Class Association Structure Derived From Linked Objects

Yuzhong Qu, Weiyi Ge, Gong Cheng, and Zhiqiang Gao
Institute of Web Science, School of Computer Science and Engineering
Southeast University, Nanjing 210096, P.R. China
{yzqu,wyge,gcheng,zqgao}@seu.edu.cn

Abstract

The Web is being extended with more and more RDF data, especially the links between objects. Object links are critical to the semantic Web just as page links to the hypertext Web. Studying the macroscopic properties of object links can help people understand the semantic Web and build (semantic) Web applications in a better manner. To achieve this, we propose a notion of class association graph (CAG) based on the object links on the semantic Web, and report the results of analyzing the complex network characteristics of a CAG constructed from a real large data set collected by the Falcons search engine. The CAG observed has the scale-free nature and small-world characteristics. Then, vertex-importance graph synopsis approach is employed to depict a landscape of such class association structure.

Keywords

Object link, class association graph, complex network analysis

1 Introduction

As pointed by [1], the semantic Web will allow programmers and users alike to refer to real-world objects. Actually, more and more RDF data have been published on the Web in the past few years, and most of them were created to describe objects by using shared classes and properties. The RDF data model, together with the decentralized linkage nature of the semantic Web, brings up an object link structure in the worldwide scope, where objects are identified by URIs, and links are attributed to the relational properties among objects. We believe that such object link structure is important to the semantic Web just as the page link structure to the hypertext Web. Therefore, the macroscopic properties of the object link structure, which could help people, in a better manner, understand the semantic Web as well as build (semantic) Web applications, are deserved to be extensively studied.

On the semantic Web, an *entity* is a resource identified by a URI in RDF data. Entities are divided into classes, properties, and objects (individuals), based on RDF(S) and OWL-DL specifications. An object o is said to be an *instance* of a named class c iff there is an RDF triple $\langle o$, rdf:type, $c\rangle$ in some RDF graph, and we also say o is an *object with type* c. Supposing o_1 and o_2 are two distinct objects, we say there exists a *link* from o_1 to o_2 iff in some RDF graph there is a simple path from o_1 to o_2 in which internal vertices, if any, are all blank nodes.

Table 1: URI namespaces and corresponding prefixes

Prefix	URI Namespace
foaf	`http://xmlns.com/foaf/0.1/`
rdf	`http://www.w3.org/1999/02/22-rdf-syntax-ns#`
skos	`http://www.w3.org/2004/02/skos/core#`
yago	`http://dbpedia.org/class/yago/`

For convenience, qualified names [2] are used to give URIs in this paper, e.g., foaf:Person for http://xmlns.com/foaf/0.1-/Person. Well-known URI namespaces and corresponding prefixes used in the paper are listed in Table 1.

Figure 1(a) shows a fragment of an RDF graph encoded by http://kmi.open.ac.uk/people/tom/rdf, from which a link from Tom Heath (with type Person) to the revyu project (with type Project) is obtained, shown by Figure 1(b). Furthermore, with the link from an instance of the class Person to an instance of the class Project, we establish an association between these two classes, shown by Figure 1(c). A formal definition of class association will be given later.

To study the macrostructure of object links, in Section 2 we propose a notion of class association graph (CAG) and then introduce a real-world CAG derived from a large data set. Several complex network characteristics of this graph are reported in Section 3, and it is visualized in Section 4. Conclusions and future work are given in Section 5.

2 Class Association Graph

Let $\mathrm{I}(c)$ be the set of instances of class c, and let C be the set of classes subject to that every $c \in C$ is with $\mathrm{I}(c) \neq \emptyset$. Then a Class Association Graph (CAG), as an edge-weighted undirected graph with C the vertex set, is represented as (C, A, W_A), where A is the edge set, and an undirected edge (c_1, c_2) between $c_1, c_2 \in C$, called an *association*, is in A iff a link exists between some $o_1 \in \mathrm{I}(c_1)$ and $o_2 \in \mathrm{I}(c_2)$ of either direction; $W_A : A \rightarrow N$ is a weighting function that maps edges to natural numbers, and the weight is given by counting distinct unordered pairs of linked objects.

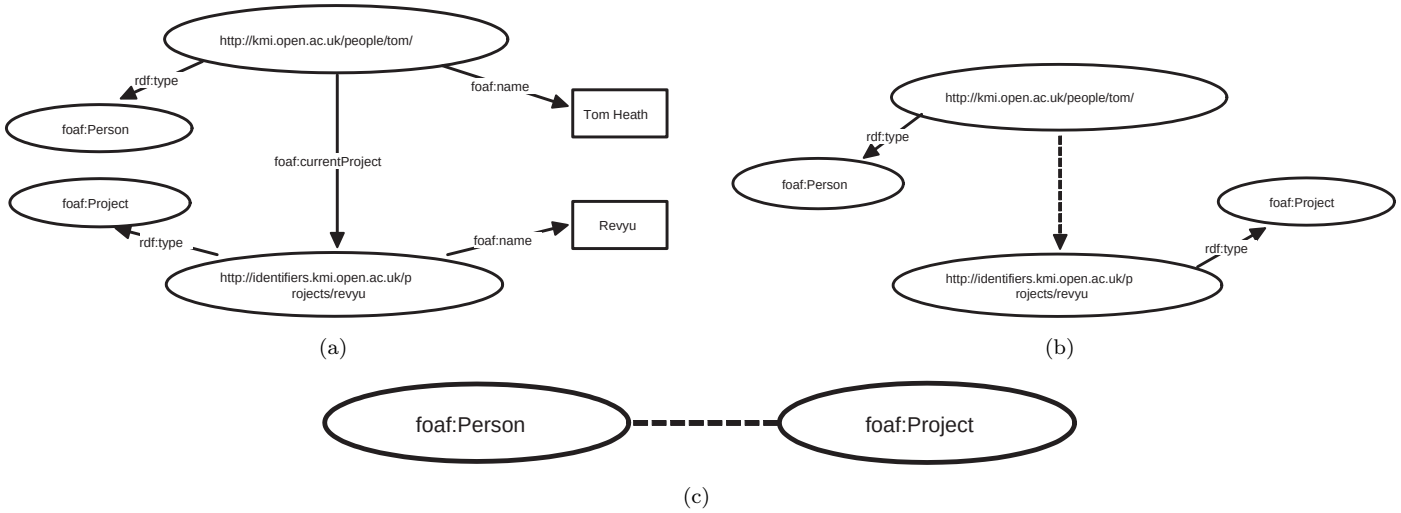

Figure 1: (a) an RDF graph, (b) a derived link (dashed line) between two objects with different types, and (c) a derived association between classes.

To perform complex network analysis of real-world class associations, we use a snapshot of the Semantic Web data collected by the Falcons search engine[1] by August 2008. The data set contains 11.7 million RDF/XML documents or 596 million RDF triples.

The data set contains 73,888,982 objects and 121,554,973 links between objects, in which 30,852,370 objects (41.8%) are with types and instantiate a total of 56,631 classes. Figure 2 depicts the distribution of classes w.r.t. the numbers of their instances, which follows a power law. In average, each class has 598 instances, and the class rdf:Statement has the largest number of instances (5,378,695). Based on these linked objects, we obtain a CAG containing 56,631 classes as vertices and 281,141 associations as edges. This graph as well as its statistics are available online.[2]

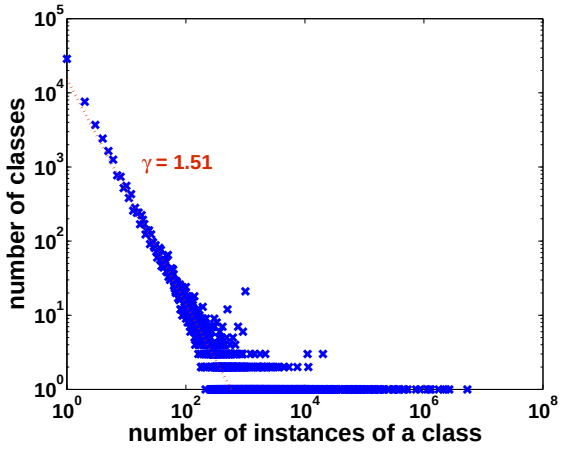

Figure 2: Distribution of the number of classes versus the number of instances of a class.

[1]http://iws.seu.edu.cn/services/falcons/
[2]http://iws.seu.edu.cn/projects/ontosearch/cag/

3 Network Characteristics of CAG

Two basic measures of vertex importance in a edge-weighted graph are degree and strength. The degree of a vertex is given by counting its associated edges, and the strength of a vertex [3] is given by summing up the weights attached to all of its associated edges. Figure 3 depicts the degree distribution and strength distribution of vertices in CAG, in which isolated vertices are excluded. Both distributions follow power laws, indicating the scale-free nature of CAG. The average degree of vertices in CAG is 9.91, and the maximal degree is 3,530, owned by foaf:Person, followed by skos:Concept and foaf:Document, as listed in Table 2. The average strength of vertices in CAG is 2,155, and the maximal strength is 12,652,946. Those vertices with high degrees are usually with high strengths, demonstrated by a positive Kendall Tau rank correlation coefficient (0.336) between degrees and strengths of vertices.

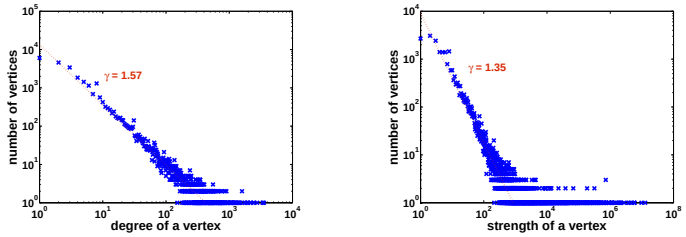

Figure 3: (a) degree distribution and (b) strength distribution of vertices in CAG.

Figure 4 shows the size distribution of connected components (CCs) of CAG. More than half of the classes (51.5%) are within trivial CCs, i.e. their instances are never linked to instances of other classes. However, these classes cover only 281,090 objects (0.9%).

Table 2: Top-10 vertices with the highest degrees

Vertex	Degree
http://xmlns.com/foaf/0.1/Person	3,530
http://www.w3.org/2004/02/skos/core#Concept	3,405
http://xmlns.com/foaf/0.1/Document	3,380
http://dbpedia.org/class/yago/Person100007846	2,759
http://smw.ontoware.org/2005/smw#Thing	2,457
http://www.w3.org/2006/03/wn/wn20/schema/NounSynset	2,435
http://dbpedia.org/class/yago/State108168978	2,173
http://dbpedia.org/class/yago/City108524735	2,167
http://dbpedia.org/class/yago/Colony108374049	1,981
http://dbpedia.org/class/yago/Alumnus109786338	1,929

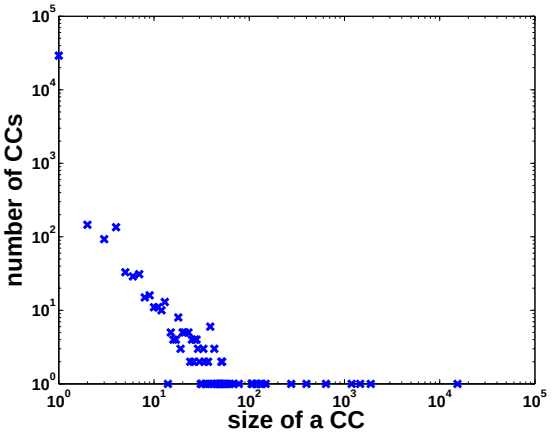

Figure 4: Size distribution of CCs.

Out of the 27,468 non-trivial CCs, the five largest ones are listed in Table 3. The largest CC (called LCC) contains 15,385 vertices (27.17%), 254,242 edges (90.56%), and covers 29,667,715 objects (96.16%), which evidently dominates all the other CCs. Therefore, we focus on LCC in the following analysis. The classes in LCC come from 698 vocabularies. Table 4 lists the top-10 vocabularies that these classes come from.

Table 3: Top-5 largest CCs

Rank	#classes	#objects
1	15,385	29,667,715
2	1,888	1,888
3	1,463	9,131
4	1,191	184,539
5	639	35,038

In graph theory, the distance between two vertices is the length of a shortest path between them. In LCC, the maximum and average distances between vertices are 16 and 3.8, respectively, indicating a closely associated structure.

We have calculated the clustering coefficient (C) and the weighted clustering coefficient (C^w) [3] of each vertex in LCC. The average clustering coefficient of vertices in LCC is 0.46, which is much larger than 0.0021, the average clustering coefficient of vertices in a random graph with the same numbers of vertices and edges. The average weighted clustering coefficient of vertices is 0.52. Besides, for most of vertices $C^w > C$ is observed, which means that those pairwise associated triplets are more likely formed by the edges with heavier weights.

Table 4: Top-10 vocabularies that the classes in LCC come from

Vocabulary	#classes
http://dbpedia.org/class/yago/	4,148
http://www.mpii.de/yago/resource/	3,843
http://www.archiplanet.org/wiki/Special:URIResolver/	1,961
http://semanticweb.org/id/	436
http://ontoworld.org/wiki/Special:URIResolver/	308
http://ontology.dumontierlab.com/	174
http://bio2rdf.org/uniprot:	120
http://oiled.man.example.net/test#	115
http://mathweb.org/wiki/Special:URIResolver/	77
http://viget.org/Special:URIResolver/	77

Let $C(k)$ be the average clustering coefficient of the vertices with degree k, and let $C^w(k)$ be the average weighted clustering coefficient of the vertices with degree k. Figure 5 shows decaying distributions of $C(k)$ and $C^w(k)$. Hub classes, i.e. the classes with high degrees, are with low clustering coefficients, which indicates that those classes associated with hub classes are usually not connected to each other.

To conclude, LCC is highly clustered yet with short distances between vertices. Therefore, it has the small-world characteristic.

4 Visualization of Class Association Structure

In order to depict a landscape of the observed class association structure, considering the large scale, we employ the vertex-importance graph synopsis approach [4]. "Graph synopses defined by the importance of vertices provide small, relatively accurate portraits, independent of the importance measure, of the larger underlying graphs and of the important vertices." We

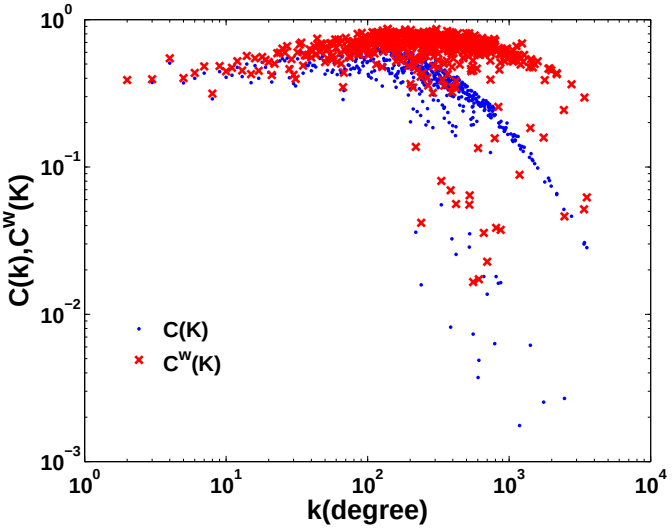

Figure 5: Distribution of (weighted) clustering coefficient of a degree.

use the following formula to measure the importance of vertices (classes):

$$\begin{aligned}\mathrm{Score}(C_i) &= |\,\mathrm{Inst}(C_i)| \cdot \sum_{C_j \in \mathrm{Neighb}(C_i)} \frac{W_{ij}}{|\,\mathrm{Inst}(C_i)| \cdot |\,\mathrm{Inst}(C_j)|} \\ &= \sum_{C_j \in \mathrm{Neighb}(C_i)} \frac{W_{ij}}{|\,\mathrm{Inst}(C_j)|}, \end{aligned} \tag{1}$$

where $\mathrm{Neighb}(C_i)$ is the set of adjacent vertices of C_i, W_{ij} is the weight of the edge (C_i, C_j), and $\mathrm{Inst}(C_i)$ is the set of instances of the class C_i.

Based on the above measure, we identify the 2,000 most important vertices, and then obtain a vertex-induced subgraph of LCC, denoted by LCC-2000. We find that LCC-2000 still has a good connectivity, and the relative ranks of the vertices in LCC-2000 are highly correlated to their ranks in LCC. These results accord with the supposition proposed in [4].

The LCC-2000 is depicted in Figure 6 by using the Fruchterman-Reingold Force-Directed layout algorithm in pajek,[3] a tool for analyzing large-scale networks. In this figure, a vertex of a large size corresponds to a class with a large number of instances, and the classes from the same vocabulary are with the same color.

Figure 6 tells that the FOAF vocabulary (in blue) is located at the center of the figure and plays a key role on the semantic Web since its classes are associated with many other classes. Many vocabularies, e.g., http://www.mpii.de/yago/ (in red, located at the bottom right part of the figure), have strong internal associations but very weak external associations. However, some other vocabularies, such as the YAGO vocabulary used by DBpedia (in green, located at the center of the figure), have both strong internal and external associations.

[3] http://vlado.fmf.uni-lj.si/pub/networks/pajek/

5 Conclusion

We have proposed a notion of CAG based on the linked objects on the semantic Web, and have captured a CAG from a large data set. We have analyzed its complex network characteristics, and have revealed its scale-free and small-world characteristics. We have also depicted a vertex-importance graph synopsis of its largest connected component, which presents, to some extent, a landscape of the class association structure brought by the linked objects on the real semantic Web.

This work has studied the macroscopic properties of linked objects at their class level. It would be very interesting to investigate the original object link structure, which is a great challenge in consider of the large scale.

6 Acknowledgments

The work is supported in part by the NSFC under Grant 60773106 and 60873153, and in part by the 973 Program of China under Grant 2003CB317004. We are grateful to Rujin Cao and Jianfeng Chen for their work on experiments. We would also like to thank anonymous reviewers for their helpful comments.

References

[1] Hendler, J., Shadbolt, N., Hall, W., Berners-Lee, T., Weitzner, D. Web science: an interdisciplinary approach to understanding the Web. Communications of the ACM, 51(7):60–69, 2008

[2] Bray, T., Hollander, D., Layman, A., Tobin, R. Namespaces in XML 1.0 (Second Edition). W3C Recommendation, 2006

[3] Barrat, A., Barthélemy, M., Pastor-Satorras, R., Vespignani, A. The architecture of complex weighted networks. In Proceedings of the National Academy of Sciences, 101(11):3747–3752, 2004

[4] Shi, X., Bonner, M., Adamic, L.A., Gilbert, A.C. The very small world of the well-connected. In Proceedings of the 19th ACM Conference on Hypertext and Hypermedia, pages 61–70, 2008

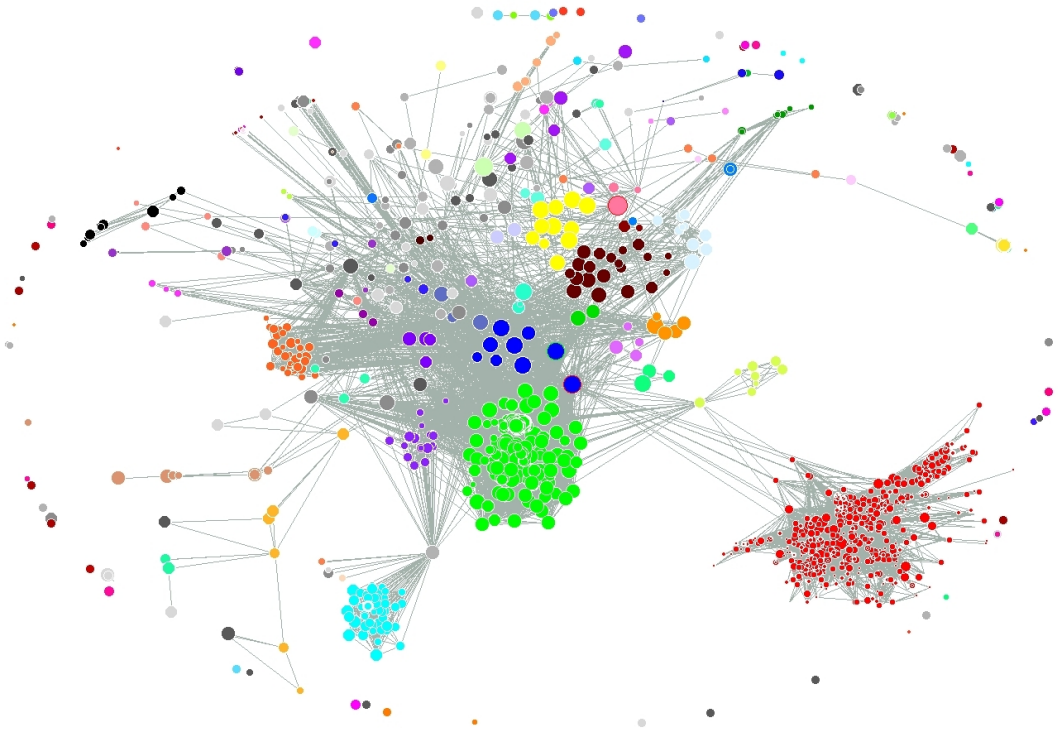

Figure 6: LCC-2000, the subgraph induced by the Top-2,000 important vertices in LCC.