The Linked Construction of Educational Resources Based on the Built-in Text Matching

Lei Huang [1, a], Chanle Wu [2, b]

[1] Computer School of Wuhan University ,Wuhan University,Wuhan, 430072,China

[2] Computer School of Wuhan University ,Wuhan University,Wuhan, 430072,China

[a]email: llei_h@163.com, [b]email: wuchlqq@qq.com

Keywords: Linked data,RDF,Knowledge Ontology,Non-mapping Linking;

Abstract. The continuous development of the Linked Data Web depends on the advancement of the relationships extraction mechanisms. This is of particular interest for the linked constructions, where currently most of the data sets are being created manually. In this article, we present built-in text matching table that enables the automatic extraction of non-mapping links between entities. (i.e., concepts, attributes and attribute values, etc.) from resourced. The experimental evaluation shows that our solution handles very well any type of Linked Course Data and improves the average extraction performance of the state of the art with around 4%, in addition to showing an increased versatility. Finally, we propose a flexible Linked Course Data -driven mechanism to be used both for refining and linking the non-mapping links between entities .

Introduction

The most important characteristics of the linked data is to set linked relationships between the same entity object in different data sets or between entities with reasoning relationships (such as owl: of sameAs). And it can use these relationships to achieve the discovery and recognition of therelevant information objects and to provide integrated services. So that people can share structured data on the World Wide Web as conveniently as sharing files. The realization of Data World Wide Web enables people to see the potential of the data open andthe universal link.

Linked Course Data

Linked Course Data (LCD) publishes course data with linked data.Course data in this paper is mainly extracted from the various courses' resource documents in the current national quality courses website. Linked data method means that various courses documents are converted to RDF data and linked with other network related data or knowledge ontology. Furthermore, the documents are processed and organized into Linked Course Data and published asthe Linked Open Data.

This article constructs Linked Course Data on the basis of the National 863 Project "research and development of electronic learning system platform and educational resources library" and the learning object base and knowledge ontology base constructed from national quality courses and national excellent resource sharing lessons"microcomputer system and interface technology". Currently, our LCD includes the Linked Course Data of three courses:Microcomputer System and Interface Technology, Computer Architecture and Principles of Computer Composition. So that a Linked Course Data set which is based on three computer courses' data is formed.

First, we should convert teaching resources documents in order to get all types of teaching resources, including text, tables, static imageand dynamic image. Then we pre-process them and convert them into RDF triples. Knowledge items are mainly developed by experts in the field. Second, we should process, standardize and mapping link the RDF data and construct links between the RDF data. Then the Linked Course Data is formed. On this basis, we can describe it with the OWL ontology language, so that the linked data evolves into knowledge ontology. Linked Course Data can also link with other knowledge data sets. The nature of the Linked Course Data set is a semantic relationship between knowledge points. Knowledgepoint is the basic unit of the

composition of the course knowledge system. Knowledgepoints are arranged in different chapters according to the knowledge system structure. A course is divided into several chapters; a chapter is divided into several sections; each section contains certain number of knowledge points.That forms a hierarchical structure of Linked Course Data. Some of the linked data concepts, which are both inside and outside the course, have semantic relations. The various relationships make Linked Course Data a mesh structure, as shown in Figure 1.

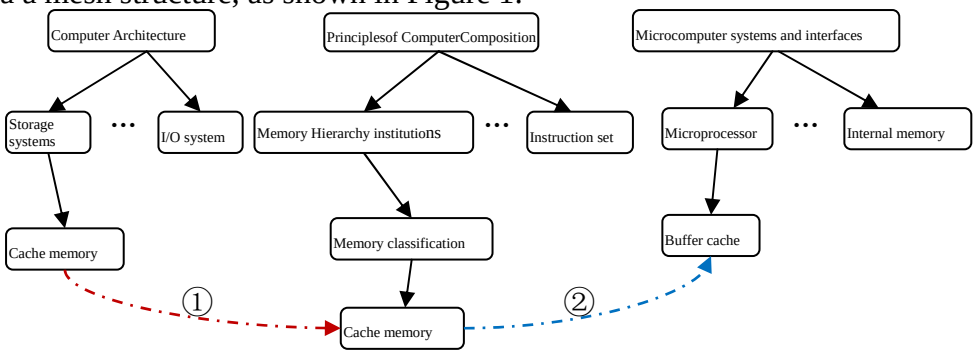

Figure1 The structure of Linked Course Data

The nature of the Linked Course Data set is a semantic relationship between knowledge points. Knowledgepoint is the basic unit of the composition of the course knowledge system. Knowledgepoints are arranged in different chapters according to the knowledge system structure. The various relationships make Linked Course Data a mesh structure , as shown in Figure 1.

The way of the construction of links

The basic idea of linked data is that: identifying things with URI, locating and searching things with URL; everything on the World Wide Web is collectively referred to as resources; describing the resources as well as the relationships between resources with concepts, attributes and attribute values. RDF describesthe network information resources witha three-groupmodel of subject, predicateand object. The subject is the URI which identifies and describes resources. The object is an attribute value of the subject or a URI in another resource relative to the subject. The predicate shows the relationship between subjects and objects. Each entity shows its attributes and the relationships with other entities with a number of triples. Generally, the construction of links is divided into two cases. One is mapping link and the other is non-mapping link.

Two different RDF data sets are referred to as the data set D1 and the data sets D2 respectively.We define certain entities E1 and E2. If we can determine that entities E1 and E2 identify the same object, we are able to establish the link between E1 and E2. A variety of applications and queries can refer to statements of the same thing in different datasets so as to get more information.The data sets D1 and D2 extend and value-add by means of suchlink.Suchlink is called mapping. Take course data as the example. Every course in the quality courses can be regarded as a sub-data set. There is resource R1 in the sub-data set of the computer architecture, labeled “cache memory”. The URI is http://myexample.org/txjg/resource/83 which can be described by RDF triple as: {“http://myexample.org/txjg/resource/83”,owl: label,“ cache memory”}. Taking “cache memory” as the object, we search the sub-data sets of Principles of Computer Composition and find resource R2. The corresponding entity name is “cache memory” which can be described by RDF triple as: {http://myexample.org/zcyl / resource/104, rdfs: name, “cache memory”}. At this time, according to the mapping of the object, we can recognizethe correlation of R1 and R2 which is denoted by {R1, owl: sameAs, R2}, as shown in Figure 1, ①.

In addition to this simple method based on text matching, we can construct a more effective link by means of increasing certain limited conditions or computing the similarity of the text. When the similarity exceeds a certain threshold, we can determine the presence of the associated. However, these methods are based on the mapping of the text. Many relative entities cannot construct links with these methods. That is, cannot be expressed byowl:sameAs.

In the same data set or between different RDF datasets,resource description objects E1 and E2

exist some non - owl:sameAs links. This kind of link is widespread: the same entity can be described by multiple languages, as a book with multilingual versions; or the same language has different names for the same entity, such as potato and pratie; entities have different carriers, as books may exist in the form of electronic documents, pictures, or Audiobooks; entities can update and transform, as books can be reprinted and adapted.Founding, recognizing and explicitly constructing these relationships is a valuable research topic, but it is also quite complicated. The premise is to obtain relevant rules or data models. The linked course data in this article also has such non-mapping link. Just as computer,microcomputer, PC, Personal Compute are the same entity, simple text mapping is difficult to establish the link. In Figure 1, “cache memory” and “buffer cache”are the same entity, but means of text mapping or text similarity computing cannot established the link.

The construction of the built-in text-matching table

To solve this problem, this paper proposes to set a text-matching table in the course datasets and include interrelated objects in the table, so that to create links quickly and accurately. In view that the course data set is a relatively standardized dataset and most objects in it are knowledge point names or scientific terms, it is less likely to cause ambiguity and can make the links more accurate which are constructed by this text-matching table.

The text-matching table can be constructed by the following three steps.

1) Data acquisition. We mainly use artificial ways here to put interrelated objects into the same set (refer to the same object, but expressed in different ways). By data acquisition, multiple object string sets generate and are denoted by S. Si is the string set of the i-th object.

2) Building the data model. This article uses string matrix, referred to as M (i, j), to store the interrelated object strings.Each line of the matrix is used to store an object, that is, to store the string of Si inthe i-th row. The number of rows of the matrix is equal to i. The number of columns of the matrix is referred to as j. J equals to the number of elements in the largest set Sk plus 1, i.e. j = size (Sk).

3) The storage mode. In order to improve the query efficiency, we make corresponding process before storing the object string. We store the string by row, and store the number of the string of each line in the first element, i.e. size (Si). Each line uses Unicode encoding sequence to sort strings. Strings are stored in ascending order. See the specific in Algorithm 1.

Algorithm 1 string object strings in the matrix

1) S is the object string set, M is the m*n string matrix
2) for i=1 to m do
3) ci=size(Si) // Calculating the number of elements of the set Si.
4) Store ci in the first element of thei-th row of the matrix.
5) Strings in Si are sorted by the Unicode encoding and stored in this row from the second element of thei-th row.
6) end for

…		…		…
7	PC	personal computer	computer	…
…		…		…
5	cache	Cache memory	buffer cache	…
…		…		…

Figure 2. the text-matching matrix

After these three steps, the text matching matrix is set up, as shown in Figure 2.

After creating the text-matching matrix, we can use it to map and to build links between objects. The mapping process is as follows: determine an entity object E1 in local data sets D1 and take keyword X1as the object name or label.Determine an entity object E2 in local data sets D2 and take keyword X2 as the object name or label. First search the text matching matrix, see if we can find the string matching with X1.ifwe cannot find, the algorithm terminates. Otherwise, search all the strings in the row with keywordto see if any can match with X2. If so, construct the link. Specific algorithm is in Algorithm 2.

By searching and matching the Principles of Computer Organization sub-data sets and computer system interface sub datasets in quality courses data, we can create the link between “cache memory” and “buffer cache”, as shown in Figure 1 ②.

Algorithm 2 the mapping algorithm based on the built-in text

1) M is the m*n text-matching matrix
2) for i=1 to m do
3) M [i, 1] is the initial length of the binary search
4) In the i-th row, from the 2th to the M [i, 1]-th element, us binary search to search if X1 match with a string
5) If so, let k = the column number of the string.Terminate algorithm 2.Call algorithm 3.
6) Otherwise, i++
7) End for

Algorithm 3 Determine whether the target object

1) According to the Unicode encoding rules, compare X1 and X2,
2) If X2>X1,for k=k+1 to n do
3) Calculated if X2 matches with M (i, j). If it matches, establish the link of the E1 and E2
4) Endfor
5) Otherwise, for k=k-1 to 2 do
6) Calculated if X2 matches with M (i, j). If it matches, establish the link of the E1 and E2
7) Endfor

Compared the storage mode in this paper to the ordinary storage mode, the former one reduces the number of visits and string matching, therefore improving the query efficiency.

Each line of the matrix stores strings in ascending order and search the elements by binary search. Compared with the common storage mode, it reduces the complexity of the algorithm from O (n2) to O (n * log (n)).

Since all object names with the same concept are stored in each row of the matrix by Unicode encoding sequence, after finding the local object attribute X1 in the matrix, the searching and matching number of the target object attribute X2 halves.

Epilogue

In summary, the method of constructing links which is based on text mapping and the text similarity calculating is a basic method. Although having a strong versatility, it sometimes misses some meaningful links. Taking Linked Course Data as the example, this paper proposes to build a text-matching table within the data set to create non-mapping links. At the same time, it constructs the corresponding data model and mapping algorithms, making up for the deficiency. The next study will consider improving the data model and relevant algorithms to improve the query efficiency. Meanwhile, automatically creating a set of non-mapping link objects is also a problem worthy of study.

References

[1] Auer, S., Feigenbaum, L., Miranker, D., Fogarolli, A., Sequeda, J. “Use casesand requirements for mapping relational databases to RDF, W3C working draft”.Technical report (2010).

[2] Z. Syed and T. Finin，“Creating and Exploiting aHybrid Knowledge Base for Linked Data”. RevisedSelected Papers Series: Communications in Computerand Information Science. Springer, April 2011.

[3] Giovanni Tummarello, Richard Cyganiak, Michele Catasta, Szymon Danielczyk,Stefan Decker, Sig.ma“live views on the Web of data,” Semantic WebChallenge (ISWC2009) (2009).Volume 8, Issue 4, November 2010, Pages 355–364.

[4] Barry Bishop, Atanas Kiryakov, Damyan Ognyanov, Ivan Peikov, Zdravko Tashev, Ruslan Velkov, Factforge: a fast track to the web of data, Semantic Web 2 (2) (2011) 157–166.

[5] Recommendation by Grouping Synonymy Tags, Journal of Computational Information Systems. 7(4), pp.1350-1357, 2011.