Apr 14, 2021

Investigating Invalid DOIs in COCI

Nooshin Shahidzadeh[1], Alessia Cioffi[1], Arianna Moretti[1], Sara Coppini[1]

[1]University of Bologna

In Development dx.doi.org/10.17504/protocols.io.bt5xnq7n

Open Science 2020/2021

N Nooshin Shahidzadeh

ABSTRACT

Purpose

The purpose of this research is to find the publishers responsible for the missing citations in COCI (the OpenCitations Index of Crossref open DOI-to-DOI citations) by sending incorrect metadata to Crossref, the publishers to whom such invalid citations point to, and the number of previously invalid citations which are currently valid.

Study design/methodology

In order to find the invalid citations, we use an already generated CSV file, containing the DOIs of invalid citations and their correct form, which is available online. These DOIs along with the COCI REST API can lead us to the responsible and referenced publishers.

Findings

We found for each individual publisher 1) the number of incorrect given citations metadata sent, and 2) the number of invalid citations received. We also extracted the total number of invalid citations that have since been corrected.

Originality/value

The results of this research may point us to publishers who generally send out incorrect citation metadata and, inversely, those who generally receive invalid citations. These findings can first of all raise awareness of the accuracy of certain publishing houses in managing their metadata (or lack thereof). Moreover, finding these trends and showcasing the labor of the corrections may lead to increasingly valid citations if the proper measures are taken.

Research limitations/implications

Based on the available data for the COCI, there may be a slight bias in our sample, causing some publishers to be incorrectly represented.

DOI

dx.doi.org/10.17504/protocols.io.bt5xnq7n

PROTOCOL CITATION

Nooshin Shahidzadeh, Alessia Cioffi, Arianna Moretti, Sara Coppini 2021. Investigating Invalid DOIs in COCI. **protocols.io** https://dx.doi.org/10.17504/protocols.io.bt5xnq7n

CREATED

Apr 12, 2021

LAST MODIFIED

Apr 14, 2021

Citation: Nooshin Shahidzadeh, Alessia Cioffi, Arianna Moretti, Sara Coppini (04/14/2021). Investigating Invalid DOIs in COCI. https://dx.doi.org/10.17504/protocols.io.bt5xnq7n

PROTOCOL INTEGER ID

49047

Reading the CSV Data

1 First we read the CSV of the invalid DOIs, containing all the invalid cited DOIs in one column and their valid citing counterparts in another.

Citations to invalid DOI-identified entities

Creating the Output JSON File

2 We create a JSON file for the output data.

2.1 We add a key called "responsible_publishers" that will include a dictionary containing all publisher data for citing articles.

2.2 We add a key called "receiving_publishers" that will include a dictionary containing all publisher data for cited articles.

2.3 We add a key called "total_number_of_corrected_dois" that will hold only a number.

Processing Each Line in the CSV File and Extracting the Needed Information

3 We use the DOI REST API to search for each of the invalid cited DOIs in the CSV. If we receive the response code "1", we know that the citation data is now valid, we now have two cases.
Step 3 includes a Step case.
Still Invalid
Now Valid

Extracting Publisher Data for Citing and Cited Articles

step case

Still Invalid

In this case we will have to extract the publisher data for both the citing and the cited articles.

4 We search through the REST API for Crossref for the metadata of each valid citing DOI. The publisher name is the value of "publisher" and the ID is the first 6 characters of the DOI of the citing article.

4.1 We check if the publisher ID exists in the "responsible_publishers" dictionary as a key. If it does not, we create a new item in the dictionary: the key is the ID of the publisher, and the value is a dictionary containing the name of the publisher and the number of invalid citation data it has sent to Crossref (which at the time of creation would be 1). If, on the other hand, it does exist, we just increment the number inside the inner dictionary of that publisher by 1.

5 We assume the first 6 characters of the invalid cited DOI to be the ID of the publisher to which the invalid citation is

Citation: Nooshin Shahidzadeh, Alessia Cioffi, Arianna Moretti, Sara Coppini (04/14/2021). Investigating Invalid DOIs in COCI. https://dx.doi.org/10.17504/protocols.io.bt5xnq7n

pointing.

6 We check the validity of this ID through the Crossref REST API. If this ID exists and belongs to a publisher, we add it to the "receiving_publisher" dictionary as a key, the value of which would be a dictionary containing the number of the citations received and the name of the publisher. (Similar to step 4.1 above)
Step 6 includes a Step case.
Not Valid
Valid

step case

Not Valid

In this case we will try to find the publisher name through other ways.

7 We search through the REST API of Crossref for the reference list of the citing article and see if there is any additional data (for example in the "unstructured" field) given in the XML that could lead us to the cited publisher.

7.1 If additional data was found, then we use the "query" attribute of the Crossref REST API to search for the newly found string. If any publication was found, we add the "publisher" field of that publication to the "receiving_publishers" dictionary, in the form explained above.

Output

8 We return the completed JSON file after processing all the lines. (CSV addition possible later.)

Citation: Nooshin Shahidzadeh, Alessia Cioffi, Arianna Moretti, Sara Coppini (04/14/2021). Investigating Invalid DOIs in COCI. https://dx.doi.org/10.17504/protocols.io.bt5xnq7n