Web Semantics: Science, Services and Agents on the World Wide Web 34 (2015) 1–12

Contents lists available at ScienceDirect

Web Semantics: Science, Services and Agents on the World Wide Web

journal homepage: www.elsevier.com/locate/websem

A bootstrapping approach to entity linkage on the Semantic Web

Wei Hu [a,b,*], Cunxin Jia [b]

[a] National Key Laboratory for Novel Software Technology, Nanjing University, China
[b] Department of Computer Science and Technology, Nanjing University, China

ARTICLE INFO

Article history:
Received 4 December 2014
Received in revised form
14 July 2015
Accepted 23 July 2015
Available online 4 August 2015

Keywords:
Entity linkage
Bootstrapping
Discriminative property
Linked data
Semantic Web

ABSTRACT

In the Big Data era, ever-increasing RDF data have reached a scale in billions of entities and brought challenges to the problem of entity linkage on the Semantic Web. Although millions of entities, typically denoted by URIs, have been explicitly linked with owl:sameAs, potentially coreferent ones are still numerous. Existing automatic approaches address this problem mainly from two perspectives: one is via equivalence reasoning, which infers semantically coreferent entities but probably misses many potentials; the other is by similarity computation between property-values of entities, which is not always accurate and do not scale well. In this paper, we introduce a bootstrapping approach by leveraging these two kinds of methods for entity linkage. Given an entity, our approach first infers a set of semantically coreferent entities. Then, it iteratively expands this entity set using discriminative property-value pairs. The discriminability is learned with a statistical measure, which does not only identify important property-values in the entity set, but also takes matched properties into account. Frequent property combinations are also mined to improve linkage accuracy. We develop an online entity linkage search engine, and show its superior precision and recall by comparing with representative approaches on a large-scale and two benchmark datasets.

1. Introduction

Over the past years, the rapid development of the Semantic Web (SW) impels the proliferation of RDF data. Due to the distributed essence of SW, it frequently happens that many different *entities*, typically denoted by URIs from distributed data sources, refer to the same real-world "thing" (called *coreferent entities* in this paper). Such examples exist in the fields of personal profiling, publication, biomedicine, multimedia, geography, etc.

Entity linkage, also known as coreference resolution or entity matching [1–3], is the process to link different entities that refer to the same resource. Driven by the Linked Open Data (LOD) Initiative, millions of entities have been explicitly linked using owl:sameAs statements. But compared with billions of entities on the current SW, there are still a significant number of entities that potentially refer to the same resource but have not been interlinked yet. For instance, more than 70 entities retrieved in the Falcons search engine [4] appear to refer to Tim Berners-Lee (the inventor of the Web), but merely six have been linked with owl:sameAs. A report about LOD [5] also shows that, out of 31 billion RDF statements less than 500 millions represent links across data sources; most just link to one another.

In the SW area, conventional automatic approaches tackle entity linkage mainly from two perspectives: one is based on *equivalence reasoning* mandated by OWL semantics, e.g., owl:sameAs and other special properties [6,7]; the other is based on the intuition that entities refer to the same resource if they share some *similar* property-values in their descriptions [8,9]. Roughly speaking, the semantics-based approaches can infer explicitly coreferent entities, but they probably miss many potential candidates; while the similarity-based ones are often inaccurate due to heterogeneous ways to describe the same thing, and do not scale well as exhaustively pairwise comparison is needed [10]. Recent work uses *machine learning* to improve linkage performance [5,11,12]. However, it can be time-consuming and labor-intensive to build a large-scale, high-quality training set. Hence, a critical question to entity linkage is: How to combine the two kinds of approaches to bridge the gap between entity links that we already have and potential candidates?

In this paper, we propose a *bootstrapping* approach, called ObjectCoref, by leveraging the semantics-based and similarity-based approaches. Bootstrapping [13] is a technique used to iteratively improve the performance of a classifier and suitable for

* Corresponding author at: National Key Laboratory for Novel Software Technology, Nanjing University, China. Tel.: +86 25 89680923.
E-mail addresses: whu@nju.edu.cn (W. Hu), jiacunxin@smail.nju.edu.cn (C. Jia).

http://dx.doi.org/10.1016/j.websem.2015.07.003
1570-8268/

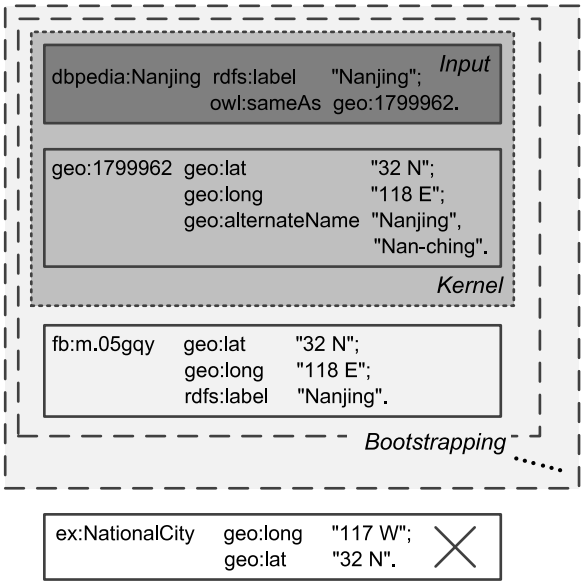

Fig. 1. Running example.

entity linkage, since there are abundant unresolved entities, but the amount of existing links is limited.

Specifically, given an entity, we begin with building a *kernel* automatically, which consists of a set of semantically coreferent entities, via equivalence reasoning on several widely-used OWL/SKOS vocabulary elements. Then, we iteratively expand the kernel using *discriminative* property-value pairs to query new coreferent entities. The discriminability of each property-value pair is learned with a statistical measure, which does not only reveal the importance of this property-value pair for characterizing the coreferent entities, but also considers matched properties by comparing the values of these entities. Moreover, we mine *frequent property combinations* (i.e., the properties often used together) to enhance the property selection criterion in bootstrapping, so that the linkage accuracy can be further improved.

Example 1. To help understanding the bootstrapping process, please consider the four data sources in Fig. 1, each of which contains a candidate entity to be linked. Let us begin with dbpedia:Nanjing. Through owl:sameAs, geo:1799962 can be inferred as coreferent and added into the kernel.

During expansion, (rdfs:label, "Nanjing") and (geo:alternateName, "Nanjing") are learned in the first iteration as two discriminative property-value pairs, in which rdfs:label and geo:alternateName are two matched properties. Consequently, fb:m.05gqy is linked due to holding the same property-value and added in the training set. In the second round, (geo:lat, "32 N") is the most discriminative property-value pair, but it causes a non-coreferent entity ex:NationalCity being incorrectly linked.

If we already mined a frequent property combination {geo:lat, geo:long}, ex:NationalCity would not be selected any more, because the values of geo:long are different ("118 E" versus "117 W"). □

We developed an online, open source entity linkage search engine for ObjectCoref (http://ws.nju.edu.cn/entity-linkage/), and evaluated its performance on a large-scale, real-world dataset from the 2011 Billion Triples Challenge (BTC) and two benchmark datasets provided in the Ontology Alignment Evaluation Initiative (OAEI). Our experimental results show that, by comparing with 11 representative entity linkage approaches, ObjectCoref achieved superior precision and recall on all the three datasets.

The proposed approach substantially improves our previous work [14] in four aspects: (1) We formalize the query-driven entity linkage problem in distributed data sources; (2) We propose a new discriminability measure based on information gain, which incorporates both coreferent and non-coreferent entities to improve linkage accuracy; (3) We comprehensively investigate various vocabularies for kernel generation and refine the filtering rules for frequent property combinations; and (4) We verify our approach on more available datasets and extensively compare it with more representative approaches at larger scale. The newly introduced aspects lead to 7% increase in precision without losing recall.

Also, the difference between the early version of ObjectCoref published in [15] and the approach proposed in this paper is significant. Although the early version used equivalence reasoning to build a kernel, it only expanded the kernel using the same labels without any bootstrapping iterations nor frequent property combinations (see Section 8 for more details).

The rest of this paper is organized as follows. The preliminaries on entity linkage are introduced in Section 2. The bootstrapping framework is shown in Section 3. In Section 4, we introduce the method for building kernels. In Section 5, we describe the bootstrapping algorithm and the discriminability measure. Section 6 presents the discovery of frequent property combinations. The experiments are reported in Section 7 and related work is discussed in Section 8. Finally, we conclude with future work in Section 9.

2. Preliminaries

An *entity* on the SW is typically denoted by a URI and described with a set of properties and values. An *RDF statement* is an ⟨entity, property, value⟩ triple, where the value can be a literal, an entity or a blank node.

The semantics of owl:sameAs dictates that two entities linked with it should be the same, such as ⟨dbpdia:Semantic_Web, owl:sameAs, fb:m.076k0⟩.

Inverse functional property (IFP), e.g., foaf:homepage, is a special kind of property that has high discriminability for equivalence reasoning. The semantics of IFP defines that different entities can be indirectly linked to be the same by holding the same value for that property. IFPs can be inferred in many ways in terms of OWL semantics. As an example, the work in [16] conducted reasoning over pD* that included rules to deal with owl:sameAs, owl:InverseFunctionalProperty, etc. However, anyone can declare anything on the SW, inferring IFPs across sources may bring errors and inconsistency. For instance, it is incorrectly inferred that foaf:name is an IFP in Falcons. The work in [7] addressed the hijacking problem of new ontologies published on the SW redefining the semantics of other existing ontologies. It suggested to use dereferenceable URIs[1] to avoid this problem.

Functional property (FP) is a kind of property that can only have one (unique) value for an entity, which can be used to infer equivalence among entities. Also, only dereferenceable FPs are used. The *cardinality constraint* owl:cardinality (or owl:maxCardinality) is a built-in OWL property that links a restriction class to a data value. The restriction using owl:cardinality (or owl:maxCardinality) constrains a class of all entities that holds exactly (at most) N semantically distinct values for the property, where N is the value of the cardinality constraint. When $N = 1$, its semantics is similar to FP's, but is localized to a particular class.

For example, let us assume that a country is limited to have exactly one capital using FP. For a country like China, if it had two

[1] *URI dereference* is a resource retrieval mechanism that uses any of the Internet protocols (e.g., HTTP) to obtain a representation of the resource it identifies. The representation retrieved by dereferencing a URI is the authoritative definition of that URI [7].