Classification of URLs Citing Research Artifacts in Scholarly Documents based on Distributed Representations

Masaya Tsunokake
Graduate School of Informatics
Nagoya University
Furo-cho, Chikusa-ku, Nagoya, Japan
tsunokake.masaya.z3@s.mail.nagoya-u.ac.jp

Shigeki Matsubara
Information and Communications
Nagoya University
Furo-cho, Chikusa-ku, Nagoya, Japan
matubara@nagoya-u.jp

ABSTRACT

This paper describes methods for classifying URLs referring to research artifacts in scholarly papers, and examines their classification performance. The methods discriminate whether a URL refers to a research artifact or not and classify the identified URL into "tool" or "data." The methods use distributed representations obtained from citation contexts of the URL. Each component of a URL can be regarded as a word, and the meaning of the entire URL can be generated by synthesizing the distributed representation of each component using compositional functions. This paper evaluates several types of compositional functions from the viewpoint of classification performance. Experiments with using URLs in international conference papers showed the effectiveness of our proposed compositional functions.

CCS CONCEPTS

• **Information systems** → **Information extraction**; *Clustering and classification*; **Digital libraries and archives**.

KEYWORDS

Open science, Data repository, Information extraction, Data citation, Scholarly document processing

1 INTRODUCTION

Open science is an activity for promoting sharing and utilizing research artifacts[1]. One strategy for promoting these activities is to develop and provide repositories for research artifacts. In recent years, repositories of research artifacts have been developed, such as Zenodo[2] and Mendeley Data[3]. National infrastructures for sharing research artifacts have also been developed, such as Australian National Data Service[4][28], European Open Science Cloud[5][5], Research Data Shared Service[6], National Data Service[7][27], and NII Research Data Cloud[8].

In order to establish a research artifact repository, it is required to register research artifacts and their metadata[9]. The number of research artifact citations in scholarly papers has been increasing in recent years. Automatically extracting information on research artifacts from a large number of scholarly papers makes the development or expansion of a repository more efficient.

This paper describes methods for classifying URLs referring to research artifacts cited in scholarly papers, all of which are extended from our previous method [29], and examines their classification performance. The methods discriminate whether a URL in scholarly papers refers to a research artifact or not, and classify identified research artifacts into the type "tool" (e.g., programs and software) or "data" (e.g., measurement data and test data).

Our previous approach uses words surrounding the URL in scholarly papers, that is, distributed representations obtained from citation contexts of the URL. The meanings of non-natural language strings such as URLs can be expressed as distributed representations. Each component of a URL, such as domain name, directory name, and file name, can be regarded as a word, and the meaning of the entire URL can be generated by synthesizing the distributed representation of each component using compositional functions.

This paper evaluated several types of compositional functions from the viewpoint of classification performance. Experiments using URLs in international conference papers showed the effectiveness of our proposed compositional functions.

2 URL REFERRING TO RESEARCH ARTIFACT

2.1 Metadata in Research Artifact Repository

Creating metadata is necessary to facilitate access to resources in repositories. The most basic metadata scheme is Dublin Core Metadata Element Set[10]. As an example, Table 1 shows the metadata

Table 1: Metadata of Penn Treebank in OLAC (in part)

Property	Value
title	Treebank-3
contributor	Mitchell P. Marcus et al.
publisher	Linguistic Data Consortium
date	1999
type (DCMI)	Text
description	This release contains the following Treebank-2 Material ... will include these missing files.
identifier	DOI: 10.35111/gq1x-j780 https://catalog.ldc.upenn.edu/LDC99T42

[1]In this paper, we denote research artifacts as materials generated or used in the course of research activities, such as tools (e.g., software, program) and data (e.g., measurement data, test data).
[2]https://zenodo.org/
[3]https://data.mendeley.com/
[4]https://www.ands.org.au/
[5]https://ec.europa.eu/research/openscience/index.cfm?pg=open-science-cloud
[6]https://www.jisc.ac.uk/rd/projects/research-data-shared-service
[7]http://www.nationaldataservice.org/
[8]https://rcos.nii.ac.jp/en/service/
[9]Information about research artifacts (e.g., name, creator, type, and usage)
[10]https://www.dublincore.org/specifications/dublin-core/dces/

Figure 1: Design of classification task

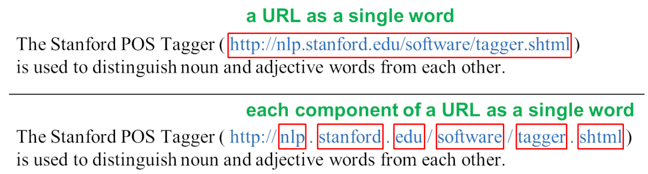

Figure 2: Example of different semantic units for giving meaning to a URL. The sentence is quoted from [30].

of Penn Treebank [14] on the Open Language Archives Community (OLAC)[11] storing information on language resources[12] (e.g., corpora, dictionaries) according to Dublin Core.

If such information can be extracted automatically, the generation of metadata can become easier. Kozawa et al. [12] have proposed a method for automatically extracting usage information about language resources from scholarly papers. The method identifies language resources using their names registered in SHACHI[13] [26] as clues. For this reason, research artifacts whose usage information can be extracted are limited to those in repositories. We aim to extract information about the type of research artifact, including ones not stored in existing repositories.

2.2 Research Artifact Citation

Recently, research artifacts, such as datasets and software, have been increasingly cited in scholarly papers. Thus, there is a growing movement to establish formal rules for data and software citations, as FORCE11 has declared "Data Citation Principles" [6] and "Software Citation Principles" [24]. However, it is a long way off before this practice is widely spread among researchers. Howison and Bullard [8] have shown that there were many informal citations appearing in biology papers. One strategy for automatic identification of the informal citations is to identify research artifact mentions in the body text [13]. Some studies address the identification of dataset names [10, 20, 23] while others do that of software names [3, 4, 22]. On the other hand, there are many cases in which research artifacts are listed in the reference section [11] or are cited by providing the corresponding URL.

Providing URLs in papers is a common form of Web citation. Yang et al. [31] have analyzed such citations, and NLPExplorer[14] [17], which is a service for searching scholarly papers, provides access to URLs cited by the papers. We also focus on URLs in scholarly papers because many published research artifacts are accessible on the Web. However, not all URLs in scholarly papers refer to research artifacts. Therefore, we aim to identify URLs referring to the research artifacts in scholarly papers.

2.3 Classification of URLs in Scholarly Papers

Fig. 1 illustrates the design of task in our study. The goal is to identify URLs referring to research artifacts from scholarly papers and categorize them. In this task, URLs in scholarly papers are classified into the following three categories[15]:

- **tool**: program, software, toolkit, etc.
  - https://nlp.stanford.edu/projects/glove/
  - https://github.com/google-research/bert
  - http://www.nltk.org/
- **data**: observation data, experimental data, data source, etc.
  - http://qwone.com/~jason/20Newsgroups/
  - http://babelnet.org
  - http://answers.yahoo.com[16]
- **other**: Not research artifacts (e.g., publications, services).
  - http://is.muni.cz/publication/884893/en
  - http://www.apple.com/ios/siri
  - https://www.mturk.com

Our previous method [29] uses words surrounding a URL for a classifier. URLs are placed on either footnote, reference section, or body text. Even if a URL is in a footnote or the reference section, the sentences referring to the corresponding footnote or reference generally exist in the body text. For example, a footnote "[6]http://lemurproject.org/clueweb09/" is referred to by the following sentence in the body text [32]:

> The ClueWeb09[6] dataset is a collection of 1 billion webpages (5TB compressed in raw HTML) in 10 languages by Carnegie Mellon University in 2009.

By observing this sentence, it turns out that the above URL is provided to refer to a corpus. This paper calls one sentence referring to a URL in the body text as "**citation context**."

3 URL CLASSIFICATION BASED ON DISTRIBUTED REPRESENTATIONS

A comprehensive view of all citation contexts for each URL allows us to classify it properly. Based on this idea, our previous approach [29] obtains a distributed representation of a URL from its citation contexts and uses it for classification. According to the distributional hypothesis [7], even for non-natural language strings such as URLs, their meaning could be obtained from words co-occurring in their surroundings. The following two approaches with different semantic units can be considered:

- regarding an entire string of a URL as a word
- regarding each component of a URL as a word, and obtaining the meaning of the URL from that of each component

[11]http://www.language-archives.org/
[12]In the field of natural language processing, the accumulation and use of the metadata of language resources have long been encouraged because the need for their repositories have been recognized [9].
[13]http://shachi.org/
[14]http://nlpexplorer.org

[15]Enumerated URLs in each category are examples of URLs belonging to the category.
[16]This URL refers to a Web page for a Q&A service. However, the scholarly papers tend to refer to the URL for pointing to a data source of question answering datasets. Therefore, in scholarly papers, the type can be considered to the "data."

Fig. 2 shows an example for two different semantic units. The acquisition unit of a distributed representation also varies depending on the employed approach.

Nanba [16] obtained distributed representations of URLs based on the former approach. He proposed a method named "W2V-URL" of giving keywords to URLs by word2vec [15]. In W2V-URL, words whose distributed representation is highly similar to that of a URL are assigned to the URL as the keywords. We employed the URL classification based on distributed representations of URLs as baseline method [29]. The procedure of the baseline is as follows:

(1) Assign a unique ID to each URL in scholarly papers and convert each URL to a tag with the corresponding ID[17]
(2) Obtain a distributed representation for each tag
(3) Classify URLs using distributed representations

On the other hand, we proposed the URL classification method based on the latter approach [29]. Thus, the method regards each component of a URL as a word and obtains its distributed representation. In some cases, the type of the target referred to by a URL can be inferred from the domain or directory name constituting the URL. For example, it can be inferred from the expressions of directory names "tools" and "TreeTagger" that a URL "http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/" points to a tagging tool. Distributed representations of these components obtained from citation contexts may be able to capture the meaning of substrings. This paper calls a component of URL as a **URL element** (e.g., host name, domain name, directory name, file name, and extension).

In our previously proposed method, URLs in scholarly papers are classified according to the following procedure[18]:

(1) Decompose each URL in scholarly papers into URL elements
(2) Assign a unique ID to each URL element and convert each URL element into a tag with the corresponding ID
(3) Obtain a distributed representation for each tag
(4) Classify each URL using the vector computed by adding distributed representation of each URL element in the URL

4 COMPOSITION OF DISTRIBUTED REPRESENTATIONS OF URL ELEMENTS

Our previous method classifies URLs with vectors which are computed by adding distributed representation of each URL element [29]. In this paper, some types of compositional functions for distributed representations of URL elements are evaluated.

Our previous method tends to misclassify URLs whose gold label are "tool" into the "data" class and vice versa. In addition, misclassified URLs tend to be short (i.e., URLs with a small number of directories). For example, the URL "https://twitter.com/," which has the "data" label, was misclassified into the "tool" class. It is considered that this disadvantage is caused by URL elements with extremely high frequency, such as host and domain names. For example, URL element "com" is a generic top-level domain appearing in many URLs. The bias of citation contexts in scholarly papers might lead to characterizing the distributed representation of "com" as "tool" even though it is not critical evidence in the URL classification. The classification of short URLs especially is affected by URL elements whose frequency is extremely high.

This paper revises the step (4) in the above procedure of the previous classification method as follows:

(4)′ Classify a URL using the vector combined by $f(v_1, \ldots, v_n)$, where $f(\cdot)$ is a compositional function, and v_i is the vector of the ith URL element in the URL

We evaluate fundamental manipulations as compositional functions, such as averaging, summation, and max-pooling[19]. In addition, we also evaluate several functions to improve our previous method.

URL elements with extremely high frequency, such as host and domain names, are considered to be less useful for classification. To weaken the influence of frequent URL elements, we extend the fundamental manipulations by weighting with the entropy of URL elements. The entropy is computed on the basis of frequency of each URL element in scholarly papers. The entropy of each URL element was computed by

$$-\log_2 \frac{Count\,(w)}{\Sigma_{w'} Count\,(w')} \tag{1}$$

where w is the target URL element, w' is an arbitrary URL element in the set of all URL elements, and $Count(\cdot)$ is a function counting its argument in the scholarly papers. In addition, the top-level domains can be considered not to contribute much to the classification of the targets referred to by URLs. Therefore, we also employed manipulations except for distributed representations of top-level domain names simply.

This task can be regarded as a sequence classification task. Using a model based on recurrent neural networks (RNN) as the compositional function may realize to get better weights for synthesizing URL elements and incorporate order information into input features. Therefore, we also verify a classification method employing the gated recurrent unit (GRU) [2] as a gated RNN. Fig. 3 shows the architecture of the verified model.

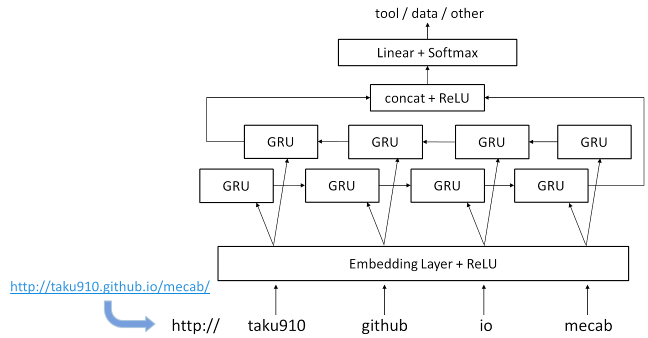

Figure 3: Architecture of our model using GRU

5 EXPERIMENT

[17] For example, every "http://nlp.stanford.edu/software/tagger.shtml" is converted to the tag "[URL2495]."

[18] For example, a URL "http://nlp.stanford.edu/software/tagger.shtml" is converted to a sequence of the URL elements: "nlp," "stanford," "edu," "software," "tagger," "shtml." In addition, each of them is converted to tags "[PARTS7070]," "[PARTS9479]," "[PARTS3891]," "[PARTS9344]," "[PARTS9680]," "[PARTS9182]," respectively.

[19] Taking the maximum value along each dimension.

Table 2: Hyperparameters of the adopted distributed representation and classification model

Compositional function	Parameters of word2vec			Classification model	Standardization
	epochs	window	dimension		
None (baseline method)	20	10	300	logistic regression with one-vs-rest	False
averaging	10	5	800	logistic regression with one-vs-one	True
summation	20	10	700	logistic regression with one-vs-rest	True
max-pooling	20	5	400	logistic regression with one-vs-one	True

Table 3: List of frequent URL elements in scholarly papers

Rank	URL element	Freq.	Rank	URL element	Freq.
1	org	4983	11	pdf	3545
2	com	3545	12	arxiv	3505
3	www	3505	13	nlp	1964
4	github	1964	14	google	1712
5	aclweb	1712	15	abs	1704
6	anthology	1704	16	ac	1491
7	edu	1491	17	net	920
8	doi	920	18	v1	757
9	html	757	19	p	318
10	cs	625	20	stanford	309

5.1 Experimental Data

Experimental data were the same as our previous study [29]. The data were generated from scholarly papers in the proceedings of ACL 2010–2019, which are the international conferences in the field of natural language processing. Concretely, we collected PDF files from ACL Anthology [25] and converted them into texts in preserving their structural information[20] by PDFNLT-1.0[21] [1]. The number of papers was 3,837. There were 12,568 URL occurrences[22] and the number of distinct URLs was 9,480. The average number of URL elements in the URLs was 4.72, and the number of distinct URL elements is 11,724. Table 3 shows the frequent URL elements.

Many URLs are provided in footnotes or references[23]. To capture citation contexts, these URLs were mechanically inserted into the body texts according to where the corresponding footnote or reference is referred to. After that, URLs or URL elements were converted to tags according to the procedure described in Section 3. For example, in the baseline method of our previous study [29], the citation context illustrated in Section 2.3 is transformed as follows:

> The ClueWeb09 [URL2164] dataset is a collection of 1 billion webpages (5TB compressed in raw HTML) in 10 languages by Carnegie Mellon University in 2009.

These processed texts of papers were used for obtaining the distributed representations.

To evaluate performances for URL classification, we labeled URLs appearing frequently in the scholarly papers with "tool," "data," or "other." The created dataset contains 500 annotated URLs. The URLs described in Section 2.3 are examples extracted from this annotated dataset. The labeling ratios of "tool," "data," and "other" in 500 URLs are 39.8%, 33.6%, and 26.6%, respectively. Of them, 100 URLs are used as a development set.

[20]Components of a scholarly paper such as title, authors, body text, figures, tables, captions, footnotes, and reference list.
[21]https://github.com/KMCS-NII/PDFNLT-1.0
[22]Strings beginning with either "http://," "https://," or "ftp://" were identified as URLs.
[23]The rates of URLs in footnotes and references are 0.767 and 0.127, respectively.

Table 4: Experimental result of basic compositional functions

Compositional function	Accuracy	Macro-averaging evaluation		
		precision	recall	F1-score
None (baseline)	0.785	0.781	0.777	0.779
averaging	0.798	0.811	0.805	0.808
summation	0.800	0.807	0.809	0.808
max-pooling	0.750	0.749	0.759	0.754

5.2 Experiment for Basic Functions

We used word2vec [15] to obtain distributed representations and Gensim[24] [21] for its implementation. Sentence segmentation and word tokenization were also performed by using gensim.

As the baseline, we also evaluate the classification method regarding a URL as a single word (described in Section 3). Since the baseline method does not decompose URLs into URL elements, the compositional function does not exist.

For each method, the best parameters of word2vec[25] were selected on the basis of the performance in the development set. Similarly, we also chose a classification model from logistic regression, linear SVM, and nonlinear SVM with RBF kernel, a multi-class classification approach from one-vs-one and one-vs-rest, and whether to standardize input features. Table 2 presents the selected parameters. We used scikit-learn[26][19] for the implementation of classifiers.

The 10-fold cross-validation was performed on the 400 URLs, excluding the development set. The development set was added to the training data for each cross-validation split. For evaluation, we computed the accuracy on the 400 URLs. We also measured precision, recall, and F1-score for each split by macro-averaging. Table 4 shows the results[27]. The results of the averaging and summation are the best and competitive with each other.

F1-score for each label in the baseline, averaging, and summation is shown in Table 5. Compared to the baseline, both compositional

[24]https://radimrehurek.com/gensim/
[25]The epoch was 10 or 20, and the window size was 5 or 10. Dimension sizes range from 100 to 1000 in increments of 100. The other hyperparameters were the default values, except that the pruning threshold for low-frequency words was set to 3.
[26]https://scikit-learn.org/stable/
[27]Precision, Recall, and F1-score are the averages over 10 splits.

Table 5: Results of extended compositional functions and GRU

Type	Composition function	Accuracy	F1-score macro average	tool	data	other
baseline	None	0.785 (314/400)	0.779	0.830	0.801	0.663
averaging-based	averaging	0.796 (319/400)	0.808	0.789	0.744	0.859
	weighted by entropy	0.790 (316/400)	0.796	0.806	0.728	0.821
	except top-level domain	0.788 (315/400)	0.799	0.793	0.729	0.842
summation-based	summation	0.800 (320/400)	0.808	0.809	0.725	0.857
	weighted by entropy	0.798 (319/400)	0.805	0.810	0.732	0.842
	except top-level domain	0.813 (325/400)	0.816	0.821	0.745	0.864
RNN-based	GRU	0.823 (329/400)	0.820	0.835	0.746	0.865

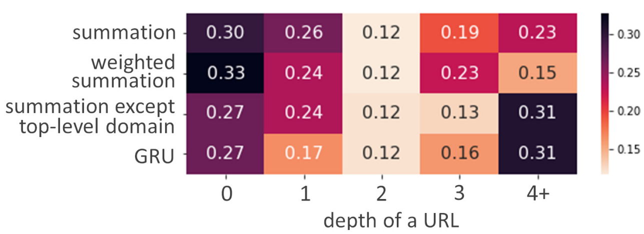

Figure 4: A heatmap of error rates for each depth of URLs

functions have the disadvantage of identifying the "tool" and "data" class. This result is the same as our previous study and it may be improved by less considering URL elements with a negative effect.

5.3 Experiment for Extended Functions

According to the results in Section 5.2, we used the averaging and summation as a basic compositional function and modified it. Therefore, the averaging and summation were extended to a weighted averaging and weighted summation by the entropy of URL elements, respectively. The entropy was computed based on frequency of each URL element in the experimental data. In addition, we also employed averaging and summation except for distributed representations of top-level domain names simply. For each function, the parameters are selected in the same way described in Section 5.2. Other experimental settings are also the same.

Table 5 shows the results. A part of extended compositional functions improved the discrimination performance of "tool" and "data" classes, which was a disadvantage of the proposed method. However, accuracy and macro-averaging F1-score of extended compositional functions are lower than that of basic compositional functions, excluding summation except top-level domain.

Fig. 4 shows error rates of each compositional function based on the summation for each depth of URLs. In Fig. 4, the higher the error rate was, the more intense the color was. As described in Section 4, summation tends to misclassify short URLs. Although the weighted summation had worse results, the performance of the summation except top-level domain was improved. This result indicates that there are frequent URL elements with useful information for the URL classification and simply excluding the top-level domains is effective for the summation. As a case study, the URL "http://www.imsdb.com/" misclassified into the "tool" class by the summation is correctly classified into the "data" class by excluding the distributed representation of "com" from the summation.

5.4 Experiment for RNN-based Function

As with the above Sections, we evaluate the classification method using GRU [2] as a compositional function. The model was implemented in PyTorch[28] [18]. In the training step, we used Adam as an optimizer and cross entropy loss. In addition, dropout was applied on inputs for GRU. The weights of the embedding layer were fixed by the pre-trained distributed representations of the URL elements described in Section 5.2. The best parameters were selected based on the classification performance on the development set[29].

After setting parameters, the method employing GRU was evaluated by 10-fold cross-validation with the same setting as in Section 5.2. The experimental results are shown in Table 5. GRU outperformed other compositional functions in the accuracy and macro-averaging F1-score as well as F1-score for "tool" and "data" class which basic compositional functions had difficulty identifying. Fig. 4 shows the error rates of GRU for each depth of URLs. GRU also improved the classification performance of short URLs compared to the basic compositional function.

6 CONCLUSION

This paper described methods for classifying URLs referring to research artifacts in scholarly papers and examined their classification performance. The methods use distributed representations obtained from citation contexts of the URL. Our approach regards each component of URLs as a word, and input features for a classifier are generated by synthesizing the distributed representation of each component using compositional functions. Experimental results showed the effectiveness of our compositional functions.

ACKNOWLEDGMENTS

This research was partially supported by the Grant-in-Aid for Scientific Research (B) (No. 21H03773) of JSPS.

[28] https://pytorch.org/

[29] We trained the model employing each combination of parameters for 300 epochs and selected the best epoch. The selected dimension size of hidden state in GRU was 50 from 25, 50, 100, and 200; the selected batch size was 32 from 4, 8, 16, 32, and 64; the selected learning rate was 1.0e-3 from 1.0e-3, 1.0e-4, and 1.0e-5; the selected dropout rate was 0.2 from 0.0, 0.2, 0.4, 0.6, and 0.8; the selected epoch was 98. The epoch, window size, and dimension size of selected distributed representations of URL elements were 20, 5, and 600, respectively.

REFERENCES

[1] Takeshi Abekawa and Akiko Aizawa. 2016. SideNoter: Scholarly Paper Browsing System based on PDF Restructuring and Text Annotation. In *Proceedings of the 26th International Conference on Computational Linguistics: System Demonstrations* (Osaka, Japan) *(COLING 2016)*. COLING 2016 Organizing Committee, 136–140. https://aclanthology.org/C16-2029

[2] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. (2014). arXiv:arXiv:1412.3555 https://arxiv.org/abs/1412.3555

[3] Caifan Du, Johanna Cohoon, Patrice Lopez, and James Howison. 2021. Softcite dataset: A Dataset of Software Mentions in Biomedical and Economic Research Publications. *Journal of the Association for Information Science and Technology* 72, 7 (July 2021), 870–884. https://doi.org/10.1002/asi.24454

[4] Caifan Du, James Howison, and Patrice Lopez. 2020. Softcite: Automatic Extraction of Software Mentions in Research Literature. In *Poster abstracts of the 1st Workshop on Natural Language Processing and Data Mining for Scientific Text Workshop (SciNLP)*. https://scinlp.org/history/2020/pdfs/softcite-automatic-extraction-of-software-mentions-in-research-literature.pdf

[5] European Commission. 2016. *European Cloud Initiative - Building a Competitive Data and Knowledge Economy in Europe*. https://ec.europa.eu/newsroom/dae/document.cfm?doc_id=15266

[6] Data Citation Synthesis Group. 2014. *Joint Declaration of Data Citation Principles*. FORCE11, San Diego, CA, USA. https://doi.org/10.25490/a97f-egyk

[7] Zellig S. Harris. 1954. Distributional Structure. *WORD* 10, 2-3 (1954), 146–162. https://doi.org/10.1080/00437956.1954.11659520

[8] James Howison and Julia Bullard. 2016. Software in the Scientific Literature: Problems with Seeing, Finding, and Using Software Mentioned in the Biology Literature. *Journal of the Association for Information Science and Technology* 67, 9 (Sept. 2016), 2137–2155. https://doi.org/10.1002/asi.23538

[9] Nancy Ide and James Pustejovsky. 2010. What Does Interoperability Mean , Anyway? Toward an Operational Definition of Interoperability for Language Technology. In *Proceedings of the Second International Conference on Global Interoperability for Language Resources (ICGL 2010)* (Hong Kong, China). https://www.cs.vassar.edu/~ide/papers/ICGL10.pdf

[10] Daisuke Ikeda, Kota Nagamizo, and Yuta Taniguchi. 2020. Automatic Identification of Dataset Names in Scholarly Articles of Various Disciplines. *International Journal of Institutional Research and Management* 4, 1 (May 2020), 17–30. hhttp://www.iaiai.org/journals/index.php/IJIRM/article/view/480

[11] Tomoki Ikoma and Shigeki Matsubara. 2020. Identification of Research Data References based on Citation Contexts. In *Proceedings of the 22nd International Conference on Asia-Pacific Digital Libraries (ICADL 2020)* (Kyoto, Japan) *(Lecture Notes in Computer Science)*. Springer, Cham, Switzerland, 149–156. https://doi.org/10.1007/978-3-030-64452-9_13

[12] Shunsuke Kozawa, Hitomi Tohyama, Kiyotaka Uchimoto, and Shigeki Matsubara. 2010. Collection of Usage Information for Language Resources from Academic Articles. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC 2010)* (Valletta, Malta). European Language Resources Association (ELRA), 1227–1232. https://lexitron.nectec.or.th/public/LREC-2010_Malta/pdf/746_Paper.pdf

[13] Frank Krüger and David Schindler. 2020. A Literature Review on Methods for the Extraction of Usage Statements of Software and Data. *Computing in Science Engineering* 22, 1 (2020), 26–38. https://doi.org/10.1109/MCSE.2019.2943847

[14] Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. *Building a Large Annotated Corpus of English: The Penn Treebank*. Technical Report. University of Pennsylvania, Philadelphia, PA, USA.

[15] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and Their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2* (Lake Tahoe, Nevada) *(NIPS'13)*. Curran Associates Inc., Red Hook, NY, USA, 3111–3119.

[16] Hidetsugu Nanba. 2018. Construction of an Academic Resource Repository. In *Proceedings of Toward Effective Support for Academic Information Search Workshop* (Hamilton, New Zealand). Kyushu University, Fukuoka, Japan, 8–14. https://doi.org/10.5109/2230668

[17] Monarch Parmar, Naman Jain, Pranjali Jain, P Jayakrishna Sahit, Soham Pachpande, Shruti Singh, and Mayank Singh. 2020. NLPExplorer: Exploring the Universe of NLP Papers. *Advances in Information Retrieval* 12036 (March 2020), 476–480. https://doi.org/10.1007/978-3-030-45442-5_61

[18] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32* (Vancouver, Canada) *(NIPS'19)*. Curran Associates Inc., Red Hook, NY, USA, 8024–8035. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[19] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (Oct. 2011), 2825–2830. https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf

[20] Animesh Prasad, Chenglei Si, and Min-Yen Kan. 2019. Dataset Mention Extraction and Classification. In *Proceedings of the Workshop on Extracting Structured Knowledge from Scientific Publications (ESSP)* (Minneapolis, Minnesota). Association for Computational Linguistics, Stroudsburg, PA, USA, 31–36. https://doi.org/10.18653/v1/W19-2604

[21] Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks* (Valletta, Malta). University of Malta, 46–50. http://is.muni.cz/publication/884893/en

[22] David Schindler, Benjamin Zapilko, and Frank Krüger. 2020. Investigating Software Usage in the Social Sciences: A knowledge Graph Approach. In *Proceedings of the 17th European Semantic Web Conference Semantic Web (The Semantic Web)* (Heraklion, Crete, Greece) *(Lecture Notes in Computer Science)*. Springer, Cham, Switzerland, 271–286. https://doi.org/10.1007/978-3-030-49461-2_16

[23] Ayush Singhal and Jaideep Srivastava. 2013. Data Extract: Mining Context from the Web for Dataset Extraction. *International Journal of Machine Learning and Computing* 3, 2 (April 2013), 219–223. https://doi.org/10.7763/IJMLC.2013.V3.306

[24] Arfon M Smith, Daniel S Katz, and Kyle E Niemeyer. 2016. Software Citation Principles. *PeerJ Computer Science* 2 (Sept. 2016), e86. https://doi.org/10.7717/peerj-cs.86

[25] ACL Anthology team. [n.d.]. ACL Anthology. https://aclanthology.org/

[26] Hitomi Tohyama, Shunsuke Kozawa, Kiyotaka Uchimoto, Shigeki Matsubara, and Hitoshi Isahara. 2008. Construction of an Infrastructure for Providing Users with Suitable Language Resources. In *Proceedings of the 22nd International Conference on Computational Linguistics: Companion Volume: Posters* (Manchester, UK) *(Coling 2008)*. Coling 2008 Organizing Committee, 119–122. https://aclanthology.org/C08-2030.pdf

[27] John Towns, Christine Kirkpatrick, Kenton McHenry, and Kandace Turner. 2016. *Towards a U.S. National Data Service - Inaugural Report*. The National Data Service, Urbana, IL, USA. http://www.nationaldataservice.org/docs/NDS_InauguralReport_Jan-Jun2016.pdf

[28] Andrew Treloar. 2009. Design and Implementation of the Australian National Data Service. *The International Journal of Digital Curation* 4, 1 (June 2009), 125–137. https://doi.org/10.2218/ijdc.v4i1.83

[29] Masaya Tsunokake and Shigeki Matsubara. 2020. Identification and Classification of Research Data Cited in Scholarly Papers. *IEEJ Transactions on Electronics, Information and Systems* 140, 12 (Dec. 2020), 1357–1364. https://doi.org/10.1541/ieejeiss.140.1357 (in Japanese).

[30] Linlin Wang, Kang Liu, Zhu Cao, Jun Zhao, and Gerard de Melo. 2015. Sentiment-Aspect Extraction based on Restricted Boltzmann Machines. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (Beijing, China). Association for Computational Linguistics, Stroudsburg, PA, USA, 616–625. https://doi.org/10.3115/v1/P15-1060

[31] Siluo Yang, Ruizhen Han, Jingda Ding, and Yanfei Song. 2012. The distribution of Web Citations. *Information processing & management* 48, 4 (July 2012), 779–790. https://doi.org/10.1016/j.ipm.2011.10.002

[32] Xuchen Yao and Benjamin Van Durme. 2014. Information Extraction over Structured Data: Question Answering with Freebase. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (Baltimore, Maryland). Association for Computational Linguistics, Stroudsburg, PA, USA, 956–966. https://doi.org/10.3115/v1/P14-1090