Earth Sci Inform
DOI 10.1007/s12145-014-0203-1

RESEARCH ARTICLE

SEM+: tool for discovering concept mapping in Earth science related domain

Jin Guang Zheng · Linyun Fu · Xiaogang Ma · Peter Fox

Received: 21 April 2014 / Accepted: 22 December 2014

Abstract The amount of Earth Science related domain concepts and vocabularies encoded in popular Semantic Web languages such as OWL and SKOS grows rapidly as more and more domain scientists realize the power of Semantic Web Technologies. The interlinking between these concepts will enable the possibility of performing data integration and identity recognition, which is crucial in developing applications that use data from multiple sources. In this paper, we discuss a new tool for performing concept mapping called SEM+. In SEM+, we designed the Information Entropy based Weighted Similarity Model to compute semantic similarity between entity data and suggest possible linking. We also adopted a blocking approach to group possible matching entities into one block and therefore reduce the computation space. We performed evaluations on SEM+ using the Integrated Ocean Observatory System ontology and the Marine Metadata Interoperability ontology and discussed the results and new findings.

Keywords Ontology matching · Instance matching · Owl:sameAs · Entity resolution · Linked data

Communicated by: H. A. Babaie

Published in the Special Issue of *Semantic e-Science* with Guest Editors Dr. Xiaogang Ma, Dr. Peter Fox, Dr. Thomas Narock and Dr. Brian Wilson

J. G. Zheng (✉) · L. Fu · X. Ma · P. Fox
Tetherless World Constellation, Computer Science Department,
Rensselaer Polytechnic Institute, 110 8th St, Troy, NY, USA
e-mail: zhengj3@rpi.edu

L. Fu
e-mail: ful2@rpi.edu

X. Ma
e-mail: max7@rpi.edu

P. Fox
e-mail: foxp@rpi.edu

Introduction

Ontologies and domain vocabularies have been widely adopted and used in Earth, environmental and geospatial related sciences, such as the Semantic Web for Earth and Environmental Terminology (SWEET) (Raskin and Pan 2005), NERC vocabularies (NERC Vocabulary Server 2999), and Global Change Master Directory (GCMD) (http://gcmd.nasa.gov/). Many of these ontologies and vocabularies make use of World Wide Web Consortium (W3C) recommended formats such as Web Ontology Language (OWL) (Bechhofer 2009), Simple Knowledge Organization System (SKOS) (SKOS (2007)), and Resource Framework Description (RDF) (Klyne and Carroll 2006). One of the advantages of using such formats is that the knowledge and concept representation can be easily processed by computers (Berners-Lee et al. 2001). These ontologies and vocabularies aimed to capture the semantics of the data and to represent domain knowledge.

Due to the large number of ontologies in the domain and different knowledge modeling scenarios and different concept term representations, there is a semantic heterogeneity issue among these ontologies. Sometimes, a same concept may be represented by different terms in two ontologies. For example, in geologic time, the term Tertiary is still used by some geologists. It equals to the time span between the Paleogene and Neogene Periods of the Cenozoic Era in the 2009 Geologic Time Scale chart by the Geological Society of America (Walker and Geissman 2009), but no longer used there. Sometimes, two terms from two different ontologies may have same meaning, such as the term Spring, which can mean the season of the year or a natural source of water. Because of this heterogeneity issue, knowledge and data represented by different ontologies are difficult to be integrated and reused by multiple systems. To reconcile such differences of semantic representations among different

Published online: 10 January 2015

ontologies, some efforts have been devoted to build a common semantic framework and to encode ontologies for Earth sciences, such as SWEET. Even though such an ontology is useful and powerful in solving the heterogeneity problem, it requires a shared understanding of the concepts in a domain and the semantic framework itself. This means for any new comers who want to use the framework, there is a huge learning effort required on the user to understand the framework. Another way to solve this semantic heterogeneity issue is to perform concept mapping among these ontologies and vocabularies. Such mapping has been performed manually by the domain scientists, such as the mapping between various geological time classifications (Haq 2007). However, performing manual mappings can be time consuming and often not scalable.

In order to solve the heterogeneity problem in a scalable manner, we built a semantic similarity based ontology and vocabulary matching system, SEM+ (Similarity-based Entity Matching). SEM+ implements a novel semantic similarity computation model called the Information Entropy and Weighted Similarity Model (IEWS Model) to suggest similarity measures between concepts from different ontologies and vocabularies. Based on the similarity measures, SEM+ creates "same as" links among those concepts. SEM+ also implements a new prefix-based blocking algorithm, which groups possible matching pairs into one block. This blocking algorithm reduces the number of concept-pairs that are needed for similarity computation, which is useful when we are required to perform mapping between two large domain ontologies. We tested and evaluated SEM+ by performing matching tasks on various Earth and environmental related ontologies. To summarize, the main contributions of the paper include:

We present a new concept matching tools for Earth and Environmental
related ontologies and vocabularies with high precision and recall.
We present a prefix-based blocking algorithm which reduces number of pair-wise matching computation and allows trade-off between efficiency and effectiveness.
Our evaluation result shows that SEM+ can be used to suggest many possible mappings across different domain ontologies. These newly discovered concept mappings can be leveraged by the domain scientists in related studies.

The remainder of this paper is organized as follows: Related work section discusses some existing works in the literature and compares them with our system. SEM+ section presents the SEM+ system. Experimental evaluation is discussed in Experiment section. Conclusions are presented in Conclusions section.

Related work

To the best of our knowledge, we are the first one who develop and apply concept matching tool to the Earth science domain. However, our approach is related to research in both instance and schema level matching conducted by the database and the Semantic Web communities.

In database community, instance matching is also known as concept resolution, record linkage (Newcombe and Kenedy 1962), deduplication (Sarawagi and Bhamidipaty 2002), reference reconciliation (Dong et al. 2005). Tools such as TAILOR (Elfeky et al. 2005), BigMatch (Yancey 2002), MOMA (Thor and Rahm 2007), and Swoosh (Benjelloun et al. 2009) have been developed. These concept resolution tools follow the single-global-threshold paradigm, where they compute match suggestion measures in either supervised or unsupervised manner and then compare the measures with a threshold to determine the match. Chaudhuri (Chaudhuri et al. 2005) proposed the compact set criterion and sparse neighborhood criterion to enable more accurate characterization of duplicated records. Compared to these systems, our proposed work is a semantic similarity driven matching system.

In the Semantic Web community, algorithms are developed to compute similarity between instance data on the Web of Data, such as those presented in papers (Nguyen et al. 2012; Volz et al. 2009; Rong et al. 2012). Volz et al. (Volz et al. 2009) used user configured information as a guide and computed similarity measure for possible matching suggestion. Compared to this approach, our approach doesn't require user configuration. SLINT (Nguyen et al. 2012) applied different similarity computation methods to different type of values such as dates similarity for date values, integer similarity for integer values, etc., then combined these similarities to get final similarity value for possible matching. Compared to SLINT, our approach takes both common and distinguishing descriptions into consideration while ignores descriptions that are not present for differentiating one concept from another. Rong et al. (Rong et al. 2012) extracted literal information from the concepts and represented this information as vectors. They then used the vector space model and other machine learning techniques to compute a similarity score. Compared to Rong et al.'s algorithm, our approach considers not only the literal information but also the structural information.

Schema level ontology matching is a more studied field compared to instance level ontology matching in the Semantic Web community. There are many impressive state-of-the-art systems developed for the purpose of performing schema level ontology matching (Shvaiko and Euzenant 2013). In this section, we will focus on reviewing some of the similarity based ontology matching tools developed in recent years, such as (Duan et al. 2012; Jean-Mary et al. 2009; Stumme and Madche 2011; Euzenat 1994; Tang et al. 2006; Cruz et al. 2009). Among these systems, ASMOV (Jean-Mary et al.

2009) computed the similarity between two concepts from different ontologies by computing their lexical similarities and structural similarities. Duan et al. (Duan et al. 2012) used Jaccard Similarity and "Edit distance" similarity as two measures in similarity computation. FCA-Merge (Stumme and Madche 2011) and T-Tree (Euzenat 1994) computed subclass similarity, superclass similarity, lexical similarity, and concepts' instance similarities to suggestion mapping. Compared to FCA-Merge and T-Tree, RiMOM (Tang et al. 2006) took more information in similarity computation such as taxonomy structure, concept names, etc. AgreementMaker (Cruz et al. 2009) first used TF*IDF model to compute cosine similarity between two concepts. It then computed descendent similarities and sibling similarities using the cosine similarities. Compared to these systems, SEM+ considers that information describing concepts has different weights and it uses a machine learning approach and an information entropy approach to compute these weights.

In terms of scalability, to the best of our knowledge, not many existing matchers investigated the problem (Jimenez-Ruiz and Grau 2012; Nguyen et al. 2012), since most of the existing matching systems were designed with focus on improving precision and recall. They have been applied on small datasets. SLINT (Nguyen et al. 2012), Rong et al.'s system (Rong et al. 2012) and LogMap (Jimenez-Ruiz and Grau 2012) are three systems that have algorithms to reduce the computation space. All those systems treated literal descriptions of concepts as bags of words and used the inverted index technique (Baeza-Yates and Ribeiro-Neto 1999) to create blocks. Compared to these computation reduction techniques, our blocking algorithm further reduces the concept-wise computation using *prefix blocks*.

SEM+

In this section, we discuss the detailed implementation of SEM+. We start by defining the problem we intend to solve. Given two sets of concepts from two ontologies, namely C and C', where concepts c ε C and c' ε C'. The concepts c and c' are described by a set of statements δs in a triple format as (*subject, predicate, object*). We then compute similarity scores s between c and c'. The score s is in the range of 0 to 1, where 0 indicates c and c' are dissimilar and 1 indicates c and c' are representing the same object. Based on this similarity score, we select possible concept matches.

Overview

SEM+ consists of two major components: 1. *Prefix-blocking* groups concepts that are likely to be similar to each other into one block, and dissimilar concepts into difference blocks based on the literal descriptions of the concepts such as rdfs:label, rdfs:comment, etc. 2. *IEWS Model* takes two or more concepts from the same block and computes semantic similarity between these concepts. *IEWS Model* consists of three sub-components. A *Property Weight Learning component* learns the importance of the properties that are used to describe the features of the given concepts, and assigns weights to each property. An *Information entropy computation component* computes information entropy of the common descriptions of two concepts. A *Triple-Wise Similarity computation component* computes similarity between two triples. An overview of SEM+ is depicted in Fig. 1.

Blocking algorithm

Giving two large sets of concepts, pairwise similarity computation becomes so expensive. Therefore reducing the number of concept pairs for which similarity scores are to be computed is important. In SEM+, we adopted a blocking algorithm to reduce the computation space. The goal of blocking is to group similar concepts into "same" blocks and dissimilar concepts into "different" blocks as fast as possible, with blocks as small as possible. Potential similar concepts should be contained in the blocks as completely as possible. More careful (thus more expensive) similarity computation is then to be performed within each block to determine those exact similarity score. In essence, the function of blocking divides concepts into blocks with restricted size and thus reduces the number of concept pairs for exact similarity score computation. Then the problem lies in how to find good indicators for potentially similar concepts without using sophisticated formulae such as $\mathrm{Sim}^F(c, c')$, see Eq. (4), in the IEWS Model.

In many concepts, parts of the descriptions of the concepts are presented in plain literal values. These literal descriptions play an important role on describing the concepts. Assuming that each concept eventually will be linked to a set of words that describe certain properties of the concept, we can leverage this information to perform keyword based indexing to help improve the performance in terms of computation speed. In SEM+, we proposed to compute the concept frequency (the number of concepts a word belongs to) of words appear in the literal descriptions (*LDs*) or non-URL descriptions, such as labels and comments, and then to compare only the prefixes of concepts. Here *prefixes* are certain number of words that have the least concept frequency. When implementing the concept blocking approach, words in each concept c are extracted and an inverted index is built to record the list of concept for each word w, and each list has size l_w. Then we filter the inverted index by removing the list for w if $l_w > l_b$, where l_b is the *blocking parameter*. The remaining words are the *prefixes*. Since we only choose less frequent words to be *prefixes*, the approach also follows the intuition that concepts sharing some least common descriptions are much more likely to be the same than those do not share, because these rare descriptions

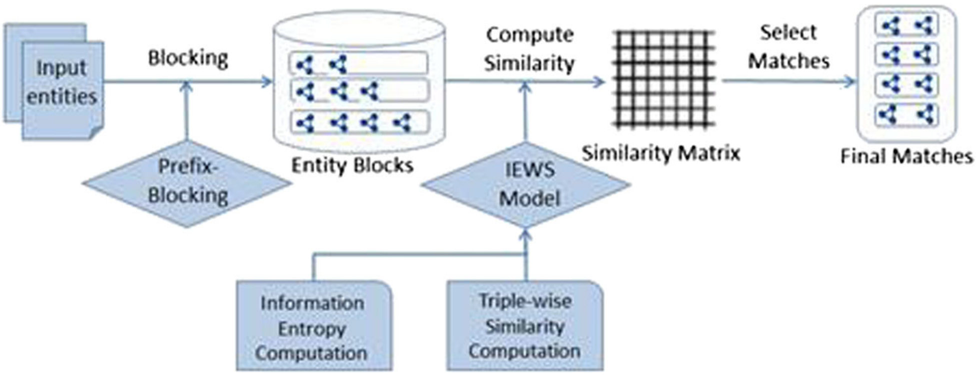

Fig. 1 System overview of SEM+: input is set of entities to be matched and output is concept matches

are usually key features of the concepts. Below is an example of the blocking.

Example 1. Consider the following four concepts and their corresponding *LDs*.

$$\begin{aligned} w &= \{A, B, C, E, K, L\} & x &= \{C, D, E, L\} \\ y &= \{B, K, E, L\} & z &= \{A, B, L\} \end{aligned}$$

If l_b=2, Then the prefixes and corresponding blocks are

$$A\ :\ \{w,\ z\}\ C\ :\ \{w,\ x\}\ D\ :\ \{x\}\ K\ :\ \{w, y\}$$

We can see this approach ignores the frequent words such as B, E and L, which appear in more than 3 documents. It also treats concepts with different number of features equally based on the rare-description-sharing criterion. Note that the blocks may contain an overlap of some of their concepts. Since the final similarity computation $\mathrm{Sim}^{\mathrm{F}}$*(c, c')* will only apply to the concepts within the same block, we reduce the number of $\mathrm{Sim}^{\mathrm{F}}$*(c, c'*) computation from 6 to 3 in this example. For each concept pair from different blocks, a similarity score 0 is assigned, in this example, similarity score 0 is assigned to concept pair x and y.

This blocking stage makes it possible to make trade-offs between efficiency and effectiveness in the matching computation. Greater blocking parameter values result more, and bigger blocks, which do not speed up the matching process dramatically but preserve more similar pairs in the blocks. Smaller parameter values speed up the similarity computation a lot, but discard more similar pairs out of the blocks.

Information entropy and weighted similarity model

Triple-wise similarity computation

The concepts in ontologies are described by a set of triples δ. Each triple describes one of the properties about the concept. Therefore, computing similarity between two concepts is the same as computing similarity between the triples that describe the concepts. In this section, we discuss how SEM+ performs triple-wise similarity computation or *pv* similarity (*Sim*^{pv}) computation.

One of the challenges when computing *pv* similarity is that sometimes the data that describe the same information of the concept are structured differently. For example, _:Boston rdfs:type _:t1

_:t1 rdfs:label ‘City’

is same as

_:Boston _:category ‘City’.

To solve this problem, SEM+ first checks if the propertis of the *pvs* are the same or there exists a property mapping between the properties of the *pvs* and then uses Eq. (1) to compute the similarity score. In the given example, a mapping between rdf:type and _:category must be established before the similarity computation. This property mapping is just a sub-problem of ontology schema matching. In case of ontology matching, property mapping between OWL, SKOS, and RDFS is pre-assigned. In case of instance matching, we differentiate the properties’ URLs to obtain ontologies that describe the properties, and thereafter perform property mapping.

$$\mathrm{Sim}^{\mathrm{pv}}(pv, pv') = \begin{cases} 1\ .\ \ \mathrm{Sim}^{l}\ (v,\ v')\ \text{ if both } v_s \text{ are literal} \\ 2\ .\ \ \mathrm{Sim}^{F}\ (v,\ v')\ \text{ if both } v_s \text{ are URL} \\ 3.\ \text{extract and comute use } \mathrm{Sim}^{l} \text{ otherwise} \end{cases} \qquad (1)$$

In Eq. (1), *pv* ε δ and *pv*’ ε δ’, *v* is the value part of *pv* pair and *p* is the property part of *pv* pair. The formula checks to see whether the value parts of both *pv* pairs are URLs or literals. Note that, URL means that the value points to another resource description. If both values are literal, SEM+ computes the *pv* similarity using Lin’s similarity (Sim^{l}) (Lin 1998). If both values are URLs, SEM+

computes *pv* similarity recursively using Eq. (4). In some cases, it is costly to recursively traverse the URLs until there is no URLs to follow and then compute similarity. Therefore, in SEM+, we only traverse URLs to the depth of three. In the case where one of the values is literal and the other is a URL, SEM+ first extracts the resource description of the resource point by using the URL, then extracts the literal contents of the other value, and finally uses Sim^{l} to compute the similarity. As a result of this process we get a vector of *pv* similarity that represents the similarity between concepts c and c'.

By applying triple-wise similarity computation algorithm and the concept of Jaccard similarity, one can compute the similarity between two concepts c and c' as:

$$\mathrm{Sim}(\mathrm{c},\ \mathrm{c'}) = \frac{\sum \mathrm{Sim}^{pv}}{\sum \mathrm{Sim}^{pv} + \alpha\left(\left|PV1\right| - \sum \mathrm{Sim}^{pv}\right) + \beta\left(\left|PV2\right| - \sum \mathrm{Sim}^{pv}\right)} \tag{2}$$

Where $|PV1|$ is the number of *pvs* in concept c and $|PV2|$ is the number of *pvs* in concept c', and α and β are coefficients of variation on the similarity measure on c and c' unique description.

Information entropy

Information entropy is a quantified measure of the uncertainty of the information content (Shannon 1948). In the previous section, we assume that each triple is equally important for describing the entity in terms of precision. However, in reality, the amount of information that each triple contains is different in terms of discriminative power. For example, the amount of information for the triple that describes the Social Security Number is higher than the triple that describes the gender. In information theory, Shannon suggests that the amount of information can be quantified as information entropy.

To compute the information entropy presented in the descriptions of concepts, we consider each property describing the concept as a *Variable X*, and possible values of the property as possible *Outcomes* x_s. By knowing the outcome of X, we eliminate the uncertainty because of variable X. For example, if we are given a triple which describes the gender of an unknown person (_:unknown _:gender ?g), then there will be two possible guesses: male or female. The probability of gender to be male or female depends on the distribution of the poll of the unknown person come from. If the poll is binomially distributed, then the expected uncertainties or probabilities of the unknown person to be male or female are both 1/2. In other words, the information entropy for property _:gender is 1/2. Therefore, if we know the value of _:gender, for example, female, we can eliminate all concepts that are male. Following is a formal definition for computing information entropy of properties describe the concepts based on Shannon's entropy (Shannon 1948):

Given a property X, with possible values $\{x_1, x_2, x_3, x_4 \ldots . x_n\}$ and probability of obtaining each value as $P(x_i)$, then the information entropy of X denoted $H(X)$ is:

$$H(X) = -\sum_{i=1}^{n} P(x_i)\ log_b\ (P(x_i)) \tag{3}$$

where b is the base for the logarithm.

Final similarity computation

In previous section, we discussed how we compute triple-wise similarities. By putting everything together, we get the final similarity computation formula:

$$\mathrm{Sim}^{\mathrm{F}}(c,\ c') = \mathrm{H(P)}\frac{\sum Sim^{pv}}{\sum Sim^{pv} + \alpha\left(\left|PV1\right| - \sum \mathrm{Sim}^{pv}\right) + \beta\left(\left|PV2\right| - \sum \mathrm{Sim}^{pv}\right)} \tag{4}$$

In this equation, H(P) is the information entropy of the common descriptions, and P is the set of properties in $\sum \mathrm{Sim}^{pv}$.

Using Eq. (4), SEM+ computes final similarity score and, based on the similarity scores, it suggests possible matches.

Experiment

A prototype of SEM+ was implemented using Java and existing frameworks such as Lucene[1] and Jena.[2] Using this prototype, we conducted two experiments: 1. Study the accuracy of SEM+ in both ontology schema matching and instance matching. During this experiment, we set blocking parameter $l_b = full$, to ensure that we covered all possible comparisons and therefore it gives a better understanding on the accuracy performance of SEM+. SEM+ was configured to perform one to one matching. We also set α and β to be 1 in the final similarity computation. In this experiment, we also analyzed the effectiveness of proposed Property Weight Learning component and Information Entropy component in order to improve the accuracy of SEM+. 2. Study how well our blocking algorithm reduces both the computation space and the cost of reduction in terms of recall. In this experiment, we set l_b to different values in order to study the effect of blocking. The experiments were carried out on a PC with 8 Intel Xeon processors of speed 2.40 GHz and 32 GB memory. Each processor has a 12 M cache.

Accuracy evaluation

The goal of concept matching is to discover and generate an alignment among entities that refer to same real-world individual or concept. In some cases, concept matchers would either match instances that are not referred to the same individual or concept, or not generate a match where two instances are actually refer to the same individual or concept. Therefore, to evaluate accuracy of concept matching, it is necessary to find out the number of incorrect match generated and correct match missed. This is done by comparing the result generated by concept matcher with a standard result. Then we can use this information to compute precision, and recall F_1 values. The formula to compute each of these values is given as follows:

$$p = \frac{\left| M \bigcap S \right|}{M} \quad r = \frac{\left| M \bigcap S \right|}{S} \quad \mathrm{F}_1 = \frac{2pr}{p+r}$$

In this formula, M indicates the set of alignments discovered by instance matcher; S indicates the set of standard alignments.

Matching between integrated ocean observatory system ontology and marine metadata interoperability ontology

We used our mapping algorithm to map vocabulary terms from IOOS Platform Vocabulary,[3] IOOS Parameter Vocabulary v2.0,[4] MMI Platform Ontology,[5] Climate and Forecast (CF) Standard Names Parameter Vocabulary[6] and DRDC Atlantic NADAS Parameter Codes.[7] We used those vocabularies and ontologies because they are open access on the Internet and we saw similar terms among them through a quick read. Mapping results were compared with reference mappings found at MMI Ontology Registry and Repository.[8] To be specific, we mapped the following 3 pairs of vocabularies.

Table 1 Mapping results compared with reference mappings

	IOOS-MMI	CF-IOOS	DRDC-CF
# Exact matches in reference	22	45	6
# Exact matches in reference SEM+ found	8	22	0

- IOOS Platform Vocabulary and MMI Platform Ontology (IOOS-MMI)
- CF Standard Names Parameter Vocabulary and IOOS Parameter Vocabulary v2.0 (CF-IOOS)
- DRDC Atlantic NADAS Parameter Codes and CF Standard Names Parameter Vocabulary (DRDC-CF)

In the experiment, we always first chose a vocabulary that has fewer terms. Then for each term in that vocabulary, we tried to find exactly one most likely match from the other vocabulary. For example, for CF-IOOS, since CF has 2524 terms while IOOS Parameter Vocabulary v2.0 has only 185 terms, we find one most likely match from the 2524 terms in CF for each of the 185 terms in IOOS Parameter Vocabulary v2.0. We obtained 185 mapping results for this pair of vocabularies, of which we found 22 were tagged as “exact matches” in the reference mappings.

Table 1 summarizes our mapping results in comparison with the reference mappings.

As we looked into the individual mappings to see what was going on, we obtained some really interesting findings. For example, our algorithm found the following match:http://mmisw.org/ont/ioos/platform/aircraft = http://www.cdi.com#Aircraft, thus it did not find the following match because only one top match was picked for each term in IOOS Platform Vocabulary:

http://mmisw.org/ont/ioos/platform/aircraft =
http://mmisw.org/ont/mmi/platform/Aircraft

, which is in the reference mappings. The match can be found by simply changing a parameter to let the algorithm find top 2

[1] http://lucene.apache.org/core/

[2] http://jena.apache.org/

[3] Available at http://mmisw.org/ont/ioos/platform, containing 34 terms.

[4] Available at http://mmisw.org/ont/ioos/parameter, containing 185 terms.

[5] Available at http://mmisw.org/ont/mmi/platform, containing 163 terms.

[6] Available at http://mmisw.org/ont/cf/parameter, containing 2524 terms.

[7] Available at http://mmisw.org/ont/drdc/parameter, containing 41 terms.

[8] http://mmisw.org/orr/#b

Table 2 Highly likely mappings between IOOS vocabularies and other ontologies or vocabularies found by SEM+

http://mmisw.org/ont/ioos/platform/submersible	http://www.cdi.com#Submersible
http://mmisw.org/ont/ioos/platform/glider	http://mmisw.org/ont/mmi/platform/Glider
http://mmisw.org/ont/ioos/platform/balloon	http://www.cdi.com#Balloon
http://mmisw.org/ont/ioos/platform	http://mmisw.org/ont/mmi/platform
http://mmisw.org/ont/ioos/platform/mooring	http://www.cdi.com#Mooring
http://mmisw.org/ont/ioos/platform/aircraft	http://www.cdi.com#Aircraft
http://mmisw.org/ont/ioos/platform/buoy	http://www.cdi.com#Buoy
http://mmisw.org/ont/ioos/parameter/wet_bulb_temperature	http://mmisw.org/ont/cf/parameter/wet_bulb_temperature
http://mmisw.org/ont/ioos/parameter/height	http://mmisw.org/ont/cf/parameter/height
http://mmisw.org/ont/ioos/parameter/air_density	http://mmisw.org/ont/cf/parameter/air_density
http://mmisw.org/ont/ioos/parameter/dew_point_temperature	http://mmisw.org/ont/cf/parameter/dew_point_temperature
http://mmisw.org/ont/ioos/parameter/visibility	http://mmisw.org/ont/cf/parameter/visibility_in_air
http://mmisw.org/ont/ioos/parameter/relative_humidity	http://mmisw.org/ont/cf/parameter/relative_humidity
http://mmisw.org/ont/ioos/parameter/significant_wave_height	http://mmisw.org/ont/cf/parameter/sea_surface_wave_significant_height
http://mmisw.org/ont/ioos/parameter/time	http://mmisw.org/ont/cf/parameter/time
http://mmisw.org/ont/ioos/parameter/precipitation_amount	http://mmisw.org/ont/cf/parameter/precipitation_amount

matches instead of the top 1. After this tuning the number of exact matches found increased from 8 to 13.

Our algorithm failed to find any correct mappings for DRDC-CF. We looked into the matter and found that the problem was related to common short function words used in descriptions, such as *in*, *at*, and *of*. For example, the term <http://mmisw.org/ont/drdc/parameter/_020> in DRDC has the description "Depth in metres at GPS position", so we matched it with <http://mmisw.org/ont/cf/parameter/sea_floor_depth_below_sea_surface> in CF, which has the description "The sea_floor_depth_below_sea_surface is the vertical distance between the sea surface and the seabed as measured at a given point in space including the variance caused by tides and possibly waves" since they share quite some words such as "depth", "in" and "at", and we failed to match it with the correct one <http://mmisw.org/ont/cf/parameter/depth>, whose description "Depth is the vertical distance below the surface" does not share many words and ends up with a slightly lower Jaccard similarity score.

Table 3 Number of computation reduced to for different l_b

l_b	Peop.	Org.	Loc.	Comb.
2	5257	2571	2613	8779
10	39,998	26,747	21,310	75,388
50	259,776	177,396	203,442	567,443
100	591,918	362,984	480,197	1,197,011

Table 4 Recalls on different block size with number of correct pair found in the same block

l_b	Peop.	Org.	Loc.	Comb.
2	0.65(3243)	0.61(1195)	0.64(1241)	0.57(4999)
10	0.92(4596)	0.895(1745)	0.9(1725)	0.87(7687)
50	0.995(4951)	0.97(1892)	0.954(1827)	0.97(8624)
100	0.996(4958)	0.97(1894)	0.96(1838)	0.98(8682)

Nevertheless, SEM+ was able to find the following highly likely mappings which were not captured by the reference mappings (Table 2).

Although we are not in the place of validating these mappings, we argue that they are good suggestions for the domain experts to find missing mappings.

In terms of scalability, due to the lack of the large benchmark dataset in the Earth science domain, we evaluated the blocking algorithm using the OAEI NYTimes to DBpedia instance matching dataset.[9] The question we asked is how well can our blocking algorithm performs in grouping matching entities into a same block. Using this dataset, we evaluated the effect of our blocking algorithm by setting l_b=2, 10, 50, 100. The number of computations reduced to are presented in Table 3 and recalls (*Rec*) are presented in Table 4. Recalls are computed with equation $Rec = \frac{M'}{M}$, where *M'* is the number of matching entity pairs found in the same block and *M* is the total number of matching entity in the standard file.

From Tables 3 and 4, we can see that the blocking algorithm enables us to make trade-offs between computation time and the number of wrongly discarded pairs. For example, using the OAEI People dataset, with l_b=50, we can use 259, 776 comparison computations to achieve 99.5 % of recalls rate, compared to 4979*4977=24,780,483 comparison computation to achieve 100 % of the recall rate, which is about 95 times faster.

[9] http://oaei.ontologymatching.org/2011/instance/index.html

Conclusions

In this paper, we presented SEM+, an automatic concept matching tool with detailed discussion of the blocking algorithm and the IEWS model for similarity computation. Manually creating mappings between two ontologies or sets of vocabularies can be time consuming and is often not scalable given the massive amount of concepts and ontologies created. Therefore, a tool that can automatically suggest possible mappings can significantly reduce the human effort in creating such mapping. We performed the evaluation using Earth science related ontologies. We compared our mapping results with existing manually created mappings and discussed related limitations and advantages of the developed methods. In the future, we are interested in using domain thesauri to improve the accuracy of performed matching. Domain thesauri offer more precise definition of concepts in a context. By using them, the SEM+ will be able to give a more accurate similarity score.

References

Baeza-Yates R, Ribeiro-Neto B (1999) Modern information retrieval. ACM Press, Addison-Wesley

Bechhofer S (2009) OWL: web ontology language. Encyclopedia of Database Sytems. Springer US. 2008–2009. Miles, Alistair, and José R. Pérez-Agüera.

Benjelloun O, Garcia-Mollina H, Menestrina D, Su Q, Whang S, Widom J (2009) Swoosh: a generic approach to entity resolution. VLDB J 18(1):255–276

Berners-Lee T, Hendler J, Lassila O (2001) The semantic web. Sci Am 284(5):28–37

Chaudhuri S, Ganti V, Motwani R (2005) Robust identification of fuzzy duplicates. In Prof. of ICDE, pp. 865–876

Cruz IF, Antonelli FP, Stroe C (2009) Agreementmaker: efficient matching for large real-world schemas and ontologies. PVLDB 2(2):1586–1589

Dong X, Halevy Y, Madhavan J (2005) Reference reconciliation in complex information spaces. In Proc. of SIGMOD, pp. 865–876

Duan S, Fokoue A, Srinivas K, Byrne B (2012) A clustering-based approach to ontology alignment, In Proc. of ISWC

Elfeky M, Elmagarmid A, Verykios V (2005) Tailor: a record linkage tool box. In Proc. of SIGMOD, pp. 85–96

Euzenat J (1994) Brief overview of T-Tree: the TROPES taxonomy building tool, in: 4th ASIS SIG/CR Workshop on Classification Research, Columbus (OH, US), pp. 69–87

Haq BU (ed) (2007) The geological time table, 6th edn. Elsevier, Amsterdam

Jean-Mary Y, Shironoshita E, Kabuka M (2009) Ontology matching with semantic verification. In Proc. of Web Semantics: Science, Services and Agents on the World Wide Web

Jimenez-Ruiz E, Grau B (2012) LogMap: logic-based and scalable ontology matching, In Proc. of ISWC

Klyne G, Carroll JJ (2006) Resource description framework (RDF): concepts and abstract syntax

Lin D (1998) An information-theoretic definition of similarity. In Proc. of 15th International Conference of machine Learning (ICML) pp. 296–304

NERC Vocabulary Server, http://vocab.ndg.nerc.ac.uk/

Newcombe H, Kenedy J (1962) Record linkage: making maximum use of the discriminating power of identifying information. Commun ACM 5(11):563–566

Nguyen K, Ichise R, Le B (2012) SLINT: a schema-independent linked data interlinking system. In Ontology Matching (OM 2012)

Raskin RG, Pan MJ (2005) Knowledge representation in the semantic web for earth and environmental terminology (SWEET). Comput Geosci 31(9):1119–1125

Rong S, Niu X, Xiang E, Wang H, Yang Q, Yu Y (2012) A machine learning approach for instance matching based on similarity metrics. In Proc. Of ISWC

Sarawagi S, Bhamidipaty A (2002) Interactive deduplication using active learning. In Proc. of KDD, pp. 269–278

Shannon C (1948) A mathematical theory of communication. Bell Sys Techn J 27(3):379–423

Shvaiko P, Euzenant J (2013) Ontology matching: state of the art and future challenges. IEEE Trans Knowl Data Eng

SKOS (2007) simple knowledge organisation for the web. Cat Classif Q 43.3–4: 69–83.

Stumme G, Madche A (2011) FCA-Merge: bottom-up merging of ontologies, In the 7th International conference on artificial Intellligence (IJCAI), pp. 225–230

Tang J, Li J, Liang B, Huang X, Li Y, Wang K (2006) Using Bayesian decision for ontology mapping. J Web Semantics Sci, Serv Agents World Wide Web, pp. 243–262

Thor A, Rahm E (2007) Moma – a mapping-based object matching system. In Proc. of CIDR, pp. 247–258

Volz J, Bizer C, Gaedke M, Kobilarov G (2009) Discovering and maintaining links on the web of data. In Proc. of ISWC, pp. 650–665

Walker JD, Geissman JW (2009) 2009 GSA geologic time scale. GSA Today 19(4–5):60–61

Yancey W (2002) Bigmatch: a program for extracting probable matches from a large file for record linkage. Statistical research report series rrc2002/01, U.S. Bureau of Census