ISSN No. 0976-5697

Volume 3, No. 3, May-June 2012

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Improvement of Supervised Machine Learning Methods in the Web of Linked Data by using Available *Owl:sameAs* Links in the LOD Cloud

Leila Namnik*
Computer engineering dept.
Science and Research Branch, Islamic Azad University
Khuzestan, Iran
Leila.namnik@gmail.com

Mehran Mohsenzadeh
Computer engineering dept.
Science and Research Branch, Islamic Azad University
Tehran, Iran
M.mohsenzadeh@srbiau.ac.ir

Mashalla Abbasi Dezfouli
Computer engineering dept.
Science and Research Branch, Islamic Azad University
Khuzestan, Iran
Masha_abbasi@yahoo.com

Abstract: The web of Linked Data is characterized by linking structured data from different sources using equivalence statements, such as *Owl:sameAs*, as well as other types of properties. Object coreference resolution is to identify "equivalent" URIs that denotes the same object. *Owl:sameAs* links will be established between coreferent URIs, identified by coreference resolution methods. In our previous work, we described an approach for object coreference resolution in the Linked Data environment which relied on standard supervised machine learning methods and support vector machines (SVMs). We proposed to employ different similarity functions and combined them with a learning scheme. In this paper, we extend our previous approach by using existing *Owl:sameAs* links already exist in the web of Linked Data by Linking Open Dataset (LOD) project. By using these links, we could substitute the process of manually labeling training examples in the learning model with an automatic one. We evaluate our approach on common datasets and obtain encourage results, that offer performance comparable with state-of-the-art non learning based systems on these datasets.

Keywords: Coreference Resolution, Data Interlinking, *Owl:sameAs*, Linked Data, Semantic Web, SVM.

I. INTRODUCTION

Linked Data is a set of standards and practices for publishing and interlinking structured data on the web [1]. It can be seen as an approach to data integration at web scale. Linked Data is built upon two simple ideas [2]: Employs the RDF data model to publish structured data on the web and to set explicit RDF links between entities within different data sources. The most important types of RDF links are 'identity links'. These links connect individuals which refer to the same real world entities with different URIs (also known as coreferent URIs). By common agreement [1], web of Linked Data uses the *Owl:sameAs* statement to state the identity links. An *Owl:sameAs* statement is an RDF triple which connects two RDF resources by mean of an *Owl:sameAs* predicate [1]. In the web of Linked Data which forms by Linking Open Data project [1], an increasing number of *Owl:sameAs* links have been published between equivalent RDF resources.

Many researchers have focused on exploring coreferent URIs between local and pair wise data resources [3,4,5,6,7]. In our previous work [8], we proposed a new method for coreference resolution in the Linked Data environment based on supervised machine learning. In this paper, we extend our previous work and present an improvement machine learning based approach for coreference resolution which learn from a small corpus, generated automatically from available *Owl:sameAs* links in the web of Linked Data. The remainder of this paper is organized as follows. Related work is discussed in Section II. In section III we present the process of our proposed method for coreference resolution. The experimental results on a real-world datasets are reported in Section IV. Finally, Section V concludes this paper and gives future work.

II. RELATED WORK

The problem of coreference resolution was originally studied in the database community where it is known as record linkage or object identification [9]. With the development of the Linked Data initiative, it gains importance in the Semantic Web community where it is studied under the names of coreference resolution [10], reference reconciliation [11], and link discovery [12]. Current frameworks for link discovery in Linked Data can be subdivided into two categories: domain-specific and universal frameworks. Domain-specific link discovery framework aim at discovering links between knowledge bases from a particular domain. For example, the RKB knowledge base uses URI lists to compute links between

[1] http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData

universities and conferences [13]. For each new mapping task, a new program has to be written, wherein the source, target and mapping function must be declared. Another domain-specific tool is GNAT [6], which was developed for the music domain. GNAT uses similarity propagation on audio fingerprinting to discover links between music data sets. Universal link discovery frameworks are designed to carry out mapping tasks independently from the domain of the source and target knowledge bases. For example, RDF-AI [7] concentrates on the data-level issues which occur when combining datasets using the same schema.

The algorithm builds on string (Monge-Elkan) and linguistic (Word-Net) similarity measures to calculate similarities between literal property values, and then invokes an iterative graph matching algorithm to calculate a distance between individuals. While RDF-AI considered datasets described under a unique ontology, the KNOFUSS architecture [14] tackles the data interlinking problem whether or not the datasets are described under the same ontology. It is based on a generic component-based approach allow to select the best appropriate method for a given interlinking task. Silk [15] provides a flexible, declarative language for specifying matching heuristics. SILK employs different string based distances. Parameters such as threshold and aggregation mechanisms for specific datasets have to be manually defined by the user. As such, it has the limitations; in particular, it ignores relevant types of evidence: the structure of the semantic data graph and knowledge defined in the ontology. Manually coming up with logic for combining similarity scores is difficult; we have used a learning based approach in this paper.

III. PROPOSED METHOD

Our previous work [8] discussed an approach for coreference resolution in the Linked Data environment. First, we have supposed that input RDF datasets are structured based on a common ontology. We have designed the procedure of our previous approach as the following steps:

A. Splitting Primary RDF Datasets by Ontology-Level Features
B. Selecting Suitable Properties
C. Applying String Similarity Measures
D. Constructing Feature Vectors
E. Learning Resource Coreferencing

In step A, We defined the usage of schema-level features (ontology classes) as a heuristic to reducing the scale of datasets. Primary RDF datasets divided to secondary RDF triple stores via SPARQL queries based on different ontology classes. In step B, first, we extracted all of the properties for each class and then we have chosen the subset of properties based on more informative properties. This selection has done by considering input datasets characteristics and problem solving domain. In step C, we used several well-known approximate string similarity metrics for computing textual similarity between the values of properties, selected in step B. A Token-Based and a Character-Based matcher have been designed in this step for this purpose. In step D, we constructed Feature Vectors for each resource pair in RDF triple store i and RDF triple store j. Therefore we had a k*d-dimensional vector for each resource pair. Each dimension shows approximate textual similarity score between values of property P_i of resources of type C_i, have been calculated by i_{th} similarity metric in the matchers. In machine learning terminology, these feature vectors forms the basis for classifying the resource pair as a “coreferent” or “non-coreferent”.

In step E, for tackling the issue of integrating different measurements, we employ a machine learning approach based on an SVM binary classifier. This binary classifier acted as a parsing function, taking a resource pair as input and generating decision value as output. If it generated positive values, the two input resources are regarded as “coreferent”; otherwise, “non-coreferent”. Then *Owl:sameAs* links can be generated between coreferent pairs of resources. In order to train the SVM classifier, we labeled manually a set of coreferent resource pairs with positive labels and a set of non-coreferent resource pairs with negative labels.

In coreference resolution problem, supervised machine learning methods are the most important solutions. Although these methods can be mapped to wide range of domains, they need to sufficient training examples to learn a decision model. Obtaining these examples is often not easy. The supervised learning system mentioned in our previous paper, relies on standard machine learning. We manually labeled examples as “coreferent” or “non-coreferent” for training the model. We proposed an approach in this paper which utilizing specific features of the web of Linked Data. In particular, large volumes of *Owl:sameAs* links are available, which makes it possible to learn and exploit data patterns not represented explicitly in the ontologies. We use these links in our coreference resolution method as positive examples in our SVM-Based training model. We substitute the process of manually labeling examples in training set with an automatic one. The procedure of our approach is as follows:

A. Load two input datasets into a local relational database
B. Convert selecting properties and values for each resource into table rows
C. Perform a join on the equivalence property (Owl:sameAs) to create resource pairs

In our approach, we use a local relational database in order to process datasets more efficiently. We load datasets into this database. Specifically, each resource in the dataset is represented as a row in a table, each property occurring in the ontology is a column, and the resource URI is the primary key. In cases of multi-valued properties, the row is replicated in such a way that each cell contains a single value but the number of rows equals the number of multiple values. Each new row however, is still identified with the same URI, thus retaining the number of distinct resources. In general, the total number of rows for each resource is the product of cardinalities of the value sets for each of its properties.

We then perform a join on the *Owl:sameAs* property (from input dataset 1 to input dataset 2) such that we get a combination of resources from both datasets. Therefore we can create some resource pairs. We label these resource pairs as positive examples. In our SVM-Based classifier, we created all possible resource pairs as Feature Vectors. With this approach mentioned here, we can label some of these Feature Vectors as positive examples.

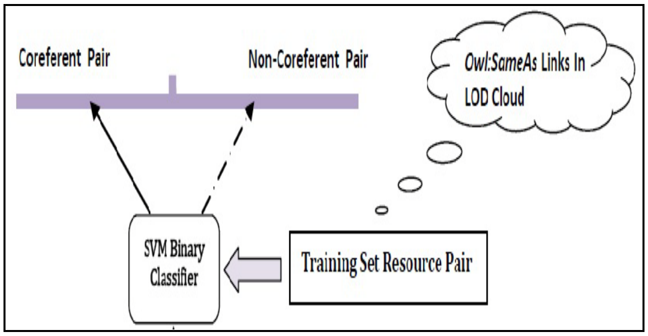

Figure1. Using the existing *Owl:sameAs* links in LOD in our approach

IV. IMPLEMENTATION AND RESULTS

A. Datasets:

We ran our coreference resolution method for Rexa and AKT EPrints dataset pair from the domain of scientific publications. Our datasets were structured according to the SWETO-DBLP [2] ontology, which extends the FOAF ontology [3] , and contained instances of three types*: foaf:Person, opus:Article and opus:Article_in_Proceedings.* The last two, are subclasses of the class *opus:Publication.*

AKT EPrints archive [4] : This dataset contains information about papers produced within the AKT research project.

Rexa dataset [5]: This dataset extracted from the Rexa search server, which was constructed in the University of Massachusetts using automatic IE (Information Extraction) algorithms.

B. Experimental Methodology and Results:

We used the *LIBSVM* [6][16], a good implementation of the SVM classifier, for learning whit radial basis function as kernel function. There are two parameters for an RBF kernel: C and gamma. Best parameter selection performed using the grid-search and cross-validation. We used 5-fold cross-validation for Article_In_Proceeding class and 4-fold cross-validation for Article class. For V-fold cross-validation, we first divided the training set into v subsets of equal size. Sequentially one subset is tested using the classifier trained on the remaining v-1 subsets. With cross-validation, each instance of the whole training set is predicted once so the cross-validation accuracy is the percentage of data which are correctly classified. Traditionally, the quality of the coreferencing output is evaluated by comparing it with the set of true corefrencing and calculating the precision and recall metrics. Our method would allow estimate the quality of a set of mappings without possessing labeled data or involving the user. Under these conditions, it is not possible to calculate the precision and recall. Therefore, the results are reported as accuracy measure:

ACCURACY = (TP + TN) / (TP + TN + FP + FN)

TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative

Table I. Results of our approach

Measure	**Accuracy on class Article**	**Accuracy on clas Article_In_Proceedings**
Our Method	C-V[7] Accuracy : 78.21% Classification Accuracy 88.56%	C-V Accuracy : 85.4% Classification Accuracy:92.2%
Our Previous Wor [8]	C-V Accuracy : 81.82% Classification Accuracy 95.45%	C-V Accuracy : 95.2% Classification Accuracy:93.87%

Table 2. Results of previous methods

Measure Method	**Precision**
KNOFUSS [14]	0.92
RDF-AI [7]	0.95

V. CONCLUSION AND FUTURE WORKS

In this paper, we discussed the problem of coreference resolution in the linked data environment. A new method, based on supervised learning has been developed to address this issue. In our method, a binary classifier based on SVM, trained to classify resource pairs as coreferent or non coreferent. In order to train a model, sufficient training examples are needed. We used the existing *Owl:sameAs* links as positive examples in training set. We could get a performance near to our previous work without needing to manually constructing the training set. Although we assumed that input datasets have been described with a common ontology, our algorithm is flexible to address the issue of different ontologies. If ontologies differ, first an automatic schema matching systems have been used and the same procedure be done for mapped class resulted by schema matching tools. So the future work of this study includes this state. Another area for future work lies in

[2] http://lsdis.cs.uga.edu/projects/semdis/swetodblp/august2007/opusaugust2007.rdf
[3] http://xmlns.com/foaf/spec/
[4] http://eprints.aktors.org/
[5] http://www.rexa.info/
[6] Software available at : http://www.csie.ntu.edu.tw/~cjlin/libsvm

[7] Cross-Validation

applying the method on larger datasets. In this state, we must use the clustering methods.

VI. REFERENCES

[1]. C. Bizer, T. Heath, T. Berners-Lee. Linked Data : The story so far. 2009, International Journal on Semantic Web and Information Systems (IJSWIS), 2009, 5(3):1-22.

[2]. T. Berners-Lee. Linked Data - Design Issues. http://www.w3.org/DesignIssues/LinkedData.html

[3]. A. Nikoliv, M. Aquin, E. Motta. Unsupervised data linking using a genetic algorithm. Technical Report kmi, 2011.

[4]. W. Hu, J. Chen, Y. Qu. A Self-Training Approach for Resolving Object Coreference on the Semantic Web. Proceedings WWW 2011, 87-96.

[5]. J. Volz, C. Bizer, M. Gaedke, and G. Kobilarov. Discovering and maintaining links on the web of data. ISWC 2009.

[6]. Y. Raimond, CH. Sutton, M. Sandler. Automatic Interlinking of Music Datasets on the Semantic Web. Linked Data on the Web workshop, 2008.

[7]. F. Scharffe, Y. Liu, C. Zhou. RDF-AI: an architecture for RDF datasets matching, fusion and interlink. Proceedings of workshop on Identity, reference, and knowledge representation (IJCAI), 2009.

[8]. L. Namnik, M. Mohsenzadeh, M. Abbasi Dezfouli. A new method for coreference resolution in web of Linked Data based on machine learning. International Journal of Advanced research in Computer Sciences,vol 3, No 1, 2012.

[9]. A.K. Elmagarmid, P.G. Ipeirotis and V.S. Verykios. Duplicate record detection: A survey. 2007, IEEE Transactions on Knowledge and Data Engineering, 19(1):1-16.

[10]. W. Hu, J. Chen, Y. Qu. A Self-Training Approach for Resolving Object Coreference on the Semantic Web. Proceedings WWW 2011, 87-96.

[11]. X. Dong, A. Halevy, J. Madhavan. Reference reconciliation in complex information spaces. 2005, ACM SIGMOD international conference on Management of data.

[12]. J. Volz, C. Bizer, M. Gaedke, and G. Kobilarov. Discovering and maintaining links on the web of data. ISWC 2009.

[13]. H. Glaser, I.C. Millard, W.K. Sung, S. Lee, P. Kim, B.J. You. Research on linked data and coreference resolution. Technical report, University of Southampton, 2009.

[14]. A. Nikolov, V. Uren, E. Motta, A. de Roeck. Handling instance coreferencing in the knofuss architecture. 5th European Semantic Web Conference (ESWC 2008).

[15]. J. Volz, Ch. Bizer, M. Gaedke, G. Kobilarov. Silk – a link discovery framework for the web of data. Workshop on Linked Data on the Web (LDOW 2009), 18th International World Wide Web Conference (WWW2009), 2009.

[16]. C.C. Chang, C.J. Lin. LIBSVM: a library for support vector machines, 2001.