# Term Dependence on the Semantic Web

Gong Cheng and Yuzhong Qu

Institute of Web Science, School of Computer Science and Engineering
Southeast University, Nanjing 210096, P.R. China
{gcheng,yzqu}@seu.edu.cn

**Abstract.** A large amount of terms (classes and properties) have been published on the Semantic Web by various parties, to be shared for describing resources. Terms are defined based on other terms, and thus a directed dependence relation is formed. The study of term dependence is a foundation work and is important for many other tasks, such as ontology maintenance, integration, and distributed reasoning on the Web scale. In this paper, we analyze the complex network characteristics of the term dependence graph and the induced vocabulary dependence graph. The graphs analyzed in the experiments are constructed from a large data set that contains 1,278,233 terms in 3,039 vocabularies. The results characterize the current status of schemas on the Semantic Web in many aspects, including degree distributions, reachability, and connectivity.

## 1 Introduction

As with the decentralized linkage nature of the Semantic Web, terms (classes and properties) are usually defined based on other terms in various vocabularies. The meaning of a term is dependent on the meanings of those terms used in its definition. In other words, a change of the meaning of a term may affect the meanings of those terms that are dependent on it. Therefore, term dependence on the Semantic Web is a fundamental problem concerned with ontology maintenance on the Web scale and the evolution of the Semantic Web. Furthermore, the term dependence topology is an important factor that influences how applications access the meanings of semantically interlinked terms, as well as distributed reasoning on the Web scale.

Recently, graph analysis of the Semantic Web has been performed from various aspects [7,9,11,12,14,17]. However, the graph structure of schemas on the Semantic Web on a large scale has not yet been well studied. In this paper, we propose a notion of term dependence on the Semantic Web, and analyze the complex network characteristics of the term dependence graph constructed from a data set that contains 1,278,233 terms defined in 3,039 vocabularies, discovered by our Falcons search engine.[1] We analyze its degree distributions, reachability, and connectivity. We also generalize the dependence from the term level to the vocabulary level, and study its characteristics.

The remainder of this paper is structured as follows. Section 2 discusses related work. Section 3 defines basic terminology used in this paper. Section 4 provides

[1] http://iws.seu.edu.cn/services/falcons/.

A. Sheth et al. (Eds.): ISWC 2008, LNCS 5318, pp. 665–680, 2008.

an overview of the data set used in the experiments. In sections 5 and 6, complex network analysis techniques are applied to the term dependence graph and the vocabulary dependence graph, respectively. Section 7 concludes the paper and presents future work.

## 2 Related Work

Graph analysis has been successfully performed to measure the World Wide Web. Albert et al. [2] analyzed the distributions of incoming and outgoing links between HTML documents on the World Wide Web, and observed power law tails. Barabási et al. [1] found similar results at the site level. As an early work, Gil et al. [9] performed graph analysis on the Semantic Web. They combined ontologies from DAML Ontology Library into a single graph, which included 56,592 vertices and 131,130 arcs. They observed that the Semantic Web is a small world with an average path length 4.37, and the degree distribution follows a power law. Ding et al. [7] studied social networks induced by over 1.5 million of FOAF documents, in which power laws were also observed and interesting patterns of connected components were revealed. Ding and Finin [6] collected 1,448,504 RDF documents and focused on aspects such as the distribution of documents over hosts and the sizes of documents. They measured the complexity of terms by counting the number of RDF triples used to define them, and measured the instance space by counting the meta-usages of terms. Power laws were observed in both experiments. Tummarello et al. [15] found that the distribution (reuse) of URIs over documents follows a power law.

Recently, graph analysis techniques have also been applied to single ontology. Hoser et al. [11] illustrated the benefits of applying social network analysis to ontologies by measuring SWRC and SUMO ontologies. They discussed how different notions of centrality (degree, betweenness, eigenvector, etc.) describe the core content and structure of an ontology, and compared ontologies in size, scope, etc. Ma and Chen [12] surveyed the topology of two TCMLS sub-ontologies. They reported that the analyzed networks, composed of concepts and instances, are typical small-world and scale-free networks. Zhang [17] studied NCI-Ontology, Full-Galen, and other 5 ontologies, and discovered that the degree distributions of entity networks fit power laws well. Theoharis et al. [14] analyzed graph features of 250 ontologies. For each ontology, they constructed a property graph and a class subsumption graph. They found that the majority of ontologies with a significant number of properties approximate a power law for total-degree distribution, and each ontology has a few focal classes that have numerous properties and subclasses.

## 3 Preliminaries

### 3.1 Term and Vocabulary

Basically, vocabularies and related definitions in this paper are in accordance with [3]. A *vocabulary* on the Semantic Web is a non-empty set of URIs (called

**Table 1.** URI namespaces and corresponding prefixes

| Prefix | URI Namespace |
|---|---|
| cyc | `http://www.cyc.com/2004/06/04/cyc#` |
| dc | `http://purl.org/dc/elements/1.1/` |
| dcterms | `http://purl.org/dc/terms/` |
| foaf | `http://xmlns.com/foaf/0.1/` |
| food | `http://www.w3.org/TR/2003/PR-owl-guide-20031209/food#` |
| owl | `http://www.w3.org/2002/07/owl#` |
| rdf | `http://www.w3.org/1999/02/22-rdf-syntax-ns#` |
| rdfs | `http://www.w3.org/2000/01/rdf-schema#` |
| skos | `http://www.w3.org/2004/02/skos/core#` |
| vcard | `http://www.w3.org/2001/vcard-rdf/3.0#` |
| vin | `http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#` |

its *constituent terms*) that denote a class or a property with a common URI namespace. For example, the URIs `http://xmlns.com/foaf/0.1/Person` and `http://xmlns.com/foaf/0.1/knows`, both containing the URI namespace `http://xmlns.com/foaf/0.1/`, are two constituent terms of the FOAF vocabulary. For convenience, qualified names [4] 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. The *authoritative description* of a vocabulary is an RDF graph (a set of RDF triples) encoded by its *namespace document* as well as those RDF documents retrieved by dereferencing the URIs of its constituent terms. Whereas anyone can say anything on the Semantic Web, the authoritative description of a vocabulary is considered to be the most trustable with regard to its constituent terms.

A vocabulary $v$ on the Semantic Web is formulated as $\langle id, C, P, G\rangle$, where $id$ is the URI namespace that identifies $v$; $C$ and $P$ are the sets of constituent classes and properties of $v$, respectively, s.t. $C \cup P \neq \emptyset$; $G$ is the authoritative description of $v$. A URI $t$ is a constituent class (property) of a vocabulary $v$ iff two conditions are satisfied: (a) the URI namespace of $t$ is $v.id$; (b) $v.G$ entails the RDF triple $\langle t$, `rdf:type`, `rdfs:Class`$\rangle$ ($\langle t$, `rdf:type`, `rdf:Property`$\rangle$). The entailment in the experiments is performed by an implemented reasoning engine, based on RDF(S) [10] and OWL DL [13] entailment rules. For example, $t$ is a constituent class of $v$ if $v.G$ contains an RDF triple whose subject is $t$ and predicate is `rdfs:subClassOf`; $t$ is a constituent property of $v$ if $v.G$ contains an RDF triple whose predicate is `owl:onProperty` and object is $t$. All such rules are not listed in the paper due to space restrictions.

### 3.2 RDF Sentence

Two RDF triples are *b-connected* [18] if they contain common blank nodes. The b-connected relation is defined as transitive. In an RDF graph, an *RDF sentence*

is a maximum subset of b-connected RDF triples. Formally, in an RDF graph $G$, an RDF sentence $\tilde{s}$ is a subset of RDF triples that satisfy the following conditions: (a) $\forall \tau_i, \tau_j \in \tilde{s}$, $\tau_i, \tau_j$ are b-connected; (b) $\forall \tau_i \in \tilde{s}, \tau_j \in G \setminus \tilde{s}$, $\tau_i, \tau_j$ are not b-connected. Figure 1 illustrates an RDF sentence. Let $U$ be the set of all URIs. For an RDF sentence $\tilde{s}$, define $\text{Subj}(\tilde{s}) = \{s | s \in U \wedge \exists \langle s, p, o \rangle \in \tilde{s}\}$. Analogously define $\text{Pred}(\tilde{s}) = \{p | p \in U \wedge \exists \langle s, p, o \rangle \in \tilde{s}\}$ and $\text{Obj}(\tilde{s}) = \{o | o \in U \wedge \exists \langle s, p, o \rangle \in \tilde{s}\}$. For example, for the RDF sentence $\tilde{s}$ depicted in Fig. 1, $\text{Subj}(\tilde{s})$ = {`food:SeafoodCourse`}, $\text{Pred}(\tilde{s})$ = {`rdfs:subClassOf`, `rdf:type`, `owl:onProperty`, `owl:allValuesFrom`, `owl:hasValue`}, and $\text{Obj}(\tilde{s})$ = {`owl:Restriction`, `food:hasDrink`, `vin:hasColor`, `food:White`}.

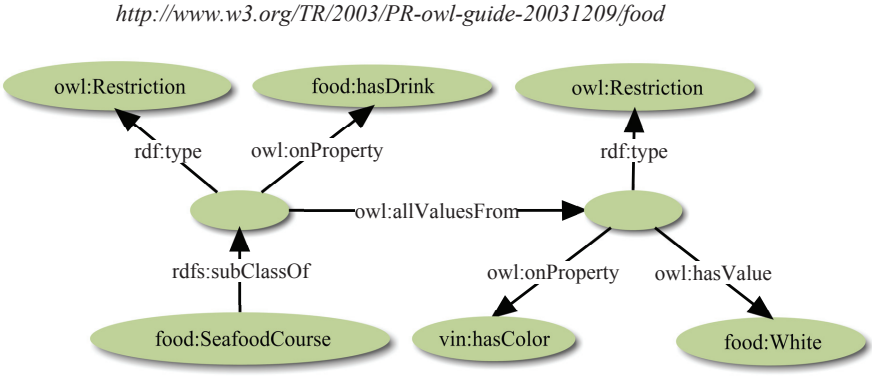



**Fig. 1.** An RDF sentence

Evidently, two distinct RDF sentences do not share blank nodes. RDF semantics [10] treats blank nodes as existential variables, which are not addressable from outside a graph and are usually created to connect URIs and literals. Besides, an RDF graph $G$ can be decomposed into a unique set of RDF sentences, denoted by $\text{Sent}(G)$. For more details of RDF sentence, refer to [18] or [16]. In [16], the Minimum Self-contained Graph is an equivalent definition of RDF sentence.

### 3.3 Term Dependence

On the Semantic Web, terms are related to each other in various ways. Most previous work analyzed specific kinds of relations between terms, such as property graph [14], class subsumption graph [14], or a combination of several specific relations [11]. This paper generalizes from these specific relations to a single relation called term dependence, and analyzes its complex network characteristics.

For term $t_1 \in v_1.C \cup v_1.P$ and term $t_2$, $t_1$ *directly depends on* $t_2$, or $t_2$ *directly influences* $t_1$, iff $\exists \tilde{s} \in \text{Sent}(v_1.G), t_1 \in \text{Subj}(\tilde{s}), t_2 \in \text{Pred}(\tilde{s}) \cup \text{Obj}(\tilde{s})$. For example, in Fig. 1, `food:SeafoodCourse` directly depends on all the other terms occurring in that RDF sentence. Using RDF sentences rather than RDF triples

to induce dependence causes that dependence is always from URIs to URIs, and blank nodes are not involved in the dependence graph, whereas they still make contributions.

Direct dependence between terms is a very general directed relation, which covers many important specific relations. For example, in RDFS expressions, a class directly depends on its super-classes, and a property directly depends on its super-properties, domain and range; in OWL expressions (after translating OWL axioms to RDF graphs according to [13]), a class directly depends on the properties and classes in its property restrictions (as shown in Fig. 1), and a property directly depends on its inverse property. And naturally, terms often directly depend on those language-level terms in RDF(S) and OWL when using the expressions thereof. Characterizing relations between terms with dependence could greatly simplify the analysis since the relations become homogeneous. Compared with specific relations, term dependence gives a more comprehensive view. However, it is limited by its origin from RDF syntax, e.g., OWL equivalence axioms will only be transformed into unidirectional dependence, whereas bidirectional dependence may be better in some cases.

Generally, to understand the meaning of a term, it is necessary to understand the terms it depends on. In other words, a change of the meaning of some term may affect the meanings of the terms that depend on it, which explains why the word "influence" is used as the inverse relation of dependence.

Direct dependence/influence can be naturally extended to more general dependence/influence in a recursive way: for terms $t_1$ and $t_2$, $t_1$ *depends on* $t_2$, or $t_2$ *influences* $t_1$, iff $t_1$ directly depends on $t_2$ or there exists a term $t_3$ satisfying that $t_1$ directly depends on $t_3$ and $t_3$ depends on $t_2$.

## 4 Data Set

All the experiments described in this paper were run on a snapshot of the Semantic Web data collected by the Falcons search engine [5] until April 2008. This section introduces how the data set is constructed, including the seed set collection and the crawling strategy, and then characterizes the distributions of the data set.

### 4.1 Crawler

RDF document, each identified by a URI, is a basic unit of the data set. The construction of the data set was bootstrapped by submitting to the crawler a set of seed URIs of RDF documents, which were obtained in two ways. Firstly, a list of phrases were extracted from the category names at the top three levels of the Open Directory Project,[2] randomly combined as keyword queries, and sent to the Swoogle search engine[3] and Google search engine (for "filetype:rdf" and "filetype:owl") to retrieve URIs of potential RDF documents. Secondly, the

[2] `http://www.dmoz.org/`.

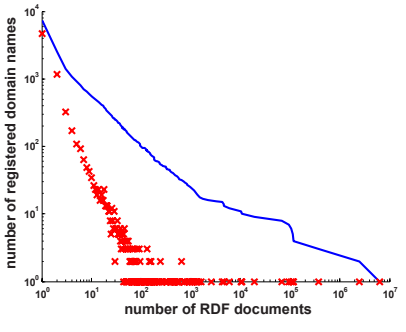
[3] `http://swoogle.umbc.edu/`.

URIs of RDF documents from several online repositories were manually added to the seed set, including Ping the Semantic Web.com,[4] SchemaWeb,[5] etc.

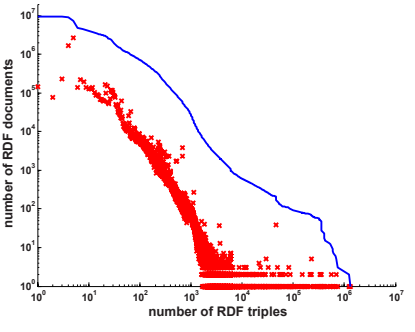
A multi-thread crawler was then implemented to dereference URIs with content negotiation and download RDF documents. For simplicity, the "Accept" field in the header of HTTP requests was always set to "application/rdf+xml", and only well-formed RDF/XML documents would be included in the data set. After parsing an RDF document by using Jena,[6] all the URIs mentioned in the document were submitted for further crawling. During a six-month running, 24 million URIs have been pinged, and 11 million documents have been downloaded, 9.8 million of them confirmed as well-formed RDF/XML documents.

### 4.2 Distributions and Statistics of the Data Set

The 9.8 million RDF documents analyzed in this paper come from 114,408 hosts, or 7,290 registered domain names.[7] The distribution of the number of RDF documents on registered domain names, shown in Fig. 2(a), approximates a power law. The long tail of the distribution is caused by several registered domain names that host large numbers of RDF documents, including `bio2rdf.org`, `dbpedia.org`, `openlinksw.com`, `buzznet.com`, `bibsonomy.org`, `l3s.de`, etc.




(a) Distribution (*crosses*) and cumulative distribution (*curve*) of the number of registered domain names versus the number of RDF documents per registered domain name.




(b) Distribution (*crosses*) and cumulative distribution (*curve*) of the number of RDF documents versus the number of RDF triples per document.

**Fig. 2.** Distributions of the data set

The data set contains 401 million RDF triples altogether. The distribution of sizes of RDF documents, shown in Fig. 2(b), also approximates a power law,

[4] `http://pingthesemanticweb.com/`.

[5] `http://www.schemaweb.info/`.

[6] `http://jena.sourceforge.net/`.

[7] A registered domain name is more general than the host part of a URI. For example, the host part of `http://iswc2008.semanticweb.org/` is `iswc2008.semanticweb.org`, but its registered domain name is `semanticweb.org`.

except for the initial segment. Actually the distribution has a maximum at 5 RDF triples, and the cumulative distribution curve exhibits that about half of the RDF documents (51.6%) in the data set contain no more than 5 RDF triples. Generally, each of these small RDF documents encodes a snippet of RDF triples to describe only one specific entity, and such style has been widely adopted in the data sources from the Linking Open Data project.[8] There are also 237 thousand RDF documents (2.4%) that do not contain any RDF triples, but may declare some URI namespaces. It is partially because several servers do not return the HTTP response code 404 for unknown URIs but return such "skeleton" RDF documents. Besides, the only two RDF documents that contain more than 1 million RDF triples are the NCI Thesaurus[9] and WordNet.[10]

Based on the definitions introduced in Sect. 3.1, a total of 3,039 vocabularies and 1,278,233 constituent terms have been recognized from the data set, including 1,158,480 classes (90.6%), 118,808 properties (9.3%), and the other 945 (0.07%) that are both classes and properties. Although RDFS and OWL Full do not require disjointness of classes and properties, such "ambiguous" definitions may on one hand become an obstacle to attract inexperienced developers into the promotion of the Semantic Web, and may on the other hand increase the complexity of computation (e.g., reasoning), especially when some popular terms fall into this group, such as `vcard:Orgname`.

Actually, if the definition of a term is relaxed from the authoritative description of its vocabulary to any description discovered on the Semantic Web, the numbers of classes and properties in the data set will increase to 2,196,855 (+89.5%) and 195,812 (+63.5%), respectively. However, to best ensure the quality, the following analysis will only focus on the previous 1,278,233 terms, denoted by $\mathbb{T}$.

Figure 3(a) shows the distribution of constituent terms of vocabularies, which approximates a power law especially when the number of constituent terms is larger than 10. The largest vocabulary observed is EthanAnimals,[11] which contains 196,591 terms, followed by FMA,[12] containing 75,245 terms. Different vocabularies are created for different domains and purposes, and they may contain more classes or more properties. Figure 3(b) shows a scatter plot of such data. There are 2,385 vocabularies (78.5%) containing at least one class and one property, 557 vocabularies (18.3%) containing only classes, and 97 vocabularies (3.2%) containing only properties. EthanAnimals, as the vocabulary that contains the most classes, does not contain any properties. Actually, out of the 23 vocabularies that contain more than 10,000 terms, 19 contain less than 10 properties, and most of these large vocabularies describe the medical domain by

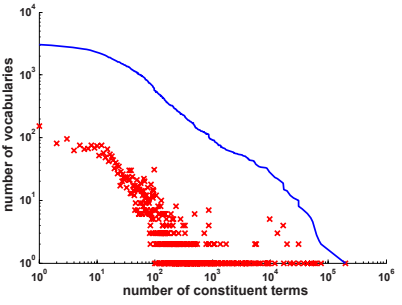
[8] `http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData`.

[9] `http://www.berkeleybop.org/ontologies/obo-all/ncithesaurus/ncithesaurus.owl`.

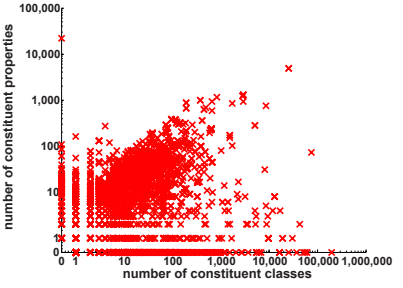
[10] `http://www.w3.org/2006/03/wn/wn20/rdf/full/wordnet-wordsensesandwords.rdf`.

[11] `http://spire.umbc.edu/ontologies/EthanAnimals.owl`.

[12] `http://onto.eva.mpg.de/fma/fma.owl`.




(a) Distribution (crosses) and cumulative distribution (curve) of the number of vocabularies versus the number of constituent terms of a vocabulary.




(b) Distribution of the number of constituent properties versus the number of constituent classes of a vocabulary.

**Fig. 3.** Distributions of terms

using only large class hierarchies. A vocabulary[13] used by DBpedia contains the most properties. Besides, there are 6 vocabularies that contain more than 1,000 classes and more than 1,000 properties, all of which are different versions of the OpenCyc ontology.[14]

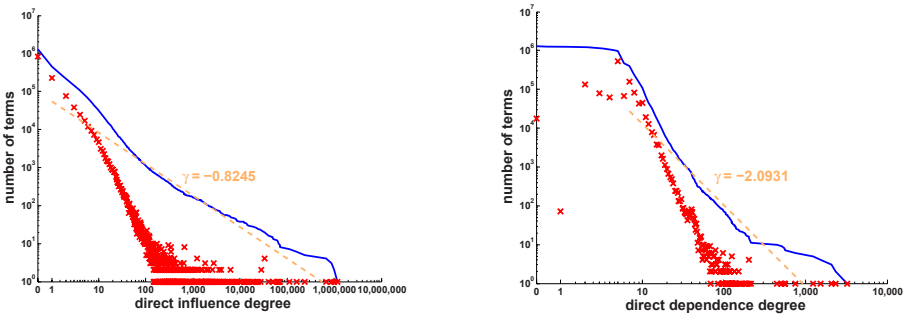
## 5 Complex Network Analysis of Term Dependence

Dependence between terms on the Semantic Web can be characterized by a directed graph, called the *term dependence graph*, denoted by $\mathrm{TDG} = \{\mathbb{T}, \mathbb{TD}\}$, where $\mathbb{T}$ is the vertex set, each vertex labeled with a term $t \in \mathbb{T}$; $\mathbb{TD}$ is the arc set, and an arc $\langle t_1, t_2 \rangle$ exists iff $t_1$ directly depends on $t_2$. The TDG analyzed in this paper includes 1,278,233 vertices and 7,312,657 arcs (after removing self-loops). The remainder of this section will analyze TDG to study its complex network characteristics and show how terms are defined and related to each other on the real Semantic Web.

### 5.1 Degree Analysis

Two basic measures of TDG are the distributions of in-degrees and out-degrees, which are called *direct influence degrees* and *direct dependence degrees* of terms, respectively. A term of a higher direct influence degree is referenced by more other terms in their definitions, and a term of a higher direct dependence degree references more other terms in its definition. It is worth noting that direct dependence is derived from explicitly specified information by data owners, which exhibits their original biases and customs of defining terms.

[13] `http://dbpedia.org/property/`.
[14] `http://www.opencyc.org/`.




(a) (b)

**Fig. 4.** Distribution (*crosses*) and cumulative distribution (*curve*) of the number of terms versus the (a) direct influence degree and (b) direct dependence degree

Figure 4 shows the distributions of direct influence degrees and direct dependence degrees on a log-log scale. The average in-degree/out-degree is 5.72. In Fig. 4(a), the cumulative distribution of direct influence degrees follows a power law with the exponent $\gamma = -0.8245$. There are 7 terms that are of a direct influence degree higher than 100,000, which are (in descending order) `rdf:type`, `rdfs:subClassOf`, `owl:Class`, `rdfs:label`, `rdfs:comment`, `rdfs:Class`, and `owl:equivalentClass`. It indicates that class hierarchy (including class equivalence) is the most observed structure when defining terms and publishing vocabularies, whereas developers are also inclined to attach human-readable information to terms by using annotation properties. Not surprisingly, all these terms are in language-level vocabularies. The most observed non-language-level terms are mainly those properties for generating unique identifiers in large vocabularies, such as `cyc:guid`. Besides, 829,101 terms (64.9%) do not directly influence any other terms, which covers 64.6% classes and 67.0% properties.

As shown in Fig. 4(b), the distribution of direct dependence degrees does not fit a power law quite well, especially for the initial segment, and has a maximum at 5 degrees, which covers 40.9% terms. It is interesting that 17,505 terms (1.4%) do not depend on any other terms. It is mainly because, some large vocabularies, such as the NEWT taxonomy,[15] do not encode all the term definitions in one RDF document but only returns information principally about just one term when its URI is dereferenced. Then, it is possible that some term definitions have not been crawled but they can still be confirmed as terms since they have been found in other term definitions in the same vocabulary. Besides, the cumulative distribution curve shows that 10 terms are of a direct dependence degree higher than 400. A case-by-case study reveals that all these terms are classes and are also of a direct influence degree higher than 1,400. Actually each of them, called a focal class in [14], is a central term in the vocabulary, depending on and being depended on by many other terms.

[15] `http://purl.uniprot.org/taxonomy/`.

The Pearson's correlation coefficient between direct influence degrees and direct dependence degrees is 0.006 (ranging from -1 to 1), which means there is almost no linear relationship between them.

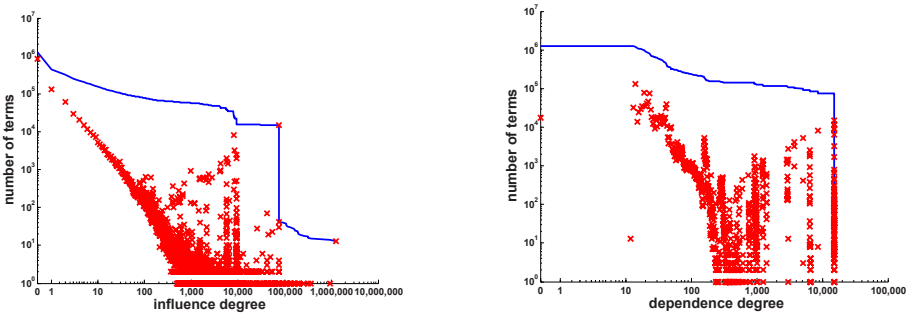
### 5.2 Reachability Analysis

The previous subsection analyzed the direct dependence and influence between terms in graph view. According to the definitions in Sect. 3.3, the more general dependence and influence can also be clearly characterized in the view of graph theory: $\forall t_1, t_2 \in \mathbb{T}$, $t_1$ depends on $t_2$, or $t_2$ influences $t_1$, iff $t_2$ is reachable from $t_1$ in TDG. For each term, the number of its reachable terms is called its *dependence degree*, and the number of the terms that can reach it is called its *influence degree*.

When retrieving a term definition or understanding a term, it is often the case that those terms it directly depends on still need to be explored and their definitions will also be retrieved, and goes on. A term of a higher dependence degree requires more steps of such retrieval. Correspondingly, the influence degree indicates how important a term exhibits on the Semantic Web because a change of the meaning of a high-influence-degree term will affect the meanings of a large amount of other terms.

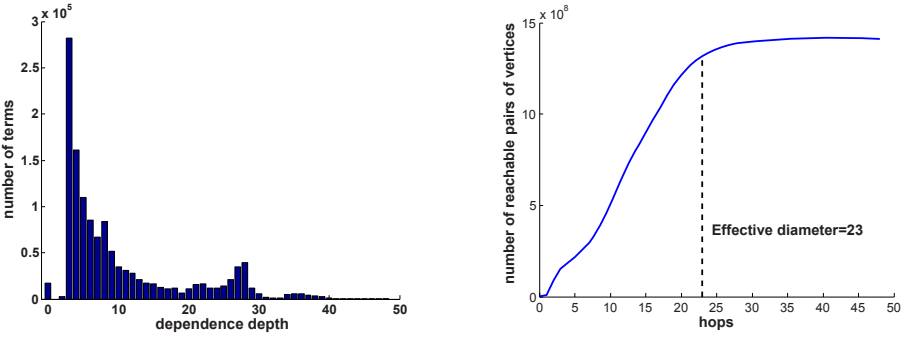
Figure 5 shows the distributions of term influence degrees and term dependence degrees on a log-log scale. In average, each term depends on 1,105 other terms. In Fig. 5(a), the initial segment of the distribution follows a power law, but the rest part is a mess. One reason is that many large strongly connected components (SCC) are observed in TDG, and all the terms in an SCC have exactly the same influence degree and dependence degree. Particularly, there are 13 terms, including `rdf:type`, `rdfs:Resource`, `rdfs:Class`, `rdfs:subClassOf`, `rdf:Property`, `rdfs:subPropertyOf`, `rdfs:domain`, `rdfs:range`, `rdfs:label`, `rdfs:comment`, `rdfs:seeAlso`, `rdfs:isDefinedBy`, and `rdfs:Literal` that compose an SCC, all of which influence almost all the terms on the Semantic Web. It also explains why few terms has a dependence degree between 1 and 12, as shown in Fig. 5(b). These results demonstrate that RDF and RDFS should be kept stable because a change of their meanings will almost change the whole Semantic Web. It is also a best practice for all the Semantic Web applications to be equipped with the ability to understand and use these terms.

In graph theory, the distance between two vertices is the length of a shortest path between them, and the eccentricity of a vertex is the maximum distance from the vertex to any other reachable vertices. In TDG, the eccentricity of a term is called its *dependence depth*. When retrieving a complete definition of a high-dependence-depth term, more rounds of breadth-first search (BFS) are required; and when understanding such terms, people are more likely to become lost in long-distance paths. Besides, it may take more steps to reflect a change of the meaning of a high-dependence-depth term caused by a change of the meaning of some term it depends on, due to the long distance. Figure 6(a) shows the distribution of dependence depths of terms on the Semantic Web. The average dependence depth is 10.05. About half of the terms (51.4%) have




(a) (b)

**Fig. 5.** Distribution (*crosses*) and cumulative distribution (*curve*) of the number of terms versus the (a) influence degree and (b) dependence degree



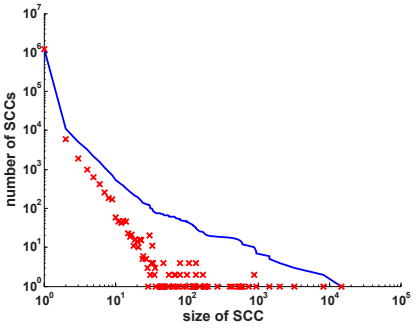

(a) Distribution of the number of terms versus the dependence depth.

(b) Hop plot and effective diameter.

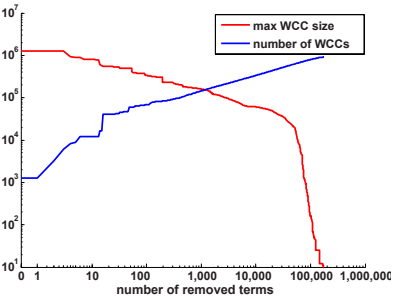
**Fig. 6.** Eccentricities and hop plot of TDG

a dependence depth not higher than 6. However, there are still 11.5% terms that have a dependence depth higher than 25, which often occur at the bottom of class hierarchies. The highest dependence depth observed is 48, owned by 4 classes at the bottom of a deep class hierarchy in FMA.

In some cases, to process a term, it is not necessary to retrieve all the terms it depends on, but instead, a significantly large subset (e.g., 90%) is enough for specific applications. In graph theory, hop plot [8] is used to measure the rate of increase of reachable vertices with increasing the distance threshold (called hops). The effective diameter of a graph is the minimum number of hops in which 90% of all reachable pairs of vertices can reach from one to the other. Figure 6(b) shows the hop plot of TDG, which approximates a linear correlation when hops is less than 23, the effective diameter of TDG. It means that in average, when retrieving the definitions of a term and those terms it depends on in a BFS way, the number of newly found terms does not remarkably decrease until 23 rounds later. Evidently, 23 seems too large a value for human beings.




(a) Distribution (*crosses*) and cumulative distribution (*curve*) of the number of SCCs versus the size of SCC.




(b) Connectivity versus the number of removed terms.

**Fig. 7.** Connectivity of TDG

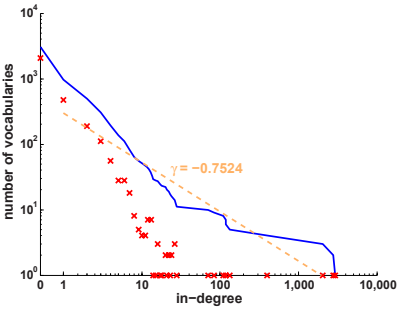
### 5.3 Connectivity Analysis

It is possible that a set of related terms are defined in respect of each other, i.e., they are reachable from each other in TDG and thus form an SCC. Figure 7(a) shows the distribution of sizes of SCCs. Most terms (93.4%) are within trivial SCCs (SCC with only one vertex), i.e., they are not involved in circular dependence. The largest SCC is with 14,883 terms in FMA, interlinked by class subsumption relations and property restriction structures. Although there are 10,994 non-trivial SCCs, only 23 of them are with terms in more than one vocabularies, out of which 22 are with terms in a "family" of vocabularies, i.e., a set of vocabularies that have a significantly long common prefix of URI namespaces, such as FOOD and VIN. The only real cross-vocabulary SCC is with 19 properties in DC, DCTERMS, and SKOS, including `dc:creator`, `dcterms:date`, `skos:note`, etc., each of which is used to describe some of the others.

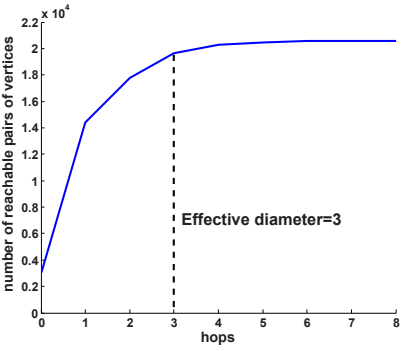
To further examine the connectivity of TDG, the terms of the highest degree are removed one at each step. Figure 7(b) shows that TDG is rapidly broken into over 40 thousand weakly connected components (WCCs) after only 16 terms are removed, revealing that the connectivity of TDG heavily depends on a few popular terms. Specifically, if all the terms in language-level vocabularies (RDF, RDFS, OWL, and DAML) are removed, the average in-degree/out-degree will decrease from 5.72 to 1.92.

## 6 Vocabulary Dependence

Out of the 7,312,657 arcs in TDG, 5,315,615 (72.7%) are between terms in different vocabularies. Thus, it is interesting to generalize the dependence from term level to vocabulary level, and study its characteristics.




(a) Distribution (crosses) and cumulative distribution (curve) of in-degrees.




(b) Hop plot and effective diameter.

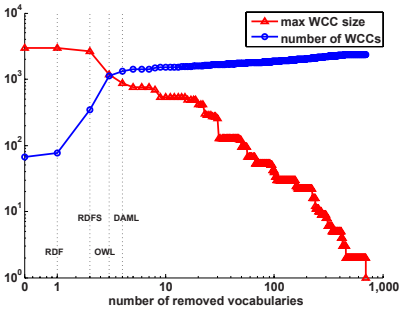
**Fig. 8.** In-degree distribution and hop plot of VDG

Dependence between vocabularies on the Semantic Web can also be characterized by a directed graph, called the *vocabulary dependence graph*, denoted by VDG $= \{\mathbb{V}, \mathbb{VD}\}$, where $\mathbb{V}$ is the vertex set, each vertex labeled with a vocabulary $v$; $\mathbb{VD}$ is the arc set, and an arc $\langle v_1, v_2 \rangle$ exists iff $\exists t_1 \in v_1.C \cup v_1.P, t_2 \in v_2.C \cup v_2.P$, $t_1$ directly depends on $t_2$. The VDG analyzed in this paper includes 3,039 vertices and 11,392 arcs (after removing self-loops). The average in-degree/out-degree is 3.75. Figure 8(a) shows the cumulative distribution of in-degrees of VDG, which approximates a power law with the exponent $\gamma = -0.7524$. The four vocabularies of the highest in-degree are RDF, RDFS, OWL, and DAML, all of which are language-level vocabularies. If these vocabularies are removed, the average in-degree/out-degree will decrease to 1.06. It exhibits that most dependence relations are attributed to the dependence to language-level vocabularies.

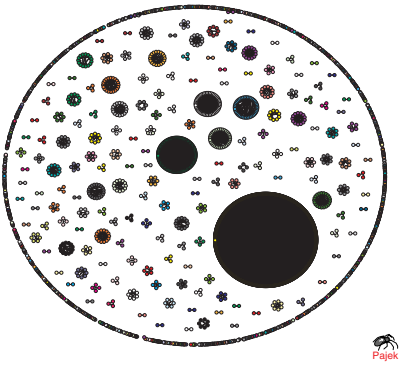
To measure the reachability and distance features of VDG, Fig. 8(b) shows its hop plot. Over half of all reachable pairs of vertices can reach from one to the other with no more than 1 hop, and the effective diameter is just 3, which is much smaller than 23, the effective diameter of TDG. It means that long-distance term dependence is principally within vocabularies.

To examine the connectivity of VDG, the vocabularies of the highest degree are removed one at each step. Figure 9(a) shows that VDG is totally fragmented to the isolation of single vertices only after 695 vocabularies (22.9%) are removed. Actually, VDG is rapidly broken into 1,320 WCCs just after four language-level vocabularies are removed, as depicted in Figure 9(b).[16] However, there is still a large WCC with 871 vocabularies (28.7%), mainly due to the use of annotation properties in DC and SKOS. Other small non-trivial WCCs are usually composed of families of vocabularies.

[16] This figure is generated by Pajek (`pajek.imfm.si`).




(a) Connectivity versus the number of removed vocabularies.




(b) WCCs of VDG after removing four language-level vocabularies.

**Fig. 9.** Connectivity of VDG

**Table 2.** Indicators of TDG and VDG before/after removing four language-level vocabularies

| Indicator | TDG | | | | VDG | | | |
|---|---|---|---|---|---|---|---|---|
| | before | | after | | before | | after | |
| | Avg. | Max. | Avg. | Max. | Avg. | Max. | Avg. | Max. |
| In-degree | 5.72 | 1,187,173 | 1.92 | 60,836 | 3.75 | 2,947 | 1.06 | 133 |
| Out-degree | 5.72 | 3,239 | 1.92 | 3,235 | 3.75 | 20 | 1.06 | 17 |
| #Reachable_from | 1,105 | 1,260,727 | 1,088 | 196,512 | 5.77 | 2,968 | 2.95 | 332 |
| #Reachable | 1,105 | 15,259 | 1,088 | 15,240 | 5.77 | 46 | 2.95 | 43 |
| Eccentricity | 10.05 | 48 | 9.55 | 48 | 1.77 | 8 | 1.25 | 8 |
| Effective diameter | 23 | | 22 | | 3 | | 3 | |
| $\gamma$ (in-degree) | -0.8245 | | | | -0.7524 | | | |

## 7 Conclusion

This paper proposed term dependence on the Semantic Web, based on the RDF sentence structure extracted from authoritative description of terms, and analyzed the complex network characteristics of the term dependence graph as well as the induced vocabulary dependence graph. Experiments were performed on a real data set collected by our Falcons search engine, which is much larger than those in previous graph analysis of the Semantic Web. The main results are summarized in Table 2. The data set, analyzed graphs, and statistical results are available online.[17]

We observed that term dependence on the Semantic Web forms a scale-free network, i.e., with a power-law degree distribution. The graph structure is very

[17] `http://iws.seu.edu.cn/projects/ontosearch/dependence_graph/`.

complex, and a change of the meaning of a term may affect a large amount of other terms (in average) through long-distance paths. However, complex structures mainly exist within vocabularies. To define terms, developers establish most cross-vocabulary dependence to language-level terms or other popular annotation properties, and they rarely link their terms to other domain vocabularies even on overlapped topics. The schema-level of the Semantic Web is still far away from a Web of interlinked ontologies, which indicates that ontologies are rarely reused and it will lead to difficulties for data integration.

In future work, as with the growth of the numbers of terms and vocabularies on the Semantic Web, their evolution model deserves to be investigated in the future. Besides, exploring the macrostructure of the instance level of the Semantic Web is also an attractive research topic.

**Acknowledgments.** The work is supported in part by the NSFC under Grant 60773106, and in part by the 973 Program of China under Grant 2003CB317004. We would like to thank Jun Ye for his effort in the experiments. We are also grateful to Weiyi Ge for his work in implementing the crawler.

## References

1. Adamic, L.A., Huberman, B.A., Barabási, A.-L., Albert, R., Jeong, H., Bianconi, G.: Power-Law Distribution of the World Wide Web. Science 287(5461), 2115a (2000)
2. Albert, R., Jeong, H., Barabási, A.-L.: Internet: Diameter of the World-Wide Web. Nature 401(6749), 130–131 (1999)
3. Berrueta, D., Phipps, J.: Best Practice Recipes for Publishing RDF Vocabularies. W3C Working Draft (2008)
4. Bray, T., Hollander, D., Layman, A., Tobin, R.: Namespaces in XML 1.0 (Second Edition). W3C Recommendation (2006)
5. Cheng, G., Ge, W., Qu, Y.: Falcons: Searching and Browsing Entities on the Semantic Web. In: 17th International Conference on World Wide Web, pp. 1101–1102. ACM Press, New York (2008)
6. Ding, L., Finin, T.: Characterizing the Semantic Web on the Web. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 242–257. Springer, Heidelberg (2006)
7. Ding, L., Zhou, L., Finin, T., Joshi, A.: How the Semantic Web is Being Used: An Analysis of FOAF Documents. In: 38th Annual Hawaii International Conference on System Sciences, pp. 113–113. IEEE Computer Society, Washington (2005)
8. Faloutsos, M., Faloutsos, P., Faloutsos, C.: On Power-Law Relationships of the Internet Topology. In: Annual Conference of the Special Interest Group on Data Communication, pp. 251–262. ACM Press, New York (1999)
9. Gil, R., García, R., Delgado, J.: Measuring the Semantic Web. AIS SIGSEMIS Bulletin 1(2), 69–72 (2004)
10. Hayes, P.: RDF Semantics. W3C Recommendation (2004)
11. Hoser, B., Hotho, A., Jäschke, R., Schmitz, C., Stumme, G.: Semantic Network Analysis of Ontologies. In: Sure, Y., Domingue, J. (eds.) ESWC 2006. LNCS, vol. 4011, pp. 514–529. Springer, Heidelberg (2006)

12. Ma, J., Chen, H.: Complex Network Analysis on TCMLS Sub-Ontologies. In: 3rd International Conference on Semantics, Knowledge and Grid, pp. 551–553. IEEE Computer Society, Washington, DC (2007)
13. Patel-Schneider, P.F., Hayes, P., Horrocks, I.: OWL Web Ontology Language Semantics and Abstract Syntax. W3C Recommendation (2004)
14. Theoharis, Y., Tzitzikas, Y., Kotzinos, D., Christophides, V.: On Graph Features of Semantic Web Schemas. IEEE Trans. Knowl. Data Eng. 20(5), 692–702 (2008)
15. Tummarello, G., Delbru, R., Oren, E.: Sindice.com: Weaving the Open Linked Data. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 552–565. Springer, Heidelberg (2007)
16. Tummarello, G., Morbidoni, C., Bachmann-Gmür, R., Erling, O.: RDFSync: Efficient Remote Synchronization of RDF Models. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 537–551. Springer, Heidelberg (2007)
17. Zhang, H.: The Scale-Free Nature of Semantic Web Ontology. In: 17th International Conference on World Wide Web, pp. 1047–1048. ACM Press, New York (2008)
18. Zhang, X., Cheng, G., Qu, Y.: Ontology Summarization Based on RDF Sentence Graph. In: 16th International Conference on World Wide Web, pp. 707–716. ACM Press, New York (2007)