Adding Semantic Scholar data to KTH’s ÅBU

Gerald Q. Maguire Jr.

Seif Haridi <[haridi@kth.se](mailto:haridi@kth.se)> raised the issue of Semantic Scholar as a source of information regarding authors and the importance of their publications. So I suggested that he look at making this data available for the KTH Annual Bibliometric Evaluation (ÅBU). This document presents some of the things that I learned when looking at Semantic Scholar.

Section 1shows the information that is available for an author and a paper. Section 2 shows the information that is available via the API for an author while Section 3 shows the information that is available for a publication via the API.

Section 4 describes how one can get the entire corpus (i.e., the underlying data that Semantic Scholar – S2) uses. It is important to note that this is the raw data and not the information computed by Semantic Scholar. To get the computed information one needs to use the API as described at <http://s2-public-api-prod.us-west-2.elasticbeanstalk.com>. So why is the corpus interesting? This information is useful to try to match up the information from DiVA about a publication – when the publication is missing one of the identifiers that Semantic Scholar uses: S2 Paper ID, DOI, ArXiv ID, MAG[[1]](#footnote-1) ID, ACL ID, PubMed ID (PMID), Corpus ID, Full details of the attributes for a publication are given in <http://s2-public-api-prod.us-west-2.elasticbeanstalk.com/corpus/>.

# What does Semantic Scholar show for an author?

Figure 1 shows an author’s page (in this case my own page: <https://www.semanticscholar.org/author/Gerald-Q.-Maguire/1681232> . Figure 2 shows the Author detail card, while Figure 3 shows the HTML for this Author detail card. Figure 4 shows an example of the page shown for a specific publication.

|  |
| --- |
|  |

Figure : View of an author page with Chrome developer

|  |
| --- |
|  |

Figure : Author detail card

|  |
| --- |
| <div class="author-detail-card\_\_main-content"><div class="author-detail-card\_\_meta-section\_\_meta-details"><div class="author-detail-card\_\_verified-check"><span class="verified-checkmark" style="width: 100%;"><div class="verified-checkmark\_\_circle"><svg class="icon-svg icon-checkmark" data-selenium-selector="icon-checkmark"><use xlink:href="#checkmark"></use></svg></div></span></div><div class="flex-row"><h1 class="author-detail-card\_\_author-name" data-selenium-selector="author-name">Gerald Q. Maguire</h1></div><div class="author-detail-card\_\_affiliation">KTH Royal Institute of Technology</div></div><div class="author-detail-card\_\_meta-section\_\_profile-details"><div class="author-detail-card\_\_meta-row author-detail-card\_\_orcid-id"><a href="https://orcid.org/0000-0002-6066-746X" target="\_blank" rel="noopener noreferrer" class="orcid-link"><svg height="14" width="14" class="icon-svg icon-orcid-id" data-selenium-selector="icon-orcid-id"><use xlink:href="#orcid-id"></use></svg><span class="orcid-link\_\_id-label">0000-0002-6066-746X</span></a></div><div class="author-detail-card\_\_meta-row author-detail-card\_\_homepage"><a href="https://people.kth.se/~maguire/" target="\_blank" rel="noopener noreferrer" class="author-detail-card\_\_homepage\_\_link"><span class="author-detail-card\_\_homepage\_\_link\_\_icon"><svg class="icon-svg icon-fa-link" data-selenium-selector="icon-fa-link"><use xlink:href="#fa-link"></use></svg></span><span class="author-detail-card\_\_homepage\_\_label">https://people.kth.se/~maguire/</span></a></div></div><div class="author-detail-card\_\_meta-section\_\_author-stats"><div class="author-detail-card\_\_stats-row"><span class="author-detail-card\_\_stats-row\_\_label">Publications</span><span class="author-detail-card\_\_stats-row\_\_value">238</span></div><div class="author-detail-card\_\_stats-row"><span class="author-detail-card\_\_stats-row\_\_label">h-index<a class="" href="/faq#h-index"><svg class="author-detail-card\_\_h-index-info icon-svg icon-information" data-selenium-selector="icon-information"><use xlink:href="#information"></use></svg></a></span><span class="author-detail-card\_\_stats-row\_\_value">30</span></div><div class="author-detail-card\_\_stats-row"><span class="author-detail-card\_\_stats-row\_\_label">Citations</span><span class="author-detail-card\_\_stats-row\_\_value">12,273</span></div><div class="author-detail-card\_\_stats-row"><span class="author-detail-card\_\_stats-row\_\_label">Highly Influential Citations</span><span class="author-detail-card\_\_stats-row\_\_value">460</span></div></div><div class="author-detail-card\_\_meta-section\_\_meta-buttons"><div class="dropdown-menu author-detail-card\_\_alert-dropdown" tabindex="0" role="button" aria-label="Dropdown menu" outline="none"><button class="icon-button button--primary dropdown-menu-trigger" data-selenium-selector="manage-author-alerts-button"><span class="flex-row-centered"><svg width="12" height="12" class="icon-svg icon-fa-bell" data-selenium-selector="icon-fa-bell"><use xlink:href="#fa-bell"></use></svg><span class="icon-button-text">Follow Author...</span></span></button></div><a class="icon-button button--secondary flex-row-centered author-detail-card\_\_claim-button-container" data-heap-id="author\_edit\_button" data-heap-author-id="1681232" data-selenium-selector="author-claim-button" href="/me/account/author-contact"><div class="flex-row-centered"><svg width="15" height="15" class="icon-svg icon-user-with-pencil" data-selenium-selector="icon-user-with-pencil"><use xlink:href="#user-with-pencil"></use></svg><span class="icon-button-text">Edit Author Page</span></div></a></div></div> |

Figure : Author detail card HTML

|  |
| --- |
|  |

Figure : Example of a page about a publication

# Author information

There is an API to get author information (called author lookup): [https://api.semanticscholar.org/v1/author/[S2AuthorId](https://api.semanticscholar.org/v1/author/%5bS2AuthorId)]

Figure 6 shows the results of this for <https://api.semanticscholar.org/v1/author/1681232> . While it include "influentialCitationCount":460 and computing the length of the 'papers' component it gives 238. However, it does not include other information (such as h-index and Citations) that is in the author detail card (see Figure 5). Citations (really total citations) can be computed by summing the citation information for each of the 238 publications. Of course, given the number of citations to each publication one can compute the h-index.

|  |
| --- |
|  |

Figure : Closeup of an author detail card

|  |
| --- |
| { 'aliases': [ 'Gerald Q Maguire',  'G. Q. Maguire',  'G Q Maguire',  'Gerald Maguire',  'Gerald Quentin Maguire',  'Jr. G Q Maguire'],  'authorId': '1681232',  'influentialCitationCount': 460,  'name': 'Gerald Q. Maguire',  'papers': [ { 'paperId': 'fb5d1bb23724d9a5a5eae036a2e3cf291cac2c1b',  'title': 'Cognitive radio: making software radios more personal',  'url': 'https://www.semanticscholar.org/paper/fb5d1bb23724d9a5a5eae036a2e3cf291cac2c1b',  'year': 1999},  { 'paperId': '53a092e0536134633110d7781378971556ea08cd',  'title': 'IP-based protocols for mobile internetworking',  'url': 'https://www.semanticscholar.org/paper/53a092e0536134633110d7781378971556ea08cd',  'year': 1991},  { 'paperId': 'e35fd2e65d915a502e40cf7ab1977ebd4acac0be',  'title': 'A class of mobile motion prediction algorithms '  'for wireless mobile computing and '  'communications',  'url': 'https://www.semanticscholar.org/paper/e35fd2e65d915a502e40cf7ab1977ebd4acac0be',  'year': 1996},  …  { 'paperId': '63b411a5415c79679526ba2994b006ce015bbc9e',  'title': 'Image Processing Requirements and Distributed '  'Networks in a Digital Imaging Environment',  'url': 'https://www.semanticscholar.org/paper/63b411a5415c79679526ba2994b006ce015bbc9e',  'year': 1982}],  'url': 'https://www.semanticscholar.org/author/1681232'} |

Figure : Example of author information (with some of the papers removed)

# Paper information

One can get information about a given paper via the paper lookup API:

https://api.semanticscholar.org/v1/paper/[Paper Identifier or URL]

Consider the paper shown in Figure 7. Table 1 shows the top level of the JSON for <https://api.semanticscholar.org/v1/paper/fb5d1bb23724d9a5a5eae036a2e3cf291cac2c1b>

We can computer the total number of citations from the length of xxxx[‘citations’ ], while xxxx[influentialCitationCount’] gives use the value 312 (by the lightbulb). The other fields give use the information shown in the figure.

|  |
| --- |
|  |

Figure : Information about one paper- as shown via the web interface

Table : Top-level information about one paper

|  |  |  |
| --- | --- | --- |
| Field | Type | Value |
| abstract | <class 'str'> | "Software radios are … of cognitive radio." |
| arxivId | <class 'NoneType'> | null |
| authors | <class 'list'> | [{"authorId":"48168298","name":"J. Mitola","url":"https://www.semanticscholar.org/author/48168298"}, {"authorId":"1681232","name":"Gerald Q. Maguire","url":"https://www.semanticscholar.org/author/1681232"}] |
| citationVelocity | <class 'int'> | 360 |
| citations | <class 'list'> | A list of the 8773 citations each of the form:  { 'arxivId': None,  'authors': [ {'authorId': '2157293', 'name': 'Narasimha Rao Banavathu'},  {'authorId': '1694137', 'name': 'Mohammed Zafar Ali Khan'}],  'doi': '10.1109/LSP.2019.2893999',  'intent': ['background'],  'isInfluential': False,  'paperId': 'd6821b5e05d75ebac6dde7dd14a39c84e58397e3',  'title': 'Optimization of $N$-out-of-$K$ Rule for Heterogeneous Cognitive '  'Radio Networks',  'url': 'https://www.semanticscholar.org/paper/d6821b5e05d75ebac6dde7dd14a39c84e58397e3',  'venue': 'IEEE Signal Process. Lett.',  'year': 2019} |
| corpusId | <class 'int'> | 28753059 |
| doi | <class 'str'> | '10.1109/98.788210' |
| fieldsOfStudy | <class 'list'> | ['Computer Science'] |
| influentialCitationCount | <class 'int'> | 312 |
| is\_open\_access | <class 'bool'> | True |
| is\_publisher\_licensed | <class 'bool'> | False |
| paperId | <class 'str'> | 'fb5d1bb23724d9a5a5eae036a2e3cf291cac2c1b' |
| references | <class 'list'> | [ { 'arxivId': None,  'authors': [{'authorId': '153325625', 'name': 'H. Maaß'}],  'doi': '10.1023/A:1019168514184',  'intent': [],  'isInfluential': False,  'paperId': 'ef17f964c037293a3c112751dd0597eb2675e3a0',  'title': 'Location-aware mobile applications based on directory '  'services',  'url': 'https://www.semanticscholar.org/paper/ef17f964c037293a3c112751dd0597eb2675e3a0',  'venue': 'Mob. Networks Appl.',  'year': 1998},  { 'arxivId': None,  'authors': [{'authorId': '144883814', 'name': 'E. Davis'}],  'doi': '10.1609/aimag.v19i4.1424',  'intent': [],  'isInfluential': False,  'paperId': '1615f35d88d2099e3ccb9bb9a3d41f6d385c0082',  'title': 'Naive Physics Perplex',  'url': 'https://www.semanticscholar.org/paper/1615f35d88d2099e3ccb9bb9a3d41f6d385c0082',  'venue': 'AI Mag.',  'year': 1998},  …  { 'arxivId': None,  'authors': [{'authorId': '1402971527', 'name': 'Itu-T'}],  'doi': None,  'intent': [],  'isInfluential': False,  'paperId': 'ee97b1813af01c730a86a7731f946ce6860de346',  'title': 'Specification and Description Language (SDL)',  'url': 'https://www.semanticscholar.org/paper/ee97b1813af01c730a86a7731f946ce6860de346',  'venue': '',  'year': 1999}] |
| title | <class 'str'> | 'Cognitive radio: making software radios more personal' |
| topics | <class 'list'> | [ { 'topic': 'Cognitive radio',  'topicId': '17597',  'url': 'https://www.semanticscholar.org/topic/17597'},  { 'topic': 'Equalization (communications)',  'topicId': '11827',  'url': 'https://www.semanticscholar.org/topic/11827'},  { 'topic': 'Field-programmable gate array',  'topicId': '1368',  'url': 'https://www.semanticscholar.org/topic/1368'},  { 'topic': 'Cognition',  'topicId': '3248',  'url': 'https://www.semanticscholar.org/topic/3248'},  …  { 'topic': 'Regular expression',  'topicId': '48721',  'url': 'https://www.semanticscholar.org/topic/48721'}] |
| url | <class 'str'> | 'https://www.semanticscholar.org/paper/fb5d1bb23724d9a5a5eae036a2e3cf291cac2c1b' |
| venue | <class 'str'> | 'IEEE Wirel. Commun.' |
| year | <class 'int'> | 1999 |

# Getting the Corpus

For information about how to download the latest Sematic Scholar Open Research Corpus (S2ORC) [1] see <http://s2-public-api-prod.us-west-2.elasticbeanstalk.com/corpus/download/>

The CORPUS is 120 GB (as of 2020-05-27) and consists of files containing lines of JSON records with the information about each publication. An example of an entry from the corpus is shown in Figure 8.

The data dictionary is at <http://s2-public-api-prod.us-west-2.elasticbeanstalk.com/corpus/>. It is important to note that the corpus does not have any of the information that Semantic Scholar computers over the whole corpus.

See also their github at <https://github.com/allenai/s2orc>

Table : DiVA number of publications with the different types of identifiers for all of KTH from 2012-2019 – total of 41 578 publications (excluding theses at all levels)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | ISBN | DOI | ISI | PMID | ScopusId |
| Number | 8705 | 31412 | 29393 | 6061 | 32807 |
| Percentage | 20.94% | 75.54% | 70.69% | 14.58% | 78.90% |

|  |
| --- |
| { 'authors': [ {'ids': ['1776025'], 'name': 'Rhaban Hark'},  {'ids': ['2836018'], 'name': 'Nieke Aerts'},  {'ids': ['47422037'], 'name': 'David Hock'},  {'ids': ['1679387'], 'name': 'Nils Richerzhagen'},  {'ids': ['145799243'], 'name': 'Amr Rizk'},  {'ids': ['145754722'], 'name': 'Ralf Steinmetz'}],  'doi': '10.1109/TNSM.2018.2876654',  'doiUrl': 'https://doi.org/10.1109/TNSM.2018.2876654',  'entities': [],  'fieldsOfStudy': ['Computer Science'],  'id': 'd5baf37d9a6039168d85c80aa37ed8222768c170',  'inCitations': ['8bc41ecb5263f7a1fae6087e6929005325361486'],  'journalName': 'IEEE Transactions on Network and Service Management',  'journalPages': '1264-1276',  'journalVolume': '15',  'magId': '2897713897',  'outCitations': [ '8ebff39102ceae8602bf52f45b59a2e329a7514f',  'c260f075acf6550fcdfd1aaed204bc8f475befe7',  '6403a9dc96d1c7bec68325aab697e891793568de',  '39d4ad26ff4359e2f8221bba036951d16168dbdf',  'c98017ae87cba1c6bef12c0b8f5768cb4d0529f2',  '5f28bf666498d5800e015f12318930ce03cd5587',  '1e8f95d143d0dcbdec04fdb6ec91e43e38a9b4c2',  '3520b8c41833d64d86cba8c429a057aabe7f08fe',  'cc71016673d4e30dbe41a8edefdcf25f04db93c4',  '394fe080a0e93c1843bb847e5d3ea81476a2cd25',  'addb787928d37893cddde17f5ac0ee946e15d8bb',  '5cc0be99c2f1270d8c91b32b8134a2553545dbe2',  '8afd17be15ec8bd31ce25076bd3a99faba1ef62c',  '80aea60eed56067392cfafecd50dae7efc55939e',  '64f3a81fff495ac336dccdd63136d451852eb1c9',  'd47bca3f9f8c86506ba842533990c2d6f2b91658',  '6adc921e32dcb8d6b01f02f47e2b5e8f243c2ebb',  'a79e6cbcabcf37e35e3f5f1638322f692973eda0',  'c589d0853a29665e22679907a77143371bc9f7c5',  '4068bfb3d65fe4d4eae6bf217d66bf1e77793e75',  'b5d404ad21473319196847a7817d33183c4d3746',  '1bf2e0f409ed0ae1342d219ffcc756128878d298',  '3ef946edf2c40ba07f6847f96e99d7ada9ed9594',  '02b5adf77d811f12ad17cc8f1a7efed7cb7b49c0',  '1cbbdf58133f763813b3a61b8faf2f5ab74464b7',  'ff0a6bf43a29c40edec3014595df013ec10839a5',  'ab50df3eb05f0033acec9cb9789135a635ef0aa1',  '0742fa40bf9be455fc6338e3a40ed6f0113d4a61',  'f7c3b5a070e3cb73708964d4d2199c1c146a5527',  '004a046ad0db05084f103e8b1e3e9fb43b2f9386'],  'paperAbstract': 'A decisive advantage of software-defined networking '  '(SDN) is its support for flexible network '  'reconfigurations. Considering that, software-defined '  'networks require accurate and timely data-plane state '  'information. Network monitoring mechanisms usually '  'require considerable resources on SDN controllers as '  'well as on the data-plane elements. In this paper, we '  'propose an optimization of the statistic transmission to '  'reduce costs on both control- and data-plane regardless '  'of the used monitoring application and statistic '  'provisioning tool. To this end, we intercept the '  'statistic message exchange and 1) aggregate multiple '  'requests coming from different monitoring '  'applications/controllers, 2) filter irrelevant statistic '  'messages with respect to their information gain before '  'delivering them to the control applications, and 3) '  'deploy statistic caching. The proposed system, denoted '  'STATISTIC REQUEST RELAY, forms a logically centralized '  'statistic relay between controllers and the managed '  'data-plane network. Our evaluation shows that the number '  'of statistics processed on controllers as well as '  'statistic requests on switches is reduced significantly '  'while the performance penalty is negligible when using '  'statistic aggregation and filtering as proposed here.',  'pdfUrls': ['https://doi.org/10.1109/TNSM.2018.2876654'],  'pmid': '',  's2PdfUrl': '',  's2Url': 'https://semanticscholar.org/paper/d5baf37d9a6039168d85c80aa37ed8222768c170',  'sources': ['DBLP'],  'title': 'Reducing the Monitoring Footprint on Controllers in '  'Software-Defined Networks',  'venue': 'IEEE Transactions on Network and Service Management',  'year': 2018} |

Figure : Example of an entry in the corpus

# Processing data from the corpus to get S2 author IDs

First one needs to get the DiVA data and then use it extract the relevant information from the corpus. In the case of all of the KTH publications (excluding theses at all levels), the script show in Figure 9 gets the data as a CSV file. The file is 92 612 826 bytes in size. It can be read into Excel and out put as an XLSX formatted file of 37 327 903 bytes. The spreadsheet is ready in and preprocessed (as shown in Figure 10).and this produces a file (targets.json) with information about the publications that have DOIs and pubmed IDs (PMID) in the spreadsheet. An example of an entry in this file is shown in Figure 11. The size of the targets.json file is 26 823 386 bytes and contains 41 577 entries.

|  |
| --- |
| wget -O kth-exluding-theses-all-level2-2012-2019.csv 'https://kth.diva-portal.org/smash/export.jsf?format=csvall2&addFilename=true&aq=[[]]&aqe=[]&aq2=[[{"dateIssued":{"from":"2012","to":"2019"}},{"organisationId":"177","organisationId-Xtra":true},{"publicationTypeCode":["bookReview","review","article","artisticOutput","book","chapter","manuscript","collection","other","conferencePaper","patent","conferenceProceedings","report","dataset"]}]]&onlyFullText=false&noOfRows=5000000&sortOrder=title\_sort\_asc&sortOrder2=title\_sort\_asc' |

Figure : Get the DiVA data for the years 2012 to 2019

|  |
| --- |
| time ./preprocess\_for\_corpus.py /z3/maguire/SemanticScholar/KTH\_DiVA/kth-exluding-theses-all-level2-2012-2019.xlsx  file\_name='/z3/maguire/SemanticScholar/KTH\_DiVA/kth-exluding-theses-all-level2-2012-2019.xlsx'  Finished reading spreadsheet  real 0m14.013s  user 0m14.012s  sys 0m0.718s |

Figure : Preprocess the spreadsheet

|  |
| --- |
| {"PID": 913155, "Name": "Olivecrona, Henrik (Karolinska Institutet, Sweden) (Department of Molecular Medicine and Surgery, Section of Orthopaedics and Sport Medicine);Maguire, Gerald Q., Jr. [u1d13i2c] [0000-0002-6066-746X] (KTH [177], Skolan f\u00f6r informations- och kommunikationsteknik (ICT) [5994], Kommunikationssystem, CoS [5998], Radio Systems Laboratory (RS Lab) [13053]);Noz, Marilyn E. [0000-0002-2442-1622] (New York University, Department of Radiology) (Nuclear Medicine);Zeleznik, Michael P. [0000-0002-0706-1805] (University of Utah) (School of Computing, College of Engineering);Kesteris, Uldis (Department of Orthopedics, Sk\u00e5ne University Hospital);Weidenhielm, Lars [0000-0002-4280-1178] (Karolinska Institutet at Karolinska University Hospital Solna) (Department of Molecular Medicine and Surgery)", "Title": "A CT method for following patients with both prosthetic replacement and implanted tantalum beads : preliminary analysis with a pelvic model and in seven patients", "DOI": "10.1186/s13018-016-0360-7"} |

Figure Example of an entry in the targets.json file

The next step is to extract the relevant entries from a shard of the corpus. Shard 185 is 652M bytes in size and contains 9 98 539 entries and is bytes in size. This step is show in Figure 12. This produces two files matches\_corpus\_185.json (the relevant entries from shard 185) and a reduced targets file called remaining\_targets.json. Since some entries have been found the size of the remaining\_targets file has decrease to 26 710 288 bytes (and contains 41 416 entries). The matches\_corpus\_185 file is 650 159 bytes in size and has only 161 entries.

|  |
| --- |
| time ./find\_in\_corpus.py targets.json 185 /z3/maguire/SemanticScholar/SS\_corpus\_2020-05-27  length of targets=41577  doi\_set length=31349, pmid\_set length=0  shard\_filename=/z3/maguire/SemanticScholar/SS\_corpus\_2020-05-27/s2-corpus-185  corpus\_shard length=998539  length of remaining targets=41416  real 0m39.365s  user 0m37.004s  sys 0m2.970s |

Figure Finding relevant entries in the corpus (in this case for shard 185)

Now that we have the relevant entries from the corpus, the next ste is to process them looking for names of users at KTH and their kthid and corresponding S2 author ID(s) and Se2name.. This is done as shown in Figure 13. This produces a file names-185.json that contains 47 214 bytes and 335 entries. The entries are of the form shown in Figure 14. It is also possible to have partial match with more changes such as that shown in Figure 15. Of course one can have extreme mismatches such as shown in Figure 16 (while the author might be a member of the Atlas Collaboration, this is beyond the ability of the program to handle.

|  |
| --- |
| time ./match\_names.py matches\_corpus\_185.json  length of matches=161  process matches  … <<lots of output>>  length of name\_info=335  real 0m6.380s  user 0m1.143s  sys 0m0.513s |

Figure : Match DiVA names with S2 names

|  |
| --- |
| {"diva\_name": "Gerald Q. Maguire Jr.", "S2\_author\_name": "Gerald Q. Maguire", "kthid": "u1d13i2c", "S2\_author\_ID": ["1681232"], "PID": 733621} |

Figure : Example of a matching name entry

|  |
| --- |
| {"diva\_name": "Karl H. Johansson", "S2\_author\_name": "Karl Henrik Johansson", "kthid": "u1h0d9k2", "partial match": 5, "S2\_author\_ID": ["1679177"], "PID": 1082962} |

Figure : A partial match

|  |
| --- |
| {"diva\_name": "G.;Morley Aad", "S2\_author\_name": "Atlas Collaboration", "kthid": "u1bjxgiu", "partial match": 16, "unlikey match": "123.07692307692308%", "S2\_author\_ID": ["40952709"], "PID": 874480} |

Figure : An "unlikely" match

Currently match\_names.py seems to handle the cases where the number of DiVA authors and the number of S2 authors for a paper are exactly equal. It will complain with an error output when it finds the lengths of these list fo not match. An example, is shown in Figure 17. If one looks for this entry in DiVA you will find urn:nbn:se:kth:diva-149304 as shown in Figure 17. Only 6 of the authors are shown in DiVA (see the screen dump in Figure 18).

|  |
| --- |
| \*\*\*\*\* len(s2\_authors=12, length of split names=6, diva\_name=Aad, G.;Kuwertz, Emma [u15kq18r] (KTH [177], Skolan för teknikvetenskap (SCI) [6091], Fysik [6128], Partikel- och astropartikelfysik [6133]);Lund-Jensen, Bengt [u13bp6vd] [0000-0003-3867-0336] (KTH [177], Skolan för teknikvetenskap (SCI) [6091], Fysik [6128], Partikel- och astropartikelfysik [6133]);Morley, Anthony [u1bjxgiu] (KTH [177], Skolan för teknikvetenskap (SCI) [6091], Fysik [6128], Partikel- och astropartikelfysik [6133]);Strandberg, Jonas [u1midk9y] [0000-0002-8913-0981] (KTH [177], Skolan för teknikvetenskap (SCI) [6091], Fysik [6128], Partikel- och astropartikelfysik [6133]);Zwalinski, L. for m['PID']=738906, S2\_publication\_ID=d52ff5de878704d29fb541ec4f02f6e89fefd85b  [ {'ids': ['152494319'], 'name': 'Georges Aad'},  {'ids': ['78330846'], 'name': 'Richard Brenner'},  {'ids': ['102772894'], 'name': 'Claus P. Buszello'},  {'ids': ['3913447'], 'name': 'Elias Coniavitis'},  {'ids': ['30619610'], 'name': 'Tord Ekelof'},  {'ids': ['119473983'], 'name': 'Mattias Ellert'},  {'ids': ['33277150'], 'name': 'Arnaud Ferrari'},  {'ids': ['3176238'], 'name': 'Charlie Isaksson'},  {'ids': ['89211416'], 'name': 'Alexander Klevedal Madsen'},  {'ids': ['152320650'], 'name': 'Henrik Ohman'},  {'ids': ['153590434'], 'name': 'Daniel Pelikan'},  {'ids': ['8683287'], 'name': 'Lukasz Zwalinski'}] |

Figure : Warning output when the number of authors does not match in DiVA and S2

|  |
| --- |
|  |

Figure : Example of a paper where only some of the names are in DiVA

# Processing DiVA CSV for author information

An interesting question is: How many KTH authors are there in a given set of DiVA records (as exported in a CSV file, in this case a 92 612 826 byte long file). The program get\_pid\_and\_names.py takes in a CSV file (see Figure 19) and process it producing the files shown in Figure 20. An example from the file of missing IDs is shown in Figure 21. A example of some entries of the the aliases are shown in Figure 22. And Some examples from the pid\_name.csv file are shown in Figure 23. Finally, some examples from the pid\_name file are shown in Figure 24 and in Figure 27. According to KTHB these PI000\* entries were for people whose bibliographic entry they imported but for whom they did not know a KTHID.

|  |
| --- |
| ./get\_pid\_and\_names.py /z3/maguire/SemanticScholar/KTH\_DiVA/kth-exluding-theses-all-level2-2012-2019.csv  file\_name='/z3/maguire/SemanticScholar/KTH\_DiVA/kth-exluding-theses-all-level2-2012-2019.csv'  Finished reading spreadsheet  Number of KTH authors with KTHIDs=4708  Number of KTH authors without KTHIDs=73  number of aliases with fake kthids=384 |

Figure : Processing the CSV file for information about authors

|  |  |
| --- | --- |
| Size (bytes) | Filename |
| 2 008 | kth-exluding-theses-all-level2-2012-2019\_missing\_kthids.csv |
| 2 436 638 | kth-exluding-theses-all-level2-2012-2019\_pid\_name\_aliases.JSON |
| 20 227 792 | 17 kth-exluding-theses-all-level2-2012-2019\_pid\_name.csv |
| 5 521 437 | kth-exluding-theses-all-level2-2012-2019\_pid\_name.JSON |

Figure : Files produces by the program

|  |  |  |
| --- | --- | --- |
| Name | ORCID | PIDs missing KTHIDs for named person |
| Abduljabbar, Meyser |  | [1218213] |
| Ahlmer, Anna-Klara |  | [1236879] |
| Anand, Kumar |  | [542605] |
| Andersson, Birger |  | [1247308] |
| Andersson, K. |  | [1071872] |
| Archenti, Andreas |  | [1304215] |
| Axelsson, M. |  | [621223] |
| Axelsson, Magnus |  | [902760] |

Figure : Examples of missing KTHIDs

|  |
| --- |
| {  "kthid": "u1d13i2c",  "entry": {  "orcid": "000-0002-6066-746",  "aliases": [  {  "Name": "Maguire Jr., Gerald Q.",  "PID": [  528381,  606323,  638177,  …  1367981,  1416571  ]  },  {  "Name": "Maguire, Gerald Q.",  "PID": [  561069  ]  },  {  "Name": "Maguire Jr., Gerald",  "PID": [  561509  ]  },  {  "Name": "Maguire, Gerald Q., Jr.",  "PID": [  913155  ]  }  ]  }  } |

Figure : Example of user information with a kTHID and aliases (some entries elided)

|  |
| --- |
| "913155","Olivecrona, Henrik (Karolinska Institutet, Sweden) (Department of Molecular Medicine and Surgery, Section of Orthopaedics and Sport Medicine);Maguire, Gerald Q., Jr. [u1d13i2c] [0000-0002-6066-746X] (KTH [177], Skolan för informations- och kommunikationsteknik (ICT) [5994], Kommunikationssystem, CoS [5998], Radio Systems Laboratory (RS Lab) [13053]);Noz, Marilyn E. [0000-0002-2442-1622] (New York University, Department of Radiology) (Nuclear Medicine);Zeleznik, Michael P. [0000-0002-0706-1805] (University of Utah) (School of Computing, College of Engineering);Kesteris, Uldis (Department of Orthopedics, Skåne University Hospital);Weidenhielm, Lars [0000-0002-4280-1178] (Karolinska Institutet at Karolinska University Hospital Solna) (Department of Molecular Medicine and Surgery)" |

Figure : An example from the pid\_name.csv file

|  |
| --- |
| {  "PID": 528381,  "entry": {  "Name": "Maguire Jr., Gerald Q.",  "kthid": "u1d13i2c",  "orcid": "000-0002-6066-746"  }  } |

Figure : An example from the pid\_name.JSON file

Note also that in the pid\_name.JSON file there are entries for “fake” KTHIDs, such as shown in Figure 26:

|  |
| --- |
| {  "PID": 537585,  "entry": {  "Name": "Johansson, Linda",  "kthid": "P0000000"  }  }  …  {  "PID": 601297,  "entry": {  "Name": "Ning, Zhijun",  "kthid": "P000000"  }  }  …  {  "PID": 606886,  "entry": {  "Name": "Malmberg, Patrik",  "kthid": "PI000000"  }  }  …  {  "PID": 666261,  "entry": {  "Name": "Tu, Yongming",  "kthid": "PI000000"  }  }  … |

Figure : Fake KTHIDs

|  |
| --- |
| {  "kthid": "PI000000",  "entry": {  "orcid": "000-0003-3871-000",  "aliases": [  {  "Name": "Ripken, Joachim",  "PID": [  528680  ]  },  {  "Name": "DesAutels, Philips",  "PID": [  537910  ]  },  {  "Name": "Dou, Shi Xue",  "PID": [  541016  ]  },  {  "Name": "Malmberg, Patrik",  "PID": [  606886  ]  },  …  ]  }  } |

Figure : Examples of the fake KTHID PI000000

|  |
| --- |
| {  "kthid": "PI000010",  "entry": {  "orcid": false,  "aliases": [  {  "Name": "Stahl, Clement",  "PID": [  1038629  ]  }  ]  }  } |

Figure : Another fake KTHID is shown for PI000010

# Finding some of the missing data for DiVA entries from the 2012-2019 period

As one could see in the previous section, there were quite a lot of missing KTHIDs. I extended get\_pid\_and\_names.py to look at the KTHID information using first the ORCID (if available) and then trying to match against the name (both of these use the kthid\_dict and the affiliation in formation from augmented\_pid\_and\_authors computed within this program). This enabled me to resolve 12 more entries automatically and one entry found the ORCID information, but not a KTHID, so I had to look this up manually. I sent this updated information to [biblioteket@kth.se](mailto:biblioteket@kth.se) in [#ID:KTH-INC-3531526#]. Using this same approach, I was able to identify 76 more authors with their KTHID and in some cases their ORCID based on other records that had this data. I sent this updated information to [biblioteket@kth.se](mailto:biblioteket@kth.se) in [#ID:KTH-INC-3531693#].

Additionally, I found several more fake KTHIDs: P000000, PI 000000, and pi000000-note that the first is missing the “I” after the “P”, the second has a space in it, and the third has a lower case “pi”.

My estimate of the total number of KTH affiliated authors for the span 2012-2019 is: 4708+73+384-1=5164. Thus the remaining set to identify is roughly 368 authors.

After quite a number of hours of manually checking for identifying information and searching, I have been able to reduce the number of missing authors to ~336 representing 408 publications. So most of those I have not found only authored a single publication.

The result is that of ~5164 KTH authors only 336 or 6.5% are missing – not that the numbers are not exact since some “authors” – appear multiple times because of different spellings of their names or other reasons for duplication. Of the 41 578 publications, with KTH author information missing for the 336 authors (of 408 publications) – to a first approximation this is 0.98% (now this number is not exact because the number of KTH authors who are missing from some publication – may have two or more on a common publication – so the percentage could be even smaller). I will only know these numbers better when I have done more analysis of all these data that I have.

After some more manually effort, the current situation (show in Table 3) as computed by:

cat kth-exluding-theses-all-level2-2012-2019\_pid\_name\_aliases-manually\_corrected.json | jq -c '.' > z1; cat z1| sort | uniq >pubs-2012-2019\_pid\_name\_aliases-manually\_corrected.json

Table : Status as of 2020-11-04

|  |  |
| --- | --- |
| 273 | missing |
| 4841 | found |

## Adding KTHIDs

I added a new program augment\_names\_from\_profile.py to use the the KTH profile API (note that you need an application key for this) to augment the information with the user’s first and last names (as stored in their KTH profile). See Figure 28 for an example of running this (note that were 6 entries where the KTHID did not have a user name (as output in the figure) – I have sent mail to IT-support and the library about this.).

|  |
| --- |
| ./augment\_names\_from\_profile.py pubs-2012-2019\_pid\_name\_aliases-manually\_corrected.json pubs-2012-2019\_augmented.json  length of input=5114  processing input  \*\*\* KTHID: xxxxxxxx missing first and last name in {'username': None}  \* alias(es)=['xx, yy]  ...  length of output=5114  time ./augment\_names\_from\_profile.py /z3/maguire/SemanticScholar/KTH\_DiVA/pubs-2012-2019\_pid\_name\_aliases-manually\_corrected.json /z3/maguire/SemanticScholar/KTH\_DiVA/pubs-2012-2019\_augmented.json |

Figure : Augmenting with user's first and last names from KTH profile

The answer from IT-Support where that these were due to typing errors with 5 of the 6 being simple single character errors such as an “l” for and “i”, etc. The 7th was a case of a two character error with two adjacent characters.

There are still 273 entries missing a KTHID. For these entries the program outputs lines of the form shown in Figure 29:

|  |
| --- |
| no kthid in {'kthid': '', 'entry': {'kth': '(KTH [177], Centra [12851], Nordic Institute for Theoretical Physics NORDITA [12850])', 'orcid': '', 'aliases': [{'Name': 'xxxx, yyyy', 'PID': [1130981, 1255535]}, {'Name': 'xxxx, y.', 'PID': [1269339]}]}} |

Figure : Example of output for a user missing a KTHID

The IT-Support response included “the missing kthids could be difficult to decide from a database.“ and “I suspect your remaining 273 names is a question of detective work,” – so detective work remains, although one case was the first and last names being exchanged. (I have ask IT-Support for the KTHID of the user with the parts of the name reversed).

Additionally, I potentially found information to resolve another one of the 273, as I found that the author had also author’s a Master’s thesis and have ask IT-Support to get the KTHID for this user.

I will also note that one of the “authors” in a KTH entity and not an individual author, but have not gotten the organizations pseudo-KTHID. Thus there remain roughly 270 author names yet to be resolved.

## Adding ORCID values

I augmented the augment\_names\_from\_profile.py program to also look at adding ORCIDs from the user’s profile (if they exist in the user’s research profile). The results of how many exist and other data are in Figure 30 and further explained in Table 4.

|  |
| --- |
| missing\_orcids=249, matching\_orcids=1749, missmatched\_orcids=14, user\_missing\_orcids=2829 |

Figure ; Some statistics from trying to add orcid to the entries

Table : Further details of the orcid statistics (the fraction of entries is the fraction of those 4841 entries with KTHIDs)

|  |  |  |  |
| --- | --- | --- | --- |
| Statistic | Number of occurrences | Fraction of entries (%) | Description |
| missing\_orcids | 249 | 5.1 | This case occurs when the incoming data from DiVA does not have an orcid value, but the user’s research profile does have one. |
| matching\_orcids | 1749 | 36.3 | This case occurs when the DiVA data and the user’s research profile have the value orcid value. |
| missmatched\_orcids | 14 | 0.3 | Of the 14, 11 have an lower case “x” for the check digit in the orcid, rather than a capital “X”, while 3 have different orcid values. |
| user\_missing\_orcids | 2829 | 58.7 | This occurs when the user’s research profile does not have an orcid value. |

It is rather interesting that 58.7% of the user’s research profiles lack ORCID IDs. The high fraction of matching orcid values is also positive. The relatively few mismatched orcids value is also quite positive, especially as most of them are only a difference in capitalization. The fact that ~41% of the entries correspond to a user research profile with an ORCID suggests that KTH’s effort to encourage users to have ORCID identifiers has been successful (however, I have not done a correlation of the numbers of these as a function of the time since the kTH campaign started with the years of the publications by the relevant author).

# Finding names and S2 authorIDs

After some effort in dealing with errors in DOIs (where a DiVA record points to a DOI for the entire proceedings rather than a specific paper) and S2 corpus entries that have only some of the authors (for example 500 out of 2884 authrs) [note the errors have been reported to KTHB or S2 as appropriate], the match\_names.py program now has special case handling to deal with the errors. Should these errors get fixed the special caes can be removed.

Processing hard 186, lead to 298 JSON records providing mappings between a name, KTH , and S2\_author\_ID, see the file names-186.json. [Note that these JSON files are not visible in the archive, due to GPDR.]

# DiVA data cleanup

Before embarking on finding mappings between KTHIDs and S2 authorIDs, I decided to clean up the DiVA data that I had and also to create a file with author information. The latter was inspired by the S2 concept of aliases for authors (i.e., other versions of their name that are used).

|  |
| --- |
| augmented\_by\_kthid for u1d13i2c={'kthid': 'u1d13i2c', 'profile': {'firstName': 'Gerald Quentin', 'lastName': 'Maguire Jr'}, 'kth': '(KTH [177], Skolan för informations- och kommunikationsteknik (ICT) [5994], Kommunikationssystem, CoS [5998])', 'orcid': '0000-0002-6066-746X', 'aliases': [{'Name': 'Maguire Jr., Gerald Q.', 'PID': [528381, 606323, 638177, 675824, 675849, 690825, 690828, 706514, 733621, 852295, 854051, 866222, 866274, 948397, 948549, 948742, 1068493, 1087906, 1177431, 1180496, 1184756, 1230455, 1314634, 1367981, 1416571]}, {'Name': 'Maguire, Gerald Q.', 'PID': [561069]}, {'Name': 'Maguire Jr., Gerald', 'PID': [561509]}, {'Name': 'Maguire, Gerald Q., Jr.', 'PID': [913155]}]} |

Figure : Example of the information that was collected for an author to show the users of aliases

## Collecting author information

To collect author data, I wrote a number of python programs:

|  |  |
| --- | --- |
| get\_pid\_and\_names.py | Process a CSV file exported from DiVA and extract authors name and the publication IDs used for a given version of the author’s name (i.e., an alias). |
| augment\_names\_from\_profile.py | Use the KTHID of the author to get their first and lat names and ORCID information (if available) using the KTH profile API. |
| check\_augmented\_names.py | Takes the augmented author JSON file and check for missing KTHIDs. |
| further\_augment\_from\_CSV\_file.py | Further extend the information about the authors – several variants of this were used to track down possible errors in the DiVA data, to identify possible aliases, and to check against the additional ORCID and KTHIDs (as described in Section 10.2). |

The augmented author information consists of JSON lines of the form shown in Figure 32.

|  |
| --- |
| {"kthid": "u1d13i2c", "entry": {"orcid": "0000-0002-6066-746X", "kth": "(KTH [177], Skolan för informations- och kommunikationsteknik (ICT) [5994], Kommunikationssystem, CoS [5998])", "aliases": [{"Name": "Maguire Jr., Gerald Q.", "PID": [528381, 606323, 638177, 675824, 675849, 690825, 690828, 706514, 733621, 852295, 854051, 866222, 866274, 948397, 948549, 948742, 1068493, 1087906, 1177431, 1180496, 1184756, 1230455, 1314634, 1367981, 1416571]}, {"Name": "Maguire, Gerald Q.", "PID": [561069]}, {"Name": "Maguire Jr., Gerald", "PID": [561509]}, {"Name": "Maguire, Gerald Q., Jr.", "PID": [913155]}]}, "profile": {"firstName": "Gerald Quentin", "lastName": "Maguire Jr"}} |

Figure : Example entry from pubs-2012-2019\_augmented.json

the "kthid" field contains a KTHID or a space (when I don’t know the KTHID of the author or a special value as a “fake” KTHID. These “fake” KTHIDs are different from those used by the publication infrastructure unit of KTHB, in that they start with special characters and are designed to be unique (for different periods of time).

Strings that start with ⚠⚠ followed by a number are my fake KTHIDs, these are uniquely assigned when I read the file in with the program (further\_augment\_from\_CSV\_file.py), they are not supposed to be meaningful - only that for the run of the program they are unique. Therefore, an entry with an empty kthid will be assigned one of these type of fake IDs.

Strings that start with ⚑ (i.e., a black flag) followed by a number are also fake KTHIDs. However, these are manually assigned to people whom I have found (via IT-support) do not have a KTHID (or more correctly IT-support has no record of them having a KTHID).

All of the other strings are KTHIDs mined from the DiVA data on non-thesis documents between 2012-and 2019 or manually found by some other means (including the DiVA data from 1st, 2nd, and 3 cycle theses).

## Using jq to extract entries from the augmented JSON file

You can use jq to get JSON records from the file such as shown in Figure 33.

|  |
| --- |
| cat authors\_pubs\_2912-2919.json | jq -c 'select(.kthid == "u1d13i2c")'  will produce  {"kthid":"u1d13i2c","profile":{"firstName":"Gerald Quentin","lastName":"Maguire Jr"},"kth":"(KTH [177], Skolan för informations- och kommunikationsteknik (ICT) [5994], Kommunikationssystem, CoS [5998])","orcid":"0000-0002-6066-746X","aliases":[{"Name":"Maguire Jr., Gerald Q.","PID":[528381,606323,638177,675824,675849,690825,690828,706514,733621,852295,854051,866222,866274,948397,948549,948742,1068493,1087906,1177431,1180496,1184756,1230455,1314634,1367981,1416571]},{"Name":"Maguire, Gerald Q.","PID":[561069]},{"Name":"Maguire Jr., Gerald","PID":[561509]},{"Name":"Maguire, Gerald Q., Jr.","PID":[913155]}]} |

Figure : Example of using jq to get an entry from the augmented JSON file

In the above example, the "profile" data comes from the KTH profile API. The "kth" data comes from the first record I found with an affiliation for this author. The "orcid" is mined from both the KTH profile API and from the DiVA records. the list of aliases is of the form:

{"Name": "last, first", "PID": [pid1, pid2, ...]}

where the pid1, pid2, are DiVA documents where I found this version of the name used. This PID data is not complete - but simply provides a document where I found the name in this form.

The affiliation information should probably also have a form similar to the aliases, as people have different affiliations in the DiVA data for different publications.

## Extra ORCID and KTHID information

On 4December 2020, I got a file of ORCID;KTHID values from Anders Wändahl of KTHB.

I quickly made a version of further\_augment\_from\_CSV\_file.py to process this information and ran the program as shown in Figure 34.

|  |
| --- |
| ./further\_augment\_from\_CSV\_file.py /z3/maguire/SemanticScholar/KTH\_DiVA/kth-exluding-theses-all-level2-2012-2019-corrected.csv /z3/maguire/SemanticScholar/KTH\_DiVA/pubs-2012-2019\_augmented.json /z3/maguire/SemanticScholar/KTH\_DiVA/orcid\_kth-id\_2020-12-04.txt  file\_name='/z3/maguire/SemanticScholar/KTH\_DiVA/kth-exluding-theses-all-level2-2012-2019-corrected.csv'  augmented\_json\_file\_name='/z3/maguire/SemanticScholar/KTH\_DiVA/pubs-2012-2019\_augmented.json'  orcid\_fileName='/z3/maguire/SemanticScholar/KTH\_DiVA/orcid\_kth-id\_2020-12-04.txt'  length of augmented\_by\_kthid=9467  augmented\_by\_kthid for u1d13i2c={'kthid': 'u1d13i2c', 'profile': {'firstName': 'Gerald Quentin', 'lastName': 'Maguire Jr'}, 'kth': '(KTH [177], Skolan för informations- och kommunikationsteknik (ICT) [5994], Kommunikationssystem, CoS [5998])', 'orcid': '0000-0002-6066-746X', 'aliases': [{'Name': 'Maguire Jr., Gerald Q.', 'PID': [528381, 606323, 638177, 675824, 675849, 690825, 690828, 706514, 733621, 852295, 854051, 866222, 866274, 948397, 948549, 948742, 1068493, 1087906, 1177431, 1180496, 1184756, 1230455, 1314634, 1367981, 1416571]}, {'Name': 'Maguire, Gerald Q.', 'PID': [561069]}, {'Name': 'Maguire Jr., Gerald', 'PID': [561509]}, {'Name': 'Maguire, Gerald Q., Jr.', 'PID': [913155]}]}  total entries=9467, number\_of\_entries\_with\_KTHIDs=8846, number\_of\_entries\_with\_KTHID\_and\_ORCID=3553, number\_of\_entries\_with\_fake\_KTHIDs=621, number\_of\_entries\_with\_fake\_KTHIDs\_with\_ORCID=34  u17jh27g: 0000-0001-5905-8467  for user=u18xqygi different ORCID=0000-0001-7020-1551 than existing=0000-0002-4830-7832  for user=u1viadzh different ORCID=0000-0001-9314-545X than existing=0000-0001-9314-545x  for user=u1b2c0lu different ORCID=0000-0001-9567-155X than existing=0000-0001-9567-155x  for user=u165ghxx different ORCID=0000-0002-0697-846X than existing=0000-0002-0697-846x  u1337frh: 0000-0002-1409-6352  for id=u1cmhq5l and orcid=0000-0002-3937-9359, found user=⚠⚠277  for user=u1opd8t3 different ORCID=0000-0002-4477-971X than existing=0000-0002-4477-971x  for user=u145n239 different ORCID=0000-0002-4501-695X than existing=0000-0002-4501-695x  for id=u1wab1qn and orcid=0000-0002-5050-5176, found user=⚠⚠280  u1qgi0an: 0000-0002-6461-439X  u17dwg3s: 0000-0002-7422-3966  for user=u1xbdlaj different ORCID=0000-0002-7987-1567 than existing=0000-0003-2432-7617  for user=u1i928wd different ORCID=0000-0002-8170-379X than existing=0000-0002-8170-379x  u1h4jl4h: 0000-0002-9432-254X  for user=u1670dm5 different ORCID=0000-0003-1377-565X than existing=0000-0003-1377-565x  for user=u1hxruli different ORCID=0000-0003-1443-403X than existing=0000-0003-1443-403x  for user=u167dmzs different ORCID=0000-0003-1654-841X than existing=0000-0003-1654-841x  u1huwj6n: 0000-0003-1693-1320  for user=u1dypakr different ORCID=0000-0003-3070-794X than existing=0000-0003-3070-794x  already\_know\_orcid=2911, new\_orcid=6, differing\_orcids=12,new\_users=740  Finished reading extra ORCID info |

Figure : Using further\_augment\_from\_CSV\_file.py with the extra ORCID and KTHID file

Note that the above run of the program (which purposely returns after processing the extra ORCID information) gave me:

* KTHIDs for two users that I did not have:

for id=u1cmhq5l and orcid=0000-0002-3937-9359, found user=⚠⚠277

for id=u1wab1qn and orcid=0000-0002-5050-5176, found user=⚠⚠280

* 2911 already known ORCIDs
* 740 new users
* 6 new ORCIDs for authors I already had:

u17jh27g: 0000-0001-5905-8467

u1337frh: 0000-0002-1409-6352

u1qgi0an: 0000-0002-6461-439X

u17dwg3s: 0000-0002-7422-3966

u1h4jl4h: 0000-0002-9432-254X

u1huwj6n: 0000-0003-1693-1320

* 12 mis-matches versus what I already had:
* for 10 of these the difference is that the DiVA record(s) had a "x" rather than an "X" in the check character:

for user=u1viadzh different ORCID=0000-0001-9314-545X than existing=0000-0001-9314-545x

for user=u1b2c0lu different ORCID=0000-0001-9567-155X than existing=0000-0001-9567-155x

for user=u165ghxx different ORCID=0000-0002-0697-846X than existing=0000-0002-0697-846x

for user=u1opd8t3 different ORCID=0000-0002-4477-971X than existing=0000-0002-4477-971x

for user=u145n239 different ORCID=0000-0002-4501-695X than existing=0000-0002-4501-695x

for user=u1i928wd different ORCID=0000-0002-8170-379X than existing=0000-0002-8170-379x

for user=u1670dm5 different ORCID=0000-0003-1377-565X than existing=0000-0003-1377-565x

for user=u1hxruli different ORCID=0000-0003-1443-403X than existing=0000-0003-1443-403x

for user=u167dmzs different ORCID=0000-0003-1654-841X than existing=0000-0003-1654-841x

for user=u1dypakr different ORCID=0000-0003-3070-794X than existing=0000-0003-3070-794x

* Two users where there is a difference:

for user=u18xqygi different ORCID=0000-0001-7020-1551 than existing=0000-0002-4830-7832

for user=u1xbdlaj different ORCID=0000-0002-7987-1567 than existing=0000-0003-2432-7617

For the first of these the users with different ORCID values, the first author actually has two ORCIDs. Her KTH profile page <https://www.kth.se/profile/olga> points to ORCID: <https://orcid.org/0000-0001-7020-1551>

However, there are many record in DiVA (such as the following that shows another ORCID for this user):

"1307966","Kordas, Olga [u18xqygi] [0000-0002-4830-7832] (KTH [177], Skolan för arkitektur och samhällsbyggnad (ABE) [5850], Hållbar utveckling, miljövetenskap och teknik [13604]) (UrbanT);

Note that <https://orcid.org/0000-0002-4830-7832> - seems to be much more complete.

For the second, the author’s KTH profile shows:

user={'defaultLanguage': 'en', 'acceptedTerms': True, 'isAdminHidden': False, 'avatar': {'visibility': 'public'}, '\_id': 'u1xbdlaj', 'kthId': 'u1xbdlaj', 'username': 'chrdan', 'homeDirectory': '\\\\ug.kth.se\\dfs\\home\\c\\h\\chrdan', 'title': {'sv': 'FORSKARE', 'en': 'RESEARCHER'}, 'streetAddress': 'OSQUARS BACKE 5', 'emailAddress': 'christina.bodin.danielsson@arch.kth.se', 'telephoneNumber': '087908541', 'isStaff': True, 'isStudent': False, 'firstName': 'Christina', 'lastName': 'Bodin Danielsson', 'city': 'Stockholm', 'postalCode': '10044', 'remark': '', 'lastSynced': '2020-09-10T06:29:20.000Z', 'researcher': {'researchGate': '', 'googleScholarId': '', 'scopusId': '', 'researcherId': '', 'orcid': '0000-0002-7987-1567'}, 'courses': {'visibility': 'public', 'codes': [], 'items': []}, 'worksFor': {'items': [{'key': 'app.katalog3.A.AD', 'path': 'a/ad', 'location': 'OSQUARS BACKE 5, 10044 STOCKHOLM', 'name': 'ARKITEKTURSKOLAN', 'nameEn': 'ARCHITECTURE'}, {'key': 'app.katalog3.A.AD.ADE', 'path': 'a/ad/ade', 'location': 'OSQUARS BACKE 5, 10044 STOCKHOLM', 'name': 'ARKITEKTUR FORSKNING EXTERN', 'nameEn': ''}]}, ...

hence I used this in my entry:

{"kthid": "u1xbdlaj", "entry": {"orcid": "0000-0003-2432-7617", "kth": "(KTH [177], Skolan för arkitektur och samhällsbyggnad (ABE) [5850], Arkitektur [5851]) (Arc plan)", "aliases": [{"Name": "Bodin Danielsson, Christina", "PID": [704686, 842404, 1055759]}, {"Name": "Danielsson, Christina", "PID": [1359413]}]}, "profile": {"firstName": "Christina", "lastName": "Bodin Danielsson"}}

While https://orcid.org/0000-0002-7987-1567 shows:

Given Names Deactivated Family Name Deactivated

ORCID iD

https://orcid.org/0000-0002-7987-1567

This account has been deprecated, please see account https://orcid.org/0000-0003-2432-7617 for the latest information

So the one ORCID has been replaced by another and the ORCID I was using was correct.

Given the programs outputs, I updated my local copy of the publication information (i.e., the CSV file) and the augmented authors’ information JSON file.

References

[1] Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld, ‘S2ORC: The Semantic Scholar Open Research Corpus’, in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, 2020, pp. 4969–4983 [Online]. DOI: 10.18653/v1/2020.acl-main.447

1. Microsoft Academic Graph [↑](#footnote-ref-1)