MioGatto: A Math Identifier-oriented Grounding Annotation Tool

Takuto Asakura[1], Yusuke Miyao[1], Akiko Aizawa[1,2] and Michael Kohlhase[3]

[1]*The University of Tokyo, Tokyo, Japan*
[2]*National Institute of Informatics, Tokyo, Japan*
[3]*FAU Erlangen-Nürnberg, Erlangen, Germany*

Abstract

We present a new annotation tool, called MioGatto, to efficiently build large corpora for grounding math formulae. While in documents in science, technology, engineering, and mathematics, math identifiers can be used in multiple meanings in a single document, corpora with annotated coreference relations between identifiers are crucial for the grounding task. Using MioGatto, annotators can produce a list of math concepts for each document, associate one of the math concepts with each occurrence of math identifiers, and annotate the text span that is the source for grounding. In general, manual annotation of coreference relations is a very tough task, but this tool is specialized for building grounding corpora and can annotate them more efficiently than existing general-purpose annotation tools. The tool can be obtained from `https://github.com/wtsnjp/MioGatto`.

1. Introduction

Recently, the authors have proposed a mathematical language processing (MLP) task called grounding of formulae [1], which has both aspects of math description alignment [2] and coreference analysis. In order to create a resource that can be used as training and evaluation data for this grounding task, we need an annotation tool that can annotate (1) a description label to each math identifier in formulae, and retain (2) information about coreference relations between math identifiers. In addition, (3) spans of text that serve as sources of grounding, i.e., natural language phrases that can be regarded as mathematical definitions and declarations, need to be annotated (Figure 1). Not surprisingly, there is no existing tool that can efficiently perform all such annotations simultaneously.

In order to efficiently create linguistic resources for the grounding tasks, we developed a novel annotation tool that has all the necessary functions. The tool is named the Math Identifier-oriented Grounding Annotation Tool (MioGatto). The core functionality of MioGatto is to annotate each math identifier with a math concept and to annotate sources of grounding, where a math concept is a description of an identifier with some extra information such as arity and math type. It has a web-based graphical user interface (GUI) that allows users to efficiently annotate math concepts and source of grounding with visual and intuitive operations (Figure 2; details will be presented in Section 3). This GUI can be seen as a tool for visualizing the annotated data, not just for information assignment. For example, in MioGatto, if an annotator mouses

13th MathUI Workshop 2021, July 26–31, 2021, Timisoara, Romania (online)

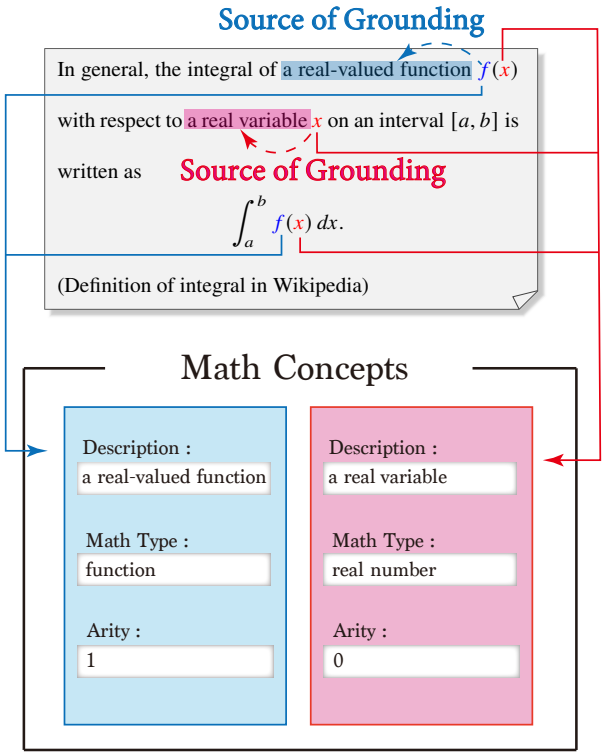

Figure 1: Math concepts and sources of grounding. The example sentence is taken from Wikipedia[1].

over a math identifier that is annotated with a math concept, the corresponding description pops up. In order to visualize a coreference relation, math identifiers referring to the same math concept are decorated with the same color. Sources of grounding are also highlighted in the same color as the corresponding math concept. It has been shown that this kind of complementary information to formulae can indeed help a reader understand the content of scientific papers [3]. During the development of the annotation tool, we will continuously discuss how to visualize annotated data. The findings will guide the development of a GUI that will be useful in supporting readers of the science, technology, engineering, and mathematics (STEM) literature in the future.

Linguistic resources containing math formulae annotated with rich information are the cornerstone for developing various MLP techniques, such as mathematical objects of interest (MOI) analysis [4], math information retrieval (MathIR), and formula search. In natural language processing (NLP), language resources that contain plenty of texts annotated with rich additional information are important for learning and evaluating statistical models. Specifically, information such as parts-of-speech, dependency structures, and coreference relations is required at the word, phrase, and sentence levels. Similar information including part-of-math tags [5],

[1] https://en.wikipedia.org/wiki/Integral

math types [6], dependency structures, and coreference relations at symbol and formula levels is useful for MLP. A key contribution of our tool is to build such a large corpus that will help advance the MLP technology.

Generally speaking, a large amount of time and economic cost is required to add such annotation information manually. Especially when annotating formulae, one needs to ask experts from various STEM disciplines who do not necessarily have a linguistic background. Therefore, in order to build a large amount of annotated formulae with high accuracy, it is crucial to have an efficient annotation tool that is easy enough to be used. Within the field of NLP, annotation tools have been developed that are easy and efficient to use for a range of purposes and tasks, e.g., brat [7] and WebAnno [8]. Some of these tools claim to be general-purpose, and if they are general enough, they can be applied to annotate math formulae as well. There are also a small number of tools that are specialized for annotating STEM documents, e.g., KAT [9]. Nevertheless, compared to the wide variety of annotation tools for creating corpora for NLP, the choice of tools that can add linguistic annotations to math formulae is very limited. Especially, it is impractical to annotate coreference relations on a large scale without using a tool with dedicated features. MioGatto is a tool that addresses this problem.

2. Related Work

There is commercial software available for the general public, which provide basic annotation functionality, such as adding free-text annotations or highlighting text spans in documents, e.g., Adobe Acrobat[2] for PDF documents and hypothes.is[3] for web pages. However, they are not designed to create linguistic resources, and do not support the annotations necessary for language processing, such as dependencies and coreference relations between words. Therefore, many specialized tools have been developed to efficiently create annotated corpora [11, 12, 7, 13, 14, 15], and these tools are preferred to be used to build language resources. These tools typically support the annotation of two main types of information: text spans and relations between regular text spans. The ability to label text spans with part-of-speech tags and whether they are proper expressions or not is also common. Most of these annotation tools accept XML or other text data as input, but there are also tools, e.g., PDFAnno [14], that can annotate PDFs directly. Such a tool is useful for scientific papers, where it is difficult to distinguish tables, equations, footnotes, etc., from the main text.

Since building an annotated corpus is a time-consuming and laborious task, the efficiency is important, and some tools are designed to build large corpora at a high speed. For example, WebAnno [13] has extensive functionalities for project and user management, which makes it easy for multiple annotators to work together. There is also research that attempts to make the annotation more efficient by focusing on specific types of annotations. SACR [15] is a specialized tool for annotating coreference relations, which compares multiple annotation UIs and adopts the most efficient one in its design.

In general, annotation tools for creating linguistic resources naturally annotate words, phrases, and sentences, and do not have special support for math formulae. In some cases, features

[2] https://acrobat.adobe.com
[3] https://web.hypothes.is

for text-span annotation can be used to annotate math formulae. However, structures such as superscripts, subscripts, and operators in formulae do not exist in natural language, and the functions that assume such structures can be used for more efficient annotation. A small number of tools were developed that specialize in annotating STEM literature with math formulae. KAT [9] is a web-based annotation tool that is specialized for annotating STEM documents. This tool allows annotators to effectively add attributes for the OMDoc format [16] to the STEM documents. Its annotation output is expressed in RDF, and thus can be used in a universal way. AnnoMathTeX [17, 18] is another annotation tool that specializes in annotating math identifiers. The tool takes either a Wikitext or a LaTeX document as input

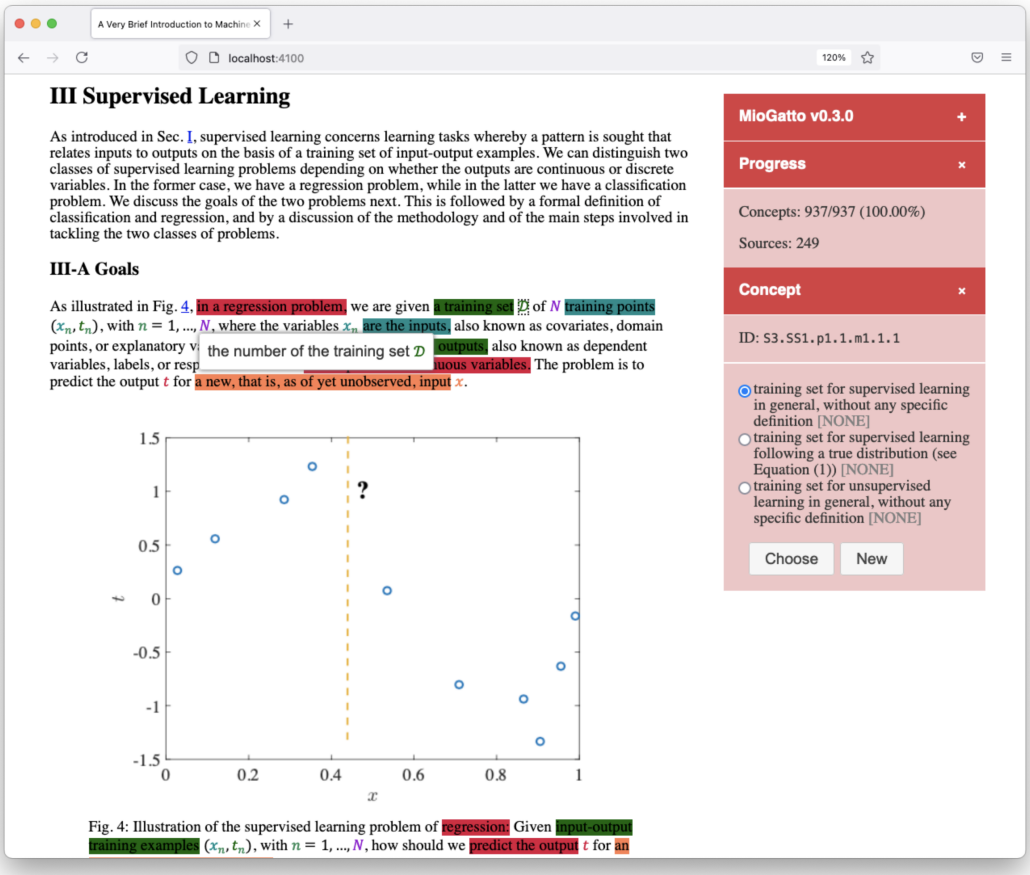

Figure 2: The screen of MioGatto. This is a captured image of annotating an arXiv paper in the field of machine learning [10]. The basic annotation operations, such as selecting the math concept that each identifier refers to, are performed in a sidebar on the right. Each occurrence of an identifier is colored according to the annotated math concept. In other words, identifiers with the same color have a coreference relation. Grounding sources are also highlighted in colors that correspond to math concepts associated with them. When the mouse is over an annotated occurrence, the tooltip with the description of the corresponding math concept is shown.

and annotates formulae in LaTeX syntax rather than the rendered result. Notably, the tool has the ability to recommend candidate math concepts to annotate for each identifier based on four resources (arXiv, Wikipedia, Wikidata, and the surrounding text). MioGatto treats all annotations as local annotations, assuming that the meanings of math identifiers change frequently within a document, while AnnoMathTeX treats them as document-global annotations unless a 'local' option is specified. This allows efficient annotation of documents whose meanings of identifiers do not change frequently. All these tools have been developed with different tasks and philosophies in mind. Since manual annotation is an arduous task, it is desirable to use a dedicated tool for efficient corpus building, and thus we needed one that is aimed at the formulae grounding task.

3. MioGatto: the Annotation Tool

Math Identifier-oriented Grounding Annotation Tool (Figure 2) is a tool specialized for annotating math identifiers. It is open source software and distributed under the terms of the MIT license. It was developed to construct a dataset for solving the grounding task for math formulae. It also has the ability to annotate text spans to aid in automating grounding of formulae, but unlike KAT, it does not aim to annotate the structure of all elements in a STEM document.

The goal of the grounding task is to disambiguate the meaning of math identifiers in a document, as an identifier can have multiple meanings in a document and its scope is ambiguous [1]. The existence of ambiguity in the meaning of an identifier in a document means that two occurrences of an identifier in a document may or may not refer to the same math concept. Therefore, in the training and evaluation data for the grounding task, the coreference relation of all occurrences of identifiers must be made explicit. An annotation tool such as those that give a free-text description to each identifier is not appropriate for this purpose. Since a math concept can be represented by many different natural language texts, extracting coreference relations from such annotated data would require solving the difficult task of determining whether two descriptions represent the same math concept. It is not efficient for a human annotator to carefully annotate every occurrence of an identifier referring to the same math concept with exactly the same description.

Instead of giving a free-text description directly to each occurrence of an identifier, MioGatto associates each occurrence with an item in a pre-defined list of math concepts. Therefore, it is easy to see that occurrences of identifiers associated with the same math concept have a coreference relationship. We call the pre-defined list of math concepts the *math concept dictionary*. The dictionary is not a global ontology, but a document-specific one. Apart from the description of a math concept, each dictionary item can have several additional attributes, such as arity, math type, and notation usage patterns (i.e., information whether the identifier is used with other tokens such as superscripts or independently). Moreover, annotators can register text spans that are useful in identifying math concepts, which are referred to by math identifiers as sources of grounding. Most sources of grounding collected in this way correspond to definitions and declarations. In the future, we intend to use the sources of grounding to automatically extract them and dynamically generate a math concept dictionary for each document.

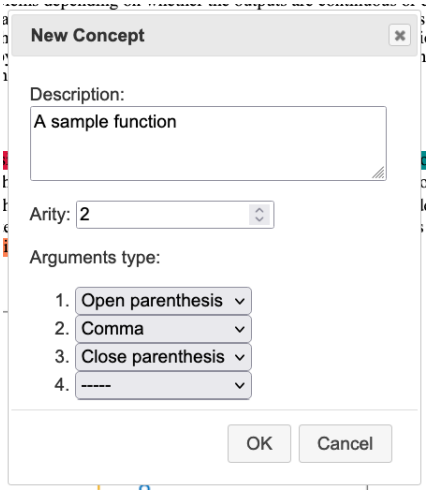

Figure 3: A dialog to add a math concept.

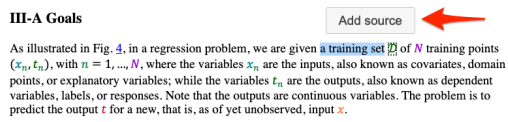

Figure 4: The button to add a source of grounding.

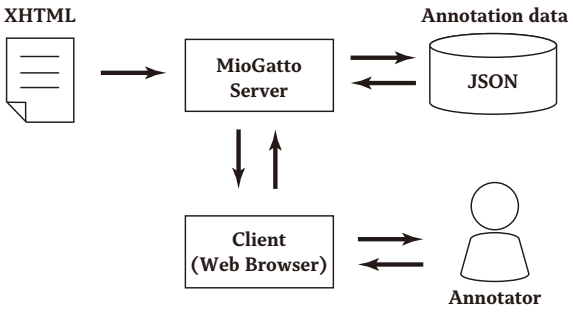

Figure 5: The architecture of MioGatto.

3.1. User Interface and Annotation Procedure

Any annotation supported by MioGatto can be done by performing intuitive operations on a web browser, without having any expertise in constructing language resources. Figure 2 shows a basic screen on MioGatto. The left side is the body of the academic paper to be annotated, and the right side is the sidebar for the MioGatto operation. Annotators can select the identifiers and text spans they want to annotate, while reading the article shown on the left. Annotators can then add the necessary information to the document by manipulating the boxes in the sidebar and the dialogs that appear as appropriate.

The annotator must first select one occurrence of a math identifier for each annotation. On the occurrence that is selected, a pointer is shown in a document shown on the left side, and the "Concept" box on the right sidebar shows the information and buttons necessary to annotate the occurrence (in Figure 2, the occurrence of identifier $\mathcal{D}$ in the first line of Subsection III-A is selected). In this state, one can either select the concept to which the occurrence refers from the list of math concepts displayed in the "Concept" box (if any in the dictionary), or create a new math concept in the dictionary. When an annotator chooses to add a new math concept, the dialog pops up with a web form to enter the required information (Figure 3). In this form, the annotator will be asked to input information such as a free-text description and arity. Once an occurrence of a math identifier is annotated with a math concept, the concept's information, most notably the description, is displayed as a tooltip when an annotator mouse over an occurrence. In addition, the annotated occurrences are colored according to the corresponding math concept, so that the coreference relation is visible to annotators.

MioGatto can also be used to annotate sources of grounding, text spans that are the basis for the grounding. After selecting a math identifier that a math concept has already been annotated, dragging the appropriate text span to select it will display a button for adding the source (Figure 4). If the annotator clicked the button, the text span will be associated with the math concept corresponding to the selected occurrence of the identifier. Most of the sources

annotated in this way correspond to definitions or declarations in mathematical terms. Within papers in mathematics, the sources are often fixed phrases, such as "Let x be *something*", whereas in the engineering literature, they are often simply apposition nouns. In the latter case, there is no one-to-one relation between math concepts and the sources of grounding, since the sources corresponding to the same math concept appear many times within the same document. Hence the annotation scheme allows annotators to annotate an arbitrary number of sources for a math concept. Similar to the occurrences of math identifiers, the text spans annotated as sources of grounding are highlighted in the color corresponding to the math concept associated with them.

3.2. Architecture and Implementation

MioGatto is a web-based annotation tool, and its entire implementation makes use of a variety of web standard technologies. The input of MioGatto is XHTML documents converted by LaTeXML [19] from LaTeX sources. To be more specific, the input must have the same additional information as the XHTML contained in the arXMLiv dataset [20, 21, 22]. In XHTML generated in this way, math formulae are written in MathML format [23], which stores more structural information inside formulae than mere image data. In order to make use of MathML, a browser that supports MathML rendering needs to be used. Firefox[4] supports MathML among the major browsers today. Each math identifier, i.e., `<mi>` element, has a unique ID in the input XHTML, and a MioGatto annotation is associated with the ID of the math identifier. The annotation information is saved and output in JSON format. For the detailed specification of the output JSON, please refer to the bundled documentation[5].

MioGatto employs a simple server-client model in terms of implementation. Figure 5 shows the architecture of MioGatto in brief. The server, implemented in Python, loads the input XHTML and stores the annotation data in JSON format. It also performs a simple preprocessing to display the input and validate the annotation data before storing them. In contrast, the client, implemented in TypeScript, is only responsible for handling UI. Such an architecture naturally scales up in the future, where a single central server will manage the annotation data and many annotators will annotate concurrently via the Internet.

4. Conclusion & Future Work

In this paper, we presented MioGatto, a dedicated tool for building datasets for the grounding task. For each occurrence of an identifier, a math concept can be annotated, and the textual spans of the sources of grounding can also be associated with the math concept. Compared to other tools dedicated to MLP, MioGatto is unique in its ability to associate math concepts with additional information such as arity and sources of grounding. This tool is also distinctive in that it assumes that the meaning of an identifier switches frequently; we have used an early version of this tool to annotate 937 math identifier occurrences for a scientific paper, and have found that semantic transitions do indeed occur frequently [1]. All the existing data we built are available from the SIGMathLing repository[6].

[4] https://www.mozilla.org/firefox/
[5] https://github.com/wtsnjp/MioGatto/wiki
[6] https://sigmathling.kwarc.info/resources/grounding-dataset/

We are now using MioGatto to annotate STEM documents with annotators from a range of disciplines, including information science, algebra, logic, and physics. Once we have a sufficient amount of identifier annotations with clear coreference relations, we begin to automate the process of the grounding task. We will continue to improve MioGatto so that it can be used by experts across a variety of domains to build the annotated dataset more efficiently. MioGatto will have a review mode so that discrepancies between annotators can be clearly shown with the GUI. It would also be valuable if comments can be added to the annotations, so that multiple annotators can discuss which annotation is better. Output format standardization should also be considered. We also obtain specific feedback from the annotators and verify that the annotated information helps the reader to read academic papers. In addition, we will explore how to display such additional information more effectively.

Acknowledgements

This work has been supported by JST, ACT-X Grant Number JPMJAX2002, Japan. We appreciate Mr. Taiga Ishii for his bug reports and feedbacks on the tool. We would like to thank Mr. André Greiner-Petter and Mr. Jan Frederik Schaefer for fruitful discussions.

References

[1] T. Asakura, A. Greiner-Petter, A. Aizawa, Y. Miyao, Towards grounding of formulae, in: Proceedings of the First Workshop on Scholarly Document Processing, 2020, pp. 138–147. doi:10.18653/v1/2020.sdp-1.16.

[2] M. Alexeeva, R. Sharp, M. A. Valenzuela-Escárcega, J. Kadowaki, A. Pyarelal, C. Morrison, MathAlign: Linking formula identifiers to their contextual natural language descriptions, in: Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020), 2020, pp. 2204–2212. URL: https://aclanthology.org/2020.lrec-1.269.

[3] A. Head, K. Lo, D. Kang, R. Fok, S. Skjonsberg, D. S. Weld, M. A. Hearst, Augmenting Scientific Papers with Just-in-Time, Position-Sensitive Definitions of Terms and Symbols, in: Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI 2021), 2021, pp. 1–18. doi:10.1145/3411764.3445648.

[4] A. Greiner-Petter, M. Schubotz, F. Müller, C. Breitinger, H. S. Cohl, A. Aizawa, B. Gipp, Discovering mathematical objects of interest—a study of mathematical notations, in: Proceedings of The Web Conference 2020 (WWW 2020), 2020, pp. 1445–1456. doi:10.1145/3366423.3380218.

[5] A. Youssef, Part-of-math tagging and applications, in: Proceedings of 10th International Conference on Intelligent Computer Mathematics (CICM 2017), 2017. doi:10.1007/978-3-319-62075-6_25.

[6] Y. Stathopoulos, S. Baker, M. Rei, S. Teufel, Variable typing: Assigning meaning to variables in mathematical text, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2018), 2018, pp. 303–312. doi:10.17863/CAM.30845.

[7] P. Stenetorp, S. Pyysalo, G. Topić, T. Ohta, S. Ananiadou, J. Tsujii, brat: a web-based tool for NLP-assisted text annotation, in: Proceedings of the Demonstrations at the 13th

Conference of the European Chapter of the Association for Computational Linguistics (EACL 2012), 2012, pp. 102–107. URL: https://aclanthology.org/E12-2021.

[8] R. Eckart de Castilho, É. Mújdricza-Maydt, S. M. Yimam, S. Hartmann, I. Gurevych, A. Frank, C. Biemann, A web-based tool for the integrated annotation of semantic and syntactic structures, in: Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities (LT4DH), 2016, pp. 76–84. URL: https://www.aclweb.org/anthology/W16-4011.

[9] D. Ginev, S. Lal, M. Kohlhase, T. Wiesing, KAT: an annotation tool for STEM documents, in: Mathematical user interfaces workshop at CICM, 2015. URL: http://www.cermat.org/events/MathUI/15/proceedings/Lal-Kohlhase-Ginev_KAT_annotations_MathUI_15.pdf.

[10] O. Simeone, A very brief introduction to machine learning with applications to communication systems, IEEE Transactions on Cognitive Communications and Networking (2018). doi:10.1109/TCCN.2018.2881442.

[11] C. Müller, M. Strube, Multi-level annotation of linguistic data with MMAX2, in: Corpus Technology and Language Pedagogy: New Resources, New Tools, New Methods, 2006, pp. 197–214.

[12] K. Bontcheva, H. Cunningham, I. Roberts, V. Tablan, et al., Web-based collaborative corpus annotation: Requirements and a framework implementation, New Challenges for NLP Frameworks (2010) 20–27.

[13] S. M. Yimam, I. Gurevych, R. E. de Castilho, C. Biemann, WebAnno: A flexible, web-based and visually supported system for distributed annotations, in: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 2013, pp. 1–6.

[14] H. Shindo, Y. Munesada, Y. Matsumoto, PDFAnno: a web-based linguistic annotation tool for pdf documents, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), 2018, pp. 1082–1086. URL: https://aclanthology.org/L18-1175.

[15] B. Oberle, SACR: A drag-and-drop based tool for coreference annotation, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), 2018. URL: https://aclanthology.org/L18-1059.

[16] M. Kohlhase, OMDoc—An Open Markup Format for Mathematical Documents [version 1.2], 2006.

[17] P. Scharpf, I. Mackerracher, M. Schubotz, J. Beel, C. Breitinger, B. Gipp, Annomathtex—a formula identifier annotation recommender system for stem documents, in: Proceedings of the 13th ACM Conference on Recommender Systems, 2019, pp. 532–533. doi:10.1145/3298689.3347042.

[18] P. Scharpf, M. Schubotz, B. Gipp, Fast linking of mathematical wikidata entities in wikipedia articles using annotation recommendation, in: Companion Proceedings of the Web Conference 2021, 2021, pp. 602–609. doi:10.1145/3442442.3452348.

[19] B. Miller, LaTeXML *The Manual*—A LaTeX to XML/HTML/MathML Converter, Version 0.8.3, 2018. URL: https://dlmf.nist.gov/LaTeXML/.

[20] H. Stamerjohanns, M. Kohlhase, D. Ginev, C. David, B. Miller, Transforming large collections of scientific publications to xml, Mathematics in Computer Science (2010).

doi:10.1007/s11786-010-0024-7.

[21] D. Ginev, H. Stamerjohanns, B. R. Miller, M. Kohlhase, The LaTeXML daemon: Editable math on the collaborative web, in: Intelligent Computer Mathematics, 2011. doi:10.1007/978-3-642-22673-1_25.

[22] D. Ginev, arxmliv:08.2018 dataset, an html5 conversion of arxiv.org, 2018. URL: https://sigmathling.kwarc.info/resources/arxmliv/, sIGMathLing.

[23] R. Ausbrooks, S. Buswell, D. Carlisle, G. Chavchanidze, S. Dalmas, S. Devitt, A. Diaz, S. Dooley, R. Hunter, P. Ion, M. Kohlhase, Mathematical Markup Language (MathML) 3.0 Specification, 2014. URL: https://www.w3.org/TR/MathML3/.