

3.1Similarity measures used

First we made a study of factors influencing the mapping of entities. This study revealed the following facts. If labels are same they are likely to be the same.If properties are equal then

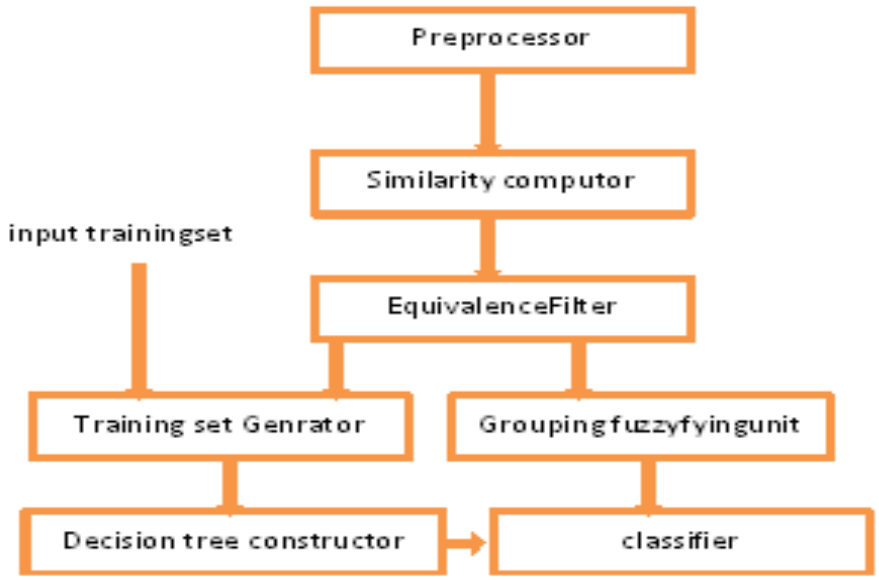

Figure 1. Components of proposed system

Concepts are likely to be equal.If domain and range for properties are equal then properties are likely to be equal.If super concepts of c1 and c2 are same then c1 and c2 are likely to be same

If sub concepts are same then their super concepts are likely to be same.If concepts have similar siblings they are likely to be similar. If super properties are same so are sub properties.If sub properties are same, so are super properties.If instances are same concepts are likely to be same

Instances that have same mother concept are same. If concepts have a similar low/high fraction of the instances then they are likely to be same.If two instances are linked to another instance via the same property, the 2 original instances are same.If two properties connect the same two instances the properties can be similar.If OWL file itself declares equality then such entities are equal. In order to cover these facts we used the following similarity measures. [Shvaiko et.al.2007][Resnik 1999][Manjula ShenoyK et.al.2012].

1) String equality checking [1, 0]

If two labels are equal their similarity measure is taken as 1 otherwise 0.

2) Hamming distance [0 1]

$$\partial(s,t) = \frac{(\sum_{i=1}^{\min(|s|,|t|)} s[i] \neq t[i]) + ||s| - |t||}{\max(|s|,|t|)}$$

Here s and t are entities to be matched.

3) Levenshteins Distance is the minimum number of insertions,deletions and substitutions of characters required to transform one string into other.

4) Substring similarity

$$\sigma(x,y) = \frac{2|t|}{|x| + |y|}$$

Here t is the largest common substring of x,y.

5)Cosine similarity

$$\sigma V(s,t) = \frac{\sum_{i\epsilon|v|} \overrightarrow{si} \; X \; \overrightarrow{vi}}{\sqrt{\sum_{i\epsilon|V|} \overrightarrow{si^2}} \; X \sum_{i\epsilon|v|} \overrightarrow{ti^2}}$$

Based upon the comments per entities we define vector for each entity and use the above formula to find their dissimilarity.

6) Path comparison

Enumerates all paths from root concept to the entities to be matched and then finds their similarity according to these paths.

Given two hierarchy of strings $< si >_{i=1}^{n}$ and $< tj >_{j=1}^{m}$ their path distance is defined as follows

$\delta(< si >_{i=1}^{n} , < tj >_{j=1}^{m}) = \lambda X \delta'((s_n, t_m) + (1-\lambda) X \delta(< s_i >_{i=1}^{n-1}, < t_j >_{j=1}^{m-1})$

Such that $\delta(<>, < tj >_{j=1}^{k}) = \delta(< si >_{i=1}^{k}, <>) = k$

δ' is the semantic similarity measure based on wordnet. $\lambda \; is \; between \; 0 \; and \; 1.$

7)Synonymy similarity

$$\sigma(s,t) = \begin{cases} 1 \; if \sum(s) \cap \sum(t) \neq \phi \\ 0 \quad otherwise \end{cases}$$

This depends on similarity of synonyms between the entities

8) Cosynonymy similarity

$$\sigma(s,t) = \frac{|\sum(s) \cap \sum(t)|}{|\sum(s) \cup \sum(t)|}$$

9)Resnik similarity

S(s,t)=IC(lcs(s,t)

10)Lin similarity

S(s,t)= 2(IC(lcs(s,t)) / IC(s)+Ic(t)

11)Dice similarity = $\frac{2.|Ic1 \cap Ic2|}{|Ic1| + |Ic2|} \in [0..1] \forall c1 \in Co1, c2 \in Co2$

12) Instance equality measure= $\begin{cases} 1, if \; |Ic1 \cap Ic2| > 0 \\ 0, if \; |Ic1 \cap Ic2| = 0 \in [0...1] \forall c1 \in Co1, c2 \in Co2 \end{cases}$

13)Modified Wu and Palmer similarity measure

Sim(C1,C2) = (2N.e-λL/D)/N1+N2.

14)Graph based similarity measure

(BSAt +BtSA) / || BSAt+BtSA||

Here A, B are adjacency matrices corresponding to the ontology graph.

15)Node attribute based similarity measure

16)Degree difference similarity measure

17)Edge attribute similarity measure

18)Lesk measure

3.2. Fuzzy decision tree construction

The measures listed above are grouped into five distinct groups. An aggregate similarity measure per group is computed as weighted sum of similarity measures in the group. Calculation of similarity measures result in a table with columns equal to each aggregate similarity measure and rows correspond to entity pair. Now In order to construct decision tree we use a table with manually mapped entity pairs and their aggregate similarity measures of ontologies in the domain as that of the ontologies to be mapped. The decision tree is called fuzzy because we use fuzzy membership function for numerical attributes which are similarity measures. The fuzzy membership function used is triangular and is explained below.Suppose we have a table of numerical attribute values as shown in table I, after applying fuzzy membership function explained in figure 1, the table becomes as in table II.

ISSN 22773061

TABLE I Similarity measure calculated for concept pair

Concept pair	Similarity measure
C1, C1	0.234
C1,C2	0.325
C2,C1	0.412
C1,C3	0.556
C2,C3	0.67
C3,C1	0.25
C3,C2	0.987

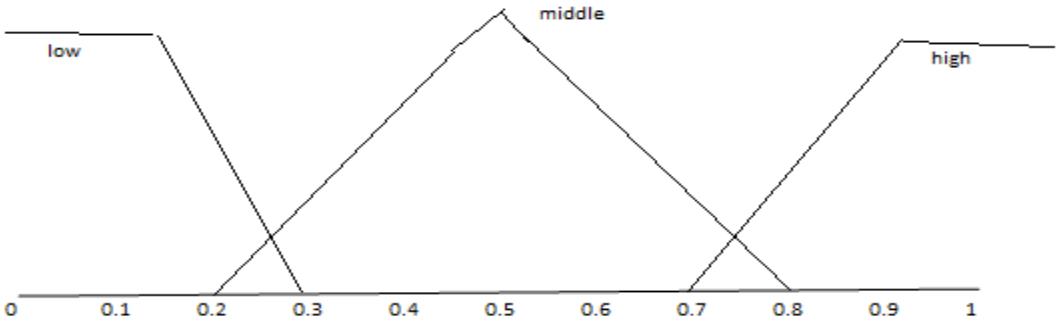

Figure 2. Fuzzy Membership function

TABLE II Similarity measure after applying fuzzy membership function

Concept pair	Similarity measure
C1, C1	low
C1,C2	low
C2,C1	middle
C1,C3	middle
C2,C3	middle
C3,C1	low
C3,C2	high

If U={u1,u2……us} is the set of data samples where C={c1,c2,….cn} is the set of n similarity measures(condition attributes) and D={d} is class label attribute. Suppose this class label attribute has m different values di for(i=1 to m), let Si be the number of samples of class di in U. Now the expected information or entropy needed to classify a given sample is given by I(S1,S2,S3…Sm)= $-\sum_{i=1}^{m} pi \log_2 pi$

Where pi is the probability that an arbitrary sample belongs to class Si and is estimated by summation of those samples entropy(m is number of all such samples). Let attribute Ci have v distinct value {A1,A2…..Av}. So this attribute can be used to partition U into v subsets {S1,S2,…Sv} where Sij(j=1 to v) contains those samples in U that have value Aj of Ci. Let Sij be the number of samples of class di in a subset Sj .The entropy of attribute Ci is given by

$$E(ci) = \sum_{j=1}^{v} \left(\frac{S1j + S2j + \cdots Smj}{S}\right) I\,(S1j, S2j \ldots Smj)$$

The term$\frac{S1j+S2j+\cdots Smj}{S}$ acts as weight of the jth subset and is the number of samples in the subset divided by the total number of samples. The smaller the entropy value ; the greater the purity of the subset partitions. Thus the attribute that leads to the largest information gain is selected as branching attribute. For a given subset Sj the information gain is expressed as

$$I\,(S1j, S2j \ldots Smj) = -\sum_{i=1}^{m} pij\ \log_2 pij$$

Where pij=$\frac{Sij}{|Sj|}$. (|Sj| number of samples in the subset Sj) and is the probability that a sample in Sj belongs to di. So information gain of attribute Ci is given by

Gain(Ci)= $I\,(S1j, S2j \ldots Smj)$-E(ci).

Example:

Table III Sample data set U with s=10.

Sl.No	Sim1	Sim2	Sim3	Sim4	Class
1	Low	middle	low	middle	1
2	Low	middle	middle	low	0
3	Low	low	low	middle	0
4	high	high	middle	middle	1
5	Low	low	low	middle	0
6	Middle	middle	high	high	1
7	Middle	high	high	high	1
8	High	high	middle	middle	1
9	high	high	middle	middle	1
10	low	low	middle	low	0

C=(Sim1,Sim2,Sim3,Sim4} n=4. D={class label} d1=1 and d2=0. m=2.

S1 =number of samples of class 1 in U. S2=number of samples of class0 in U.

There are six samples of class 1 and 4 samples of class 0.

I(S1,S2)= $-\frac{6}{10}\log_2\frac{6}{10}-\frac{4}{10}\log_2\frac{4}{10}$=0.97. {Note: I(S1,S2,S3…Sm)= -$\sum_{i=1}^{m} pi\ \log_2 pi$}

Compute entropy for each attribute.

For Sim1,U can be partitioned into 3 subsets S1,S2 and S3 since it has 3 distinct value. v=3.

For Sim1=high S11=number of samples with Sim1=high and in class d1=3. S21=number of samples with Sim1=high and in class d2=0.

I(S11,S21)=0

$\{Note: I\,(S1j, S2j \ldots Smj) = -\sum_{i=1}^{m} pij\ \log_2 pij\}$

IIIly for Sim1=middle S12= 2 S22=0 I(S12,S22)=0;

For Sim1=low S13=1 S23=4

I(S13,S23)= $-\frac{1}{5}\log_2\frac{1}{5}-\frac{4}{5}\log_2\frac{4}{5}$=0.721.

E(Sim1)= $\frac{3}{10}I(S11,S21)+\frac{2}{10}I(S12,S22)+\frac{5}{10}I(S13,S23)$=0.3605

Gain(Sim1)=I(S1,S2)-E(Sim1)= 0.97-.3605=0.6095.

Calculate similarly Gain(Sim2),Gain(Sim3),Gain(sim4) Whichever is maximum we select that as the root of the decision tree. To choose the next node we continue in the same way.

3.3. Formal concept analysis for finding other than 1:1 relationships

In order to find other than 1:1 relationship we use the method of formal concept analysis. The requirement for this is that the input ontologies should contain instances. We form formal concept table based upon each common instance belonging to different concepts of two ontologies as follows. That is row of the formal context table will be matched instances and columns will be the concepts of ontology1 having this matched instance plus concepts of ontology2 having this matched instance.

Example:

TABLE IV Formal context

	Book(a)	Science(b)	Popular©	Pocket(d)	Essay(e)	Biography(f)	Auto(g) biography	Literature(h)	Novel(i)	Poetry(j)
MyLife(1)	X	X	X		X	X	X			
Logic(2)	X	X			X					
LaChute(3)	X			X				X	X	
MesProprietes(4)	X							X		X

Concept lattice as shown in Figure 3is built for above formal context defined in TableIV , by noting down following.

(1,2,3,4)(a) (1,2)(a,b,e) (1) (a,b,c,e,f,g) (3,4) (a,h) (3)(a,d,h,i) (4)(a,h,j)

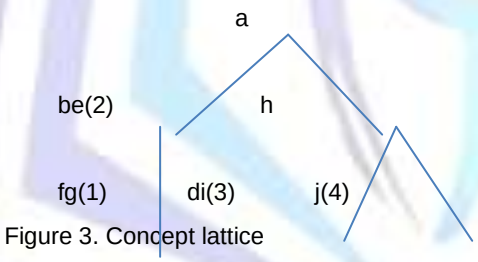

Figure 3. Concept lattice

'a' is common for all the instances, so it becomes the root concept. Next common pattern is be for 1, 2 instances. So they become next root. Among 3, 4 h is common so it becomes another root for 3, 4. The forest is constructed similarly. From the above tree it is clear that book is the root concept. Science and Essay are equivalent concepts and are sub concepts of book. And so on.

3.4. Result evaluation

The goal of the benchmark data set is to provide a stable and detailed picture of each algorithm. For that purpose, algorithms are run on systematically generated test cases. The systematic benchmark test set is built around seed ontology and many variations of it. Variations are artificially generated, and focus on the characterization of the behavior of the tools rather than having them compete on real-life problems.They are organized in three groups:

Simple tests (1xx) such as comparing the reference ontology with itself;

Systematic tests (2xx) obtained by discarding/modifying features from the reference ontology. Considered features are names of entities, comments, the specialization hierarchy, instances, properties and classes.

Real-life ontologies (3xx) found on the web.

The results of the approach on benchmark data sets of OAEI 2011 are as shown in table V.

TABLE V Results of benchmark test

Group	precision	recall	f-measure
1xx	1	1	1
2xx	0.93	0.68	0.78
3xx	0.90	0.59	0.75

It is also tested on benchmark data sets of 2012. The seed ontology concerns bibliographic references and is inspired freely from BibTeX. Itcontains 33 named classes, 24 object properties, 40 data properties, 56 named individualsand 20 anonymous individuals. Data set finance is about finance ontology, which contains 322 classes,247 object properties, 64 data properties and 1113 named individuals.Among the recent ontology mapping methods, results of MapSSS, YAM++ and AROMA are compared with the proposed method and the proposed method gave good result with respect to them. The table VI depicts the resulting precision and recall. Plot of the same is shown in Figure 4. Precision and recall have improved. This is due to the consideration of both instance and metadata measures present in ontology.

TABLE VI Overall results

dataset	biblio				finance			
accuracy	AROMA	YAM++	MapSSS	FDT	AROMA	YAM++	MapSSS	FDT
Precision	0.98	0.98	0.99	1	0.94	0.97	0.99	1
recall	0.64	0.72	0.77	0.8	0.58	0.84	0.71	0.85
fmeasure	0.77	0.83	0.87	0.9	0.72	0.9	0.83	0.91

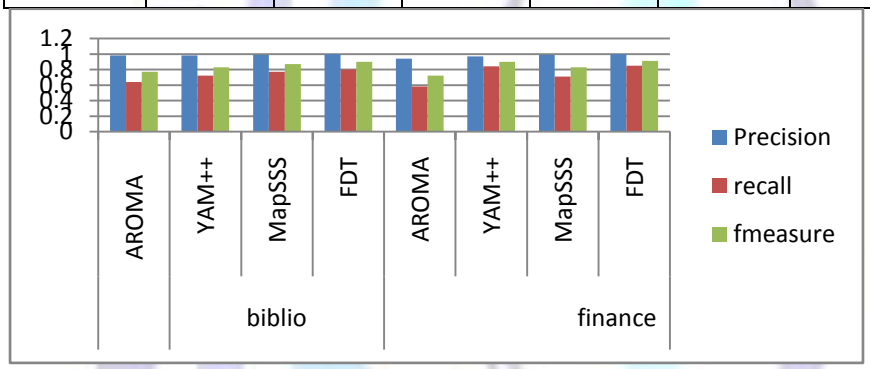

Figure 4. Result

4.CONCLUSION

In this paper we have outlined a method to map ontologies using fuzzy decision tree approach. The method so proposed has given good precision and recall compared with leading systems in this area. Future work is to find precision and recall for other data sets of OM-2012 workshop.

REFERENCES

[1]Shvaiko P., Euzenat J., “Ontology Matching State of the Art and Future Challenges” IEEE transactions on Knowledge and Data Engineering. 2013

[2] AmrouchSiham,MostefaiSihem , “Survey on the Literature of Ontology Mapping, Alignment and Merging” Proc. International Conference on Information Technology and e-services(ICITes) 2012.

[3]Shvaiko P.,Euzenat J., “Ten Challenges for Ontology matching” Proc. Seventh International Conference on Ontologies ,Databases and Application of Semantics(ODBASE) 2008.

[4]ManjulaShenoy K, K.C.Shet, U.DineshAcharya, ”Secured ontology mapping” in International Journal of Web and Semantic Technology(IJWEST). October 2012.

[5]ManjulaShenoy K, K.C. Shet ,U.DineshAcharya,”A new similarity measure for taxonomy based on edge counting”, in International Journal of Web and Semantic Technology(IJWEST). October 2012.

[6]Philip Resnik, Semantic Similarity in a Taxonomy:An Information Based Measure and its Application to Problems of Ambiguity in Natural Language” Tournal of Artificial Intelligence Research,1999.

[7] Jesus Oliva et.al., “SyMSS: A syntax Based Measure for Short Text Semantic Similarity” Journal of Data and Knowledge Engineering,Elsevir,2011.

[8] Jiawel Han and MichelineKamber,” Data Mining Concepts and Techniques”, Morgan Kaufmann publishers,Second Edition 2006.

[9] Jerome Euzenat and PavelShvaiko ,” Ontology matching”,Springer-Verlag,2007.

[10] PavelShvaiko, et.al. Proceedings of ISWC-2012 workshop on ontology matching.