Linked Open Knowledge Organization Systems*

Definition of a method for reducing the traversing†

Janneth Chicaiza
Univ. Técnica Particular de Loja
Loja, Ecuador
jachicaiza@utpl.edu.ec

Mariela Tapia-Leon
Universidad de Guayaquil
Guayaquil, Ecuador
mariela.tapial@ug.edu.ec

Nelson Piedra
Univ. Técnica Particular de Loja
Loja, Ecuador
nopiedra@utpl.edu.ec

Jorge Lopez-Vargas
Univ. Técnica Particular de Loja
Loja, Ecuador
jalopez2@utpl.edu.ec

Edmundo Tovar-Caro
Universidad Politécnica de Madrid
Madrid, Spain
etovar@fi.upm.es

ABSTRACT

The Web is the most used Internet's service to create and share information. In large information collections, Knowledge Organization plays a key role in order to classify and to find valuable information. Likewise, Linked Open Data is a powerful approach for linking different Web datasets. Today, several Knowledge Organization Systems are published by using the design criteria of linked data, it facilitates the automatic processing of them. In this paper, we address the issue of traversing open Knowledge Organization Systems, considering difficulties associated with their dynamics and size. To fill this issue, we propose a method to identify irrelevant nodes on an open graph, thus reducing the time and the scope of the graph path and maximizing the possibilities of finding more relevant results. The approach for graph reduction is independent of the domain or task for which the open system will be used. The preliminary results of the proof of concept lead us to think that the method can be effective when the coverage of the concept of interest increases.

CCS CONCEPTS

• **Computing methodologies** → **Knowledge representation and reasoning**; • **Information systems** → *Web searching and information discovery*; Information retrieval query processing;

KEYWORDS

Graph traversing, RDF, KOS, DBPedia, SPARQL

ACM Reference Format:
Janneth Chicaiza, Mariela Tapia-Leon, Nelson Piedra, Jorge Lopez-Vargas, and Edmundo Tovar-Caro. 2019. Linked Open Knowledge Organization Systems: Definition of a method for reducing the traversing. In *2nd International Conference on Applications of Intelligent Systems (APPIS 2019), January 7–9, 2019, Las Palmas de Gran Canaria, Spain.* ACM, New York, NY, USA, Article 40, 6 pages. https://doi.org/10.1145/3309772.3309804

*Produces the permission block, and copyright information
†The full version of the author's guide is available as `acmart.pdf` document

APPIS 2019, January 7–9, 2019, Las Palmas de Gran Canaria, Spain

ACM ISBN 978-1-4503-6085-2/19/01…$15.00
https://doi.org/10.1145/3309772.3309804

1 INTRODUCTION

The Web is the most used Internet's service to create and share information. Due to the large volume of heterogeneous and unstructured content, users have problems to integrate, reuse and find valuable information. In this context, Knowledge Organization (KO) plays a key role [9] because of digital libraries and other large information collections require adequate and thorough structuring systems in order to organize the increasing amount of available data and to avoid loss of information [11].

Likewise, in order to facilitate the automatic processing of data, Semantic Web technologies can be used to publish data according to the Linked Data (LD) design criteria proposed by Berners-Lee. Currently, Linked Data on the Web represents an immense source of knowledge suitable to be automatically processed and queried [3].

The potential of Linked Data lies in leveraging the properties that connect two entities (e.g., people, organizations, places, digital resources) described by RDF as a graph-based data model [3]. Therefore, the discovery of such connecting properties (or paths) is often a fundamental task in Web of Data[6] and it is a problem to resolve considering the increasing trend in sharing and interlinking pieces of structured linked data on the Web [3].

In LD, to navigate or query RDF datasets, the W3C recommends the SPARQL language. In the latest version, SPARQL has been extended with a navigational feature called property paths (PPs). However, traversing open KOS, considering their size, hinder to find nodes or optimized paths. Furthermore, the results can be imprecise or ambiguous for users, especially when the knowledge area is broad. This restriction limits the applicability of PPs for querying KOSs published as Linked Data. To fill this issue, we propose a method to identify irrelevant nodes on an open KOS, thus reducing the time and the scope of the graph path and maximizing the possibilities of finding more relevant results.

Next, we first introduce a family of reachability-based query semantics for PPs that distinguish between navigation on the Web and navigation at the data level. Thereafter, we consider another, alternative query semantics that couples Web graph navigation and data level navigation; we call it context-based semantics. Given these semantics, we find that for some PP-based SPARQL queries a complete evaluation.

Next, we first introduce KOSs and the semantic web technologies used to publish them and query. Thereafter, we define the underlying problem to the open KOSs navigation; in this context, we present the method to traverse the underlying graphs. Finally, we performed preliminary tests using the topic recommendation as a proof of concept.

2 THEORETICAL BACKGROUND

2.1 Knowledge Organization Systems

According to Hjorland [10], Knowledge Organization (KO) can be defined from two points of view: 1) in the narrow meaning, a KO is an intellectual or cognitive organization of knowledge, i.e. KO is about activities such as document description, indexing and classification performed in libraries, bibliographical databases, archives and other kinds of institutional memories, and; 2) in the broader meaning, KO is about the social division of mental labor, i.e. the organization of universities and other institutions for research and higher education, the structure of disciplines and professions, the social organization of media, the production and dissemination of knowledge, etc.

Knowledge Organization as a field of study is concerned with the knowledge organizing systems (KOS) [10]. A KOS is a system of related concepts of a given knowledge domain [12]. According to Hodge [4] this term is intended to encompass all types of schemes for organizing information and promoting knowledge management, such as thesauri, classification schemes, gazetteers, lexical databases, subject heading lists, ontologies and folksonomies.

Many institutions develop and maintain KOSs as backbone structures for their information systems. The potential of such KOSs is to serve as components in knowledge-rich applications [1]. In the Web, one key role of a controlled vocabulary is to improve precision of information retrieval from an indexed collection. "The hierarchical and associative relationships between concepts of a thesauri enable users to browse for search terms, and information provided by search engines that use this structure to automatically expand queries, thus improving recall." [1]

2.2 Publication of KOS as Linked Data

Linked Open Data (LOD) is a powerful approach for linking different Web datasets, by creating new value of data through mash-up over such various datasets [13]. Today, several datasets are published using the Linked Data design criteria. According to *The Linked Open Data (LOD) Cloud*[1], until august of 2018, there are 1.224 data collections published. Data in domains such as Geography, Government, Life Sciences, Linguistics, Media, Publications and others are part of the Web of Data.

In Linked Data, KOSs are published in RDF format according to Simple Knowledge Organization System (SKOS) Vocabulary. Since 18 August 2009, SKOS is the W3C Recommendation[2] to publish KOS on the web. In [7], SKOS Core is introduced, "an RDF vocabulary for expressing the basic structure and content of concept schemes"

In the W3C page about SKOS/Datasets[3], several concept schemes described with SKOS are identified. Some KOSs are general, i.e. cover wide knowledge areas (e.g. the UNESCO Thesaurus[4]), other systems are specific for describing a given domain (e.g. AGROVOC Agricultural Thesaurus[5]). In these cases, KOSs are named controlled vocabularies because of a group of experts have elaborated a consensual schema to knowledge organization.

Apart from controlled schemes, in the Web of Data, there are open knowledge systems created collaboratively by people. DBPedia[6] is the most popular project based-on the Wikipedia's categories. Currently, there are more than 1 million, 400 thousand linked concepts can be queried on DBPedia.

When the concepts of two schemes of different nature (one controlled and the other open) are linked, fundamental tasks of semantic information retrieval such as annotation and classification of Web content [2] can be improved.

2.3 Querying KOS through property paths

Since March 2013, the W3C recommends SPARQL[7] as the language to access and query RDF data. By means of web services named *SPARQL endpoints*, a mashup of linked data connected through various linking properties can be retrieved [6].

The last version of SPARQL 1.1 supports basic operations for RDF data management such as query, aggregation, update and delete. Also, this language supports Paths Properties (PPs)[8] which are used to traverse iteratively RDF graphs. Kostylev, Reutter, Romero, and Vrgoč define a PP as a feature to "search through the RDF graph for a sequence of IRIs that form a path conforming to a regular expression." This kind of features consists in apply operators on the graph's properties in order to build SPARQL queries that are agnostic of the underlying RDF graph structure [8].

There are eight basic SPARQL PPs, four can be applied to a unique property and the remaining require two properties. Using unary operators such as *, ?, +, and {} forms, we can retrieve all successors or ancestors of a given SKOS concept. For example, the pattern:

dbc:Artificial_intelligence skos:broader{1, 5} ?child

returns a list of all the subjects (sub-concepts) that are within a distance of up to 5 jumps of the *Artificial Intelligence* (AI) discipline.

From a theoretical point of view, a PP enables indefinitely to traverse link properties until to reach a target node [5]. However, if a graph is very deep or spread, traverse it can be much costly. Therefore, the performance of queries with PP should try to be optimized. In the next section, we present the issues related to navigate open KOS, and we propose a method for reducing the coverage of traverse the underlying graph.

[1] https://lod-cloud.net/
[2] https://www.w3.org/TR/skos-reference/
[3] https://www.w3.org/2001/sw/wiki/SKOS/Datasets
[4] http://skos.um.es/unescothes/
[5] http://aims.fao.org/
[6] http://dbpedia.org
[7] https://www.w3.org/TR/sparql11-overview/
[8] https://www.w3.org/TR/sparql11-property-paths/

3 TRAVERSING OPEN KOS

3.1 The problem statement

There is numerous research carried out in traversing and/or finding paths in a graph. Although it is a well-studied problem, it is also highly intractable [6]. Particularly, accessing and traversing open KOSs can be a challenging task, especially when path properties are applied to traverse the graph.

Open KOSs, as DBPedia's categories, remain updated thanks to people define new concepts and new associations between them. But, an issue with them is the number and variety of concepts (nodes) and relationships. This increases the extension and dispersion of the graph, making it difficult to converge to a target concept and to find a resource of interest.

In this paper, the focus is to reduce the task of traversing rich and open-domain KOS by identifying nodes that do not support information of value to the domain or that are irrelevant for a given context.

Traversing the sub-graph of the *Artificial_Intelligence* concept (AI), according to DBPedia, 28 direct successors (i.e. sub-disciplines) are retrieved. If the goal is to use hierarchies of concepts to recommend those related to a given subject of interest, around 36% is irrelevant information. Concepts related to fiction, laboratories, researchers, associations, conferences, publications, philosophy or history of AI could be discarded if the goal is to identify the closest topics related to IA as a knowledge field.

3.2 General approach

In Information Retrieval, some words (called *stopwords*), which are the most common in the text, are not generally indexed due they lack meaning by themselves. Search engines need to work quickly; therefore, if they do not process stopwords it represents a considerable saving. Similarly, if during the search or navigation by a topics network, we identifying the nodes (SKOS concepts) that are not relevant to the task or the context in which they will use, then the exploration of KOS graphs could be optimized. In this section, the proposed method to achieve this goal is explained.

The Figure 1 shows the general approach which includes two stages: 1) to identify and mark stop-concepts thus to avoid to visit them when the graph is explored; and, 2) to traverse the bounded graph using property paths.

3.2.1 Graph reduction (GR). The graph reduction consists in analyzing the graph connectivity associated with open KOS in order to identify the nodes (concepts) which are more popular in a dataset but they are irrelevant for a domain or to do a given task. For example, in the figure 1, these nodes are filled with orange color and corresponding to those with the greatest amount of relationships with semantic resources.

This stage could be performed at the first time and to be updated when it be necessary. As a result of this analysis the set of stop-concepts will remain marked with a flag, and thus, they will be not visited when traversing a graph of open KOS.

To mark stop-concepts, we propose two main steps:

(1) To identify the set of nodes with the greatest degree on the graph. An understanding of the data model used in the dataset is fundamental in order to define the best property patterns (including single properties or a combination of them) to be used to summarize results.
(2) To identify irrelevant entities (stop entities) linked to the most popular concepts. A simple method could be used to do this task. First, tokenize the labels of the most linked concepts (identified in the previous task), as a result, a list of the most common words on the dataset will be generated. Later, a manual or semi-automatic exploration by the top results will help to mark the irrelevant entities (i.e. a group of people, objects, etc.).

From the list of entities not relevant to the domain, or goal in which the open KOS will be used, it is possible to optimize the graph traversing, avoiding to expand nodes which refer to stop entities. For example, to recommend topics related to the DBPedia's category *Artificial Intelligence*, not-relevant entities such as associations, people, companies, etc., will not be visited when the graph is traversed.

3.2.2 RDF graph traversing (GT). To explore RDF graphs, SPARQL provides unary operators that traverse it using arbitrary iterations. Although the use of these properties is very easy, we propose to iterate the graph but visiting, at each step, the nodes that are at a progressive distance from the objective, starting with 1 until reaching a limit, that implies:

(1) To establish the max length (l) to navigate the graph (n)
(2) To apply operator m to traverse the graph. The value of m starts with 1, and increments in 1 progressively until it reaches l.
(3) In each iteration by the graph, the concepts related to stop-entities are marked to be not visited in the next step, i.e. these concepts will be considered as a kind of stop-nodes.

Following the three-step process described, the path through the graph could be optimized. The figure 1 (part b) identifies stop-nodes with orange color. Although these nodes have offspring, they will not be visited by the algorithm.

4 A PROOF OF CONCEPT: TRAVERSING DBPEDIA'S CATEGORIES

Open KOS have many applications, especially in the field of information retrieval. DBPedia is one of the data sources that offers great support to annotate, classify [2] and find valuable information. In this section we explain the path optimization process of this graph, considering as a potential application the recommendation of topics related to a term of interest. Specifically, we chose topics of different granularity: starting with Computer Science, then Artificial Intelligence, and finally Machine Learning.

4.1 Graph reduction (GR)

To identify irrelevant nodes (stop-concepts) for the recommendation of topics, we applied the method explained in section 3.2.1.

4.1.1 Identify the set of nodes with the greatest degree of the graph. In this step, we select the property *subject* of Dublin Core Terms (dct) to identify concepts more popular. In DBPedia, *dct:subject* is used to connect any resource to its the topic, i.e., an instance of *skos:Concept*. Executing the next query, we collect 30 thousand of concepts with the greatest connectivity with DBPedia resources.

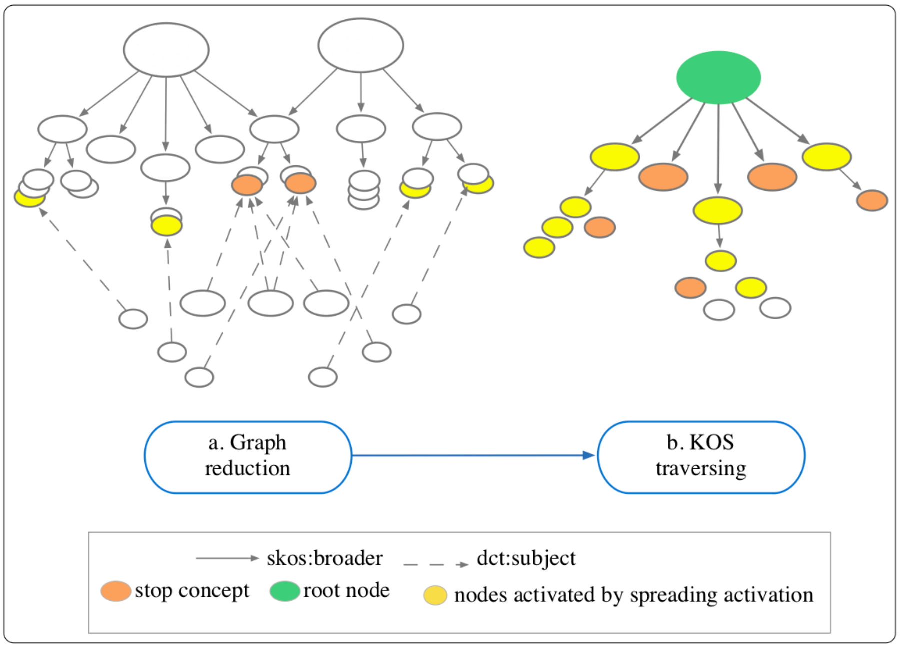

Figure 1: General approach for traversing open KOS

```
SELECT ?c COUNT(*) as ?n WHERE {
    ?c a skos:Concept .
    [] dct:subject ?c .
} GROUP BY ?c ORDER BY DESC(?n)
```

The top-ten concepts more linked to DBPedia resources are listed below. The number between parenthesis stats the amount of individuals which are connected to each *skos:Concept*.

(1) *dbc:Living_people* (795708)
(2) *dbc:Year_of_birth_missing_(living_people)* (61612)
(3) *dbc:American_films* (42991)
(4) *dbc:Place_of_birth_missing_(living_people)* (38289)
(5) *dbc:English-language_films* (36403)
(6) *dbc:Association_football_midfielders* (22063)
(7) *dbc:Minor_planet_redirects* (21730)
(8) *dbc:English_Football_League_players* (21072)
(9) *dbc:Year_of_birth_unknown* (20797)
(10) *dbc:Named_minor_planets* (20244)

4.1.2 Identify stop-concepts. We tokenized the label of each concept obtained in the previous step and filtered the most popular words in order to identify the least relevant. In this case, words such as players, researchers, conferences, people, films, etc. were setting with the status of stop-entity. From this list, which can be completed or update later, the next step is performed. We have validated and manually marked a list of less than 200 stop-entities.

As we mentioned in 3.2.1, this stage can be performed only once to initialize the system. Then, the execution of the method for traverse the DBPedia graph was performed for each concept to analyze: Computer Science (CS) -> Artificial Intelligence (AI) -> Machine Learning (ML).

4.2 RDF Graph traversing (GT)

To identify concepts related to other of interest, we established several limits to traverse the graph. The next query was execute for 4 times, i.e. the parameter for the property *skos:broader* began with a value of 1 until 4. In addition, for each concept of interest (CS, AI and ML), the variable *?r* was updated.

```
SELECT ?c COUNT(*) AS ?n WHERE {
    VALUES ?r {dbc:Artificial_intelligence}.
    ?c skos:broader{4} ?r .
} GROUP BY ?c ORDER BY DESC(?n)
```

In each iteration, the concepts related to the stop-entities were not expanded. The table 1 summarizes the quantity of nodes visited in each iteration before and after we applied the reduction method (GP).

4.3 Results

The Figure 2 shows performance of the method based-on SPARQL queries using PP. The best results (highest rate of reduction) was obtained for the most general concept, Computer Science (CS). At the contrary, for the most specific concept, Machine Learning (ML), the graph reduction were almost imperceptible. Therefore, the approach to traverse an open graph is more effective when analyzing broader concepts.

In the case of Computer Science, regardless of the iteration, the graph path is better defined, avoiding traverse around 50% of the neighboring nodes.

In order to identify the top concepts s related to each field of interest (CS, AI, LM), we define a basic metric (see expression (i))

Table 1: Nodes by concept and iteration

	Computer Science (CS)		Artificial Intelligence (AI)		Machine learning (ML)	
iteration	After GR	With GP	After GR	With GP	After GR	With GP
1	14	5	28	21	27	25
2	145	70	200	177	20	20
3	1086	359	457	384	10	10
4	5571	2218	1159	738	3	3

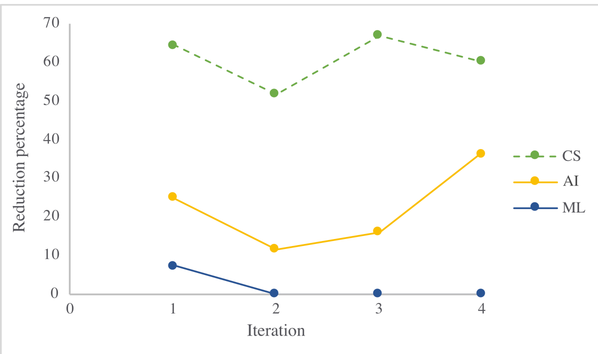

Figure 2: Results of the method

to compute the weight of each concept retrieve during the graph traversing.

Definition 4.1. If s_i is the ith successor (sub-concept) found when traversing the open graph, then its weight W is computed according to the sum of all occurrences F of this node in a given iteration j multiplied by the weight W associated with each iteration p^j:
$W(s_i) = \sum_{j=1}^{4} F_j(s_i) * W(p^j)$

For this proof of concept, we established weights W for the set of iterations P according to the distance to the root node, i. e.:
$(\vec{P}, \vec{W}) = \{(1, 1), (2, 0.75), (3, 0.5), (4, 0.25)\}$

The Table 2 presents a list of recommendations for Computer Science. Note the differences between the concepts recommended before and after the method were applied.

In an intuitive way, it can be seen that the best results were obtained after making the delimitation of the graph path. This is possible considering that from the stop-entities (conferences, publications, awards, year, etc.), helped to identify the stop-nodes, i.e. concepts such as Formal methods publications, Artificial intelligence conferences, among others.

5 CONCLUSIONS

In this paper, we address the main issue related to query open Knowledge Organizations Systems by reducing the potential paths to navigate. KOSs created collaboratively by people can be complex, considering their depth, breadth or connectivity.

Using external links between different KOSs, the machines can automatically annotate and classify digital resources. With the enriched dataset, search or recommendation engines will be able to discover more interesting facts in order to offer more accurate and complete results to the users of the network.

In this context, we propose a method to reduce the navigation by a graph and then traverse it using SPARQL property paths. The approach for graph reduction is independent of the domain or task for which the open KOS will be used.

Finally, we expose a scenario in which the proposed method is very useful. The preliminary results of the proof of concept lead us to think that as the coverage of the concept of interest increases, the application of pruning or reduction of the path is a fundamental task in order to obtain more precise results and in the best possible performance.

Currently, we continue testing the proposed method in tasks related to information retrieval and later, we will expand the validation including other usage scenarios.

ACKNOWLEDGMENTS

This research has been partially funded by the scholarship provided by the "Secretaría Nacional de Educación Superior, Ciencia y Tecnología" of Ecuador (SENESCYT).

REFERENCES

[1] Thomas Baker, Sean Bechhofer, Antoine Isaac, Alistair Miles, Guus Schreiber, and Ed Summers. 2013. Key choices in the design of Simple Knowledge Organization System (SKOS). *Web Semantics: Science, Services and Agents on the World Wide Web* 20 (2013), 35 – 49. https://doi.org/10.1016/j.websem.2013.05.001
[2] Janneth Chicaiza, Nelson Piedra, Jorge Lopez-Vargas, and Edmundo Tovar-Caro. 2014. Domain Categorization of Open Educational Resources Based on Linked Data. In *Knowledge Engineering and the Semantic Web*. Springer International Publishing, 15–28.
[3] Olaf Hartig and Giuseppe Pirr š. 2016. SPARQL with property paths on the Web. *Semantic Web* 8, 6 (2016), 773–795. Cited By :1.
[4] Gail Hodge. 2000. *Systems of knowledge organization for digital libraries: beyond traditional authority files.* Digital Library Federation Washington, D.C. vi, 37 p. ; pages. http://www.clir.org/cpa/reports/pub91/contents.html
[5] Egor Kostylev, Juan Reutter, Miguel Romero, and Domagoj Vrgoč. 2015. SPARQL with Property Paths. In *The Semantic Web - ISWC 2015*, Marcelo Arenas, Oscar Corcho, Elena Simperl, Markus Strohmaier, Mathieu d'Aquin, Kavitha Srinivas, Paul Groth, Michel Dumontier, Jeff Heflin, Krishnaprasad Thirunarayan, and Steffen Staab (Eds.). Springer International Publishing, Cham, 3–18.
[6] Qaiser Mehmood, Alokkumar Jha, Dietrich Rebholz-Schuhmann, and Ratnesh Sahay. 2018. FedS: Towards Traversing Federated RDF Graphs. In *Big Data Analytics and Knowledge Discovery - 20th International Conference, DaWaK 2018, Regensburg, Germany, September 3-6, 2018, Proceedings*. 34–45. https://doi.org/10.1007/978-3-319-98539-8_3
[7] Alistair Miles, Dan Brickley, Brian Matthews, and Michael Wilson. 2005. SKOS core: Simple knowledge organisation for the web. In *Proceedings of the International Conference on Dublin Core and Metadata Applications*. Madrid, 3–10.
[8] Ioannis Papadakis, Michalis Stefanidakis, Phivos Mylonas, Brigitte Niggemeyer, and Spyridon Kazanas. 2016. *Top-k shortest paths in large typed RDF datasets challenge.* Communications in Computer and Information Science, Vol. 641. 191–199 pages. Cited By :3.
[9] Vreda Pieterse and Derrick Kourie. 2014. Lists, Taxonomies, Lattices, Thesauri and Ontologies: Paving a Pathway Through a Terminological Jungle. *Knowl. Org.* 41, 3 (2014), 217–229. https://doi.org/10.5771/0943-7444-2014-3-217
[10] Birger Hj ÿrland. 2008. What is Knowledge Organization (KO)? *Knowl. Org.* 35, 2 (2008), 86–101. https://doi.org/10.5771/0943-7444-2008-2-3-86

Table 2: Recommendations for Computer Science

Rank	Top concepts without GR	Top concepts with GP
1	Formal methods publications	Formal methods
2	Artificial intelligence conferences Formal methods	Automated theorem proving Cross-platform software Models of computation
3	Automated theorem proving Cross-platform software Models of computation	Computer programming tools
4	Computer programming tools Software by year Turing Award laureates	Denotational semantics Formal methods tools Program derivation Quantum computing Substitution (logic) Turing machine
5	20th-century software 21st-century software Denotational semantics Formal methods organizations Formal methods people Formal methods tools Mac OS Program derivation Programming languages created in the 21st century Quantum computing Substitution (logic) Turing machine Video games by year	Abstract data types Adobe Flash Artificial intelligence Formal methods terminology Multi-agent systems Program analysis Program logic Programming languages Web development software

[11] Lena-Luise Stahn, Ingetraut Dahlberg, and Ernesto William De Luca. 2017. Knowledge Organisation for Digital Libraries. In *Proceedings of the 17th European Networked Knowledge Organization Systems Workshop co-located with the 21st International Conference on Theory and Practice of Digital Libraries 2017 (TPDL 2017), Thessaloniki, Greece, September 21st, 2017*. 6–13.

[12] Wolfgang G. Stock. 2010. Concepts and Semantic Relations in Information Science. *J. Am. Soc. Inf. Sci. Technol.* 61, 10 (Oct. 2010), 1951–1969. https://doi.org/10.1002/asi.v61:10

[13] Atsuko Yamaguchi, Kouji Kozaki, Kai Lenz, Yasunori Yamamoto, Hiroshi Masuya, and Norio Kobayashi. 2016. Data Acquisition by Traversing Class-Class Relationships over the Linked Open Data. In *Proceedings of the ISWC 2016 Posters & Demonstrations Track co-located with 15th International Semantic Web Conference (ISWC 2016), Kobe, Japan, October 19, 2016.* http://ceur-ws.org/Vol-1690/paper95.pdf