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

Masaya Tsunokake (Presenter), Shigeki Matsubara
Nagoya University, Japan

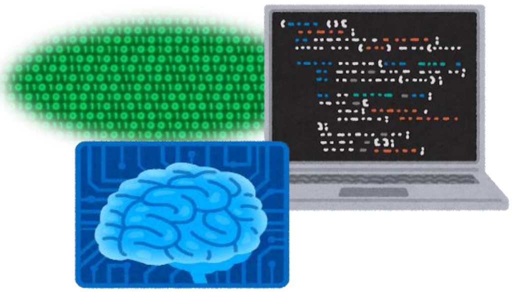
September 30, 2021

Outline

1. Introduction
2. Methodology
3. Experiments
4. Conclusion

Background

- **Research Artifacts**
  - digital objects created or used in the course of research work
    - software, toolkits, programs, observation/experimental data
  - increasingly cited in scholarly papers
    and gathering attention as one of the research results



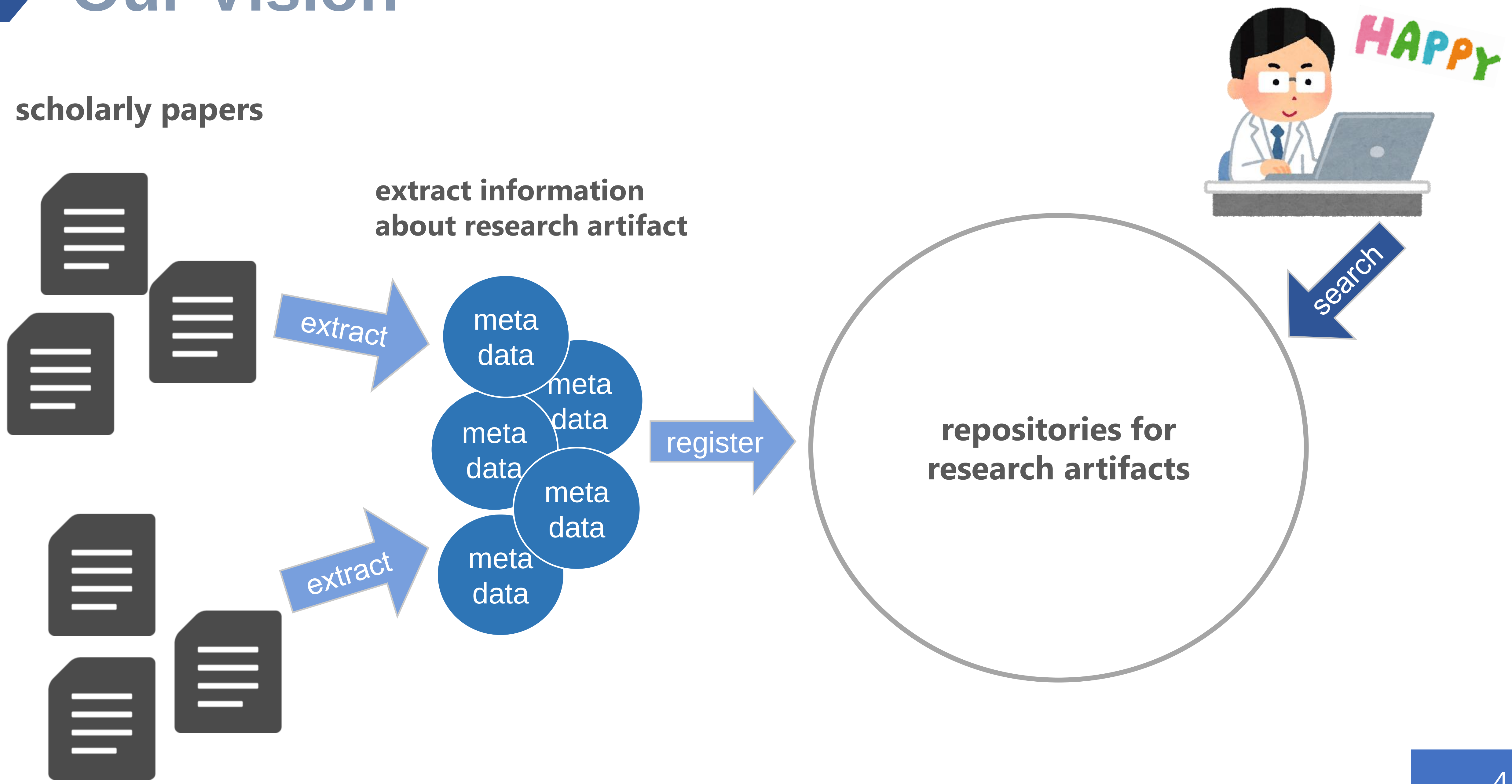
- **Repositories for research artifacts**
  - facilitate to share and utilize research artifacts
  - it is required to register metadata of research artifacts
  - metadata in the repositories make research artifacts more accessible and findable

An example metadata
(from Open Language Archive Community)

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 … will include these missing files.
identifier	DOI: 10.35111/gq1x-j780, https://catalog.ldc.upenn.edu/LDC99T42

Automatic generation of metadata
makes developing and expanding repositories more efficient

Our Vision


scholarly papers
extract information
about research artifact
extract
extract
meta data
meta data
meta data
meta data
meta data
register
repositories for
research artifacts
search
HAPPY

Related Work

- **Automatic generating metadata**
  - Kozawa et al. [1] have proposed a method for extracting usage information from scholarly papers
    - using resource names in SHACHI [2] as clue
    - target resources were limited to ones in repositories
  - Our targets include ones not stored in existing repositories

title	WordNet
creator	George A. Miller, Princeton University, etc.
publisher	The Global WordNet Association MIT Press
type	Text
identifier	http://wordnet.princeton.edu/
usage	NLP, word sense disambiguation, query expansion, cluster its senses

*An example from [1,2]

- **Identification of citations for research artifacts in scholarly papers**
  - Some method identifies dataset [3-6] or software [3,7-9] names in the body text

Example 2 *All statistical procedures were performed using IBM SPSS Statistics software version 22. Task accuracy and response times were analyzed using the SPSS software package (SPSS v17.0, Chicago, Illinois, USA).*

*quoted from [9]

  - On the other hand, there are other ways for citing them
    - listed in the reference section [10]

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

I. Dan Melamed. 2004. Statistical machine translation by parsing. In *Proceedings of the 42nd Meeting of*

*quoted from [18]

    - providing the corresponding URL

This approach uses a maximum entropy classifier[3] with $L1$ regularisation. In early experiments we found that the constituent-based approach per-

[3] http://scikit-learn.org/

*quoted from [19]

Contribution

1 We proposed the methods realizing the following tasks automatically

- identification of URLs citing research artifact in scholarly papers
- generating information about the type of the research artifacts

*quoted from [20]

> nologies, where appropriate. A video illustrating most of the user-facing features in action is currently available at https://www.youtube.com/watch?v=Efsl1ZMWFkE.

➢ not research artifact

*quoted from [21]

> requirement is not as strict as that in human languages.
> [2]In our experiments, we extract the antonym dictionary from the WordNet lexicon http://wordnet.princeton.edu/.

➢ used lexicon (research artifact)

generate

Metadata

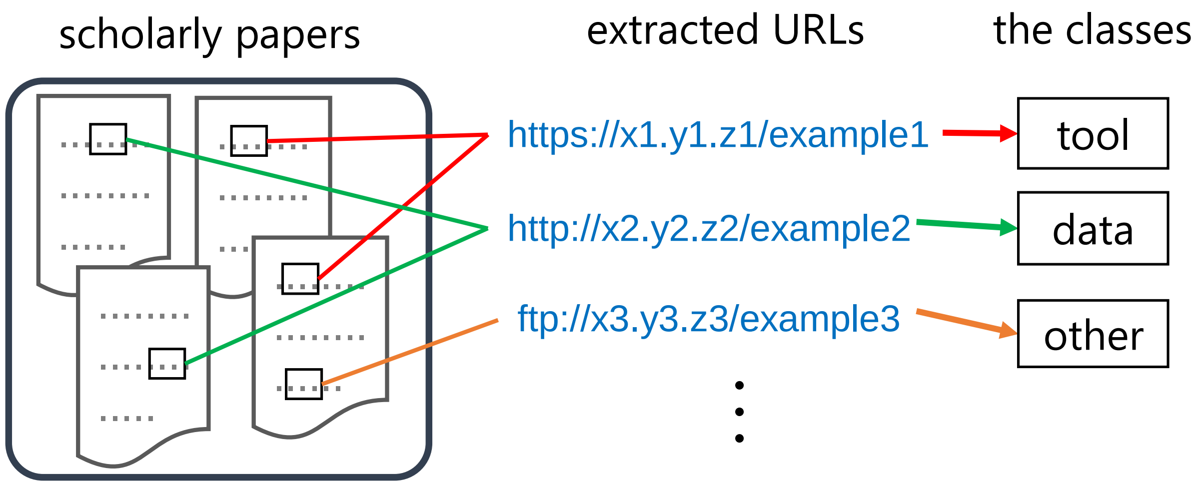
title	WordNet
creator	George A. Miller, Princeton University, etc.
publisher	The Global WordNet Association MIT Press
type	Text
identifier	http://wordnet.princeton.edu/
usage	NLP, word sense disambiguation, query expansion, cluster its senses

*An example from [1,2]

2 We evaluated the classification performances of the methods

Task Definition

- **URL classification**
  - Our goals
    - identify URLs citing research artifacts
    - detect the type of research artifacts.
  - Each URL in scholarly papers is classified based on the type of objects which the URL refers to



- **The definition of each class**

1. tool: programs, software, toolkit etc.

http://www.csie.ntu.edu.tw/~cjlin/libsvm (software)
https://www.tensorflow.org/ (framework)

2. data: observation/experimental data, data source, etc.

http://qwone.com/~jason/20Newsgroups/ (corpus)
http://babelnet.org (dictionary)

research artifacts

3. other: not research artifacts (e.g., publications, services)

http://is.muni.cz/publication/884893/en (publication)
http://www.apple.com/ios/siri (product)

Approach

- **the Citation Context of a URL:** the corresponding sentence in the body text

(referring to footnote or reference where the URL are provided)

URLs in footnotes

ter of multiple topic-related documents has gained much attention during the Document Understanding Conference[1] (DUC) and the Text Analysis Conference[2] (TAC) series. Despite a lot of re-

[1] http://duc.nist.gov/

[2] http://www.nist.gov/tac/

*quoted from [22]

URLs in bibliographic information

second model is a neural network trained using Keras (Chollet et al., 2015). The network passes the attribute vector through two dense layers, one for reducing the vector's dimension to 150 and the

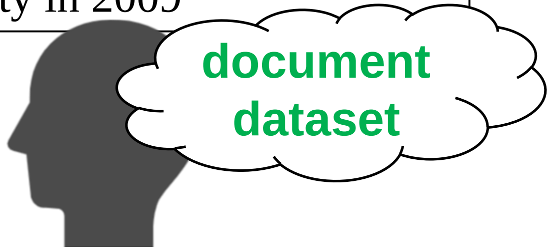
François Chollet et al. 2015. Keras. https://keras.io.

*quoted from [23]

- **intuitiveness**: reading citation contexts, we can know what resources a URL refers to
  - ➢ the system can classify an URL properly if it can captures all citation contexts of the URL

*quoted from [24]

The ClueWeb09 \footnote{http://lemurproject.org/clueweb09/} dataset is a collection of 1 billion webpages (5TB compressed in raw HTML) in 10 languages by Carnegie Mellon University in 2009



We obtain distributed representations of URLs and use them for input features in URL classification

Distributed Representations of URLs

- two approaches to obtain distributed representations of URLs **with different semantic units**

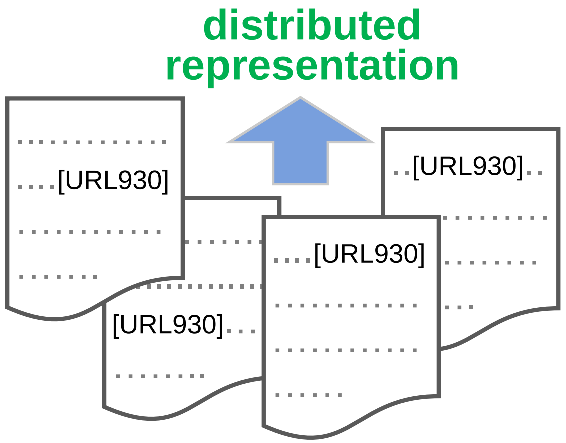
regarding each URL as a word [11]

➢ This approach converts each URL to the tag and obtains distributed representations of the tags

a tag (e.g., [URL930])

The Stanford POS Tagger (http://nlp.stanford.edu/software/tagger.shtml) is used to distinguish noun and adjective words from each other.

*quoted from [25]



regarding each component of URLs as a word (our original approach)

dataset about tweets?

➢ some components are considered to contain any meaning e.g., http://trec.nist.gov/data/tweets/

➢ This approach converts each component to the tag, obtains distributed representations of the tags, and **synthesizes them for obtaining overall representations of URLs**

The Stanford POS Tagger (~~http://~~ nlp . stanford . edu / software / tagger . shtml) is used to distinguish noun and adjective words from each other.

a tag (e.g., [COMP930])

*quoted from [25]

➢ we define components as domain, directory, filename, and extension
  - ❑ we call each component URL element

Methods for URL Classification

*quoted from [25]

1 if each URL is regarded as a word

The Stanford POS Tagger http://nlp.stanford.edu/software/tagger.shtml is used to distinguish noun and adjective words from each other.

The Stanford POS Tagger [URL2495] is used to distinguish noun and adjective words from each other.

Step 1
convert each URL to the tag

Step 2
obtain distributed representations of tags (URLs)

Step 3
classify URLs using the distributed representations as input features

2 if each URL element is regarded as a word (proposed approach)

The Stanford POS Tagger [COMP7070] [COMP9479] [COMP3891] [COMP9344] [COMP9680] [COMP9182] is used to distinguish noun and adjective words from each other.

*original sentence in [25]

Step 1
convert each URL element to the tag

Step 2
obtain distributed representations of tags (URL elements)

Step 3
create a feature of each URL by synthesizing distributed representations of the URL elements

Step 4
classify URLs using the features created in Step3

http:// $[e_1]$ $[e_2]$ … $[e_{n-1}]$ $[e_n]$

$f(v_{e_1}, \dots, v_{e_n})$

an input feature

Some Compositional Functions

1 Summation (in our previous study [12])

- add vectors element-wise
- overly affected by frequent URL element in scholarly papers

2 Summation **weighted by the entropy** of each URL element

- weaken the influence of frequent URL elements
- entropy is computed according to the frequency in papers

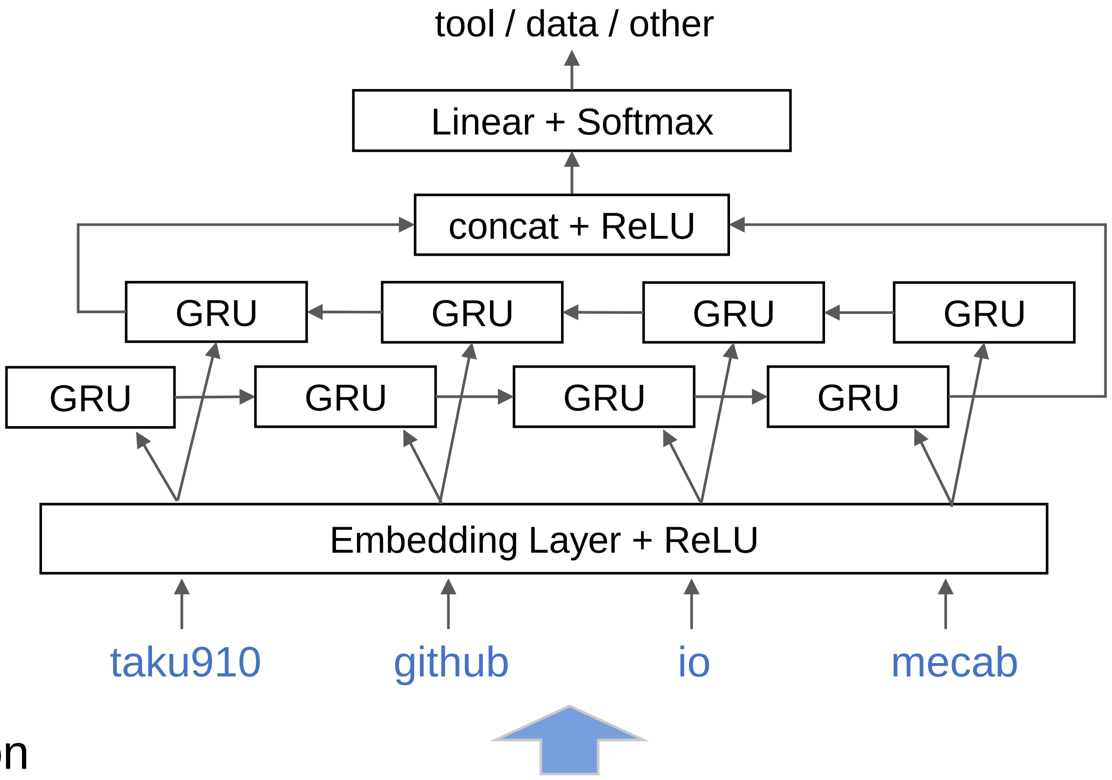
$$-\log_2 \frac{Count(w)}{\Sigma_{w'} Count(w')}$$

3 Summation **except top-level domains**

- top-level domains may be not useful for the classification
- exclude top-level domains from the computation

4 GRU [13]

- to get better weights for synthesizing
- incorporate order information

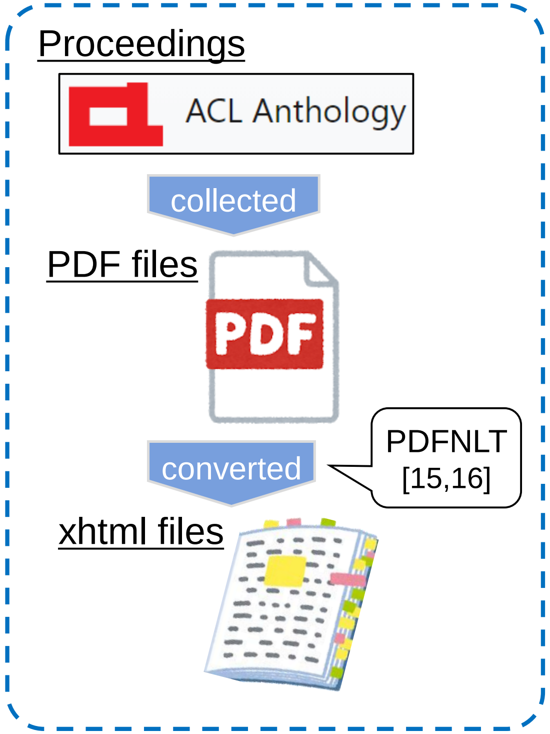


http://taku910.github.io/mecab/

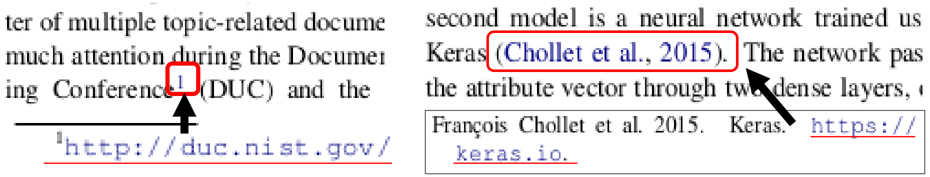
Experimental Setup

Purpose: to evaluate classification performances of the methods

Dataset: based on collected papers of the international conferences in the Natural Language Processing [14]



1. **Text dataset** for obtaining distributed representations
   - ➢ URLs were inserted into body texts


ter of multiple topic-related docume
much attention during the Documer
ing Conference[1] (DUC) and the

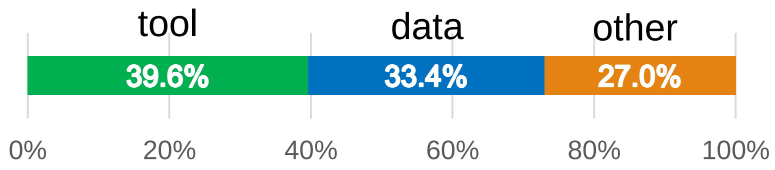
[1]http://duc.nist.gov/

second model is a neural network trained us
Keras (Chollet et al., 2015). The network pas
the attribute vector through two dense layers,

François Chollet et al. 2015. Keras. https://keras.io.

*quoted from [22,23]

2. **Annotated URLs** for evaluating classification performances
   - ➢ we labeled 500 URLs appearing frequently in the collected papers
   - ➢ 100 URLs are development set



Setup

- Obtaining distributed representations: word2vec [17]
- For each method, the following parameter are selected based on the performance for the development set:
  - parameters of word2vec (and GRU)
  - classification model
  - whether to standardize input features

Evaluation

- 10 fold cross-validation for 400 annotated URLs
- metric
  - macro-averaged F1-score
  - F1-score for each label

Experimental Result (1/2)

	Method	F1-score			
		macro-ave	tool	data	other
	baseline (regarding each URL as a word)	0.779	0.830	0.801	0.663
our approach	summation	0.808	0.809	0.725	0.857
	summation weighted by entropy	0.805	0.810	0.732	0.842
	summation except top-level domains	0.816	0.821	0.745	0.864
	GRU	0.820	0.835	0.746	0.865

- Obtaining distributed representations is effective for this task as a whole
- baseline vs our approach
  - our approach got better results on macro-averaged F1 consistently
  - our approach was not good at discriminating the “data”

Experimental Result (2/2)

	Method	F1-score			
		macro-ave	tool	data	other
	baseline (regarding each URL as a word)	0.779	0.830	0.801	0.663
our approach	summation	0.808	0.809	0.725	0.857
	summation weighted by entropy	0.805	0.810	0.732	0.842
	summation except top-level domains	0.816	0.821	0.745	0.864
	GRU	0.820	0.835	0.746	0.865

- Comparing Compositional functions
  - Compared to the summation, weighting by entropy got worse results on some metrics
  - Compared to the summation, excluding top-level domains got better results on all metrics
  - GRU got the best results

there are useful URL elements in frequent URL elements and we should exclude top-level domains only

Conclusion & Future Work

- **Conclusion**
  - We formulate the URL classification task to realize the following things:
    - identification of URLs citing research artifacts in scholarly papers
    - generating information about the type of the research artifacts
  - Using distributed representations of URLs was effective, and using those of URL elements got better results
  - When synthesizing distributed representations of URL elements, excluding top-level domains is effective

- **Future Work**
  - reveal why our approach is not good at discriminating the “data”
  - more complex functions (e.g., using Transfer Encoder)
  - multi-label classification
    - there are URLs distributing tools and datasets simultaneously

References (1/2)

[1] 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), 1227–1232.

[2] Hitomi Tohyama, Shunsuke Kozawa, Kiyotaka Uchimoto, Shigeki Matsubara, and Hitoshi Isahara. 2008. Construction of an Infrastructure for Providing Users with Suitable Language Resources. In Proc. the 22nd International Conference on Computational Linguistics: Companion Volume: Posters, 119–122.

[3] 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.

[4]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 (2020), 17–30.

[5] Animesh Prasad, Chenglei Si, and Min-Yen Kan. 2019. Dataset Mention Extraction and Classification. In Proc. the Workshop on Extracting Structured Knowledge from Scientific Publications (ESSP), 31–36.

[6] 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 (2013), 219–223.

[7] 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 (2021), 870–884.

[8] 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)

[9] David Schindler, Benjamin Zapilko, and Frank Krüger. 2020. Investigating Software Usage in the Social Sciences: A Knowledge Graph Approach. (2020). arXiv:arXiv:2003.10715

[10] Tomoki Ikoma and Shigeki Matsubara. 2020. Identification of Research Data References based on Citation Contexts. In Proc. the 22nd International Conference on Asia-Pacific Digital Libraries (ICADL 2020), 149–156.

[11] Hidetsugu Nanba. 2018. Construction of an Academic Resource Repository. In Proc. Toward Effective Support for Academic Information Search Workshop, 8–14.

[12] 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 (2020), 1357–1364. (in Japanese).

[13] 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

[14] ACL Anthology team. ACL Anthology. https://aclanthology.org/

[15] Takeshi Abekawa and Akiko Aizawa. 2016. SideNoter: Scholarly Paper Browsing System based on PDF Restructuring and Text Annotation. In Proc. the 26th International Conference on Computational Linguistics: System Demonstrations (COLING 2016). 136–140.

[16] Aizawa Laboratory. PDFNLT 1.0. https://github.com/KMCS-NII/PDFNLT-1.0

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and Their Compositionality. In Proc. the 26th International Conference on Neural Information Processing Systems - Volume 2 (NIPS’13), 3111–3119.

References (2/2)

[18] Lane Schwartz, Chris Callison-Burch, William Schuler, acd Stephen Wu. 2011. Incremental Syntactic Language Models for Phrase-based Translation. In Proc. the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 620–631.

[19] Silvia Pareti, Tim O'Keefe, Ioannis Konstas, James R. Curran, and Irena Koprinska. 2013. Automatically Detecting and Attributing Indirect Quotations. In Proc. the 2013 Conference on Empirical Methods in Natural Language Processing, 989–999.

[20] Nitin Madnani, Beata Beigman Klebanov, Anastassia Loukina, Binod Gyawali, Patrick Lange, John Sabatini, and Michael Flor. 2019. My Turn To Read: An Interleaved E-book Reading Tool for Developing and Struggling Readers. In Proc. the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 141–146.

[21] Rui Xia, Cheng Wang, Xin-Yu Dai, and Tao Li. 2015. Co-training for Semi-supervised Sentiment Classification Based on Dual-view Bags-of-words Representation. In Proc. the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 1054–1063.

[22] Avinesh P.V.S and Christian M. Meyer. 2017. Joint Optimization of User-desired Content in Multi-document Summaries by Learning from User Feedback. In Proc. the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 1353–1363.

[23] Zixuan Ke, Hrishikesh Inamdar, Hui Lin, and Vincent Ng. 2019. Give Me More Feedback II: Annotating Thesis Strength and Related Attributes in Student Essays. In Proc. the 57th Annual Meeting of the Association for Computational Linguistics, 3994–4004.

[24] Xuchen Yao and Benjamin Van Durme. 2014. Information Extraction over Structured Data: Question Answering with Freebase. In Proc. the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 956–966.

[25] Linlin Wang, Kang Liu, Zhu Cao, Jun Zhao, and Gerard de Melo. 2015. Sentiment-Aspect Extraction based on Restricted Boltzmann Machines. In Proc. the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 616–625.

Thank you for listening !

Masaya Tsunokake (Presenter), Shigeki Matsubara
Nagoya University, Japan