

The Computer Science Ontology: A Large-Scale Taxonomy of Research Areas

Angelo A. Salatino, Thiviyan Thanapalasingam, Andrea Mannocci,
Francesco Osborne, Enrico Motta

Knowledge Media Institute, The Open University, MK7 6AA, Milton Keynes, UK
`{firstname.lastname}@open.ac.uk`

Abstract. Ontologies of research areas are important tools for characterising, exploring, and analysing the research landscape. Some fields of research are comprehensively described by large-scale taxonomies, e.g., MeSH in Biology and PhySH in Physics. Conversely, current Computer Science taxonomies are coarse-grained and tend to evolve slowly. For instance, the ACM classification scheme contains only about 2K research topics and the last version dates back to 2012. In this paper, we introduce the Computer Science Ontology (CSO), a large-scale, automatically generated ontology of research areas, which includes about 26K topics and 226K semantic relationships. It was created by applying the Klink-2 algorithm on a very large dataset of 16M scientific articles. CSO presents two main advantages over the alternatives: i) it includes a very large number of topics that do not appear in other classifications, and ii) it can be updated automatically by running Klink-2 on recent corpora of publications. CSO powers several tools adopted by the editorial team at Springer Nature and has been used to enable a variety of solutions, such as classifying research publications, detecting research communities, and predicting research trends. To facilitate the uptake of CSO we have developed the *CSO Portal*, a web application that enables users to download, explore, and provide granular feedback on CSO at different levels. Users can use the portal to rate topics and relationships, suggest missing relationships, and visualise sections of the ontology. The portal will support the publication of and access to regular new releases of CSO, with the aim of providing a comprehensive resource to the various communities engaged with scholarly data.

Resource Type: Dataset, Ontology of Research Areas

Permanent URL: <https://cso.kmi.open.ac.uk/>

License: CC BY 4.0 International License

Keywords: Scholarly Data, Ontology Learning, Bibliographic Data, Scholarly Ontologies.

1 Introduction

Ontologies have proved to be powerful solutions to represent domain knowledge, integrate data from different sources, and support a variety of semantic applications [1–5]. In the scholarly domain, ontologies are often used to facilitate the integration of large datasets of research data [6], the exploration of the academic landscape [7], information extraction from scientific articles [8], and so on. Specifically, ontologies

that describe research topics and their relationships are invaluable tools for helping to make sense of research dynamics [7], to classify publications [3], to characterise [9] and identify [10] research communities, and to forecast research trends [11].

Some fields of research are well described by large-scale and up-to-date taxonomies, e.g., MeSH in Biology and PhySH in Physics. Conversely, current Computer Science taxonomies are coarse-grained and tend to evolve slowly. For instance, the current version of ACM classification scheme, containing only about 2K research topics, dates back to 2012 and superseded its 1998 release.

In this paper, we present the *Computer Science Ontology (CSO)*, a large-scale, granular, and automatically generated ontology of research areas which includes about 26K topics and 226K semantic relationships. CSO was created by applying the Klink-2 algorithm on a dataset of 16M scientific articles in the field of Computer Science [12]. CSO presents two main advantages over the alternatives: i) it includes a very large number of topics that do not appear in other classifications, and ii) it can be updated automatically by running Klink-2 on recent corpora of publications. Its fine-grained representation of research topics is very useful for characterising the content of research papers at the granular level at which researchers typically operate. For instance, CSO characterises the Semantic Web according to 40 sub-topics, such as Linked Data, Semantic Web Services, Ontology Matching, SPARQL, OWL, SWRL, and many others. Conversely, the ACM classification simply contains three related concepts: “Semantic web description languages”, “Resource Description Framework (RDF)”, and “Web Ontology Language (OWL)”.

CSO was initially created in 2012 and has been continuously updated over the years. During this period, it has supported a range of applications and approaches for community detection, trend forecasting, and paper classification [10, 11, 13]. In particular, CSO powers two tools currently used by the editorial team at Springer Nature (SN): Smart Topic Miner [3] and Smart Book Recommender [14]. The first is a semi-automatic tool for annotating SN books both by means of topics drawn from CSO and tags selected from the internal classification used at Springer Nature. The latter is an ontology-based recommender system that suggests books, journals, and conference proceedings to market at specific venues.

We are now releasing the Computer Science Ontology, so that the relevant communities can take advantage of it and use it as a comprehensive and granular semantic resource to support the development of their own applications. To facilitate its uptake, we have developed the *CSO Portal*, a web application that enables users to download, explore, and provide granular feedback on CSO. The portal offers three different interfaces for exploring the ontology and examining the network of relationships between topics. It also allows users to rate both topics and relationships between topics, as well as suggesting new topics and relationships. The feedback from the community will be considered by an editorial board of domain experts and used to generate new versions of CSO.

We intend to regularly release new versions of CSO that will incorporate user feedback and new knowledge extracted from recent research output. Our aim is to provide a comprehensive solution for describing the Computer Science landscape that will benefit researchers, companies, organisations, and research policy makers.

The paper is structured as follows. In Section 2, we discuss the related work, pointing out the existing gaps. In Section 3, we describe the Computer Science Ontology, the applications that adopted it, and how it was evaluated. In Section 4, we discuss the CSO

Portal and the relevant use cases. Finally, in Section 5 we summarise the main conclusions and outline future directions of work.

2 Related Work

Ontologies and taxonomies of research topics can support a variety of applications, such as dataset integration, the exploration process in digital libraries, the production of scholarly analytics, and modelling research dynamics [3].

In the field of Computer Science, the best-known taxonomy is the ACM Computing Classification System¹, developed and maintained by the Association for Computing Machinery (ACM). However, this taxonomy suffers from several limitations: in particular, it contains only about 2K research topics and it is developed manually. This is an extremely slow and expensive process and, as a result, its last version dates back to 2012. Hence, while the ACM taxonomy has been adopted by many publishers, in practice it lacks both depth and breadth and releases quickly go out of date.

In the field of Physics and Astronomy, the most popular solution used to be the Physics and Astronomy Classification Scheme (PACS)², replaced in 2016 by the Physics Subject Headings (PhySH)³. PACS used to associate alphanumerical codes to each subject heading to indicate their position within the hierarchy. However, this setup made its maintenance quite complex and the American Institute of Physics (AIP) discontinued it in 2010. Afterwards, the American Physical Society (APS) developed PhySH, a new classification scheme that has the advantage of being crowdsourced with the support of authors, reviewers, editors and organisers of scientific conferences, so that it is constantly updated with new terms.

The Mathematics Subject Classification (MSC)⁴ is the main taxonomy used in the field of Mathematics. This scheme is maintained by Mathematical Reviews and zbMATH and it is adopted by many mathematics journals. It consists of 63 macro-areas classified with two digits: each of them is further refined into over 5K three- and five-digit classifications representing their sub-areas. The last version dates back to 2010 and typically a new official version is released every ten years.

The Medical Subject Heading (MeSH)⁵ [15] is the standard solution in the field of Medicine. It is maintained by the National Library of Medicine of the United States and it is constantly updated by collecting new terms as they appear in the scientific literature.

The JEL⁶ classification scheme is the most used classification in the field of Economics. The JEL scheme was created by the Journal of Economic Literature of the American Economic Association. Its last major revision dates back to 1990, but in the last years there have been many incremental changes to reflect the advances in the field [16].

¹ The ACM Computing Classification System: <http://www.acm.org/publications/class-2012>.

² Physics and Astronomy Classification Scheme: <https://publishing.aip.org/publishing/pacs>.

³ PhySH - Physics Subject Headings: <https://physh.aps.org/about>.

⁴ 2010 Mathematics Subject Classification: <https://mathscinet.ams.org/msc/msc2010.html>.

⁵ MeSH - Medical Subject Headings: <https://www.nlm.nih.gov/mesh>.

⁶ Journal of Economic Literature: <https://www.aeaweb.org/econlit/jelCodes.php>.

The Library of Congress Classification⁷ is a system of library classification that encompasses many areas of science. It was developed by the Library of Congress and it is used to classify books within large academic libraries in USA and several other countries. However, it is much too shallow to support the characterisation of scientific research at a good level of granularity. For instance, the field of Computer Science is covered by only three topics: Electronic computers, Computer science, and Computer software.

A common limitation of most of these taxonomies is that, being manually crafted and maintained by domain experts, they tend to evolve relatively slow and therefore become quickly outdated. To cope with this issue, some institutions (e.g., the American Physical Society) are crowdsourcing their classification scheme. However, the crowdsourcing strategy also suffers from limitations, such as trust and reliability [17].

A complementary strategy is to automatically or semi-automatically generate these classifications using data driven methodologies. In the literature, we can find a variety of approaches for learning taxonomies or ontologies based on natural language processing [18], clustering techniques [19], statistical methods [20], and so on. For instance, Text2Onto [18] is a framework for learning ontologies from a collection of documents. This approach identifies synonyms, sub-/superclass hierarchies, etc. through the application of natural language processing techniques on the sentence structure, where phrases like “such as...” and “and other...” imply a hierarchy between terms. This method presents some similarities with the Klink-2 algorithm [12], but requires the full text of documents. TaxGen [19] is another approach to the automatic generation of a taxonomy from a corpus by means of a hierarchical agglomerative clustering algorithm and text mining techniques. The clustering algorithm first identifies the bottom clusters by observing the linguistic features in the documents, such as co-occurrences of words, names of people, organisations, domain terms and other significant words from the text. Then the clusters are aggregated creating higher-level clusters, which form the hierarchy. This strategy is similar to the one adopted by Klink-2 for inferring the *relatedEquivalent* relationships. Another approach to automatically create categorisation systems is the subsumption method [20], which computes the conditional probability for a keyword to be associated with another based on their co-occurrence. Given a pair of keywords, this system tries to understand whether there is a subsumption relationship between them, according to certain heuristics. However, this approach is limited to the statistical analysis on the co-occurrence keywords, while Klink-2 goes further by also taking advantage of external sources. It is also possible to combine ontology learning and a crowdsourcing strategy by developing approaches that take in account both statistical measures and user opinions [21, 22]. For instance, Wohlgenannt et al [21] combine human effort and machine computation by crowdsourcing the evaluation of an automatically generated ontology with the aim of dynamically validating the extracted relations.

3 The Computer Science Ontology

The Computer Science Ontology is a large-scale ontology of research areas that was automatically generated using the Klink-2 algorithm [12] on the Rexplore dataset [7].

⁷ Library of Congress Classification: <https://www.loc.gov/catdir/cpsolcc.html>.

This consists of about 16 million publications, mainly in the field of Computer Science. Some relationships were also refined manually by domain experts during the preparation of two ontology-assisted surveys in the fields of Semantic Web [23] and Software Architecture [13].

The current version of CSO includes 26K topics and 226K semantic relationships. The main root of CSO is Computer Science; however, the ontology includes also a few secondary roots, such as Linguistics, Geometry, Semantics, and so on.

The CSO data model⁸ is an extension of the BIBO ontology⁹, which in turn builds on SKOS¹⁰. It includes five semantic relations:

- *relatedEquivalent*, which indicates that two topics can be treated as equivalent for the purpose of exploring research data (e.g., Ontology Matching and Ontology Mapping). For the sake of avoiding technical jargon, in the CSO Portal this predicate is referred to as *alternative label of*.
- *skos:broaderGeneric*, which indicates that a topic is a super-area of another one (e.g., Semantic Web is a super-area of Linked Data). This predicate is referred to as *parent of* in the portal. The inverse relation (*child of*) is instead implicit.
- *contributesTo*, which indicates that the research output of one topic contributes to another. For instance, research in Ontology Engineering contributes to Semantic Web, but arguably Ontology Engineering is not a sub-area of Semantic Web – that is, there is plenty of research in Ontology Engineering outside the Semantic Web area.
- *rdf:type*, this relation is used to state that a resource is an instance of a class. For example, a resource in our ontology is an instance of topic.
- *rdfs:label*, this relation is used to provide a human-readable version of a resource's name.

The Computer Science Ontology is available for download in various formats (OWL, Turtle, and CSV) from <https://cso.kmi.open.ac.uk/downloads>. This ontology is licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0)¹¹ meaning that everyone is allowed to:

- copy and redistribute the material in any medium or format;
- remix, transform, and build upon the material for any purpose, even commercially.

In the following subsection, we will discuss the automatic generation of CSO with the Klink-2 algorithm (Section 3.1), the applications adopting it (Section 3.2), and how it was evaluated (Section 3.3).

3.1 CSO generation

CSO was automatically generated by Klink-2 [12], an algorithm that produces an ontology of research topics by processing scholarly metadata (titles, abstracts, keywords, authors, venues) and external sources (e.g., DBpedia, calls for papers, web

⁸ <http://technologies.kmi.open.ac.uk/rexplorer/ontologies/BiboExtension.owl>.

⁹ <http://purl.org/ontology/bibo/>.

¹⁰ <http://www.w3.org/2004/02/skos>.

¹¹ CC BY 4.0 International License <https://creativecommons.org/licenses/by/4.0>.

pages). Klink-2 can produce a full ontology including all the topics represented in the input dataset or focus on some branches under seed keywords (e.g., “Semantic Web”).

In Algorithm 1, we report the pseudocode of Klink-2. The algorithm takes as input a set of keywords and investigates their relationship with the set of their most co-occurring keywords. Klink-2 infers the semantic relationship between keyword x and y by means of three metrics: i) $H_R(x, y)$, which uses a semantic variation of the subsumption method for measuring the intensity of a hierarchical relationship; ii) $T_R(x, y)$, which uses temporal information to do the same; and iii) $S_R(x, y)$, which estimates the similarity between two topics. The first two are used to detect *skos:broaderGeneric* and *contributesTo* relationships, while the latter is used to infer *relatedEquivalent* relationships. Klink-2 then removes loops in the topic network (instruction #9). Finally, it merges keywords linked by a *relatedEquivalent* relationship and splits ambiguous keywords associated to multiple meanings (e.g., “Java”). The keywords produced in this step are added to the initial set of keywords to be further analysed in the next iteration and the while-loop is re-executed until there are no more keywords to be processed. Finally, Klink-2 filters the keywords considered “too generic” or “not academic” according to a set of heuristics (instruction #13) and generates the triples describing the ontology.

<p>Input : List of keywords <i>keywords</i>, User feedbacks <i>feedbacks</i> Output: Ontology <i>CSO</i></p> <pre> 1 relationships={}; // Initialise an empty set 2 while some keywords yet to process do 3 foreach k1 in keywords do 4 candidates = GetCandidates(k1, feedbacks); 5 foreach k2 in candidates do 6 relationship = InferRelationship(k1, k2, feedback, 7 relationships); 8 end foreach 9 end foreach 10 relationships = RemoveLoops(relationships); 11 new.keywords = MergeAndSplitKeywords(keywords, relationships); 12 keywords = AddNewKeywords(new.keywords); 13 end while 14 keywords = FilterTopics(keywords, feedbacks, relationships); 15 CSO = GenerateSemanticRelationships(relationships); 16 return(CSO); </pre>

Algorithm 1. The Klink-2 algorithm used to generate CSO.

Klink-2 was evaluated on the task of generating an ontology of Semantic Web topics using the metadata in the Rexplore dataset in a previous work [12]. For this purpose, we generated with the help of three senior researchers a gold standard ontology¹² including 88 research topics in the field of the Semantic Web. Klink-2 outperformed significantly the alternative algorithms ($p=0.0005$) yielding a precision of 86% and a recall of 85.5%. For further details about Klink-2 and its evaluation, please refer to [12].

¹² Gold Standard: <http://technologies.kmi.open.ac.uk/rexplore/data>.

3.2 Applications using CSO

The Computer Science Ontology has been used to support a variety of applications and algorithms. In this section, we discuss a selection of these systems and how they use the ontology, with the aim of showing the practical value of CSO and inspiring further applications.

Smart Topic Miner [3] (STM)¹³ is a tool developed in collaboration with Springer Nature for supporting its editorial team in classifying editorial products according to a taxonomy of research topics drawn from CSO and the Springer Nature internal taxonomy. STM halves the time needed for classifying proceedings from 20-30 to 10-15 minutes and allows this task to be performed also by assistant editors, thus distributing the load and reducing costs. It is currently used to classify about 800 proceedings books every year, including the ones published in the well-known Lecture Notes in Computer Science (LNCS) series family.

Smart Book Recommender [14] (SBR)¹⁴ is an ontology-based recommender system that takes as input the proceedings of a conference and suggests books, journals, and other conference proceedings which are likely to be relevant to the attendees of the conference in question. It builds on a dataset of 27K Springer Nature editorial products described with CSO research topics. SBR allows editors to investigate why a certain publication was suggested by means of an interactive view that compares the topics of the suggested publications and those of the input conference.

Augur [11] is an approach that aims to effectively detect the emergence of new research areas by analysing topic networks and identifying clusters associated with an overall increase of the pace of collaboration between research areas. Initially, Augur creates evolutionary networks describing the collaboration between research topics over time. Then it uses a novel clustering algorithm, the Advanced Clique Percolation Method (ACPM), to identify portions of the network that exhibit a significant increase in the pace of collaboration. Each identified clusters of topics represent an area of the network that is nurturing a new research area that should shortly emerge.

Rexplore [7] is a system that leverages novel solutions in large-scale data mining, semantic technologies and visual analytics, to provide an innovative environment for exploring and making sense of scholarly data. Rexplore uses CSO for characterising research papers, authors, and organisations according to their research topics and for producing relevant views. For instance, Rexplore is able to plot the collaboration graph of the top researchers in a field and to visualise researchers in terms of the shifting of their research interests over the years. Rexplore also describes each topic in CSO with a variety of analytics, and allows users to visualise the trends of its sub-topics.

The **Technology-Topic Framework** [24] is an approach that characterises technologies according to their propagation through research topics drawn from CSO, and uses this representation to forecast the future propagation of novel technologies across research fields. The aim is to suggest promising technologies to scholars and accelerate the flow of knowledge from one community to another and the pace of technology propagation. The system was evaluated on a set of 1,118 technologies in the Artificial Intelligence field yielding excellent results.

¹³ Demo of Smart Topic Miner: http://rexplore.kmi.open.ac.uk/STM_demo.

¹⁴ Demo of Smart Book Recommender: http://rexplore.kmi.open.ac.uk/SBR_demo.

EDAM [13] is an expert-driven automatic methodology for creating systematic reviews that limits the amount of tedious tasks that have to be performed by human experts. Typically, systematic reviews require domain experts to annotate hundreds of papers manually. EDAM is able to skip this step by i) characterising the area of interest using an ontology of topics, ii) asking domain experts to refine this ontology, and iii) exploiting this knowledge base for classifying relevant papers and producing useful analytics. The first implementation of EDAM adopted CSO for analysing the field of Software Architecture.

3.3 CSO Evaluation

Since its introduction in 2012, the Computer Science Ontology has been used in several studies and proved to effectively support a wide range of tasks such as:

- forecasting new research topics [11];
- exploration of scholarly data [7];
- automatic annotation of research papers [13];
- detection of research communities [10];
- ontology forecasting [25].

In this section, we will briefly report the results of these studies and highlight the role of CSO.

Forecasting new research topics. The evaluation of the Augur system [11] proved that semantically enriching topics networks with CSO yields a significant improvement in performance on the task of predicting the emergence of novel research areas. Table 1 shows precision and recall obtained in the period 1999-2009 by a version of Augur using CSO and by an alternative version that uses only keywords to represent research topics¹⁵.

Exploration of scholarly data. The Rexplore system was shown to be able to support users in performing specific tasks more effectively than Microsoft Academic Search (MAS), thanks to its organic representation of research topics [7]. We conducted a user study and asked 26 users to complete three tasks using one of the systems. The users adopting Rexplore completed the task more quickly and with higher success rate, as reported in Table 2.

Automatic annotation of research papers. The aforementioned Expert-Driven Automatic Methodology [13] uses CSO for automatically classifying research papers by categorising under a topic all papers that contain in the title, abstract, or keyword field the label of the topic, its *relatedEquivalent*, or its *skos:narrowerGeneric*. We applied this approach to the field of Software Architecture¹⁶ and found that its performance in classifying papers was not statistically significantly different from those of six senior researchers in the field ($p=0.77$). Table 3 shows the agreement between the annotators, computed as the ratio of papers which were tagged with the same category by both annotators. The approach adopting CSO yielded the highest average agreement and also obtained the highest agreement with three out of six domain experts.

Detection of research communities. The Temporal Semantic Topic-Based Clustering (TST) is an approach for detecting research communities by clustering researchers according to their research trajectories, defined as distributions of topics

¹⁵ The evaluation material of Augur can be found at <http://rexplore.kmi.open.ac.uk/JCDL2018>.

¹⁶ The evaluation material of EDAM can be found at <http://rexplore.kmi.open.ac.uk/data/edam>.

over time. We evaluated the full version of TST that characterises the researcher's interests according to CSO against 25 human experts in the fields of Semantic Web and Human Computer Interaction, finding no significant differences ($p > 0.14$). Conversely, an alternative version that simply uses keywords was outperformed by both TST and human experts ($p < 0.0001$).

Ontology Forecasting. The Semantic Innovation Forecast model (SIF) [25] is an approach to predict new concepts of an ontology at time $t+1$, using only data available at time t . The full version of SIF, learning from concepts in CSO, was able to significantly outperform¹⁷ several variations of LDA [26], as reported in Table 4.

Table 1. Performance of Augur [11] when characterising topics with keywords or CSO.

	Keywords		CSO	
	Precision	Recall	Precision	Recall
1999	0.68	0.49	0.86	0.76
2000	0.62	0.39	0.78	0.70
2001	0.69	0.49	0.77	0.72
2002	0.65	0.50	0.82	0.80
2003	0.72	0.54	0.83	0.79
2004	0.70	0.47	0.84	0.68
2005	0.62	0.49	0.71	0.66
2006	0.32	0.32	0.43	0.51
2007	0.06	0.21	0.28	0.44
2008	0.06	0.08	0.15	0.33
2009	0.05	0.59	0.09	0.76

Table 2. Experimental results (in min:secs) using Rexplore and MAS to perform three different tasks.

Rexplore (CSO) (17 participants)			
	Average Time	Standard Deviation	Success Rate
Task 1	03:06	00:45	100%
Task 2	08:01	02:50	94%
Task 3	07:51	02:32	100%
MAS (no CSO) (9 participants)			
	Average Time	Standard Deviation	Success Rate
Task 1	14:46	00:24	33%
Task 2	13:52	01:35	50%
Task 3	15:00	00:00	0%

Table 3. Agreement between annotators (including EDAM) and average agreement of each annotator.

	EDAM (CSO)	User1	User2	User3	User4	User5	User6
EDAM (CSO)	-	56%	68%	64%	64%	76%	64%
User1	56%	-	40%	56%	36%	48%	44%
User2	68%	40%	-	64%	52%	76%	64%
User3	64%	56%	64%	-	52%	64%	68%
User4	64%	36%	52%	52%	-	64%	52%
User5	76%	48%	76%	64%	64%	-	72%
User6	64%	44%	64%	68%	52%	72%	-
AVG	66%	45%	58%	59%	51%	63%	60%

Table 4. Mean average precision @10 for SIF [25] and other four alternative algorithms based on LDA [26].

YEAR-FORECAST	YEAR-TRAINED	YEAR-PRIOR	SIF (CSO)	LDA	LDA-A	LDA-I	LDA-IA
2000	1999	1997-1999	0.7031	0.125	0.4761	0	0.408
2002	2001	1999-2001	0.875	0	0.8227	0.6428	0.7486
2004	2003	2001-2003	0.906	0	0.5822	0.5726	0.6347
2006	2005	2003-2005	0.8755	0.3069	0.7853	0.8385	0.6893
2008	2007	2005-2006	0.988	0.398	0.681	0.5661	0.7035
AVG			0.8695	0.1659	0.6694	0.524	0.6368

¹⁷ The evaluation material of SIF can be found at <http://technologies.kmi.open.ac.uk/rexplore/ekaw2016/OF>.

4 The CSO Portal

The CSO Portal is a web application that enables users to download, explore, and provide granular feedback on CSO. It is available at <http://cso.kmi.open.ac.uk>.

Figure 1 shows an overview of the CSO Portal. We consider three kinds of users: unregistered users, registered users, and members of the editorial board. Unregistered users can download the ontology and browse it by using three alternative interfaces. Registered users are also allowed to post feedback regarding the full ontology or specific topics or relationships. The members of the editorial board have the task of reviewing the user feedback and select the changes to be incorporated in the new releases of CSO.

In the following sections, we will discuss how users can explore CSO and leave feedback at different levels of granularity.

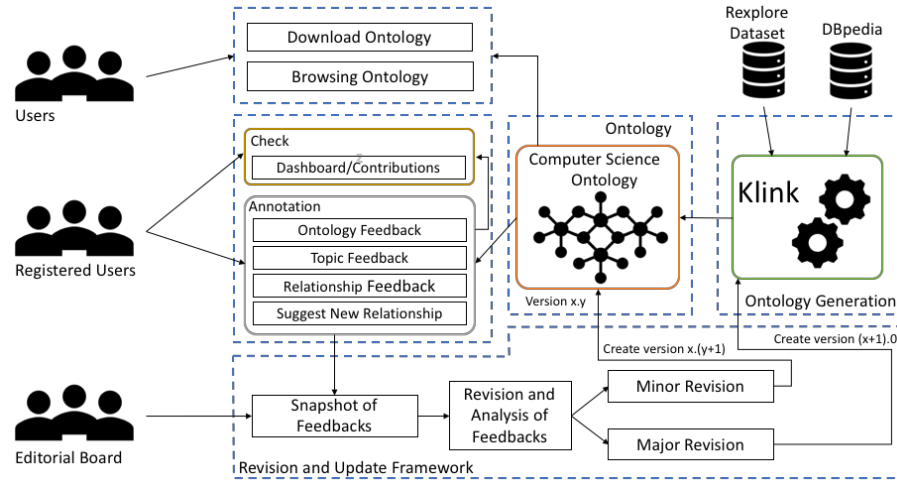


Figure 1. Overview of the Computer Science Ontology Portal.

4.1 Exploring CSO

An important functionality of the CSO Portal is the ability to search and navigate the about 26K research topics in CSO. The homepage of the portal (Figure 2) provides a simple search bar as a starting point. The user can type the label of any topic (e.g., “Semantic Web”) and submit it to be redirected to that topic page.

For a given topic, this page shows its *skos:broaderGeneric* and *relatedEquivalent* relationships with the relevant topics. For the sake of clarity, these relationships are presented to the users as **parent of/child of** and **alternative label of**. For instance, the relationships:

- *semantic web* **skos:broaderGeneric** *RDF*
- *ontology mapping* **relatedEquivalent** *ontology alignment*

are presented as:

- *semantic web* **parent of** *RDF* or *RDF* **child of** *semantic web*
- *ontology mapping* **alternative label of** *ontology alignment*

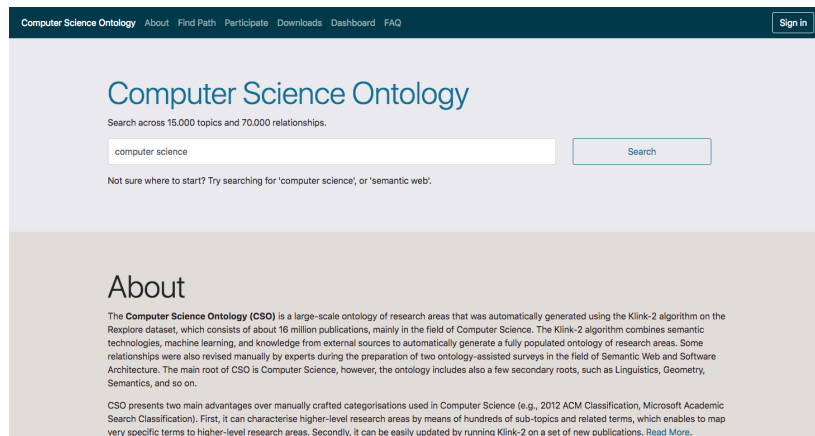


Figure 2. Homepage of the Computer Science Ontology Portal.

The CSO Portal offers three different interfaces to visualise and explore the topic relationships: the *graph view*, the *detailed view*, and the *compact view*. Figure 3, Figure 4, and Figure 5 show how these three views represent the topic “*semantic web*”¹⁸.

The **graph view** is an interactive interface that allows users to seamlessly navigate the network of topics within CSO. In this view, each topic is represented as a node and the *skos:broaderGeneric* relationships are represented as links. Initially, the view focuses on the topic searched by the user and its direct relationships. The user can also explore the ontology by expanding nodes, hiding unwanted branches, and zooming in and out. The nodes can be expanded or collapsed by left clicking on them. The user can also utilise a checkbox for highlighting the 15 key topics in the branch. This feature allows the user to quickly identify the most significant topics, making use of an approximate count of the relevant papers within the Rexplore dataset [7]. When users right-clicks on a specific node, they are prompted with a menu containing the following two options: i) *Inspect* – This opens a sidebar window, as shown in Figure 3, providing more information about the topic (description and equivalent topics), and ii) *Explore in new page* – This redirects the user to another page where the selected topic is the central node in the graph. The user can also right-click on links, which also opens a sidebar window, to find more details about that particular relationship. The graph view is generated dynamically using the D3 library¹⁹.

The **detailed view** presents each relevant triple in a separate row. The user can click on the name of a topic to jump to that topic page and navigate the ontology. Finally, the **compact view** shows the same information in a more condensed format, by grouping topics according to their relationship with the main one.

A topic page also provides a short description of the topic in question and a hyperlink to the corresponding Wikipedia article. In order to do so, we associated each topic to the relevant DBpedia entity by feeding a sentence listing the label of the topic and its direct sub and super topics to the DBpedia Spotlight API [27]. The subsequent JSON response contains a list of likely DBpedia pages for the selected topic, each with a number of probability statistics. A data analysis showed that by filtering out candidates with a similarity score less than 1 and an offset value greater than 0 it is possible to

¹⁸ <http://cso.kmi.open.ac.uk/topics/semantic%20web>

¹⁹ D3.js, <https://d3js.org>.

identify the correct DBpedia entity with nearly 100% precision. Naturally, not all CSO topics are described in DBpedia.

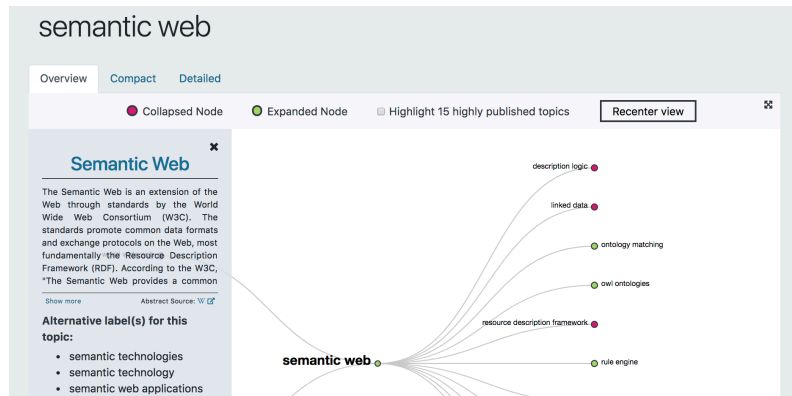


Figure 3. Screenshot of the resource page related to the topic “*semantic web*” (Overview).

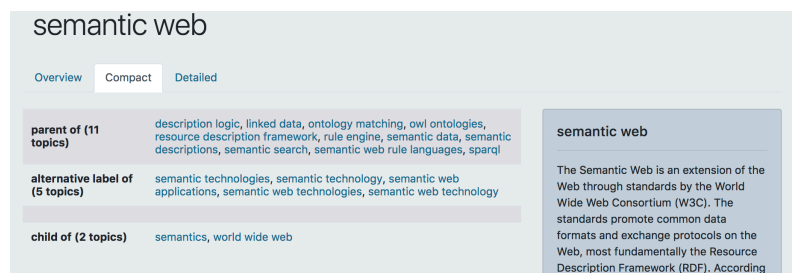


Figure 4. Screenshot of the resource page related to the topic “*semantic web*” (Compact).

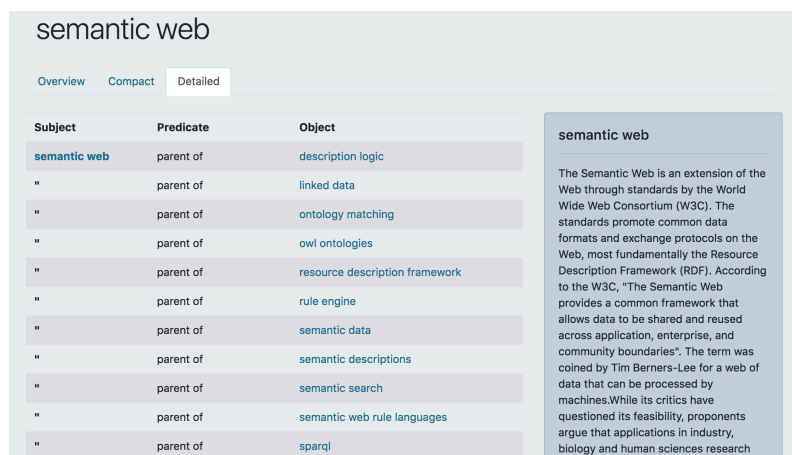


Figure 5. Screenshot of the resource page related to the topic “*semantic web*” (Detailed).

The portal supports content negotiation and yields different representations of the resources according to the content-type specified in the request. It currently supports ‘text/html’, ‘application/rdf+xml’, ‘text/turtle’, ‘application/n-triples’, and ‘application/ld+json’.

4.2 User feedback

Registered users can provide feedback about the ontology and its relationships in all the alternative views, to be considered for future releases of the Computer Science Ontology (see Section 5). In particular, users can offer feedback at i) ontology level, ii) topic level, and iii) relationship level.

The ontology level feedback is a general assessment expressing thoughts and criticisms about CSO. The user can provide it by clicking the feedback tab in the top menu and filling a text form.

Users can give feedback on specific topics by means of a form that can be triggered by clicking an icon near the topic name. Figure 6 shows as example the feedback form for the topic “ontology mapping”. Users can rate the topic as “correct”, “incorrect” or “is complicated” and comment their rating in a text field. In the same form, users can also suggest one or more relationships that are currently missing from the ontology or a new topic that should be linked by this relationship. Figure 7 shows the form for suggesting new relationships for the topic “ontology mapping”. The users can choose the predicate from “parent of”, “alternative label of”, “and child of”. The object could be either a topic that already exists in CSO or a new one.

Finally, users can offer feedback on specific relationships by means of an alternative form. As in the previous case, they can rate the relationship and add a short comment.

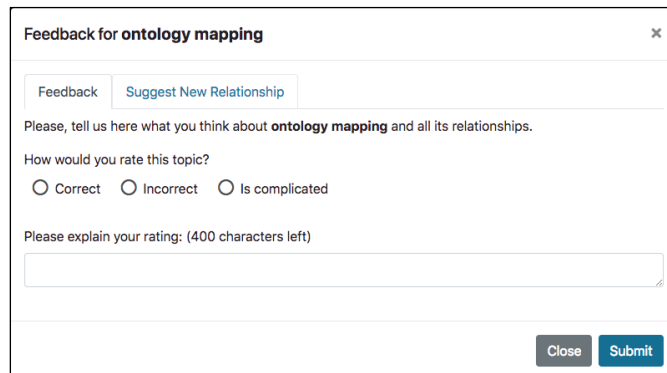


Figure 6. Form for providing feedback about the topic “ontology mapping”.

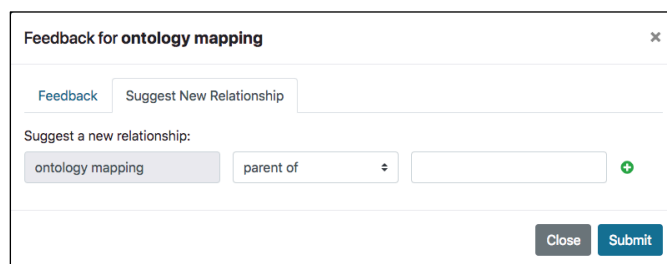
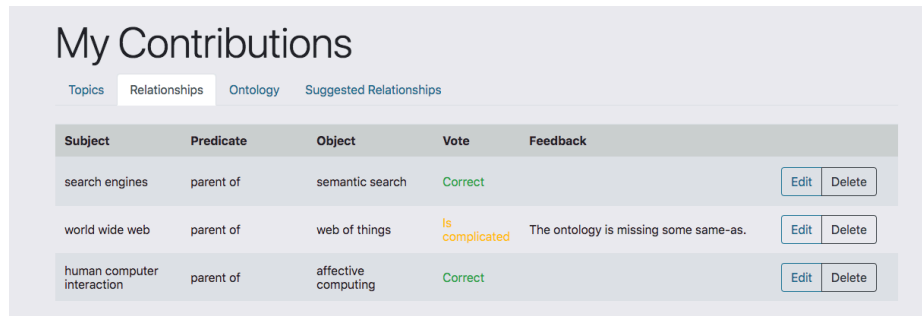


Figure 7. Form for suggesting new relationships about the topic “ontology mapping”.

The CSO Portal allows users to review their own feedback entries. In the “My Contributions” page (Figure 8) users can inspect, edit, and delete any previously given suggestion. The feedback entries are organised by typology (ontology level, topic level,

relationship level, and recommendation of new relationships), and they can be either retracted or modified.



The screenshot shows a web interface titled "My Contributions". Below the title are four tabs: "Topics", "Relationships" (which is active), "Ontology", and "Suggested Relationships". The main content is a table with five columns: "Subject", "Predicate", "Object", "Vote", and "Feedback". There are three rows of data, each with "Edit" and "Delete" buttons at the end.

Subject	Predicate	Object	Vote	Feedback
search engines	parent of	semantic search	Correct	
world wide web	parent of	web of things	Is complicated	The ontology is missing some same-as.
human computer interaction	parent of	affective computing	Correct	

Figure 8. *My Contribution* page where users can review their own feedback.

5 Future updates

We plan to periodically release new versions of the CSO ontology. The editorial board will supervise this process and review user feedback, distilling a list of recommendations to be implemented in future versions. The composition of the editorial board is currently being finalised. Initially it will comprise a small number of individuals drawn from the Open University and our industrial collaborators (between 4 and 6 people in total). Depending on the success and impact of the initiative we expect that the board will grow significantly in the future and will expand to include representatives of a variety of organizations. Both minor and major revisions will be released on a regular basis.

Minor revisions will be produced by directly implementing in the ontology the changes suggested by users and confirmed by the editorial board. The changes may include: i) removal of a topic, ii) removal of a relationship, iii) inclusion of a relationship, and iv) inclusion of a topic. In this phase, we will focus on correcting specific errors rather than expanding the ontology.

Major revisions will be produced by generating a new full ontology by feeding the Klink algorithm an up-to-date corpus of publications and the set of “correct relationships” suggested by users and confirmed by the editorial board. Indeed, the current version of Klink-2 is already able to take as input user defined relationships and incorporate them in the automatically generated ontology. The goal is to make sure that major revisions of CSO include all significant research areas that have emerged in the interval since the previous major release.

We aim to produce at least one major revision every year. The timing on the other revisions will depend on the number and quality of feedback entries. For instance, a significant number of negative feedback entries on a certain branch would trigger a comprehensive revision of it. In such a case, we will contact domain experts and invite them to review the associated branch on the CSO Portal. For instance, in a recent study [13], we assessed the CSO branch regarding Software Architecture by generating a spreadsheet representation of it and having it reviewed by three senior researchers. The CSO Portal should make this process simpler and easier to track.

6 Conclusions

In this paper, we presented the Computer Science Ontology (CSO), a large-scale, automatically generated ontology of research areas, which provides a much more comprehensive and granular characterisation of research topics in Computer Science than what is currently available in other state-of-the-art taxonomies of research areas. We discussed its characteristics, briefly introduced several applications which use it, and showed that it successfully supports several useful tasks, such as classifying research papers, exploring scholarly data, forecasting new research topics, detecting research communities, and so on. We also introduced the CSO Portal, a web application that enables users to download, explore, and provide feedback on CSO. We intend to take advantage of the CSO Portal to involve the wider research community in the ontology evolution process, with the aim of periodically releasing up-to-date revisions of CSO and allow members of the community to provide feedback. In this sense, the version of CSO presented in this paper can be considered simply as a starting point.

As future work, we are currently developing a new version of Klink-2 that will consider the quantity and the sentiment of the user feedback on previous versions of the ontology. We also intend to apply our ontology learning techniques to other research fields, such as Biology and Engineering. The ultimate goal is to create a comprehensive set of large-scale and data driven ontologies describing most branches of science.

References

1. Saif, H., He, Y., Alani, H.: Semantic Sentiment Analysis of Twitter. In: The Semantic Web -- ISWC 2012. pp. 508–524. Springer, Berlin, Heidelberg (2012).
2. Ding, L., Kolari, P., Ding, Z., Avancha, S.: Using Ontologies in the Semantic Web: A Survey. In: Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems. pp. 79–113. Springer US, Boston, MA (2007).
3. Osborne, F., Salatino, A., Birukou, A., Motta, E.: Automatic Classification of Springer Nature Proceedings with Smart Topic Miner. *Semant. Web -- ISWC 2016*. 9982 LNCS, 383–399 (2016).
4. Middleton, S.E., Roure, D. De, Shadbolt, N.R.: Ontology-Based Recommender Systems. In: Handbook on Ontologies. pp. 779–796. Springer Berlin Heidelberg, Berlin, Heidelberg (2009).
5. Hotho, A., Staab, S., Stumme, G.: Ontologies improve text document clustering. In: Third IEEE International Conference on Data Mining. pp. 541–544. IEEE Comput. Soc.
6. Livingston, K.M., Bada, M., Baumgartner, W.A., Hunter, L.E.: KaBOB: ontology-based semantic integration of biomedical databases. *BMC Bioinformatics*. 16, 126 (2015).
7. Osborne, F., Motta, E., Mulholland, P.: Exploring scholarly data with rexplore. *Semant. Web -- ISWC 2013*. 8218 LNCS, 460–477 (2013).
8. Fathalla, S., Vahdati, S., Auer, S., Lange, C.: Towards a Knowledge Graph Representing Research Findings by Semantifying Survey Articles. In: Research and Advanced Technology for Digital Libraries. pp. 315–327. Springer, Cham (2017).
9. Bettencourt, L.M.A., Kaiser, D.I., Kaur, J.: Scientific discovery and topological transitions in collaboration networks. *J. Informetr.* 3, 210–221 (2009).
10. Osborne, F., Scavo, G., Motta, E.: Identifying diachronic topic-based research communities by clustering shared research trajectories. In: The Semantic Web: Trends and Challenges. pp. 114–129. Springer International Publishing (2014).
11. Salatino, A.A., Osborne, F., Motta, E.: AUGUR: Forecasting the Emergence of New

- Research Topics. In: Joint Conference on Digital Libraries 2018, Fort Worth, Texas. pp. 1–10 (2018).
12. Osborne, F., Motta, E.: Klink-2: Integrating Multiple Web Sources to Generate Semantic Topic Networks. In: The Semantic Web - ISWC 2015. pp. 408–424 (2015).
13. Osborne, F., Muccini, H., Lago, P., Motta, E.: Reducing the Effort for Systematic Reviews in Software Engineering. Pre-Print: <https://bit.ly/2sobCkI>.
14. Thanapalasingam, T., Osborne, F., Birukou, A., Motta, E.: Ontology-Based Recommendation of Editorial Products. In: International Semantic Web Conference 2018. , Monterey, CA (USA) (2018).
15. Lipscomb, C.E.: Medical Subject Headings (MeSH). Bull. Med. Libr. Assoc. 88, 265–6 (2000).
16. Cherrier, B.: Classifying Economics: A History of the JEL Codes. J. Econ. Lit. 55, 545–579 (2017).
17. Clough, P., Sanderson, M., Gollins, T.: Examining the Limits of Crowdsourcing for Relevance Assessment. IEEE Internet Comput. 17, 32–38 (2013).
18. Cimiano, P., Völker, J.: Text2Onto. In: Natural Language Processing and Information Systems. pp. 227–238. Springer, Berlin, Heidelberg (2005).
19. Muller, A., Dorre, J., Gerstl, P., Seiffert, R.: The TaxGen framework: automating the generation of a taxonomy for a large document collection. In: Proceedings of the 32nd Annual Hawaii International Conference on Systems Sciences. 1999. HICSS-32. Abstracts and CD-ROM of Full Papers. p. 9. IEEE Comput. Soc (1999).
20. Sanderson, M., Croft, B.: Deriving concept hierarchies from text. In: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR '99. pp. 206–213. ACM Press, New York, New York, USA (1999).
21. Wohlgenannt, G., Weichselbraun, A., Scharl, A., Sabou, M.: Dynamic Integration of Multiple Evidence Sources for Ontology Learning. J. Inf. Data Manag. 3, 243–254 (2012).
22. Mortensen, J.M., Musen, M.A., Noy, N.F.: Crowdsourcing the verification of relationships in biomedical ontologies. AMIA Annu. Symp. Proc. 2013, 1020–9 (2013).
23. Kirrane, S., Sabou, M., Fernández, J.D., Osborne, F., Robin, C., Buitelaar, P., Motta, E., Polleres, A.: A decade of Semantic Web research through the lenses of a mixed methods approach. Semant. Web J. - Prepr. (2018).
24. Osborne, F., Mannocci, A., Motta, E.: Forecasting the Spreading of Technologies in Research Communities. In: Proceedings of the Knowledge Capture Conference (2017).
25. Cano-Basave, A.E., Osborne, F., Salatino, A.A.: Ontology forecasting in scientific literature: Semantic concepts prediction based on innovation-adoption priors. In: Knowledge Engineering and Knowledge Management. pp. 51–67 (2016).
26. Blei, D.M., Edu, B.B., Ng, A.Y., Edu, A.S., Jordan, M.I., Edu, J.B.: Latent Dirichlet Allocation. J. Mach. Learn. Res. 3, 993–1022 (2003).
27. Daiber, J., Jakob, M., Hokamp, C., Mendes, P.N.: Improving efficiency and accuracy in multilingual entity extraction. In: Proceedings of the 9th International Conference on Semantic Systems - I-SEMANTICS '13. p. 121. ACM Press, New York, New York, USA (2013).