On Building and Publishing Linked Open Schema from Social Web Sites

Tianxing Wu[a,*], Haofen Wang[b], Guilin Qi[a], Jiangang Zhu[c], Tong Ruan[d]

[a]*Southeast University, Nanjing, 211189, China*
[b]*Gowild Robotics Co. Ltd, Shenzhen, 518057, China*
[c]*Microsoft, Suzhou, 215123, China*
[d]*East China University of Science & Technology, Shanghai, 200237, China*

Abstract

Schema-level knowledge is important for different semantic applications, such as reasoning, data integration and question answering. Compared with billions of triples describing millions of instances, current Linking Open Data has only a limited number of triples representing schema-level knowledge. To facilitate multilingual schema-level knowledge mining, we propose a general approach to learn Linked Open Schema (LOS) in different languages from social Web sites, which contain rich sources (i.e. taxonomies composed of categories and folksonomies consisting of tags) for mining large-scale schema-level knowledge. The core part of the proposed approach is a semi-supervised learning method integrating rules to capture `equal`, `subClassOf` and `relate` relations among the collected categories and tags. We respectively apply the proposed approach to the selected English social Web sites and the Chinese ones, resulting in an English LOS and a Chinese LOS. We publish the English LOS and the Chinese one as open data on the Web with three access levels, i.e. *data dump*, *lookup service* and *SPARQL endpoint*. Experimental results show the high accuracy of the relations in the English LOS and the Chinese one. Compared with DBpedia, Yago, BabelNet, and Freebase, both the English LOS and the Chinese one not only have large-scale concepts, but also contain the largest number of `subClassOf` relations.

*Corresponding author. Tel: +86-25-52090910
Email addresses: `wutianxing@seu.edu.cn` (Tianxing Wu), `wang_haofen@gowild.cn` (Haofen Wang), `gqi@seu.edu.cn` (Guilin Qi), `jiangazh@microsoft.com` (Jiangang Zhu), `ruantong@ecust.edu.cn` (Tong Ruan)

Keywords: Linked Data, Linked Open Schema, Schema-Level Knowledge, Social Web Sites

1. Introduction

Schema-level knowledge is important for different semantic applications, such as reasoning [1, 2], data integration [3, 4] and question answering [5, 6]. There are over 1,100 datasets within the current Linking Open Data (LOD)[1] Cloud, where the number of triples representing schema-level knowledge is limited when compared with the billions of triples describing millions of instances. For example, this characteristic of data distribution exists in each of the core datasets DBpedia [7], Yago [8] and Freebase [9] in LOD as well as the well-known Chinese LOD Zhishi.me [10]. DBpedia has a small ontology containing 685 classes which form a concept hierarchy and 2,795 properties. Yago links Wikipedia leaf categories to WordNet [11] synsets to build a taxonomy. Although the Yago taxonomy has about 350,000 classes, the number of `subClassOf` relations is relatively sparse. Freebase has a very shallow taxonomy with dozens of domains and hundreds of types. Zhishi.me uses Zhishi.schema [12] to refine its original category system, which results in a taxonomy with only 39,774 `subClassOf` relations.

Current social Web sites contain different kinds of taxonomies (e.g. product catalogues and Web site directories) composed of categories and folksonomies (e.g. the collaborative tagging systems in Instagram[2] and Stackoverflow[3]) consisting of tags, which are rich and important sources for schema-level knowledge mining.

A question here is, why are different kinds of taxonomies and folksonomies important? Actually, they are different local schemata applicable in various scenarios, where even the concepts (denoted by categories and tags) of the same label may have different meanings. For example, when concept "*Sports*" is the child of concept "*Shopping and Service*" in eBay[4], it means sports goods; when "*Sports*" is the child of concept "*Recreation*" in Wikipedia, it represents kinds of physical activities. Since it is unrealistic to design a general schema from scratch for all applications in all areas, more local

[1] `http://linkeddata.org/`
[2] `http://instagram.com/`
[3] `https://stackoverflow.com/`
[4] `https://www.ebay.com/`

schemata such as different taxonomies and folksonomies are required for domain-specific needs.

The idea of learning schema-level knowledge from social Web sites was first discussed in our previous work [12], which aims at building Linked Open Schema (LOS) by automatically discovering different relations among categories in taxonomies and tags in folksonomies, but there exist three main problems in the proposed approach as follows:

- This approach relies on language-specific features and rules, which can only be applied to Chinese social Web sites. This limits the broader use of this approach for constructing the LOS in different languages, which is important to facilitate multilingual schema-level knowledge mining.

- This approach depends on manually labeled data for applying machine learning techniques, causing lots of manual work which should be performed before using this approach.

- This approach separately uses rules or machine learning techniques to learn relations among categories and tags, i.e. it does not consider jointly exploiting the advantages of rules and machine learning techniques to acquire better learning results.

To solve the above problems, we substantially extend our conference papers [12, 13] by designing a new general approach which can be applied to the social Web sites of different languages to learn relations among categories and tags. This approach not only automatically generates labeled data for machine learning, but also encodes rules into the machine learning process to get better learning results. More specifically, we first use a blocking mechanism to reduce the number of concept pairs (each pair consists of two categories, or two tags, or a category and a tag) to be calculated to ensure that our approach can be applied to a large-scale scenario. Then, we present an automatic strategy to generate labeled data from the given concept pairs. After that, we propose a semi-supervised learning method to detect `equal`, `subClassOf` and `relate` relations from concept pairs, and a post-processing step based on generic rules is leveraged to revise the misclassified results in each iteration of the learning process.

After applying this new proposed approach to the English social Web sites and the Chinese ones, we acquire an English LOS and a Chinese LOS. Compared with DBpedia, Yago, BabelNet [14] and Freebase, both the

resulting English LOS and the Chinese one have large-scale concepts and contain the largest number of `subClassOf` relations.

In summary, the main contributions of this work are listed as follows:

- We propose a *general* approach to construct LOS in different languages from social Web sites to facilitate *multilingual schema-level knowledge mining*.
- We *publish the data dump* of constructed LOS as open data for *public access*, including the English and Chinese versions. We not only directly offer the downloads of the dataset, but also provide *Lookup Service* and *SPARQL Endpoint*, which respectively allow querying with concept labels and the SPARQL language for exploring the linked open schema.
- We carry out *a comprehensive set of experiments* to evaluate our approach. Experimental results show that the proposed approach not only harvests the large-scale and high-quality English LOS and Chinese LOS, but also significantly outperforms the designed comparison methods in terms of precision, recall and F1-score.

The rest of the paper is organized as follows. Section 2 gives an overview of our approach. Section 3 describes the technical details. Section 4 shows the experimental results. Section 5 demonstrates the Web access of LOS. Section 6 outlines some related work and finally we conclude the paper and describe the future work in Section 7.

2. Overview

In this section, we give an overview of our approach to building linked open schema in any language. We start with a brief introduction of the problem, and then provide the overall workflow of our proposed approach.

2.1. Problem Defintion

Input: Given a set of social Web sites $WS = \{ws_1, ws_2, \ldots, ws_n\}$ in a certain language, where each Web site ws might contain a set of categories $CA_{ws} = \{ca_1, ca_2, \ldots, ca_m\}$ as well as a set of tags $TA_{ws} = \{ta_1, ta_2, \ldots, ta_o\}$. These categories are organized in a hierarchical way. In a *category hierarchy*, a category might be associated with zero or more parent categories as well as child categories. In Figure 1, we show an example of the categories in Google

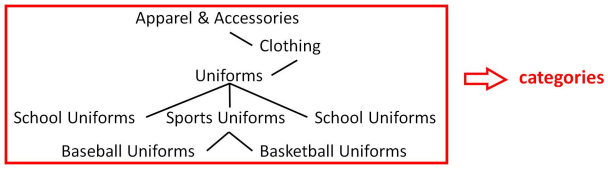


Figure 1: An example of the categories in Google Product Taxonomy

Figure 2: An example of the tags in Stackoverflow

Product Taxonomy[5], in which category "*Uniforms*" has a parent category "*Clothing*" and child categories "*School Uniforms*", "*Sports Uniforms*", and etc. The tags are organized in a flat manner without relying on a controlled vocabulary to annotate resources (e.g. Web pages, images and videos). The flat organization means that tags do not have a previously defined hierarchical structure. Here is an example in Figure 2, tags "*nlp*", "*terminology*", "*semantics*" and "*semantic-web*" form a tag group to annotate a question in Stackoverflow. In this paper, we define that the categories and tags collected from social Web sites denote concepts. A category ca_i is defined to denote a *static concept* as it is predefined by the Web site and cannot be freely modified. A tag ta_j is defined to denote a *dynamic concept* because it is often created on the fly by Web users and can be modified at any time according to their own needs.

Output: We aim at building a linked open schema (in any language) composed of concepts from the input Web sites. The generated linked open schema contains three types of semantic relations, i.e. `equal`, `subClassOf`, and `relate`. We define these relations according to the definitions of `owl:equivalentClass`, `rdfs:subClassOf` and `skos:related`. Two concepts are equal if and only if they contain exactly the same set of instances. One concept is a subclass of another if and only if all the instances of the former one are instances of the latter one. Two concepts are related if there is an associative link between them and they do not have the `equal` or

[5]`https://www.google.com/basepages/producttype/taxonomy.en-US.txt`

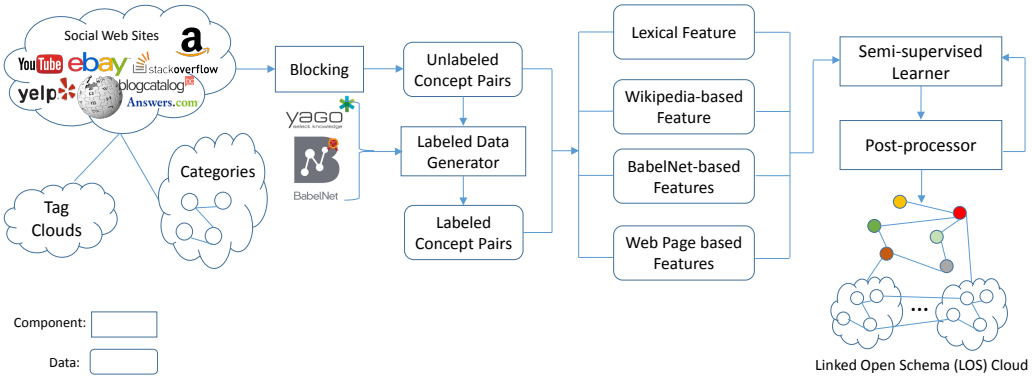


Figure 3: The workflow to generate linked open schema (LOS) cloud

`subClassOf` relation. The `relate` relation is the weakest semantic relation among the three types. The `subClassOf` relations between the concepts from different social Web sites form an integrated concept taxonomy while all the extracted semantic relations build a large semantic network.

2.2. Workflow

We now provide a workflow to explain the whole process and how different components interact with each other. As shown in Figure 3, we have four main components, namely *Blocking*, *Labeled Data Generator*, *Semi-supervised Learner* and *Post-processor*. The input of *Blocking* are the concepts (denoted by categories and tags) collected from different Web sites. *Blocking* divides all concepts into blocks and only forms unlabeled concept pairs within the same block. Compared with the number of 2-combinations from the set of all collected concepts, *Blocking* outputs a relatively small number of concept pairs for further processing, which guarantees the scalability and efficiency of our approach. The *Labeled Data Generator* generates labeled data automatically by using the existing `equal` and `relate` relations in BabelNet and `subClassOf` relations in Yago. Afterwards, we capture the designed language-independent features of the labeled and unlabeled concept pairs, including the lexical feature, Wikipedia-based feature, BabelNet-based features and Web page based features to measure the relatedness between concepts. A *Semi-supervised Learner* is then adapted to discover `equal`, `subClassOf` and `relate` relations. The learned classifier can be updated iteratively by adding new labeled data of high confidence. In each iteration, a *Post-processor* is applied. During the post-processing step, we use two

general and effective rules to filter out those misclassified pairs. Finally we build the linked open schema composed of `equal`, `subClassOf` and `relate` relations between concepts from multiple social Web sites.

3. Approach

3.1. Blocking

It is impossible to enumerate all concept pairs as candidates for semantic relation detection. *Blocking* is to divide concepts into *blocks* where each block only contains similar concepts for further processing. Here, we assume that 1) concepts are similar when they share at least one feature, and 2) there may exist a semantic relation between two concepts only from the same block. A good blocking mechanism should be much cheaper than the subsequent semi-supervised learning method, but it should still guarantee a high recall.

The blocking method consists of the following steps. Firstly, each concept is represented by some representative features. Secondly, if two concepts share one feature, they are put into the same block. Finally, an inverted index is built so that each feature with the associated concepts is a block. We use the sense of the lexical head of a concept to search BabelNet and retrieve all hypernyms and hyponyms as the features of the given concept. The lexical head is a basic concept in linguistics and refers to the word that determines the syntactic category of the given phrase [15], e.g. "*Physicists*" is the head of concept "*Chinese Physicists*". If a head has multiple senses in the given language, we simply union these feature sets as a single set. We do not disambiguate the exact sense of the head because blocking cares more about the recall instead of precision. We can tolerate noises in blocks and use further steps to filter them out.

For instance, there are four concepts w, x, y, and z, the feature set of each concept is as follows.

$$w=\{A\}$$
$$x=\{A, B\}$$
$$y=\{B\}$$
$$z=\{C\}$$

Then we build the following inverted index.

$$A=\{w, x\}$$
$$B=\{x, y\}$$
$$C=\{z\}$$

The generated blocks would be $\{w,x\}$, $\{x,y\}$, and $\{z\}$. After obtaining the generated blocks, we extract all concept pairs from each block. Besides, to remove the incorrect `subClassOf` relations in the existing hierarchies, we extract the concept pairs which already hold the parent-child relations in the existing hierarchies for further detection. For example, given a hierarchy: "*Chinese Physicists* `is a child of` *Physicists*, *Physicists* `is a child of` *Scientists*", we extract additional concept pairs as follows.

$$\{(Chinese\ Physicists, Physicists), (Physicists, Scientists)\}$$

3.2. Generating Labeled Data

Labeled data is essential for the subsequent semi-supervised learning as the training data. Here, we automatically generate labeled data by using Yago and BabelNet. To our knowledge, compared with several large-scale open knowledge bases (i.e. DBpedia, BabelNet and Freebase) in LOD, Yago contains the largest and high-quality taxonomy consisting of `subClassOf` relations. Similarly, BabelNet is the largest online synonym thesaurus and also has large-scale `Semantically_Related` relations which are similarly defined as our `relate` relation. Therefore, we choose these two knowledge bases to help automatically generate labeled data. Here, we randomly selected 200 unlabeled concept pairs, each of which hold the `equal` relations in BabelNet, and 800 unlabeled concept pairs with the `subClassOf` relations in Yago. Additionally, we randomly selected 3,000 concept pairs which hold the `relate` relations in BabelNet, while do not hold the `equal` relations in BabelNet or the `subClassOf` relations in Yago. These 4,000 concept pairs are treated as the labeled data for the subsequent semi-supervised learning.

3.3. Feature Engineering

To measure the relatedness between concepts from different aspects, we define six features which are divided into two groups namely *Basic Features* and *Semantic Features*. *Basic Features* refer to the lexical feature and Wikipedia-based feature. *Semantic Features* include the BabelNet-based features and Web page based features. The features of all concept pairs will be fed to the *Semi-supervised Learner* to perform ternary classification.

3.3.1. Basic Features

a) Lexical Feature: To get the linguistic relatedness between concepts, we use a token-based longest common sub-string asymmetric similarity as

the lexical feature. The label of a concept c is denoted as l_c, and the word sequence of c is $seq(l_c)$. Then the Concept Label Similarity (CLSim) between two concepts c_1 and c_2 is defined as

$$\text{CLSim}(c_1, c_2) = \frac{|LCS(seq(l_{c_1}), seq(l_{c_2}))|}{|seq(l_{c_1})|} \tag{1}$$

where $|.|$ returns the length of a word sequence, and LCS[6] is a function to calculate the longest common sub-string sequence between two concept labels.

b) Wikipedia-based Feature: Inspired by ESA (Explicit Semantic Analysis) [16], we map a concept into several categories in Wikipedia, and then use these categories to represent the concept. The benefits are threefold. First, the concept representation is enriched from its label into a set of categories. Second, the dimension of categories is usually much lower than that of text features so that we avoid curse of dimensionality and enable efficient processing. Third, the categories are of higher quality than texts with less ambiguities.

The ESA vector of a concept c is $ESA_c = \langle wc_1(c),\ wc_2(c),\ ...,\ wc_n(c)\rangle$ where wc_i is a Wikipedia category and $wc_i(c)$ is the corresponding weight which indicates the relevance between the concept c and the Wikipedia category wc_i. Then the ESA Vector Similarity (ESAVSim) between concepts c_1 and c_2 is defined as

$$\text{ESAVSim}(c_1, c_2) = \frac{\sum_{wc} wc(c_1) \cdot wc(c_2)}{\sqrt{\sum_{wc} wc(c_1)^2 \cdot \sum_{wc} wc(c_2)^2}} \tag{2}$$

$\text{ESAVSim}(c_1, c_2)$ is actually the cosine similarity between the ESA vectors of two concepts.

3.3.2. Semantic Features

Since basic features are computed only by the information brought by the concept label, the semantics of the given concept are not considered. Thus, we propose several semantic features which try to capture the semantic relatedness between concepts.

c) BabelNet-based Features: Given two concepts c_1 and c_2, we map them to BabelNet and compute the Wu & Palmer (WUP) similarity [17],

[6] `https://en.wikipedia.org/wiki/Longest_common_substring_problem/`

which can be used to calculate relatedness with the depths of two synsets in the BabelNet taxonomy, along with the depth of the Lowest Common Ancestor (LCA). This similarity between c_1 and c_2 is denoted as WUPSim(c_1, c_2) and computed by

$$\text{WUPSim}(c_1, c_2) = \frac{2 * depth(LCA(c_1, c_2))}{depth(c_1) + depth(c_2)} \tag{3}$$

where $LCA(c_1, c_2)$ denotes the lowest common ancestor of c_1 and c_2 in BabelNet, and *depth* is a function to calculate the depth in BabelNet taxonomy for the corresponding synset of the given concept. Note that a concept may map to different synsets, hence we may get more than one WUP similarity between two given concepts. Here, we only choose the maximum WUP similarity, because it could lower the negative effects of ambiguity for measuring the relatedness between two similar concepts in the same block and guarantee a high recall for mining semantic relations.

In addition, we define another asymmetric similarity feature called Relative Depth Similarity (RDSim) to measure the relative depth difference between concepts. Here, we also choose the same synsets used in the maximum WUP similarity for the given concepts c_1 and c_2. The RDSim between c_1 and c_2 is defined as follows:

$$\text{RDSim}(c_1, c_2) = \frac{depth(LCA(c_1, c_2))}{depth(c_2)} - \frac{depth(LCA(c_1, c_2))}{depth(c_1)} \tag{4}$$

If the depth of c_1 is deeper than that of c_2, it indicates that c_2 is more likely to be a parent concept of c_1. For example, given two concepts "*Product*" and "*Stroller*", their lowest common ancestor is "*Artifact*". The depth of "*Stroller*" is 10 and that of "*Product*" is 8, there exists a possibility that "*Stroller*" is a sub-concept of "*Product*".

d) Web page based Features: The context information such as the parent concept of a static concept (i.e. category) or other concepts in the same tag group of a dynamic concept (i.e. tag) can help reveal the real meaning of each concept. For example, when concept "*Sports*" is the child of concept "*Shopping and Services*" in eBay, it means sports goods; when "*Sports*" is the child of concept "*Recreation*" in Wikipedia, it represents kinds of physical activities. However, since such context information is limited and heterogeneous for static and dynamic concepts, it is hard to get reasonable relatedness between concepts only with this context information. Thus, we

try to acquire more context information by querying the Web with the search engine Google[7].

To accurately get the context information returned by Google for each category, the labels of the given category *ca* and its parent category *pca* are jointly submitted to Google. However, the root categories in different taxonomies do not have parent categories, but they are usually unambiguous, otherwise users will be easily confused when exploring each taxonomy in a top-down manner. Therefore, we simply submit the label of each root category to Google to get its context information. Different from categories, given a tag *ta*, we randomly select one tag *ota* that co-occurs in the same tag group with *ta* and then jointly submit the labels of *ta* and *ota* to Google.

In each of the top 20 returned snippets of Web pages, we extract the words co-occurred with *ca* (*ta*) in the same sentence except *pca* (*ota*), because *pca* (*ota*) is a part of the query, thus it occurs quite a lot of times. After removing the stopwords and the words with frequency less than 3, we adopt TF-IDF (Term Frequency-Inverse Term Frequency) [18] for word weighting. As a result, a concept (i.e. category or tag) c can be denoted as one n-dimension context vector $CV(c) = \langle w_1(c), w_2(c), ..., w_n(c)\rangle$, where the weight of the i-th term $CV(c)_i$ is $w_i(c)$ and n is the number of all the words of all concepts. If a word w does not co-occur with c, the corresponding value in $CV(c)$ is zero. Given two concepts c_1 and c_2, we compute their Context Vector Similarity (CVSim) by

$$\mathrm{CVSim}(c_1, c_2) = \frac{\sum_{i=1}^{n} CV(c_1)_i \cdot CV(c_2)_i}{\sqrt{\sum_{i=1}^{n} CV(c_1)_i^2 \cdot \sum_{i=1}^{n} CV(c_2)_i^2}} \tag{5}$$

where $\mathrm{CVSim}(c_1, c_2)$ is actually the cosine similarity between $CV(c_1)$ and $CV(c_2)$. Besides, c can also be represented by a context set $CS(c) = \{w_1, w_2, ..., w_m\}$, where w_i is the i-th remained word of c and m is the number of all remained words of c. According to this representation of two concepts c_1 and c_2, we further define an asymmetric similarity called Relative Context Set Similarity (RCSSim) to measure the relative coverage difference of Web

[7] `http://www.google.com/`

page context set $CS(c_1)$ and $CS(c_2)$ as follows:

$$\text{RCSSim}(c_1, c_2) = \frac{|CS(c_1) \cap CS(c_2)|}{|CS(c_1)|} - \frac{|CS(c_1) \cap CS(c_2)|}{|CS(c_2)|} \quad (6)$$

This equation is based on a basic assumption that c_1 is a sub-concept of c_2 if most of the returned pages of c_1 are similar to those of c_2 but only a part of the returned pages of c_2 are similar to those of c_1. A similar assumption is first proposed in [19]. Here, we use the context word set extracted from the snippets to denote the returned Web pages of c_1 and c_2.

3.4. Semi-supervised Learning

While we generate labeled data by using Yago and BabelNet automatically, the size of the sample of labeled data (only 4,000) in each language is much smaller than that of the unlabeled concept pairs even after blocking. So a natural idea is to use some kind of semi-supervised learning algorithm to predict new relations in each unlabeled pair.

Although there are many existing algorithms such as label propagation that can be used, we select the simplest and the most efficient one – self-training [20]. In each iteration, self-training accepts the labeled data as training data and learns a classifier. Then the classifier is applied to the unlabeled data and adds concept pairs of high confidence to the labeled data to train a new classifier for the next iteration. The whole process will terminate if the difference between the predicted labels (i.e. `equal` or `subClassOf` or `relate`) of the concept pairs given by classifiers in the two consecutive iterations is smaller than a threshold or the maximal number of iterations is achieved.

Note that we use the Support Vector Machine (SVM) [21] algorithm to train the ternary classifier, which is known as the one of the best single classifiers [22]. Moreover, our approach is slightly different from the standard self-training algorithm. We do not directly add test results of high confidence into the labeled data. We add a post-processing step to filter some misclassified pairs using rules, which are introduced in the next subsection.

3.5. Post Processing

In order to guarantee the quality of the English LOS and the Chinese one derived from the *Semi-supervised Learner*, we integrate a post-processing step with the learning process to filter some misclassified pairs using rules. Two general and effective rules are designed as follows:

Rule 1: Given a concept pair (c_1, c_2), if c_1 and c_2 are of different concept labels, and c_1's the lexical head $h(c_1)$ has the same label with c_2 itself, then c_1 `subClassOf` c_2.

Rule 2: Given a concept pair (c_1, c_2), if c_2 and c_1 already hold a parent-child relation in some social Web site, and they share the same lexical head, then c_1 `subClassOf` c_2.

When these two rules are applied, we need to ignore case and the difference in singular and plural forms for some languages (e.g. English). As general rules, they can adapt to different languages. For example, in English, concept "*Chinese Physicists*" has the lexical head "*Physicists*" which owns the same label with concept "*Physicists*", so we can infer that concept "*Chinese Physicists*" `subClassOf` concept "*Physicists*" according to ***Rule 1***; In Chinese, concept "江苏学校" (school in Jiangsu) and "中国学校" (school in China) share the same lexical head "学校" (school), and they already hold a parent-child relation in Wikipedia, then we infer that concept "江苏学校" (school in Jiangsu) `subClassOf` "中国学校" (school in China) based on ***Rule 2***.

Although these rules are generic to any language, the solutions of deriving the lexical head of each concept in different languages are different. Since we only focus on English and Chinese in this paper, we apply the following strategies for these two languages respectively.

For English: We use the same method used in [23] to find lexical heads of English concepts. We parse the concept labels using the Stanford parser [24], and constrain the output of the head finding algorithm [25] to return each lexical head labeled as either a noun or a 3rd person singular present verb. In addition, we modify the algorithm to return both nouns for NP coordinations as lexical heads (e.g. both "*Buildings*" and "*Infrastructures*" are returned as the lexical heads of concept "*Buildings and Infrastructures in Japan*").

For Chinese: We use the same method used in [26] to find lexical heads of Chinese concepts. We take the last noun as the lexical head after performing POS tagging on each concept with FudanNLP [27]. For example, we parse concept " 中国足球运动员 (Chinese football player)" to get the result "中国 (Chinese)/LOC 足球 (football)/NN 运动员 (player)/NN", then, word "运动员 (player)" is treated as the lexical head.

4. Experiments

4.1. Data Statistics

We selected twenty one popular social Web sites in English and fifty one in Chinese. The data were crawled in September, 2015. The detailed statistics of each site are shown in Table 1 and Table 2. From the table, we listed the site name, its URL, the site type, the category number, the tag number, and the average depth of the category taxonomy. If some site does not contain any category or tag, we used ／ to indicate that the value of that column is missing. Since the semantics of tags are less stable than those of categories within an existing taxonomy, we did not take all tags from these sites to build the linked open schema (LOS), including an English LOS and a Chinese LOS. Instead, we only selected popular tags during August. For English, we collected 242,303 labels in which 229,184 are categories and 13,119 are tags. For Chinese, there are 399,853 labels in which 328,248 are categories and 71,605 are tags.

Table 1: Statistics for 21 popular social Web sites in English

Site	URL	Type	#Category	#Tag	Avg Depth
Amazon	http://www.amazon.com	E-Commerce	3,047	/	2.23
Yahoo Answer	https://answers.yahoo.com	Q&A	977	/	2.54
Youtube	http://www.youtube.com	Video Sharing	127	/	2.50
Wikipedia (EN)	http://en.wikipedia.org	Wiki	103,476	/	5.00
MSN	http://www.msn.com	Portal	75	/	1.90
Foursquare	http://foursquare.com	SNS	360	/	2.75
Ebay	http://www.ebay.com	E-Commerce	10,536	/	4.40
Expedia	http://www.expedia.com	Wiki	46	/	2.00
Answers	http://wiki.answers.com	Q&A	8,535	/	5.77
Thisnext	http://www.thisnext.com	E-Commerce	3,177	/	2.63
Match	http://www.match.com	Dating	211	/	3.00
Yelp	http://www.yelp.com	Customer Review	277	/	2.12
Epinions	http://www.epinions.com	Customer Review	656	/	3.31
Bigboards	http://www.big-boards.com	BBS	771	/	3.92
Slideshare	http://www.slideshare.net	Document Sharing	39	/	1.00
Blogcatalog	http://www.blogcatalog.com	Blog	353	/	2.02
Craiglist	http://www.craigslist.org	Classified	96,443	/	5.00
Groupon	http://www.groupon.com	E-Commerce	73	/	1.75
Zynga	http://zynga.com	Social Gaming	5	/	2.00
Instagram	http://instagram.com/	SNS	/	9,519	/
Stackoverflow	http://stackoverflow.com/	Q&A	/	3,600	/

4.2. Effectiveness Study

In this section, we study the effectiveness of our proposed approach from two different perspectives: 1) analysing the effectiveness of the proposed features and rules in our approach; 2) evaluating the accuracy of the English LOS and the Chinese one obtained by the proposed approach.

Table 2: Statistics for 51 popular social Web sites in Chinese (MPA is short for mobile phone assistant in the 2nd and 3rd row)

Site	URL	Type	#Category	#Tag	Avg Depth
360 MPA	http://sj.360.cn/	App Market	49	/	1.69
91 MPA	http://zs.91.com/	App Market	76	/	1.55
Amazon	http://www.amazon.cn/	E-commerce	3,310	/	3.65
Android Market	http://apk.hiapk.com/	App Market	279	/	2.56
Apple App Store	http://www.apple.com/cn/	App Markets	90	/	1.69
Baidu Baike	http://baike.baidu.com/	Wiki	11,743	/	2.67
Baidu Tieba	http://tieba.baidu.com/	BBS	213	/	1.57
Baidu Wenku	http://wenku.baidu.com/	Document Sharing	298	/	1.87
Baidu Zhidao	http://zhidao.baidu.com/	Q&A	2,117	/	3.24
BaiXing	http://www.baixing.com/	Classified	55,179	/	4.08
DangDang	http://www.dangdang.com/	E-commerce	6,847	/	2.59
DianDian	http://www.diandian.com/	Light Blog	/	8,105	/
DingDing Map	http://www.ddmap.com/	Customer Review	26,993	/	2.50
Docin	http://www.docin.com/	Document Sharing	734	/	1.60
Douban	http://www.douban.com/	Social Network	13,168	/	4.04
FanTong	http://www.fantong.com/	Customer Review	3,842	/	2.61
XianGuo	http://xianguo.com/	RSS	36	/	1.62
GanJi	http://www.ganji.com/	Classified	25,274	/	3.81
Guang	http://guang.com/	Social E-commerce	293	/	2.61
Hudong Baike	http://www.baike.com/	Wiki	32,293	/	5.72
JiangNanQingYuan	http://www.88999.com/	Dating	153	/	2.02
ShiJiJiaYuan	http://www.jiayuan.com/	Dating	77	/	1.83
JingDong	http://www.jd.com/	E-commerce	31,140	/	3.59
KaiXing	http://www.kaixin001.com/	Social Network	124	/	2.45
Lvping	http://www.lvping.com/	Online Travel	40,475	/	3.57
MeiLiShuo	http://www.meilishuo.com/	Social E-commerce	316	/	2.57
Mop	http://www.mop.com/	BBS	22	/	1.55
PPS	http://www.pps.tv/	Video Sharing	288	/	1.50
QieKe	http://www.qieke.com/	LBS	6,224	/	3.51
QiongYou	http://www.qyer.com/	Online Travel	107	7,400	1.68
RenHe	http://www.renhe.cn/	Business Social Network	249	/	2.55
RenRen	http://www.renren.com/	Social Network	118	/	1.98
RenRen Game	http://wan.renren.com/	Social Gaming	43	/	1.70
RenRen XiaoZhan	http://zhan.renren.com/	Light Blog	/	7,038	/
RuoLin	http://www.wealink.com/	Business Social Network	62	/	1.56
Sina iAsk	http://iask.sina.com.cn/	Q&A	5,247	/	3.24
Sina Blog	http://blog.sina.com.cn/	Blog	27	16,190	1.56
Sina Game	http://games.sina.com.cn/	Social Gaming	52	/	1.67
Sina GongXiang	http://ishare.sina.com.cn/	Document Sharing	234	/	1.57
Sina Micro Blog	http://weibo.com/	Microblogging	183	/	2.66
TaoBao	http://www.taobao.com/	E-commerce	1,843	/	3.34
Tencent Blog	http://blog.qq.com/	Blog	23	/	1.65
Tencent Micro Blog	http://t.qq.com/	Microblogging	15	/	1.00
TianYa	http://www.tianya.cn/	BBS	1,706	/	3.18
Tudou	http://www.tudou.com/	Video Sharing	755	/	1.64
TuiTa	http://www.tuita.com/	Light Blog	/	3,135	/
Netease Blog	http://blog.163.com/	Blog	19	/	1.60
Netease Micro Blog	http://t.163.com/	Microblogging	/	29,737	/
Netease Reader	http://yuedu.163.com/	RSS	46	/	1.83
Chinese Wikipedia	http://zh.wikipedia.org/	Wiki	55,122	/	3.71
Youku	http://www.youku.com/	Video Sharing	744	/	1.62

4.2.1. Feature and Rule Contribution Analysis

To analyse the effectiveness of the designed features and rules for predicting semantic relations, we used two groups of the labeled data automatically generated from Yago and BabelNet as the ground truth (introduced in Section 3.2). One is for English and the other is for Chinese. Each group contains 200 concept pairs labeled `equal`, 800 concept pairs labeled `subClassOf` and 3,000 labeled `relate`. We applied three methods

Table 3: Basic vs. Basic+Semantic vs. All (**English**)

Relation	Method	Precision	Recall	F-Measure
`relate`	Basic	0.865	0.883	0.874
	Basic+Semantic	0.878	0.911	0.894
	All	**0.888**	**0.929**	**0.908**
`subClassOf`	Basic	0.654	0.620	0.637
	Basic+Semantic	0.806	0.710	0.755
	All	**0.910**	**0.750**	**0.822**
`equal`	Basic	0.860	0.770	0.813
	Basic+Semantic	0.907	0.806	0.864
	All	**0.930**	**0.935**	**0.932**

Table 4: Basic vs. Basic+Semantic vs. All (**Chinese**)

Relation	Method	Precision	Recall	F-Measure
`relate`	Basic	0.832	0.876	0.853
	Basic+Semantic	0.836	0.886	0.860
	All	**0.882**	**0.922**	**0.902**
`subClassOf`	Basic	0.711	0.590	0.645
	Basic+Semantic	0.769	0.623	0.688
	All	**0.879**	**0.724**	**0.794**
`equal`	Basic	0.876	0.775	0.822
	Basic+Semantic	0.914	0.795	0.850
	All	**0.922**	**0.940**	**0.931**

based on different combinations of features and rules for each language. The first method (denoted as *Basic*) used a SVM classifier with only basic features, i.e. the lexical feature and the Wikipedia-based feature. The second method (i.e. *Basic+Semantic*) utilized a SVM classifier with not only basic features, but also semantic features including BabelNet-based features and Web page based features. The third method (*All, i.e. Basic+Semantic+Rule*) used the second classifier and the rules proposed in the post processing step.

We applied 5-fold cross validation to train the classifiers. Precision, recall, and F-measure are evaluation metrics. As shown in Table 3 and Table 4, the method using the classifier with all features and rules performs best no matter in English or Chinese labeled data. This reflects that our proposed features and rules are quite effective in predicting English and Chinese semantic relations. We can also find that the methods considering rules improve the performance greatly, which indicates the necessity of the simple and general rules in the post-processing step.

4.2.2. Accuracy of Three Semantic Relations in LOS

We ran the proposed iterative semi-supervised learning approach with all features and rules until convergence to build an English LOS and a Chinese LOS, respectively. The English LOS contains 25,474 `equal` relations, 1,047,801 `subClassOf` relations and 1,327,631 `relate` relations. The Chinese LOS contains 11,095 `equal` relations, 947,645 `subClassOf` relations and 217,881 `relate` relations. Since there are no ground truths available, we have to verify these relations manually. Due to the large number of semantic relations, it is impossible to evaluate all of them manually. Therefore, we first randomly selected a subset of relations (called *samples*) which can reflect the distribution of the whole dataset, and then we performed manual labeling to evaluate the correctness of samples. The accuracy assessment on samples are used to approximate the correctness of the English LOS and Chinese LOS.

Four graduate students participated in the labeling process. We provided them three choices namely *agree*, *disagree* and *unknown* to label each sample. After each student labeled all samples, we computed the average accuracy. Finally, similar to Yago, the Wilson interval [28] at $\alpha = 5\%$ was used to generalize our findings on the subset to the whole dataset. Wilson interval [28] is a kind of binomial proportion confidence interval for the probability of success calculated from the outcome of a series of Bernoulli trials. Wilson interval has shown good accuracy even for a small number of trials [28], and here α is the significance level.

After applying the above evaluation strategy on the English LOS, the results are as follows:

- We randomly selected 500 `equal` relations. After labeling, the average number of *agree* votes is 483, and the precision achieves 96.24%±1.62%.
- For the randomly selected 500 `relate` relations, the average number of *agree* votes is 448, and the precision is $89.29\% \pm 2.68\%$.
- We also randomly selected 500 `subClassOf` relations. After labeling, the average number of *agree* votes is 427, and the precision achieves $85.13\% \pm 3.09\%$.

The similar evaluation strategy is applied to the Chinese LOS, and we got the following results.

- We randomly selected 500 `equal` relations. After labeling, the average number of *agree* votes is 474, and the precision achieves 94.45%±1.96%.

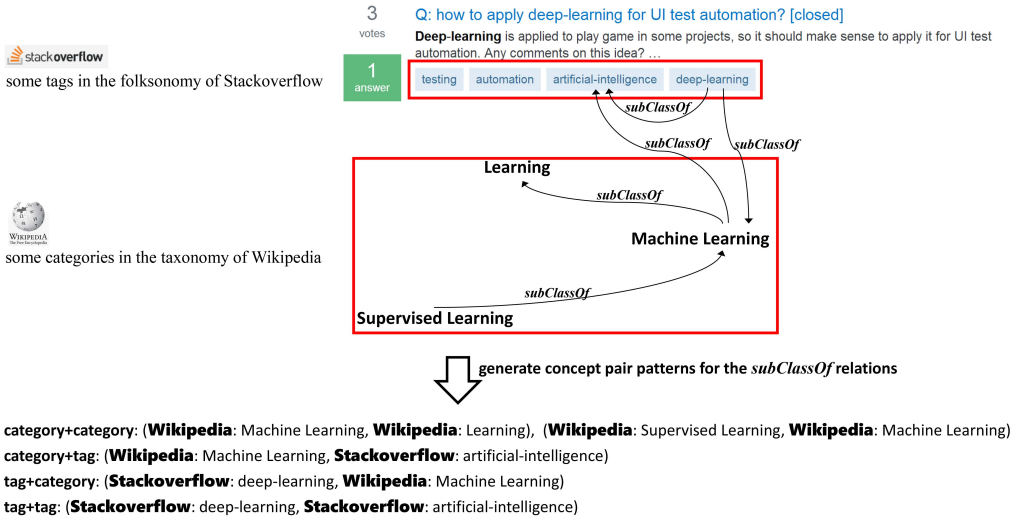


Figure 4: An example of generating concept pair patterns for `subClassOf` relations

- For the randomly selected 500 `relate` relations, the average number of *agree* votes is 461, and the precision is $91.87\% \pm 2.36\%$.
- We also randomly selected 500 `subClassOf` relations. After labeling, the average number of *agree* votes is 440, and the precision achieves $87.71\% \pm 2.85\%$.

4.3. Data Distribution of LOS

Each concept pair may be composed of two categories (denoted as `category+category`), or a category and a tag (denoted as `category+tag`), or two tags (denoted as `tag+tag`). For an asymmetric `subClassOf` relation between a category and a tag, we used `tag+category` to represent that a tag is a sub-concept of a category while `category+tag` to denote that a category is a sub-concept of a tag. Figure 4 gives an example of different concept pair patterns for the given `subClassOf` relations. Table 5 and Table 6 respectively show the concept pair pattern distribution in the three types of English and Chinese semantic relations. From these two tables, it is unsurprising that `category+category` contributes to the largest proportion (more than 0.56) of relations for any kind of semantic relations in both languages, because the number of categories we collected is much larger than the number of tags and categories has better semantic stability compared with tags. Meanwhile,

Table 5: Concept pair pattern distribution in English semantic relations

Concept Pair Pattern	`equal`	`subClassOf`	`relate`
`category+category`	0.564	0.755	0.909
`category+tag`	0.373	0.141	0.071
`tag+category`	/	0.080	/
`tag+tag`	0.063	0.024	0.020

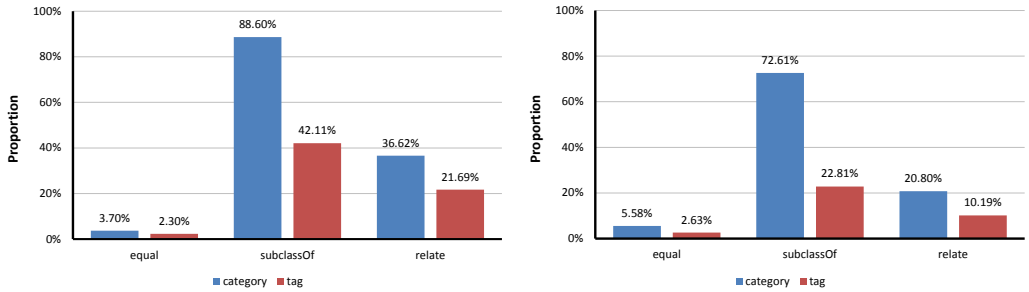
Table 6: Concept pair pattern distribution in Chinese semantic relations

Concept Pair Pattern	`equal`	`subClassOf`	`relate`
`category+category`	0.769	0.865	0.883
`category+tag`	0.201	0.086	0.102
`tag+category`	/	0.041	/
`tag+tag`	0.030	0.008	0.015

the relatively small number of tags can also contribute lots of semantic relations, which do complement the schema-level knowledge only existing among categories.

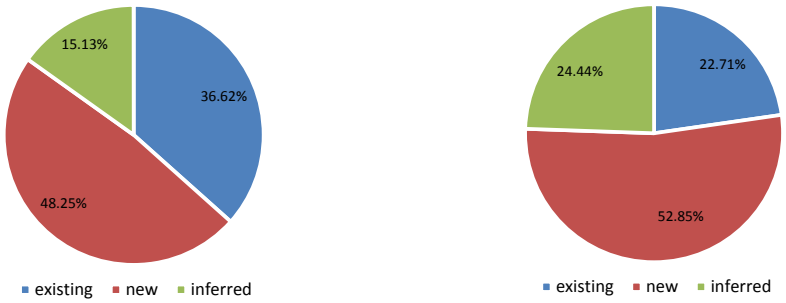
As shown in Figure 5(a) (i.e. in the English LOS), 88.60% categories and 42.11% tags appear in `subClassOf` relations, 36.62% categories and 21.69% tags contribute to `relate` relations, and 3.70% categories and 2.30% tags are for `equal` relations. In Figure 5(b) (i.e. in the Chinese LOS), 72.61% categories and 22.81% tags appear in `subClassOf` relations, 20.80% categories and 10.19% tags contribute to `relate` relations, and 5.58% categories and 2.63% tags are for `equal` relations. The high proportion of `subClassOf` relations among categories (or tags) enables the English LOS or the Chinese LOS to form a large concept taxonomy. The ratio of `equal` relations is pretty low because it is the most strict semantic relation and thus the concept pairs in the same block seldom satisfy such relation.

We also checked the number of `subClassOf` relations already defined in some existing category hierarchies. As shown in Figure 6(a) and Figure 6(b), the proportion of existing English and Chinese `subClassOf` relations are respectively 36.62% and 22.71%. 15.13% English `subClassOf` relations and 22.44% Chinese `subClassOf` relations can be inferred from the existing category hierarchies via intermediate paths. Notice that 48.25% English `subClassOf` relations and 52.85% Chinese `subClassOf` relations are newly discovered, which shows the value of the English LOS and Chinese LOS.



(a) English (b) Chinese

Figure 5: English and Chinese semantic relation proportions



(a) English (b) Chinese

Figure 6: English and Chinese `subClassOf` relation distributions

4.4. Comparison with Other Datasets

4.4.1. Overlap of Concepts and subClassOf Relations

We compared the English LOS and Chinese LOS with other well-known multilingual datasets in LOD, namely DBpedia[8], Yago[9], BabelNet[10] and Freebase[11] in terms of concepts and `subClassOf` relations. Table 7 and Table 8 show the information of concepts and `subClassOf` relations of each dataset in English and Chinese, respectively. They also list the concept overlap and `subClassOf` relation overlap between the English (or Chinese) LOS and the other datasets. As for the concept number, the Chinese LOS is

[8] `http://dbpedia.org/About`
[9] `http://www.mpi-inf.mpg.de/yago-naga/yago/`
[10] `http://babelnet.org/`
[11] `https://www.freebase.com/`

Table 7: Overlap between the English LOS and other datasets

	English LOS	DBpedia	Yago	BabelNet	Freebase
Concept Number	242,303	1,213,462	408,467	1,042,196	133,075
Concept Overlap	/	102,083	47,814	46,034	8,003
`subClassOf` relation Number	**1,047,801**	684	458,242	89,074	23,878
`subClassOf` relation Overlap	/	40	53,168	10,759	70

Table 8: Overlap between the Chinese LOS and other datasets

	Chinese LOS	DBpedia	Yago	BabelNet	Freebase
Concept Number	399,853	142,411	48,621	463,485	2,035
Concept Overlap	/	86,981	24,200	33,714	447
`subClassOf` relation Number	**947,645**	3	40,937	57,407	1,092
`subClassOf` relation Overlap	/	2	8,608	2,612	36

larger than DBpedia, Yago and Freebase, and the English LOS is only larger than Freebase. The concept overlap between the English (or Chinese) LOS and other datasets is small, which shows that there may exist many new concepts in social Web sites and they are good supplements to these existing knowledge bases.

Regarding `subClassOf` relations, both of the English LOS and the Chinese one contain the largest number (at least more than twice as large as the number of `subClassOf` relations in other datasets). When looking at the `subClassOf` relation overlap, we find only small overlaps between the English (or Chinese) LOS and other datasets. Thus, combining the English LOS and the Chinese one with these datasets could form a larger linked open schema.

4.4.2. Overlap of equal Relations with BabelNet

The English LOS contains 25,474 `equal` relations where 14,849 of them represent the same meaning with different labels. The Chinese LOS contains 11,095 `equal` relations where 2,139 of them are of different labels. Since BabelNet is the largest multilingual synonym thesaurus in current LOD, we would like to check how many extracted `equal` relations are covered by BabelNet. For each concept in BabelNet, it is organized in form of a synset in which there are synonyms representing the same concept in different labels or languages. Here, we did not count a `equal` relation when concepts in a pair have the same string. In this way, we get 10,160 English `equal` relations and 1,013 Chinese `equal` relations covered by BabelNet. Due to the small overlap of `equal` relations, how to complement `equal` relations in the English LOS and the Chinese one leveraging BabelNet is worthy to study in the

future. Additionally, we also checked the number of synonymous concept labels between the English LOS and the Chinese one using the multilingual synsets of BabelNet. The number of bilingual synonymous concept labels is 23,204, a small proportion of the concepts in English or Chinese LOS, which indicates that there may exist much room for us to mine the emerging cross-lingual `equal` relations hidden in these concepts within social Web sites.

5. Web Access to Linked Open Schema

Besides publishing the data dump of the English LOS and the Chinese one as open data for public access, we also provide users with two ways for querying, i.e. *Lookup Service* and *SPARQL Endpoint*.

5.1. Linked Data

According to the Linked Data principles[12], LOS creates IRIs for all categories and provides sufficient information when someone looks up an IRI by the HTTP protocol. Since LOS contains concepts of different languages from different sites, we design an IRI pattern to indicate what the language of a concept is, where it comes from and whether it is a category or a tag. The pattern `http://los.linkingopenschema.info/[language]/[site]/[concept type]/[label]` has five parts. `http://los.linkingopenschema.info/` is the namespace. The second part gives the language information of the concept. If it is English (Chinese), the second part is `EN` (`ZH`). The third part tells the provenance of the concept. If it is a tag, the fourth part is `dynamic`. Otherwise, it is `static`. The last part is the concept label.

When publishing LOS, we follow the best practice recipes [29] and try to reuse existing RDF vocabularies which have clear semantics and are widely used. Particularly, we use `rdfs:subClassOf` for `subClassOf` relations, `skos:related` for `relate` relations and `owl:equivalentClass` for `equal` relations. When the Semantic Web agents that accept "application/rdf+xml" content type access our server, resource descriptions in the RDF format will be returned.

5.2. Lookup Service

Lookup Service is provided for users to access LOS. The service is available at `http://los.linkingopenschema.info/LookUp.jsp`. Given a query, all

[12] `http://www.w3.org/DesignIssues/LinkedData.html`

Figure 7: An example page of integrated concepts

concepts whose labels exactly match the query are returned. If two categories are `equal`, they are automatically merged as an integrated view for browsing.

If a user searches for a Chinese concept "净水机" (water purifier), as shown in Figure 7, we return a page integrating two equivalent concepts from two e-commerce Web sites (i.e. JingDong[13] and DangDang[14]). From the page, we can see provenances of two concepts, other equivalent concepts with different labels, their parent concepts, child concepts, related concepts, and links to their original pages in Web sites. These information are organized in the `Resource Site Label`, `EqualClass`, `SuperClass`, `SubClass`, `RelatedClass` and `Link` sections respectively.

We can click on any parent concept or child concept to switch to another page view. Such an interaction stands for navigation in the integrated concept taxonomy of the English (or Chinese) LOS. A click on one related concept or an equivalent concept corresponds to traversal on the semantic network of the English or Chinese LOS.

5.3. SPARQL Endpoint

SPARQL Endpoint is also provided for querying LOS. Professional users can submit customized queries at `http://los.linkingopenschema.info/`

[13] `https://www.jd.com/`
[14] `http://www.dangdang.com/`

SPARQL.jsp. We use AllegroGraph RDFStore[15] as the backend triple store.

6. Related Work

There are two lines of research related to this work. They are ontology learning and ontology alignment, which will be discussed in details respectively.

6.1. Ontology Learning

Ontology learning, especially taxonomic knowledge learning has aroused extensive attention from the research community. Taxonomic knowledge learning can be encyclopedic-based or Web-based. For the encyclopedic-based approaches, they mainly focus on extracting concept hierarchies from Wikipedia. WikiTaxonomy [30] builds a taxonomy from the Wikipedia category system. It contains 105,000 `subclassOf` relations with the accuracy of 88%. Kylin Ontology Generator (KOG) [31] uses Markov Logic Network (MLN) to predict subsumption relations between Wikipedia infobox classes. Yago [8] interlinks Wikipedia categories to WordNet synsets. The integrated taxonomy contains WordNet synsets as hypernyms and Wikipedia categories as hyponyms. There are over 350,000 classes and 450,000 `subClassOf` relations in Yago and the accuracy is estimated to be 96%.

Regarding Web-based approaches, Hearst patterns [32] are widely used. The most recent effort is Microsoft Concept Graph [33, 34]. It builds a large-scale taxonomy which contains over 5 million concepts and 80 million `IsA` relations, but does not distinguish between `subClassOf` relations and `instanceOf` relations. Mianwei Zhou et al. [35] introduced an unsupervised model to automatically derive hierarchical semantics from social annotations. Jie Tang et al. [36] proposed a learning approach to capture the hierarchical semantic structure of folksonomies which are collections of user-defined tags. Huairen Lin et al. [37] described an integrated method for extracting ontological structure from folksonomies that exploits the power of low support association rule mining supplemented by an upper ontology such as WordNet. A recent survey paper [38] compares different approaches of discovering semantics of tags. The main focus of these approaches is to capture the hierarchical semantic structure of folksonomies.

[15] `http://www.franz.com/agraph/allegrograph/`

Our approach is more similar to the Web-based approaches because we detect three types of semantic relations among categories and tags collected from various social Web sites (not limited to Wikipedia only). Furthermore, compared with previous works, we put an emphasis on mining semantic relations between concepts (including categories and tags) and designing a general approach incorporating different features and rules for semi-supervised learning.

6.2. Ontology Alignment

Ontology alignment is very active in the Semantic Web community as well as the database field. In recent years, many alignment tools [39] have been developed. We will introduce some representative examples as follows. Falcon-AO [40] is an automatic ontology matching system leveraging linguistic matching (called LMO) and graph matching (called GMO) to generate reliable alignments between heterogeneous ontologies. However, Falcon-AO only finds equivalence relations. So it cannot be used to tackle our problem. BLOOMS [41] is another ontology alignment tool relying on an external knowledge base such as WordNet or Wikipedia to construct a BLOOMS forest for every class in an ontology. After that, it defines a function to compute the overlap of each tree pair in the two BLOOMS forests. Based on the rate of overlap, BLOOMS determines whether an alignment should be added. Supposing that we use BLOOMS in our scenario, if Wikipedia is used as the knowledge base, when dealing with a large number of concept pairs, BLOOMS will invoke too many calls to Wikipedia, which causes unaffordable time cost. If WordNet is used, too few concepts from different social sites can be aligned because BLOOMS treats each concept label as a whole without analyzing its structure. PARIS [42] aligns not only classes but also relations and instances, it measures the degree of matching based on probability estimates and runs without any parameter tuning. It is noticed that Paris aligns classes based on instance matching, so it will fail in our scenario since we only have concepts without any instances.

In this work, we need to deal with the categories within existing taxonomies and tags without explicit structure, so it is hard to directly use existing ontology alignment tools to learn semantic relations among categories and tags.

7. Conclusions and Future Work

In this paper, we built and published the English linked open schema (LOS) and the Chinese one. We applied a new general semi-supervised learning method integrating rules to the categories and tags collected from different popular English and Chinese social Web sites to detect `equal`, `subClassOf`, and `relate` relations among them. Overall, our proposed approach can be generalized to any language. The semi-supervised learning method and rules are language-independent, but generating the lexical heads used in rules depends on different linguistic characteristics of languages. To transfer our approach to another language, we only need to extract the lexical heads of the categories (or tags) according to the grammar and syntax of that language, and all other parts are the same in any language. The experiments show not only the effectiveness of our proposed approach, but also the high quality of LOS. We also provided several mechanisms for users to access LOS, including the data dump, lookup service and SPARQL endpoint.

As for the future work, in order to avoid the data in LOS becoming outdated, we will update LOS by continuous crawling the most popular social Web sites, and implement new algorithms of semantic relation detection to support incremental updates. We also plan to extract categories and tags from the social Web sites in other languages (e.g. Japanese, Germany and French) to contribute multilingual LOS. Moreover, we will try to mine cross-lingual `equal` relations between concepts in different languages to build a global LOS.

8. Acknowledgements

This work is supported in part by the National Natural Science Foundation of China under Grant No. 61672153 and the 863 Program under Grant No. 2015AA015406. We also thank Qiu Ji and Yuncheng Hua for their valuable feedback and detailed comments during the proofreading process.

References

[1] I. Horrocks, U. Sattler, Ontology reasoning in the SHOQ (D) description logic, in: IJCAI, Vol. 1, 2001, pp. 199–204.

[2] J. Z. Pan, A flexible ontology reasoning architecture for the semantic web, IEEE Transactions on Knowledge and Data Engineering 19 (2).

[3] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, M. Rodriguez-Muro, R. Rosati, M. Ruzzi, D. F. Savo, The mastro system for ontology-based data access, Semantic Web 2 (1) (2011) 43–53.

[4] M. Rodriguez-Muro, R. Kontchakov, M. Zakharyaschev, Ontology-based data access: Ontop of databases, in: ISWC, 2013, pp. 558–573.

[5] V. Lopez, V. Uren, E. Motta, M. Pasin, Aqualog: An ontology-driven question answering system for organizational semantic intranets, Web Semantics: Science, Services and Agents on the World Wide Web 5 (2) (2007) 72–105.

[6] O. Ferrández, R. Izquierdo, S. Ferrández, J. L. Vicedo, Addressing ontology-based question answering with collections of user queries, Information Processing & Management 45 (2) (2009) 175–188.

[7] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, S. Hellmann, DBpedia-a crystallization point for the web of data, Web Semantics: Science, Services and Agents on the World Wide Web 7 (3) (2009) 154–165.

[8] J. Hoffart, F. M. Suchanek, K. Berberich, G. Weikum, Yago2: a spatially and temporally enhanced knowledge base from Wikipedia, Artificial Intelligence 194 (2013) 28–61.

[9] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, J. Taylor, Freebase: a collaboratively created graph database for structuring human knowledge, in: SIGMOD, 2008, pp. 1247–1250.

[10] X. Niu, X. Sun, H. Wang, S. Rong, G. Qi, Y. Yu, Zhishi.me - weaving Chinese linking open data, in: ISWC, 2011, pp. 205–220.

[11] G. A. Miller, Wordnet: a lexical database for English, Communications of the ACM 38 (11) (1995) 39–41.

[12] H. Wang, T. Wu, G. Qi, T. Ruan, On publishing Chinese linked open schema, in: ISWC, 2014, pp. 293–308.

[13] T. Wu, G. Qi, H. Wang, Zhishi. schema explorer: A platform for exploring Chinese linked open schema, in: CSWS, 2014, pp. 174–181.

[14] R. Navigli, S. P. Ponzetto, Babelnet: Building a very large multilingual semantic network, in: ACL, 2010, pp. 216–225.

[15] J. E. Miller, J. Miller, A critical introduction to syntax, A&C Black, 2011.

[16] E. Gabrilovich, S. Markovitch, Computing semantic relatedness using Wikipedia-based explicit semantic analysis, in: IJCAI, Vol. 7, 2007, pp. 1606–1611.

[17] Z. Wu, M. Palmer, Verbs semantics and lexical selection, in: ACL, 1994, pp. 133–138.

[18] R. Baeza-Yates, B. Ribeiro-Neto, Modern information retrieval, Vol. 463, ACM press New York, 1999.

[19] D. Shen, M. Qin, W. Chen, Q. Yang, Z. Chen, Mining web query hierarchies from clickthrough data, in: AAAI, Vol. 7, 2007, pp. 341–346.

[20] K. Nigam, R. Ghani, Analyzing the effectiveness and applicability of co-training, in: CIKM, 2000, pp. 86–93.

[21] V. Vapnik, The nature of statistical learning theory, Springer science & business media, 2013.

[22] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, Do we need hundreds of classifiers to solve real world classification problems, J. Mach. Learn. Res 15 (1) (2014) 3133–3181.

[23] S. P. Ponzetto, M. Strube, Deriving a large scale taxonomy from Wikipedia, in: AAAI, Vol. 7, 2007, pp. 1440–1445.

[24] D. Klein, C. D. Manning, Fast exact inference with a factored model for natural language parsing, in: NIPS, 2002, pp. 3–10.

[25] M. Collins, Head-driven statistical models for natural language parsing, Ph.D. thesis, University of Pennsylvania (1999).

[26] T. Wu, S. Ling, G. Qi, H. Wang, Mining type information from Chinese online encyclopedias, in: JIST, 2014, pp. 213–229.

[27] X. Qiu, Q. Zhang, X. Huang, Fudannlp: A toolkit for Chinese natural language processing, in: ACL, 2013, pp. 49–54.

[28] L. D. Brown, T. T. Cai, A. DasGupta, Interval estimation for a binomial proportion, Statistical Science (2001) 101–117.

[29] D. Berrueta, J. Phipps, A. Miles, T. Baker, R. Swick, Best practice recipes for publishing RDF vocabularies, Working draft, W3C.

[30] S. P. Ponzetto, M. Strube, Wikitaxonomy: A large scale knowledge resource, in: ECAI, Vol. 178, 2008, pp. 751–752.

[31] F. Wu, D. S. Weld, Automatically refining the Wikipedia infobox ontology, in: WWW, 2008, pp. 635–644.

[32] M. A. Hearst, Automatic acquisition of hyponyms from large text corpora, in: COLING, 1992, pp. 539–545.

[33] W. Wu, H. Li, H. Wang, K. Q. Zhu, Probase: A probabilistic taxonomy for text understanding, in: SIGMOD, 2012, pp. 481–492.

[34] Z. Wang, H. Wang, J.-R. Wen, Y. Xiao, An inference approach to basic level of categorization, in: CIKM, 2015, pp. 653–662.

[35] M. Zhou, S. Bao, X. Wu, Y. Yu, An unsupervised model for exploring hierarchical semantics from social annotations, in: ISWC/ASWC, 2007.

[36] J. Tang, H.-f. Leung, Q. Luo, D. Chen, J. Gong, Towards ontology learning from folksonomies, in: IJCAI, Vol. 9, 2009, pp. 2089–2094.

[37] H. Lin, J. Davis, Y. Zhou, An integrated approach to extracting ontological structures from folksonomies, in: ESWC, 2009, pp. 654–668.

[38] A. Garcia-Silva, O. Corcho, H. Alani, A. Gomez-Perez, Review of the state of the art: Discovering and associating semantics to tags in folksonomies, The Knowledge Engineering Review 27 (01) (2012) 57–85.

[39] J. Euzenat, P. Shvaiko, Ontology Matching, Vol. 333, Springer, 2007.

[40] N. Jian, W. Hu, G. Cheng, Y. Qu, Falcon-AO: Aligning ontologies with falcon, in: Proceedings of K-CAP Workshop on Integrating Ontologies, 2005, pp. 85–91.

[41] P. Jain, P. Hitzler, A. P. Sheth, K. Verma, P. Z. Yeh, Ontology alignment for linked open data, in: ISWC, 2010, pp. 402–417.

[42] F. M. Suchanek, S. Abiteboul, P. Senellart, Paris: Probabilistic alignment of relations, instances, and schema, in: VLDB, Vol. 5, VLDB Endowment, 2011, pp. 157–168.