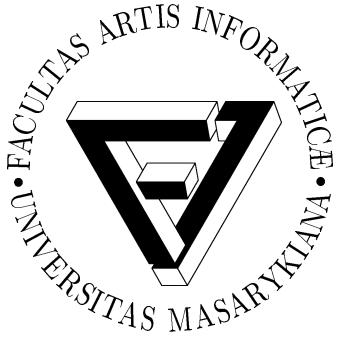
DML 2011
Towards a Digital Mathematics Library

muni
PRESS



http://www.fi.muni.cz/~sojka/dml-2011.html

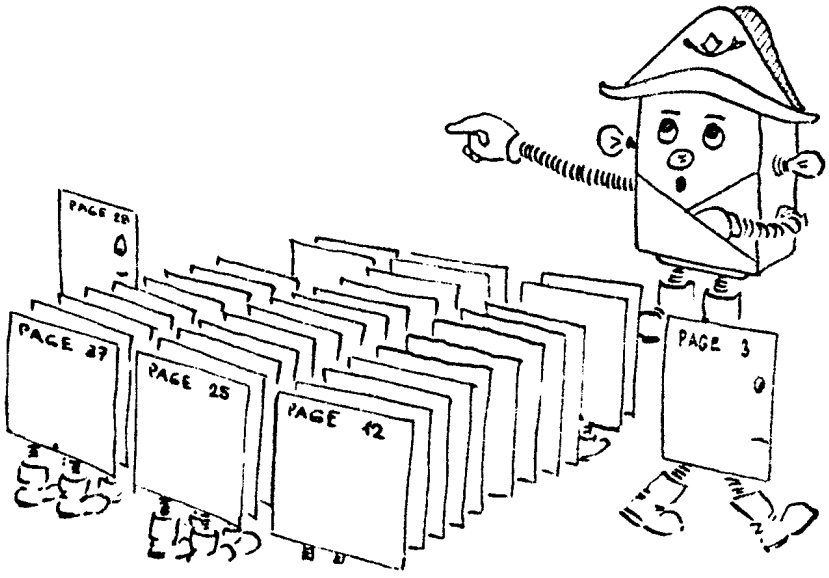
Petr Sojka, Thierry Bouche (editors)

DML 2011

Towards a Digital Mathematics Library

Bertinoro, Italy
July 20–21, 2011

Proceedings



Masaryk University, Brno, 2011

Proceedings Editors

Petr Sojka
Faculty of Informatics, Masaryk University
Department of Computer Graphics and Design
Botanická 68a, CZ-602 00 Brno, Czech Republic
Email: sojka@fi.muni.cz

Thierry Bouche
Cellule Mathdoc (UMS 5638), Université Joseph-Fourier (Grenoble 1)
B.P. 74, 384 02 Saint-Martin d'Hères, France
Email: thierry.bouche@ujf-grenoble.fr

CATALOGUING-IN-PUBLICATION – NATIONAL LIBRARY OF THE CZECH REPUBLIC

DML 2011 (Bertinoro, Italy)

DML 2011 : Towards a Digital Mathematics Library : Bertinoro, Italy, July 20-21, 2011 : proceedings / Petr Sojka, Thierry Bouche (editors). - 1st ed. - Brno : Masaryk University, 2011. - VIII+111 p.

ISBN 978-80-210-5542-1

025:004.08 * 930.25:004.08 * 51:81'42'373.46 * 002.2:004
- digital libraries
- digital archives
- digitization of documents
- mathematical texts
- proceedings of conferences
- digitální knihovny
- digitální repozitáře
- matematické texty
- digitalizace dokumentů
- sborníky konferencí

020 - Library and information sciences [12]

02 - Knihovnictví. Informatika [12]

ISBN 978-80-210-5542-1

Organization

DML 2011 was organized by Faculty of Informatics, Masaryk University, Brno, Czech Republic with the help of University of Bologna, Italy. Web page of the workshop is `http://www.fi.muni.cz/~sojka/dml-2011.html`.

Program Committee

José Borbinha (Technical University of Lisbon, IST, PT)
Thierry Bouche (University Grenoble I, Cellule Mathdoc, FR) [co-chair]
Michael Doob (University of Manitoba, Winnipeg, CA)
Thomas Fischer (Goettingen University, Digitization Center, DE)
Yannis Haralambous (Télécom Bretagne, FR)
Václav Hlaváč (Czech Technical University, Faculty of Engineering, Prague, CZ)
Michael Kohlhase (Jacobs University Bremen, DE)
Janka Chlebíková (Comenius University, MFF, Bratislava, SK)
Enrique Maciás-Virgós (University of Santiago de Compostela, ES)
Jiří Rákosník (Academy of Sciences, Institute of Mathematics, Prague, CZ)
Eugénio Rocha (University of Aveiro, Dept. of Mathematics, PT)
David Ruddy (Cornell University, Library, US)
Petr Sojka (Masaryk University, Faculty of Informatics, Brno, CZ) [co-chair]
Volker Sorge (University of Birmingham, UK)
Masakazu Suzuki (Kyushu University, Faculty of Mathematics, JP)

Additional Referees

Josef Baker, Mark Lee, Martin Líška, Jorge Machado, Zuzana Nevěřilová, Michal Růžička, Alan Sexton

Organizing Committee

Andrea Asperti *(local organization)*, Michal Růžička *(data import into DML-CZ)*, and Petr Sojka *(chair, Proceedings)*

Sponsors and Support

The DML workshop and preparation of the Proceedings was partly supported by the Masaryk University, Brno, and by EU project EuDML, project number 250503.

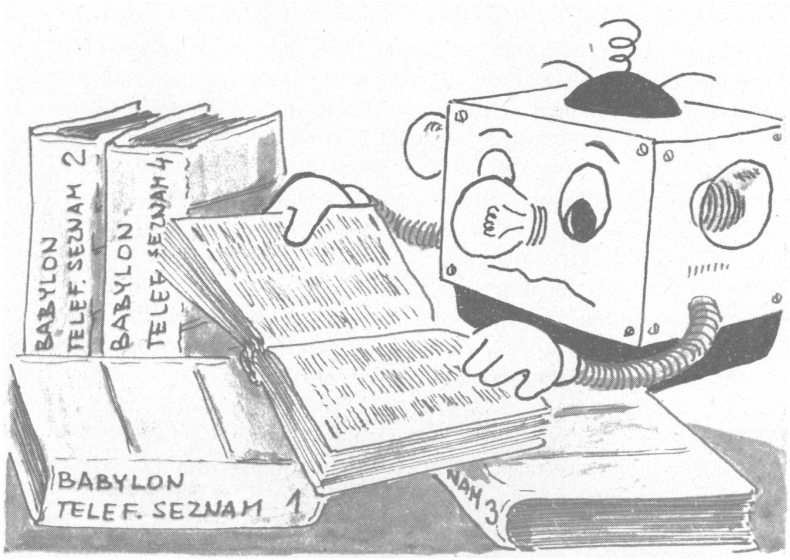

BABYLON
TELEF. SEZNAM 2
BABYLON
TELEF. SEZNAM 4
BABYLON
TELEF. SEZNAM 1
NAH 3

Table of Contents

III DML Building Technologies

Towards a Digital Mathematics Library
On the Crossroad

Petr Sojka

Masaryk University, Faculty of Informatics, Botanická 68a, 60200 Brno, Czech Republic
sojka@fi.muni.cz

Abstract. The DML workshop's objectives were to formulate the strategy and goals of a global mathematical digital library and to summarize the current successes and failures of ongoing technologies and related projects.
There is already experience with building regional DMLs or building big thematic scientific digital libraries. EuDML project reached it half-life period. While there are already big fulltext digital libraries in some domains like PubMed Central in the biomedical domain, Inspire in high-energy physics, why did not these emerge in other scientific areas? Will EuDML project in mathematics follow their success? Which crucial decisions has to be taken so that the heritage of mathematics would be sustainably at fingertips of scientists? We pose such and other questions, and try to find some answers in papers of this proceedings.

You are now at a crossroads. This is your opportunity to make the most important decision you will ever make. Forget your past. Who are you now? Who have you decided you really are now? Don't think about who you have been. Who are you now? Who have you decided to become? Make this decision consciously. Make it carefully. Make it powerfully. (Anthony Robbins)

1 The Dream

Mathematicians dream of a digital archive containing all peer-reviewed mathematical literature ever published, properly linked, with validated and verified content and form. It is estimated that the entire corpus of mathematical knowledge published over the centuries does not exceed 100,000,000 pages, an amount easily manageable by current information technologies.

"There is no royal road to mathematics" was reportedly said to Alexander the Great by fellow mathematicians as an answer when Alexander asked for a shortcut to understanding mathematics. There is no royal road to a general Digital Mathematics Library either. To make the dream a reality, concerted action of **D**igital specialists (computer scientists), **M**athematicians (topical experts), and **L**ibrarians (curators, information specialists) is needed. Mathematicians should have their say on what should constitute a DML of their choice, librarians of digital age should tell whether digital realization of the classical library metaphor of collecting, cataloguing and providing classified documents is enough for library to sustain today, and computer scientists should say what is possible technically, technologically and how. The

Petr Sojka, Thierry Bouche (editors): DML 2011, Towards a Digital Mathematics Library, pp. 1–6.
 ISBN 978-80-210-5542-1

shape of the target dream the three groups might end up with may be quite different. It may be 'just' virtual digital library mimicking the 'good old ages' of working with printed catalogue cards or with searching in basic metadata fields mathematicians are used to work with in referative mathematical databases. However, the technological progress speeds up so quickly that today there exist production systems starting to cope with semantically disambiguated texts in different languages on the fulltext level. To realize several different types of DML visions and wait for users' decision is simply beyond the socioeconomic possibilities of heterogenious mathematical community.

There are scientific domains, where the concerted action of DML preparation already happened — e.g. high-energy physics or [bio]medical scientists now work with DLs as Inspire or PubMed Central in ways that were never possible before. DLs allow for different search and linking strategies, providing new level of exploatation of scientific heritage. These communities have managed to agree on what is the common dream and how to realize it. Mathematics community is still on its road to reach the consensus and realize their DML. To help to pave the road for future DL in the domain of mathematics was the objective for setting up the DML workshop series.

2 DML Workshop Series

DML workshop series objective is to formulate the strategy and goals of a global mathematical digital library and to summarize the current successes and failures of ongoing technologies and related projects, asking such questions as:

* What technologies, standards, algorithms and formats should be used and what metadata should be shared?
* What business models are suitable for publishers of mathematical literature, authors and funders of their projects and institutions?
* Is there a model of sustainable, interoperable, and extensible mathematical library that mathematicians can use in their everyday work?
* What is the best practice for
  - retrodigitized mathematics (from images via OCR to MathML or TEX);
  - retro-born-digital mathematics (from existing electronic copy in DVI, PS or PDF to MathML or TEX);
  - born-digital mathematics (how to make needed metadata and file formats available as a side effect of publishing workflow [CEDRAM model, Euclid])?

The intention was to have the workshop as a forum for presentation and discussion of the latest developments in the the field of digitization of mathematics, based on the previous bilateral discussions and successful workshops. DML workshops have been held as satellite event of CICM multi-conferences in previous years: DML 2008 in Birmingham, UK, DML 2009 in Grand Bend, Ontario, Canada, and DML 2010 in Paris, France.

Topics of the DML workshops included

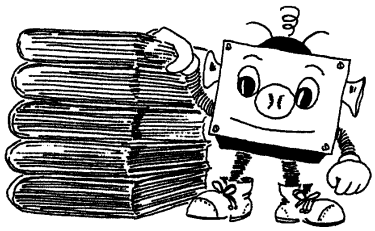
* search, indexing and retrieval of mathematical documents;

* ranking of mathematical papers, similarity of mathematical documents;
* math OCR with MathML and TEX output;
* document conversions from and to MathML, OpenMath, LATEX, PostScript and [tagged] PDF;
* mathematical document compression;
* processing of scanned images;
* algorithms for crosslinking of bibliographical items, intext citations search;
* mathematical document classification, MSC 2010;
* mathematical text mining;
* mathematical documents metadata exchange via OAI-PMH or OAI-ORE;
* long term archiving, data migration:
* reports and experience from math digitization projects;
* math publishing with long term archival goal;
* software engineering aspects of creating, handling MathML, OMDoc, OpenMath documents, and displaying them in web browsers.

3 On the Crossroad

There are already ongoing projects as the European Digital Mathematics Library (EuDML) project, where DML is aimed to be built in a bottom-up way from smaller repositories of contributing partners. This year, six out of nine Proceedings contributions do acknowledge EuDML. The Proceedings volume is divided into three parts:

1. Digital Mathematics Library Reports,
2. Digitization Workflows and Standards, and
3. DML Building Technologies.



EuDML is designed as a virtual library over existing regional repositories and publisher archives. One of participating EuDML data providers is the project DML-CZ. On page 9, Jiří Rákosník overviews current status quo of DML-CZ content and technological achievements and experience gained in the last two years.

EuDML invites joining any content providers that curate mathematical scientific content and bdim repository is an example of a smaller DML designed for smooth joining the club. Current state of this repository development is presented in a short paper by Vittorio Coti Zelati on page 19.

Talk by Takao Namiki speaking about tools to time stamp preprints brings Japanese know-how of running electronic journal server environment — for more see pages 19–23.

Digitization Workflows and Standards are covered in the second block of papers, all motivated and prepared as parts of the EuDML project. Researchers from Warsaw University's Interdisciplinary Centre for Mathematical and Computational Modelling are working on important problems of author name

disambiguation and metadata extraction from retro-born digital documents. They present their solutions in two papers on pages 27–37 and pages 39–44, respectively.

Standard EuDML metadata schema aimed to be used as rich data container for data exchange between EuDML and its partners is described in a paper by Thierry Bouche, Claude Goutorbe, Jean-Paul Jorda and Michael Jost. It is the important standard to follow for all publishers and other entities offering data for EuDML. Only slight modifications done to the widely used NLM Journal Archiving and Interchange Tag Suite warrants wide acceptance among publishers, as most of them already archive their holdings with Portico in this format.

There is a third part of proceedings named *DML Building Technologies*. As for most DML papers only metadata and PDF will be available, for tools working with full texts and math one needs a technology to get the texts and formulae out of the PDF. Progress report by Josef Baker et al. about the development of a tool to reverse engineering the PDF documents is presented on pages 65–75. At the end, this tool should possibly allow mathematics retrieval from PDF-only sources.

Tough arena of mathematical formulae indexing and search might be entered on the page 77 with a paper by Masaryk University group about WebMIaS system that allows math-aware structure respecting scalable retrieval of mathematical documents.

Finally, the respected reader can enjoy a case study about using discourse context to interpret object-denoting mathematical expression. Although the goal of semantic disambiguation with respect to math is far on the horizon of current natural language processing technologies, Magdalena Wolska et al. took the courage to tackle the problem of interpretation of mathematical expressions given the context.

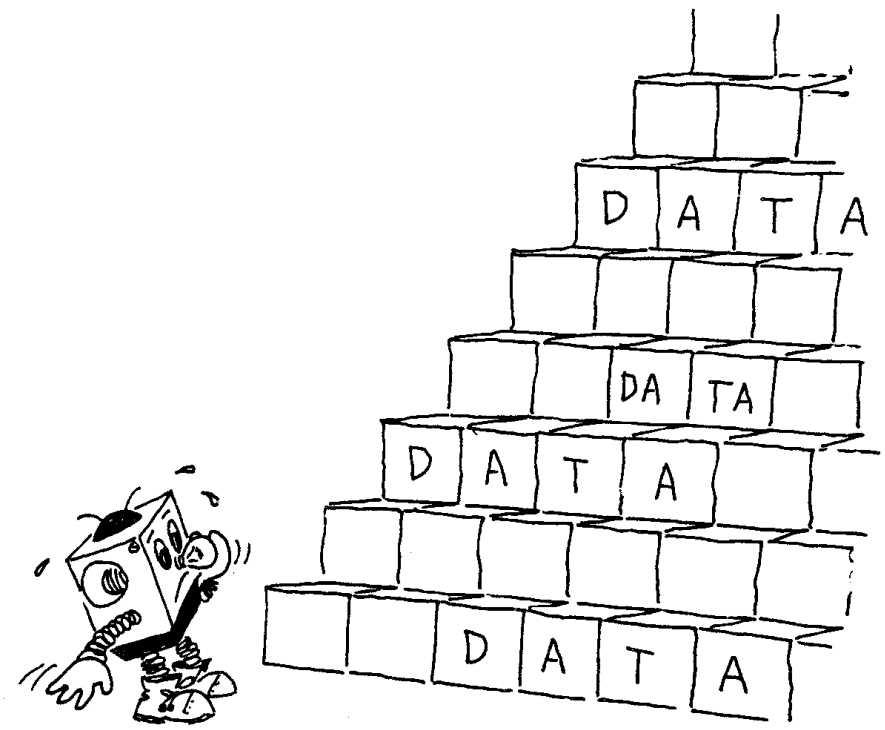
Ring the bells that still can ring
Forget your perfect offering
There is a crack in everything
That's how the light gets in.
(Leonard Cohen: Anthem)

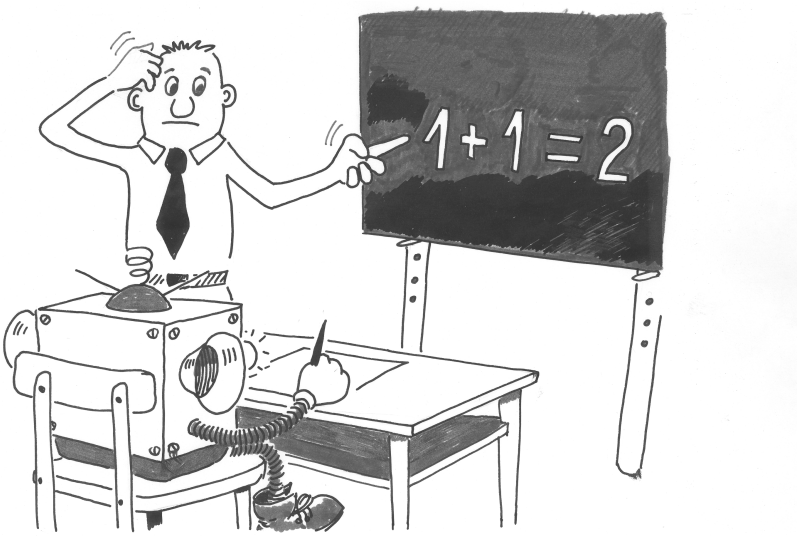
4 Summary

This volume contains the Proceedings of the Workshop *Towards a Digital Mathematics Library (DML 2011)*, organized by the Faculty of Informatics, Masaryk University and held on July 20–21, 2011 in Bertinoro, Italy, as a satellite event of CICM 2011 (Conference on Intelligent Computer Mathematics). DML 2011 offered nine presentations, EuDML session and [panel] discussion. We hope that it has helped to choose the right direction on the crossroad towards fulfilling the common dream of the Digital Mathematics Library.

Acknowledgements. My very special thanks go to the workshop PC co-chair Thierry Bouche, to Programme Committee members and additional referees for their hard work during review period. Most of the submitted papers were reviewed by three members of the Programme Committee, some even by four.

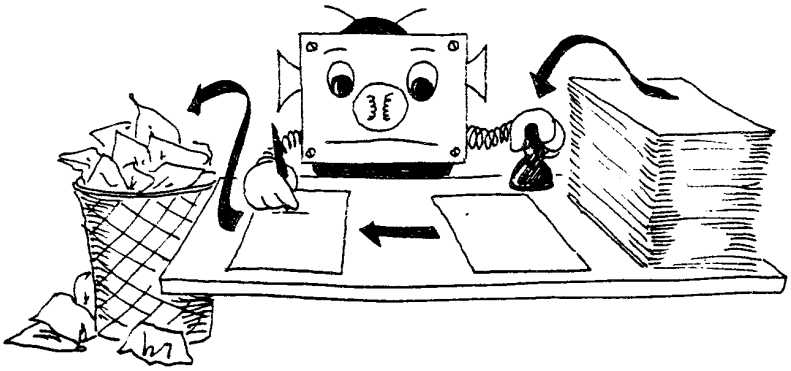
I would also like to express my appreciation to the CICM local chair Andrea Asperti for assuring conference facilities support on the workshop site. Last but not least, the cooperation of Masaryk University as a publisher of these Proceedings is gratefully acknowledged.




1+1=2

Part I

Digital Mathematics Library Reports



Recent Development of the DML-CZ and Its Current State

Jiří Rákosník

Institute of Mathematics AS CR, Žitná 25, 115 67 Praha 1, Czech Republic
rakosnik@math.cas.cz

Abstract. The project *DML-CZ: The Czech Digital Mathematics Library* has been implemented since 2005 and in 2010 switched over to routine operation. This report describes progress, growth and usage of the DML-CZ, the experience from cooperation with content providers in the designed editorial workflow, some newly implemented features, adjustments of the workflow following from both the ongoing practical experience and the requirements of the advancing EuDML project, the general public acceptance and attendance and the suggested economic model for sustainable development.

Keywords: digital mathematics library, retrodigitization, DML-CZ

1 The Original Project

The Czech Digital Mathematics Library DML-CZ [1] has been developed in frames of the project supported by the Academy of Sciences of the Czech Republic in the period 2005–2009 [2]. The project resulted in a full-featured digital library providing a free access to more than 275,000 pages of digital content in 11 journal titles, 6 conference proceedings series and 32 monographs including a collection of 25 books by the famous Czech mathematician Bernard Bolzano (1781–1848). Structured lists of references contained in metadata of journal articles represented about 160,000 items.

2 Routine Operation and Sustainability

When the funding of the project ended, the project coordinator, the Institute of Mathematics AS CR in Prague, undertook the responsibility for the maintenance and development of the DML-CZ cooperating with other project partners on a non-profit basis. The teams in the Masaryk University in Brno ensure the operation of the DML-CZ on their servers and provide technical service, process new additions and develop further tools. Colleagues from the Charles University in Prague effectively cooperate on provision and enhancement of metadata and the Digitization Centre in the Library of the Academy of Sciences employs the developed workflow for acquisition of new retrodigitized content.

The toolsets and workflow that have been implemented and verified in the project provide technical conditions for regular semiautomatic supply of new journal issues. Validation procedures enable editors to check completeness

Petr Sojka, Thierry Bouche (editors): DML 2011, Towards a Digital Mathematics Library, pp. 9–14.
 ISBN 978-80-210-5542-1

and integrity of data produced for the DML-CZ. The cooperation is based on formal contracts concluded between the Institute of Mathematics and journal publishers. Running costs in the first year of routine operation were covered by the Institute of Mathematics and starting with 2011 they will be shared by all journal publishers proportionally to the volume of delivered material.

The smooth passage to the routine operation of the DML-CZ confirms how important and forethoughtful was the timely decision to devote essential part of the project to development of efficient tools and implementation of a reliable workflow.

3 New Content

The DML-CZ content is growing faster than it has been expected. The journal *Pokroky matematiky, fyziky a astronomie* (*Advances in Mathematics, Physics and Astronomy*) published by the Union of Czech Mathematicians and Physicists since 1956 has been included. This brought new questions into consideration as the journal has rather heterogeneous content comprising original papers for general audience, translations from other journals, discussions on actual problems of mathematics and physics education, announcements, news etc. Major part of the journal published in the pre-TEX era had to be retrodigitized. Including such not purely scientific journal to DML-CZ has an important impact on increasing awareness of mathematics and of the library itself among teachers and students.

The scientific journal *Acta Universitatis Carolinae, Mathematica et Physica* has been recently processed and will be accessed soon. In fact, part of the journal has already been displayed during the project before 2009 in frames of the proceedings series of the *Winter School on Abstract Analysis*. Successful development and public acceptance of the DML-CZ encouraged the publisher to ask for enlistment of the whole journal in the DML-CZ.

The recent growth of DML-CZ is seen from the comparison of its content at the end of project in 2009 and its current content in Table 1.

Table 1. DML-CZ content

	December 2009	June 2011
Journals	11	12
Conference series	6	6
Monographs	32	65
Pages	275 220	313 707
Articles/Chapters	25 784	30 475
Issues/Volumes	2 223	2 619

4 Collected Works

A series of brand new questions emerged with the decision to include a special section devoted to collected works of eminent personalities of Czech mathematics. The first collection (see Figure 1) will be devoted to private archive of late Professor Otakar Borůvka (1899–1995), one of the most important Czech mathematicians in the 20th century who coincidentally worked at two affiliations of DML-CZ partners: the Masaryk University and the Institute of Mathematics AS CR.

Otakar Borůvka's archive contains respectable amount of 209 items representing 3,983 pages. Some of them have already been captured in the DML-CZ but the major part had to be retrodigitized. The collection consists of three main types of works: research works (9 monographs and 81 papers), other works (2 university textbooks and 51 journal and newspaper articles) and works about him (1 monograph, 1 thesis, 64 articles). This structure exceeds the scope of a standard DML focused mainly on research works. However, it meets the public demand and we believe that it belongs to the general mathematical heritage. Collections of further personalities are under consideration.

Investigating the structure of Borůvka's archive we learned that we are facing new problems like treatment of different editions of the same book, different manifestations of the same work (offprint vs. preprint, conference proceedings paper vs. working papers etc.). It came out that it is necessary to switch to the FRBR model (*Functional Requirements for Bibliographic Records*, see [3]) and to work with "creations" rather than with "publications" only. The copyright issues became more complex as well because many items in the archive come from sources which have not been treated in the DML-CZ so far.

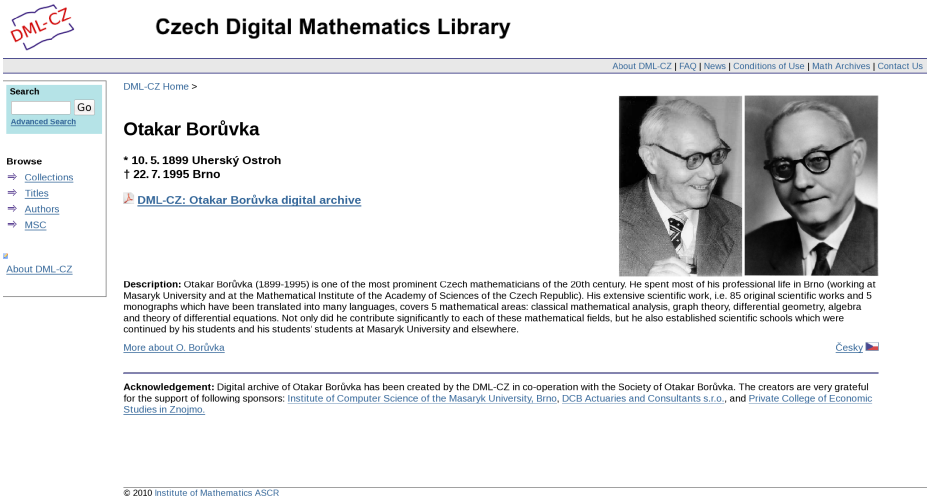


Fig. 1. Otakar Borůvka in DML-CZ

5 Technical Issues

The viability of a digital library rests above all with new acquisitions emerging mainly in the form of born-digital publications. Therefore, the DML-CZ project has been experimenting with automatic born-digital workflows since 2008 [7].

Editors of all journals included in DML-CZ are using tools and workflows that have been tailored to their individual publishing practice and that enable them to produce inputs for DML-CZ in a semiautomatic way. The formal consistency and integrity of the data are controlled by several validating procedures that have been developed in the project. This eliminates the majority of possible defects and decreases the demandingness of the final visual control.

Changes to the original editorial workflow were generally minimal. Automated procedures for validation of data of new journal issues are being gradually improved to catch irregularities, and full primary sources are archived in DML-CZ for internal use and development. Based on limiting the namespace of allowed TEX macros to those supported by our Tralics configuration, the recent improvements aimed at getting all metadata including abstracts, keywords and references transformed into representation using MathML. This was motivated by recent EuDML developments, namely by the possibility of math indexing [8].

Procedures developed during the project phase also helped in adding new journals and monographs or even complex constructs like collected works. Of course, it is less straightforward because many particular problems have to be tackled from the beginning — the structure, individual editorial practice, copyright issues etc.

The careful verification of data preceding their presentation in DML-CZ is done on different levels by different people using the Metadata Editor — a complex web-based system supporting all essential steps in the development of the library, see [4]. It appeared practical to implement a working copy of the DML-CZ presentation in which all the changes are first realized and checked. The working copy is being regularly transferred to the public DML-CZ after the final approval. This arrangement virtually prevents introducing errors to the public version and improves the stability of the public version.

The DML-CZ is one of content providers in the project of the European Digital Mathematics Library [6] and the Institute of Mathematics AS CR and the Masaryk University are partners in the project. Technical requirements stipulated in the project, developed tools and the close cooperation with other project partners have important impact on further development of DML-CZ itself. For the transfer of our metadata to the EuDML core via OAI-PMH a detailed metadata format in the form of tagged Dublin Core has recently been created. However, we find it rather limiting solution as we want to provide much more data than usually available via OAI-PMH, for instance references or even the full texts for indexing. Thus we will switch to providing data in the NLM format [5] in the future, and establish bidirectional secure channels for data exchange between DML-CZ and EuDML.

The Metadata Editor has been recently internationalized so that we can put it at the disposal of those EuDML partners who do not have such tool yet. It has been made more portable and further configuration support has been added. It is available as open source application at `http://sourceforge.net/projects/dme`.

6 Public Acceptance

Thanks to various promotion actions, lectures for mathematicians, librarians, students and teachers, journal articles, radio and TV interviews, the DML-CZ became well-known and exploited in the Czech Republic. Activities of the team in the EuDML project [6] and the fact that DML-CZ is highly ranked by Google Scholar due to the negotiated metadata interface increases its awareness abroad. According to Google Analytics there is a rather stable visit rate around 400 accesses per day. Most of them are naturally from the Czech Republic followed by the USA, Germany, India, Slovakia, China, Iran, France, Poland and the United Kingdom. Approximately 70 % of web traffic to DML-CZ is generated by Google itself. In one month, the site gets about 7,000 unique visitors.

The increasing awareness of DML-CZ is valued not only by the public but also by the cooperating journal editors. Even though it is yet too early for credible conclusions there are indications that the presence of journals in the DML-CZ helps to improve their publicity, increasing access rate to their papers and consequently number of citations.

7 Conclusion

DML-CZ is a living digital library which is growing up from its infancy to maturity while increasing its content as well as the extent and quality of services. Procedures implemented during the project phase proved efficient for inclusion of new material. The structure of DML-CZ has been recently extended with the section of collected works of eminent Czech mathematicians. This required some new arrangements, especially in the metadata scheme. The DML-CZ is profiting from its partnership in the EuDML project and preparing provision of its digital content for the upcoming EuDML.

Acknowledgement. This work is partly financed by the European Union through its Competitiveness and Innovation Programme (Information and Communications Technologies Policy Support Programme, "Open access to scientific information", Grant Agreement No. 250503).

References

1. *The Czech Digital Mathematics Library.* `http://dml.cz`.
2. *The Czech Digital Mathematics Library.* Project funded by the Academy of Sciences of the Czech Republic, 2005–2009. `http://project.dml.cz`.

3. *Functional Requirements for Bibliographic Records.* `http://en.wikipedia.org/wiki/Functional_Requirements_for_Bibliographic_Records`.
4. Miroslav Bartošek, Petr Kovář and Martin Šárfy: *DML-CZ Metadata Editor. Content Creation System for Digital Libraries.* In: Sojka, P. (ed.) DML 2008 – Towards Digital Mathematics Library. Proceedings of the workshop held in Birmingham, UK, July 27th, 2008. Brno: Masaryk University, 2008, pp. 139–151.
5. *National Library of Medicine Journal Archiving and Interchange Tag Suite.* `http://dtd.nlm.nih.gov/`
6. *The European Digital Mathematics Library.* CIP-ICT-PSP project No. 250503. `http://eudml.eu`.
7. Michal Růžička: *Automated Processing of TEX-Typeset Articles for a Digital Library*. In: Sojka, P. (ed.) DML 2008 – Towards Digital Mathematics Library. pp. 167–176 (2008), Birmingham, UK, July 27th, 2008.
8. Sojka, P., Líška, M.: Indexing and Searching Mathematics in Digital Libraries – Architecture, Design and Scalability Issues. In: Davenport, J.H., Farmer, W., Rabe, F., Urban, J. (eds.) Proceedings of CICM Conference 2011 (Calculemus/MKM). Volume 6824 of Lecture Notes in Artificial Intelligence, LNAI, Berlin, Germany, Springer-Verlag (2011) 228–243.

An Update on bdim
the Italian Digital Mathematical Library

Vittorio Coti Zelati

Dipartimento di Matematica e Applicazioni
Università degli Studi di Napoli "Federico II"
via Cintia, M.S. Angelo, 801 26 Napoli, Italy

Abstract. bdim has been in operation since the fall of 2010 and has been slowly growing in the last year. I will report here: 1) on what is new in bdim with respect to the first presentation in DML 2010; 2) on some of the technical aspects of our implementation; 3) on the projects for the near future; 4) on some of the issues related to possible integration of bdim with EuDML.

Keywords: bdim, digital libraries, metadata, mathematics

1 The State of bdim

The *Biblioteca Digitale Italiana di Matematica*, bdim, is a joint project of SIMAI (Società Italiana di Matematica Applicata e Industriale) and UMI (Unione Matematica Italiana) and has been in operation since the fall of 2010. The goal is to offer to the many Italian mathematics journals a common repository.

Up to now it contains part of the journal *Bollettino dell'Unione Matematica Italiana*, more precisely all the third series (years 1946–1967) and part of eight series (years 1998–2004).

The collection can be accessed at the web site `http://www.bdim.eu`.

Since the presentation of the project at DML 2010 (see [5]), bdim has got a stable url, and started operation. The collection has grown, in particular we have added part of the eight series of BUMI. We have at the moment 1,670 articles in the collection.

We have a new search engine, Lucene based, through which a full text search has been implemented. We have added MathJax to display math, and we will soon offer MathML as an option.

The collection has also been integrated in the mini-DML project.

2 The bdim Project: Some Technical Aspects

We started our project with the aim to make it compatible both with the international effort to produce a Digital Mathematics Library and with the Italian project of the Biblioteca Digitale Italiana (see [1]). To this end we have followed

– the international guidelines contained in *Some Best Practices for Retrodigitization*, see [4]

Petr Sojka, Thierry Bouche (editors): DML 2011, Towards a Digital Mathematics Library, pp. 15–18.
 ISBN 978-80-210-5542-1

– indications from NUMDAM see [3], which has been very helpful in starting and developing our project;
– the Italian guidelines for the Biblioteca Digitale Italiana, see [6] (in Italian), with their required set of metadata (in MAG format).

For these purpose we have decided to describe the metadata related to objects in our repository (where for us objects are not only articles but also issues, volumes and journals) with XML files in a particular format which we have developed, see the schema file `http://www.bdim.eu/schema/bdim.xsd`.

Examples of XML files describing an article and the corresponding issue can be found at `http://www.bdim.eu:8080/fedora/get/bdim:BUMI_2004_8_7B_1_23_0/FILE_BDIM`, and `http://www.bdim.eu:8080/fedora/get/bdim:BUMI_2004_8_7B_1/FILE_BDIM`

We have collected in these files all the information that we have gathered, even that which is not meant to be distributed, like the references (in BibTex format) which we have downloaded from MathSciNet while looking for matches for the bibliographies.

For articles, the XML file also gives information (size, format, dimension, md5, location) on the files associated to the given article (at least the PDF files, the DjVu file and the XML file containing the OCR of the article). For issues, volumes and journals the XML file also gives additional information about their internal structure (the articles that make up an issue, etc.). This is required by the standards of the Biblioteca Digitale Italiana.

We store and disseminate our collection using a FedoraCommons repository (see [2]). FedoraCommons uses a "compound digital object" design which aggregates one or more content items into the same digital object. Following this approach we created a Fedora object for each article, issue, volume and journal. Each Fedora object aggregates different contents (datastreams in the FedoraCommons language): the XML file describing the object, the relations involving the object itself and, for articles, their pdf and djvu files.

The repository `http://www.bdim.eu:8080/fedora/` runs on a dedicated tomcat web server. One can get the different datastreams of an object from the repository as follows: `http://www.bdim.eu:8080/fedora/get/bdim:BUMI_2004_8_7B_1_23_0/BUMI_2004_8_7B_1_23_0.pdf`.

The FedoraCommons repository is quite flexible and gives us complete control on access to the single objects and datastreams.

Moreover FedoraCommons provides an OAI server, which has been implemented at the address `http://www.bdim.eu:8080/oaiprovider/?verb=Identify`, and which disseminates our objects in oai_dc and minidml formats.

There is also a search engine distributed with FedoraCommons, based on Lucene, which indexes the objects in our repository. Using this, we have implemented the advanced search at the address `http://www.bdim.eu/ricerca`, a highlighting mechanism for hits and different methods to sort them (by relevance or "score", author and year).

We have set up our web site in such a way that the user does not have to interact with the FedoraCommons repository directly. The web site

at http://www.bdim.eu, written in PHP, fetches the relevant data from the repository, transforms it via XSL an builds the web page.

3 The Projects for the Future

Our aim is to include in our collection all the Italian mathematical journals willing to do so. Several have already expressed their desire to join bdim:

- *Rendiconti di Trieste* (since 1969, 11,000 pages),
- *Le Matematiche, Catania* (since 1945 23,000 pages),
- *Rivista di Matematica, Parma* (23,000 pages),
- *Ricerche di Matematica, Napoli* (up to 2005),
- *Rendiconti Lincei* (up to 2004, 80,000 pages)
- *Rendiconti dell'Accademia delle Scienze Fisiche e Matematiche, Napoli* (since 1862, 40,000).

Unfortunately, due to lack of funds, we have not yet been able to enlarge our collection. But we have been applying for funding to several places and we hope to get some funds in the near future.

We have also started a project for digitization of collected works of Italian mathematicians. We are working on the collected works of Salvatore Pincherle (1853–1936), Italian mathematician, first president of the Unione Matematica Italiana. This project presents several interesting aspects, in particular related to copyright issues and to the fact that we will not be dealing with journal articles only. Also, part of the relevant material has already been digitized within other digitization projects, so we have to decide how to proceed in these cases.

4 Integration with EuDML

The next step we would like to take is to integrate our collection in EuDML. We have been looking at the standard for data exchange, and we believe that our metadata could easily be cast in the required format. We are very interested in the EuDML enhancer toolset and we would like to share ideas and methods concerning the tools we have developed in our project. In particular, our expertise in the area of MathML is very limited and we could certainly benefit from the international cooperation.

References

1. *Biblioteca Digitale Italiana*, http://www.bibliotecadigitaleitaliana.it/.
2. *FedoraCommons*, http://www.fedora-commons.org/.
3. NUMDAM, http://www.numdam.org/.
4. Committee on Electronic Information and Communication (CEIC) of the International Mathematical Union (IMU), *Some best practices for retrodigitization*, Endorsed on August 20, 2006 at Santiago de Compostela by the General Assembly of the International Mathematical Union. http://www.mathunion.org/CEIC/Publications/retro_bestpractices.pdf.

5. V. Coti Zelati, *bdim: the Italian Digital Mathematical Library*, Towards a Digital Mathematics Library. Paris, France, July 7–8, 2010 (Brno, Czech Republic), (P. Sojka, ed.), Masaryk University Press, 2010, pp. 79–81, http://dml.cz/dmlcz/702576.
6. Istituto Centrale per il Catalogo Unico, *Standard mag – versione 2.0.1*, http://www.iccu.sbn.it/opencms/opencms/it/main/standard/metadati/pagina_267.html.

Time Stamping Preprint and Electronic Journal Server Environment

Takao Namiki[1], Kazutsuna Yamaji[2], Toshiyuki Kataoka[3], and Noboru Sonehara[4]

[1] Department of Mathematics, Hokkaido University
nami@math.sci.hokudai.ac.jp
[2] R & D Center for Academic Networks, National Institute of Informatics
yamaji@nii.ac.jp
[3] R & D Center for Academic Networks, National Institute of Informatics
kataoka@nii.ac.jp
[4] Information and Society Research Division, National Institute of Informatics
sonehara@nii.ac.jp

Abstract. The exchange of preprints and journals plays an important role to communicate new research ideas and results in many academic fields. Distribution of preprints and journal articles by electronic file via the Internet has become a primary method in addition to paper publication. Electronic preprints and articles in the paperless era should be certified in terms of existence proof and tamper resistance because they are easily modified by their site administrator. We developed a secure preprint and electronic journal service environment that uses an electronic signature and timestamp technique.

Keywords: long-term electronic signature, electronic signature, timestamp, preprint, archives.

1 Introduction

In the last decade electronic preprints and mathematical journals become an infrastructure of mathematical researches, however, it is not familiar to us that electronic files of articles should be preserved with its timestamp because we might assure community of the originarity even if the modified versions were floating. On the other hand, in order to establish the priority of research ideas and results, preprints are published in several research fields. For example, arXiv.org, which is operated by Cornell University Library, is a major preprint server that covers physics, mathematics, non-linear science, computer science, quantitative biology, and statistics [1]. Moreover, while many universities operate local preprint servers, the security of these digital documents may be at risk. Currently, most preprints still have the printed issue in addition to digital file distribution. In this dual publication scheme, the printed issue is regarded as the trusted original version. However, in paperless publication, it is difficult to distinguish a copy from the original. Electronic journal has the same problem. That is, the security level of the preprint and journal article files has to be raised in order to protect the research results. In the field of business, the security of

Petr Sojka, Thierry Bouche (editors): DML 2011, Towards a Digital Mathematics Library, pp. 19–23.
 ISBN 978-80-210-5542-1

digital documents containing patentable ideas and intellectual property are guaranteed by means of an electronic signature and timestamp technique [2]. These technologies can be applied to academic publishing to ensure reliable digital content. This study proposes a secured preprint server environment using EPrints 3 software and describes its application to the mathematical preprint and electronic journal service at the Hokkaido University.

2 Long-term Electronic Signature

The electronic signature (ES) ensures integrity and signer of a document. An ES (Figure 1) is composed of signature policy, other signed attributes and digital signature [3]. The digital signature is created from an encrypted hash value of a digital document. The recipient (client application) can detect a falsification of the document by comparing hash values calculated from the original document and decrypted from the digital signature [4]. The timestamp (TS) technology guarantees existential evidence of digital documents [5]. The combination of ES and TS, as indicated by ES-T in Figure 1, ensures the authenticity of the digital documents.

The ES and TS, each of which is based on the digital signature technology, have a validity period and revocation functions against compromise of the hash algorithm and leak of the private key. However, the temporary nature of these functions causes a problem for long-term preservation. To solve this problem, a long-term signature has been proposed [3]. This signature format embeds a complete certificate and revocation references shown as ES-C in Figure 1; therefore, ES and TS can be verified even after the signature expires. This study employed international standard RFC3126 as a long-term signature format and applied it to the article PDF documents.

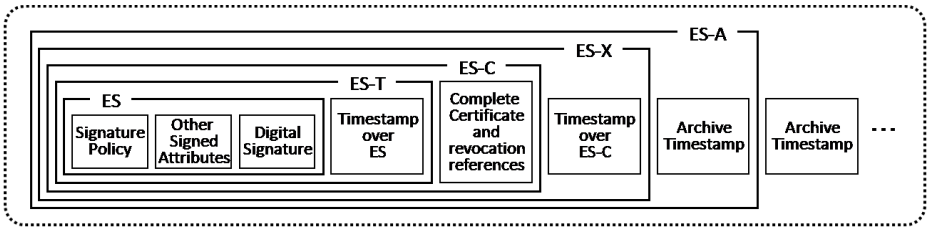


Fig. 1. Long-term signature format defined by RFC 3126 [3]

3 Application for Preprint Server

3.1 System Environments

The system architecture of the developed preprint and electronic journal server environment is shown in Figure 2. Application of the long-term signature

requires a certification authority (CA) server and long-term signature server. The CA issues a digital certificate to a user who would like to register a document on the server. CA was established by NAREGI-CA which is an open source software originally developed for grid computing [6]. NAREGI-CA provides a variety of utilities including key generation, certification issuance, verification, and storage. The long-term signature server obtains a timestamp token from the time stamping authority managed by a trusted third party. EPrints 3 was chosen for the server software.

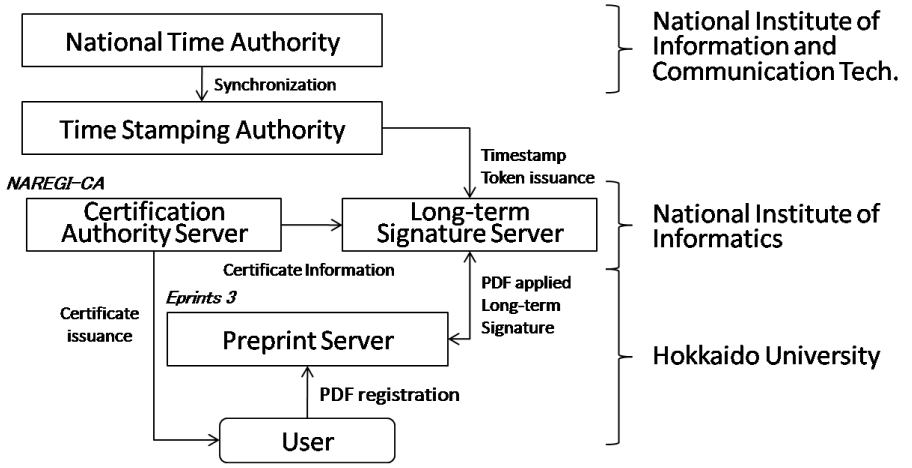


Fig. 2. System architecture of the server environment

3.2 Data and Work Flow

To attach a long-term signature to an article, a registrant must do the following procedure.

1. Obtain a digital certificate
2. Prepare an article PDF
3. Attach an electronic signature using the digital certificate to the article PDF
4. Register the article PDF file on the server

Once the PDF file has been registered, the following steps will be performed automatically.

1. The server sends the PDF file to the long-term signature server.
2. The long-term signature server obtains a timestamp token from the time stamping authority and applies it to the PDF file.
3. The server obtains the above PDF file.
4. The server makes the PDF file public.

In the EPrints system, the moderator receives an email whenever a new item has been submitted. This workflow is executed by utilizing the perl script named send_alerts. The procedures from 5 to 8 are inserted just before the workflow. Besides the additional task that the article registrant needs to apply an electronic signature, the proposed system does not have more manual operations than the conventional one.

An example of an article PDF secured by a long-term signature is shown in Figure 3. The emblem of the Hokkaido University and minimum information for the long-term signature are superimposed on the article. The position, size, and form of this information can be customized by the client application. The long-term signature can be verified, as shown on the right of the Figure 3.

Fig. 3. Example of preprint PDF file guaranteed by long-term signature and its verification results

4 Conclusion

Since the EPrints system stores the registered file under the certain directory of the OS file system, the proposed system simply transfers the file between EPrints and the long-term signature server. This process can be used in different systems.

In Japan, the university public key infrastructure (UPKI) project is in progress, and it will be one of the key technologies of the cyber science infrastructure (CSI) framework promoted by the National Institute of Informatics. That is, the security of digital contents will be ensured by means of digital certificates issued by the UPKI project. We believe that this scheme

of enhanced security will accelerate the exchange of scholarly content via the Internet and will boost scientific research and education activities.

Finally we remark that the long-term signature framework works well in electronic journals without print edition because titles of small scale electronic journals are published from small publishers in mathematics. We have already experimented in Hokkaido Mathematical Journal, volume 38, no. 2. As access to this journal is restricted by IP address for subscribed institution, we place a PDF file of the issue on the URL: `http://www.math.sci.hokudai.ac.jp/~nami/note/hmj38-2-1.pdf` to access it from any place.

References

1. McKiernan, G. (2000). *arXiv.org: the Los Alamos National Laboratory e-print server.* The International Journal on Grey Literature 1(3): 127–138.
2. Haber, S. and Stornetta, W. S. (1991). *How to time-stamp a digital document*. Journal of Cryptology 3(2): 1432–1378.
3. Pinkas, D., Ross, J. and Pope, N. (2001). *Electronic Signature Formats for long term electronic signatures.* IETF RFC 3126.
4. Housley, R. (2004). *Cryptographic Message Syntax (CMS).* IETF RFC 3852.
5. Adams, C., Cain, P., Pinkas, D. and Zuccherato R. (2001). *Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP).* IETF RFC 3161.
6. Miura, K. (2006). *Overview of Japanese science Grid project NAREGI* Progress in Informatics 3: 67–75.
7. Millington, P. and Nixon, W. J. (2007). *EPrints 3 Pre-Launch Briefing* Ariadne 50.
8. Sakauchi, M., Yamada, S., et al. (2006). *Cyber Science Infrastructure Initiative for Boosting Japan's Scientific Research.* CTWatch Quarterly 2(1): 20–26.

Part II

Digitization Workflows and Standards

Towards a Flexible Author Name Disambiguation Framework

Łukasz Bolikowski[1] and Piotr Jan Dendek[1,2]

[1] Interdisciplinary Centre for Mathematical and Computational Modelling
University of Warsaw, ul. Pawińskiego 5A blok D, 02-106 Warsaw, Poland
[2] Faculty of Electronics and Information Technology, Warsaw University of Technology
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland

Abstract. In this paper we propose a flexible, modular framework for author name disambiguation. Our solution consists of the core which orchestrates the disambiguation process, and replaceable modules performing concrete tasks. The approach is suitable for distributed computing, in particular it maps well to the MapReduce framework. We describe each component in detail and discuss possible alternatives. Finally, we propose procedures for calibration and evaluation of the described system.

Keywords: name disambiguation, problem decomposition, scoring functions, single-linkage clustering, MapReduce framework, machine learning

1 Introduction

A person's name may be presented in several different forms, for example: "M. Brown", "Michael Brown", "M. A. Brown", "Michael A. Brown", or "Michael Arthur Brown". On the other hand, the same name form may, depending on the context, refer to several different people. Furthermore, when processed by a computer system, a name form may be distorted due to deficiencies of the software used, e.g. due to OCR errors, or incompatible handling of diacritics.

In a digital library storing scientific publications, such as the European Digital Mathematics Library (EuDML), it is of great interest to associate the names of a document contributors with their identities. This piece of information is crucial for several purposes, including: presentation of all the works of a particular author in a concise form; analysis of researchers' co-operation network and identification of communities [12]; or assessing the impact of individual researchers.

The problem of associating names with identities is commonly referred to as name disambiguation. Torvik and Smalheiser [13] show that almost 2/3 of authors in MEDLINE have an ambiguous name (surname + first initial), thus providing a good motivation for further study.

Several approaches to the problem have been proposed in the literature. Han et al. [4] offer two solutions based on supervised learning: one using naîve Bayes, the other using Support Vector Machines. Mann and Yarowsky [8], as well

Petr Sojka, Thierry Bouche (editors): DML 2011, Towards a Digital Mathematics Library, pp. 27–37.
 ISBN 978-80-210-5542-1

as Pedersen et al. [11] and Han et al. [5] propose various unsupervised clustering approaches. Kang et al. [7] stress the importance of co-authorship analysis in the name disambiguation process. Torvik and Smalheiser [13] propose a 5-step procedure to disambiguate names in an entire collection of documents.

In related research, Galvez and Moya-Anegón [3] employ finite-state graphs to standarize name variants, while Pavelec et al. [10] address the problem of finding the author of an anonymous document using stylometry.

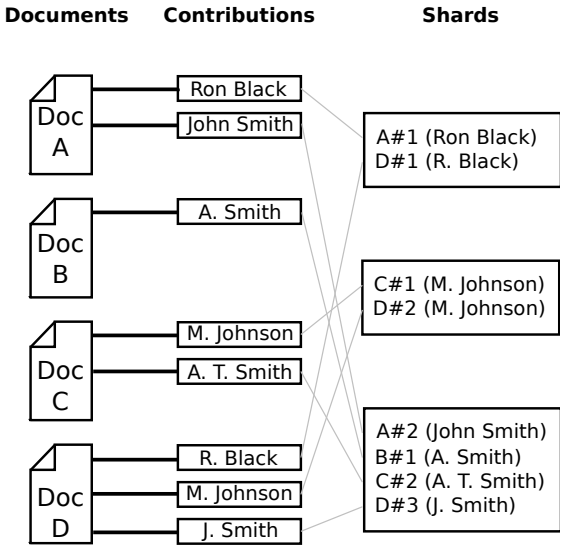


Fig. 1. Documents, contributions, and shards. Decomposition of contribution space into shards is performed according to some hash function, for example: surname of the contributor.

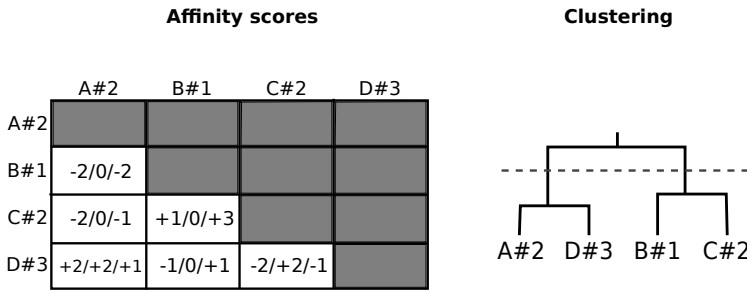


Fig. 2. For each pair of contributions within a shard, affinity is calculated based on several attributes. Next, a clustering algorithm is executed to extract author identities.

2 Author Name Disambiguation Framework

2.1 Vocabulary and assumptions

Let us begin by establishing a vocabulary that will be used throughout the rest of this paper and stating certain assumptions about the objects discussed.

A **document** is an article or a book, referenced by an identifier that is unique within the collection. We assume to have an access to each document's metadata, which at the very least contain names of all the authors.

A **contribution** is an occurrence of an author's name in a document's metadata. Thus, a contribution refers to exactly one document and to exactly one person. For the sake of clarity, let us identify a contribution by concatenating the document's identifier with "`#`" and with (one-based) position of the name on the list of authors of the document. For example, if a document with identifier `124532` has three authors: J. Stone, M. Black and A. Smith, then the contribution of M. Black to the document is identified by `124532#2`.

Next, a **shard** is a group of contributions such that a **hash function** yields the same value for all the contributions in the shard. We are free to choose any hash function which satisfies the following condition: all the contributions of the same person should have the same hash. The converse does not have to be true, though: contributions of several different people may have the same hash. A case satisfying the above conditions is shown in Figure 3. There are contributions (visualised as "document" icons) gathered into shards (ellipses). Papers written by the same person ("human" icon) should be in the same shard. It may occur that papers of more then one person will appear in the same shard, which is valid and expected. The described situation is the result of a well-constructed hash function. In contrast, shards in Figure 4 are the result of a defective hash function, which splits contributions of same person into more then one shard.

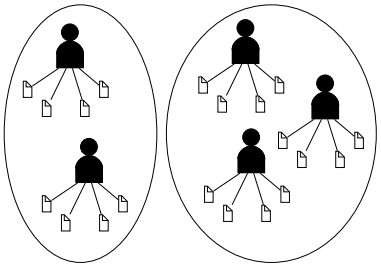


Fig. 3. Contributions (document icons) of the same person (human icons) should be present in the same shard (ellipses).

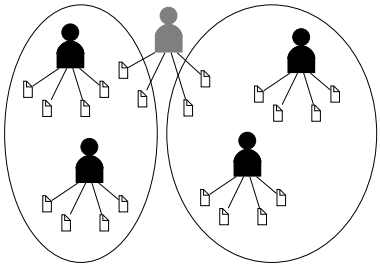


Fig. 4. The result of a defective hash function is a division of contributions (document icons) of the same person (human icons) into different shards (ellipses). The grey person's papers are split in such a manner.

The purpose of a hash function is to decompose the problem space into manageable shards. A good example of a hash function is a function which returns the lower-cased surname of a name with all the diacritic marks removed (cf. Fig. 1).

An **attribute** (or a **feature**) is a function that extracts some feature from each contribution, such as: year of publication of the document, list of keywords in the document, or e-mail address of the person. Such an attribute may be either taken from the document's metadata, or inferred from the full-text. Whenever we write "an attribute of a contribution", we mean the value of the attribute for the contribution. When a given contribution lacks a given attribute, we shall denote the value by "$\perp$".

Affinity of a pair of contributions is a real number which is a measure of our confidence that the two contributions are made by the same person. It is a weighted sum of **atomic affinities** calculated for individual attributes. We assume that for each atomic affinity is a function that returns values from the range $[-1, 1]$. When returned value is equal 1 it means that according to that function two contributions share the same personality for sure. Zero value point out that function could not determine if contributions share the same person or not. Finally, -1 value indicate that two contributions are made by two different persons. The relative imporance of atomic affinities (and thus, indirectly, of attributes) is controlled by **weights**, which are non-negative real numbers.

Finally, a cluster of contributions believed to be made by the same person will sometimes be referred to as an **identity**.

2.2 Name disambiguation flow

Our name disambiguation procedure, presented in Figure 5, takes a collection of documents on input, and yields sets of contributions likely to be made by the same person on output. The procedure consists of three steps:

1. all the contributions in the input collection of documents are partitioned into shards (cf. Figure 1);
2. for each pair of contributions within each shard, its affinity is calculated (cf. Figure 2);
3. contributions within each shard are clustered (cf. Figure 2).

As a result of the last step, we have groups of contributions suspected to be made by the same person. The entire solution maps well to the Google's MapReduce framework [2].

Each part of the process is described in the following part of the article.

2.3 Partition of the input collection

The problem domain of name disambiguation is typically in the order of millions of contributions. Since a part of our solution employs pairwise comparison (quadratic computational complexity), it is crucial to decompose the domain beforehand, thus reducing the computational effort.

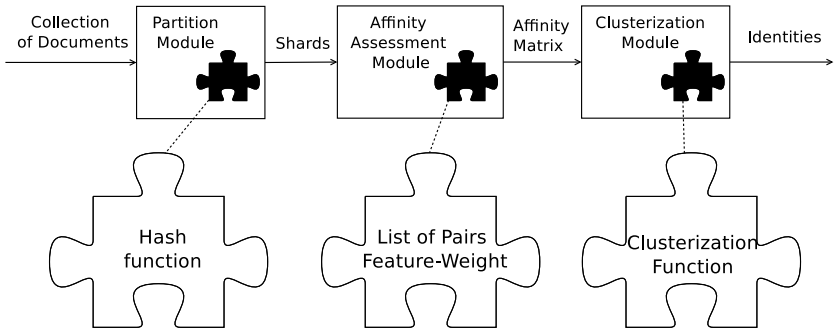


Fig. 5. Name disambiguation procedure stages. Puzzles represent replaceable parts of the solution.

When looking for same authorship of contributions, it is also apparent that some contributions do not need to be compared (e.g. they were written in time interval of two hundred years). Using this information, we can extract from the collection a working set of interest, in which documents are similar in some way, simultaneously filtering out other contributions, unlikely to be made by the same author.

For the sake of simplicity, one can restrict our attention to filters which partition the contributions into disjunctive working sets. Let us call such filters "hash functions", and the resulting working sets—"shards".

The framework is flexible and allows us to plug in any hash function satisfying the conditions stated in Subsection 2.1. We have chosen a basic hash function which takes the surname, lower-cases it, and removes all the diacritic marks.

Torvik and Smalheiser [13] found that misspellings, differences in authors' surnames notations or differences of complete surnames (e.g., "Olle Goig" vs. "Olle-Goig", "Le Roith" vs. "LeRoith", or "J. Benson" and "J. Flynn" both referring to Judy Benson Flynn) appear in approximately 1.8% of contributions. Moreover, the hash function described earlier is resilient to the majority of these discrepancies. Therefore, we have decided to ignore such mistakes.

However, if one chooses to account for misspellings, it can be achieved by using a more sophisticated hash function, for example one feeding the surname to the Soundex algorithm. (although differences in pronunciation across countries may spoil the result). It is not advised to use editorial distance, because in this case shards may overlap. Another way to extend the basic hash function is to include the initial of the first name in the hash and removing non-alphabetic characters (e.g.dashes) and diacritic marks from a surname. The biggest disadvantage of the described hash function is that it works well with surnames written with Latin alphabet. If one needs to proceed surnames written in different alphabets, they may write appropriated function to translate them into same alphabet representation. Whereas if one works on a set of surnames written in same alphabet, they do not need to employ any sophisticated surname

translation function. On a side note, this illustrates the flexibility of the proposed framework.

The final product of the partition performed using the basic hash function is a large number of small shards and a small number of larger shards. In the collection of Polish History Museum (about 100 thousand documents, mostly arts and humanities), we have found that roughly 55% of all contributions are in shards of size 10 or less, while the largest shard was in the order of a thousand. In MEDLINE (over 15 million documents, mostly life science) when using surname + initial of the first name as the hash, Torvik and Smalheiser [13] found that the largest shard ("J. Lee") contains almost 16 thousand contributions. Given this order of magnitude of shard sizes, it is feasible to run an analysis of quadratic time complexity on each of the shards.

To sum up, the described decomposition of contributions dramatically reduces the computational complexity of the second step in the procedure. As an added benefit, the individual shards can be processed in parallel, further reducing the wall-clock time of the entire process.

2.4 Pairwise contribution comparison

Once the domain is decomposed into shards, we need to establish affinities of pairs of contributions in each shard. The total affinity is the sum of atomic affinities, assessed from an extensible set of features (a list of feature examples is presented in Table 1 on the facing page).

A feature is a method, which takes two contributions on input and returns a real number from the $[-1, 1]$ range. The resulting number is multiplied by the weight of the feature, giving an atomic affinity. Finally, atomic affinities are summed up to one number—total affinity.

For each shard, the output of the procedure is a matrix of contribution affinities. The matrix is passed to the next stage of the name disambiguation process.

Feature weight Some features, such as e-mail, can alone prove that two contributions share the same identity. Other features are only weak indicators, e.g. contributing to the same journal. The weight of a feature is introduced to reflect the feature's impact on the name disambiguation process.

Feature aspects Referring to the Table 1 we can point out a few aspects of a feature:

- Discretization level
  1. **Discrete.** Some features are highly discrete, e.g. e-mail feature.
  2. **Continuous.** Some features give a continuous result, e.g. continuous time distance feature.
- Polarisation level
  1. **Polarised.** A feature can be highly positive (negative) indicator and simultaneously weak negative (positive) indicator, e.g. e-mail or journal feature.

Table 1. The list of the feature examples

Name	Description
Time distance (continuous)	$= \begin{cases} 0 & \text{year}(c_1) = \perp \vee \text{year}(c_2) = \perp \\ -1 & \lvert\text{year}(c_1) - \text{year}(c_2)\rvert > 70 \\ 1 - \left(\frac{\text{year}(c_1) - \text{year}(c_2)}{70}\right)^2 & \text{otherwise} \end{cases}$
Time distance (discrete)	$= \begin{cases} 0 & \text{year}(c_1) = \perp \vee \text{year}(c_2) = \perp \\ -1 & \lvert\text{year}(c_1) - \text{year}(c_2)\rvert > 70 \\ 1 & \text{otherwise} \end{cases}$
Journal	$= \begin{cases} 0 & \text{journal}(c_1) = \perp \vee \text{journal}(c_2) = \perp \\ 1 & \text{journal}(c_1) = \text{journal}(c_2) \\ -0.1 & \text{otherwise} \end{cases}$
Email	$= \begin{cases} 0 & \text{email}(c_1) = \perp \vee \text{email}(c_2) = \perp \\ 1 & \text{email}(c_1) = \text{email}(c_2) \\ -0.1 & \text{otherwise} \end{cases}$
Language	$= \begin{cases} 0 & \text{language}(c_1) = \perp \vee \text{language}(c_2) = \perp \\ 0.05 & \text{language}(c_1) = eng \vee \text{language}(c_2) = eng \\ 0.1 & \text{language}(c_1) = \text{language}(c_2) \\ -1 & \text{otherwise} \end{cases}$
Keywords (discrete)	$= \begin{cases} 0 & \text{keyword}(c_1) = \emptyset \vee \text{keyword}(c_2) = \emptyset \\ -1 & \frac{\lvert\text{keyword}(c_1) \cap \text{keyword}(c_2)\rvert}{\lvert\text{keyword}(c_1) \cup \text{keyword}(c_2)\rvert} < 0.25 \\ 1 & \text{otherwise} \end{cases}$
Keywords (continuous)	$= \begin{cases} 0 & \text{keyword}(c_1) = \emptyset \vee \text{keyword}(c_2) = \emptyset \\ \frac{\lvert\text{keyword}(c_1) \cap \text{keyword}(c_2)\rvert}{\lvert\text{keyword}(c_1) \cup \text{keyword}(c_2)\rvert} * 2 - 1 & \text{otherwise} \end{cases}$
Self-citation	$= \begin{cases} 1 & name(c_1) = name(reference(c_1)) \\ 0 & \text{otherwise} \end{cases}$
Co-authorship	$= \begin{cases} 0.7 & \lvert\text{coauthors}(c_1) \cap \text{coauthors}(c_2)\rvert = 1 \\ 1 & \lvert\text{coauthors}(c_1) \cap \text{coauthors}(c_2)\rvert > 1 \\ 0 & \text{otherwise} \end{cases}$

2. **Fair.** A feature can be equally important as positive and negative indicator, e.g. discrete time distance features.

– Structure
1. **Flat structure.** A feature can focus on two documents connected to given contributor, e.g. year feature.
2. **Graph structure.** A feature can check connections between larger quantity of documents, e.g. co-authorship, self-citation features.

One of advantages of the framework is a possibility of flexible feature addition and weight assignation, thanks to the usage of Spring Framework. The only restriction is that feature methods must be written in Java programming language.

2.5 Clustering process

Last part of the name disambiguation process is clusterization of contributors based on similarity matrix obtained in the previous step. As previously, one can use specially prepared clusterizer to reach desired result. In our case, we decide to use Single-linkage Hierarchical Agglomerative Clustering (described in [9]) with customization. This algorithm takes in each step two "active" contributors with top level score. If this score is below a given threshold (referred as T) the procedure is ended. Otherwise similarity level of contributions (σ) is recalculated in following manner:

$$\forall_{1<i<N}\forall_{i\neq a}\forall_{i\neq b}\sigma(c_a,c_i)=\sigma(c_b,c_i)=\begin{cases} -\infty & \sigma(c_a,c_i)<T \\ & \vee\ \sigma(c_b,c_i)<T \\ \sigma(c_a,c_i) & \sigma(c_a,c_i)>\sigma(c_b,c_i) \\ \sigma(c_b,c_i) & \sigma(c_a,c_i)\leqslant\sigma(c_b,c_i) \end{cases} \quad (1)$$

After recalculation, one of the contributors is deactivated. The process is repeated till either threshold is reached or there is only one active contributor left.

We choose simple-linkage clustering because it has desirable behaviour. If A is close to B, which is close to C, then merging A and B does not set the merged cluster further apart from C (which may be the case in Complete Link Clustering). Moreover, this type of clusterization provides $\mathcal{O}(N^2)$ time complexity, which is well acceptable. However, in other approaches other clusterizers can be applied. In Figure 6 one can see an example of how the customized Single-Linkage Clusterizer works.

2.6 Framework structure

Each part of the name disambiguation framework is written in Java programming language, with usage of Spring Framework and Sesame RDF Store. Thanks to usage of Spring Framework one can easily add their own elements of solution or replace them. Sesame RDF Query Language (SeRQL), in which database queries are written, is designed to support operations over graph structures.

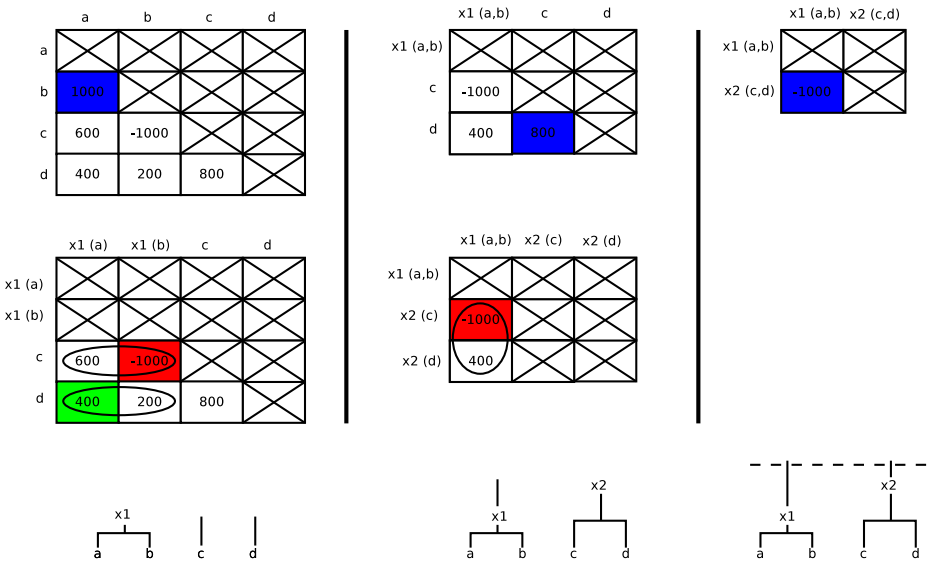


Fig. 6. Example of customized single-linkage clustering. First step is finding biggest affinity of two components in the affinity matrix. Then, if the value is negative clusterizing process is ended, whereas when the value is positive, contributions are merged into the same cluster. Affinities of the new cluster and other contributions or clusters are calculated based on recently taken components according to Equation 1.

2.7 Distributed computation

The presented framework is well-suited for distributed computation. Each shard is processed independently from all the others, and thus each can be processed on a different computing node.

The entire process can be implemented using Google's MapReduce [2]. In the "map" phase, for each document all its contributions are emitted, with the result of the hash function as the key. In the "reduce" phase, all the contributions with the same key, i.e. with the same hash function (a shard!) are processed together. Therefore, affinity assessment and clusterization are both performed during the "reduce" phase.

3 Future Work

3.1 Training and evaluation

We have developed evaluation tools for measuring precision and accuracy of the framework output for a range of parameters. We are currently implementing a supervised learning algorithm (AdaBoost [6]) to automatically calibrate the weights associated with atomic affinities. However, we need an authority file to compare the results produced by our framework with the reality. Thanks to our

co-operation with Zentralblatt MATH, we have recently obtained a high-quality authority file for the purpose of training and evaluating our framework.

3.2 Attributes from motifs

Another direction of framework extension is automatic feature generation, including weights. This approach is based on search for graph sequences with Apriori [1] algorithm. One of stages in Apriori algorithm is check of support and confidence coefficient, which can be treated as weight. A closer look at the support coefficient helps us determine how discriminative an indicator is. Does it generally give positive (negative) weights, or only to contributions of the same author?

4 Summary

We have presented a framework for author name disambiguation in a collection of documents. The framework has three "degrees of freedom" (cf. Figure 5 on page 31): one may freely choose a hash function, feature functions together with their weights, and a clusterization function.

We were primarily interested in presenting the framework itself: establishing a vocabulary, defining components and their roles, defining workflows, proposing evaluation procedures. Presenting or evaluating a particular instance of the framework was not our goal. Nevertheless, we did hint at possible implementations of the individual components of the framework, and we outlined a plan of evaluation of an implementation that is currently under way at ICM.

The presented solution might be integrated into the European Digital Mathematics Library (EuDML) to handle contributions that are not present in the Zentralblatt MATH authority file.

Acknowledgements. EuDML project is partly financed by the European Union through its Competitiveness and Innovation Programme (Information and Communications Technologies Policy Support Programme, "Open access to scientific information", Grant Agreement No. 250503).

The authors would like to thank Zentralblatt MATH for providing their authority file for the purpose training and evaluation of our name disambiguation module, and the anonymous reviewers for their valuable comments.

References

1. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. In: Proc. 20^{th} Int. Conf. Very Large Data Bases, VLDB. vol. 1215, pp. 487–499. Citeseer (1994).
2. Dean, J., Ghemawat, S.: MapReduce: Simplified Data Processing on Large Clusters. Communications of the ACM 51(1), 1–13 (2004).

3. Galvez, C., Moya-Anegón, F.: Approximate personal name-matching through finite-state graphs. Journal of the American Society for Information Science and Technology 58(13), 1960–1976 (Nov 2007).
4. Han, H., Giles, L., Zha, H., Li, C., Tsioutsiouliklis, K.: Two supervised learning approaches for name disambiguation in author citations. Proceedings of the 2004 joint ACM/IEEE conference on Digital libraries – JCDL '04, p. 296 (2004).
5. Han, H., Zha, H., Giles, C.L.: Name disambiguation in author citations using a K-way spectral clustering method. In: JCDL '05: Proceedings of the 5th ACM/IEEE-CS joint conference on Digital libraries. pp. 334–343. ACM, New York, NY, USA (2005).
6. Hastie, T., Tibshirani, R., Friedman, J.: Elements of Statistical Learning. Springer (2009).
7. Kang, I., Na, S., Lee, S., Jung, H., Kim, P., Sung, W., Lee, J.: On co-authorship for author disambiguation. Information Processing & Management 45(1), 84–97 (Jan 2009).
8. Mann, G. S., Yarowsky, D.: Unsupervised personal name disambiguation. In: Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003. pp. 33–40. Association for Computational Linguistics, Morristown, NJ, USA (2003).
9. Manning, C., D., Raghavan P., Schütze, H.: Introduction to Information Retrieval. (2008).
10. Pavelec, D., Oliveira, L. S., Justino, E., Nobre Neto, F. D., Batista, L. V.: *Compression and stylometry for author identification.* 2009 International Joint Conference on Neural Networks, pp. 2445–2450 (Jun 2009).
11. Pedersen, T., Kulkarni, A., Angheluta, R., Kozareva, Z., Solorio, T.: An unsupervised language independent method of name discrimination using second order co-occurrence features. pp. 208–222 (2006).
12. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: Arnetminer: Extraction and mining of academic social networks. In: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 990–998. ACM (2008).
13. Torvik, V. I., Smalheiser, N. R.: Author name disambiguation in MEDLINE. ACM Transactions on Knowledge Discovery from Data 3(3), 1–29 (Jul 2009).

Workflow of Metadata Extraction from Retro-Born Digital Documents

Dominika Tkaczyk and Łukasz Bolikowski

Interdisciplinary Centre for Mathematical and Computational Modelling
University of Warsaw, ul. Pawińskiego 5A blok D, 02-106 Warsaw, Poland

Abstract. In this work-in-progress report we propose a workflow for metadata extraction from articles in a digital form. We decompose the problem into clearly defined sub-tasks and outline possible implementations of the sub-tasks. We report the progress of implementation and tests, and state future work.

Keywords: metadata extraction, page segmentation, zone classification, Hidden Markov Model

1 Introduction

Whenever a digital library acquires a document without metadata, or with metadata of poor quality, there is a need for extracting metadata from the content at hand. In this paper we focus on extracting metadata of scientific articles. Our goal is to extract as much information as possible, including: title, authors, affiliations, abstract, parsed bibliographic references, journal, volume, issue, pages, and year of publication. At this stage, we are not interested in inferring missing information based on the text of the document, such as: language of the document, keywords or categories (unless they are explicitly listed in the front matter).

The problem of extracting metadata and content from a document is well-studied in the literature. We shall follow the nomenclature presented by Sojka [10]. Older approaches assume that an image of a document is available on input, and execute full digitisation from bitmap image. This was a reasonable assumption in the past, when documents were scanned and retro-digitised. For example, the Medical Article Records System (MARS) [4] works on document scans in the form of TIFF images.

Nowadays, we see more and more retro-born-digital documents, and there is no need to recognise individual characters. This difference has an impact on both the workflow and the performance of metadata extraction process. For example, Cui and Chen [5] employ a Hidden Markov Model to extract metadata from PDF documents, while page segmentation and text extraction is done by Pdftohtml, a third-party open-source tool. Marinai [7] uses JPedal package for extracting characters from PDF, performs rule-based page segmentation, and employs neural classifier for zone classification.

Petr Sojka, Thierry Bouche (editors): DML 2011, Towards a Digital Mathematics Library, pp. 39–44.
 ISBN 978-80-210-5542-1

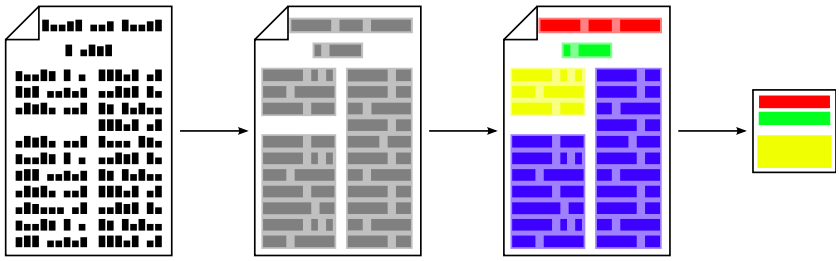


Fig. 1. From a PDF file to a metadata record. The leftmost document contains characters with bounding boxes, as extracted from the PDF file. After page segmentation, the characters are clustered into words, lines and zones, as shown in the second document. Next, the zones are classified as title, authors, affiliations, abstract, body, page number, etc. A metadata record (first to the right) is built based on the labelled zones.

2 Metadata Extraction Workflow

In this section we describe all steps of the metadata extraction process. The process begins with an electronic document (our current implementation supports only PDF format) and its output is the metadata extracted from it. There are three main stages of the process:

1. Building a tree structure that stores the document's content, described in subsections 2.1 and 2.2.
2. Analysing and enhancing the document's content based on its structure, described in subsections 2.3, 2.4 and 2.5.
3. Generating the document's metadata based on enhanced content, described in subsection 2.6.

2.1 Character extraction

The purpose of the first step of the process is to build an initial, flat structure storing the document's content. During character extraction an electronic document is processed. The output of character extraction is a list of document's pages, each of which contains a set of individual characters with their bounding boxes.

Our current implementation of character extraction is based on iText library [1] and is able to process documents in PDF format. iText is used to iterate over PDF's text-showing operators found in the document. During the iteration, we extract individual characters, their position on page, their width and height, and gather them to build initial document's structure. It is worth to notice that the character boxes we extract are in fact not the smallest rectangles enclosing characters, as iText does not provide such accurate information. In particular, all characters of the same font and size will have the same height. Moreover, extracted character widths can differ from the exact values depending on characters and fonts used.

Apart from iText, we also considered using another open source PDF library PDFBox [3]. Our final choice was iText, due to its convenient and well organized API and satisfying character extraction results.

2.2 Page segmentation

In this step we group individual characters into words, lines and zones to build a tree structure representing document's content. After page segmentation the document consists of pages, each page consists of zones, each zone consists of text lines, each line consists of words, and finally each word consists of individual characters. All zones, lines and words can be described by their content, position on the page, width and height.

Our first implementation of page segmentation was a top-down approach based on X-Y cut algorithm [8]. In this solution, the document's page is recursively divided into rectangular blocks by horizontal or vertical cuts.

In the future we plan to replace the first solution with an implementation of a bottom-up Docstrum algorithm [9]. In this approach, the distances and angles between nearest-neighbour pairs of individual characters on the page are analysed, which allows to estimate the text line orientation angle, and also in-line and between-line spacing. Based on these information, we can group individual characters into words and lines, and finally group lines into zones.

In contrast to X-Y cut, Docstrum is independent from text line orientation and text spacing used in the document. Thus, we expect to get better results from it.

2.3 Zone classification

Different zones can have different meaning: a zone can represent document's author, title, abstract, etc. To classify zones means to associate them with labels from a predefined label set. Labels we use in our zone classifier are: *abstract*, *affiliation*, *author*, *body*, *footer*, *header* and *title*. Our implementation of the classifier is based on a Hidden Markov Model [6] with all probability information obtained from a training set.

The classifier processes sequences of all zones of a page sorted accordingly to their position on the page. Correct labels of zones are unknown, but each zone can be described by a vector of features calculated from zone's content, position and dimensions. We treat such sequences of zones with unknown labels as sequences of hidden states in a Hidden Markov Model. The vectors of features are messages emited in every state. We use Viterbi algorithm to calculate the most probable states (labels) of a sequence of objects (zones) based on emitted messages (feature vectors).

To allow the Viterbi algorithm to perform its task, we have to calculate initial and transition state probability, and be able to calculate emission probability for every feature vector. All probability information needed is obtained from a training set. Each training element has a form of a sequence of zones with known labels and vectors of features. Initial and transition probability can

be calculated directly from the training set. To be able to calculate emission probability for every possible feature vector, we build a decision tree based on feature vectors of all training elements. The emission probability of a given feature vector can be calculated from those training elements, that are classified in the same leaf of the decision tree as given feature vector.

Our classifier uses 37 features to describe the document's zones. Some of the features refer to the zone's bounding box and its position on the page, e.g. zone's absolute and relative dimensions, horizontal and vertical position. We also used features related to the inner structure of the zone, e.g. absolute and relative number of text lines and words, mean text line height, width and position. Finally, some of the features are based on the text of the zone, e.g. the number of characters, digits, lowercase/uppercase letters, punctuation marks, etc.

We used documents obtained from MARG repository [2] for both training and test sets. The training set we used consisted of 317 elements and 1,379 zones. The tests we have performed on 1,236 documents with 5,359 zones gave the accuracy rate 95.5%.

It is worth to notice that apart from Hidden Markov Models there are many other supervised learning approaches available, e.g. Conditional Random Fields [11] or simple rule-based classifiers. We chose Hidden Markov Models approach because of relatively simple algorithms needed, high maintainability of the solution and good quality of results.

2.4 Bibliographic references extraction

Extracting bibliographic references from a document is a first stage of references processing, which also includes references parsing and matching.

Our current bibliographic references extractor processes only the text content of a document and is based on simple character frequency heuristic. First, we select lines with a sufficiently high frequency of digits and punctuation. Next, we remove isolated lines and fill the gaps. Finally, extracted lines are concatenated and references split.

In the future we plan to implement a better bibliographic reference extractor, that makes use not only of the text content, but also of the document's tree structure constructed in previous steps. We believe that taking into account text positioning parameters, such as lines' positions in the document, the between-line spacing or line indentation will result in better bibliographic references lines detecting and grouping.

2.5 Bibliographic references parsing

To make further bibliographic references analysis (e.g. matching references with documents) possible, we have to parse extracted references and identify their fragments containing author, title, journal, etc. Our implementation of bibliographic references parser is based on a Hidden Markov Model with all probability information obtained from a training set. Labels we use to tag

fragments of references are: *author*, *title*, *journal*, *volume*, *series*, *number*, *publisher*, *location*, *edition*, *pages*, *url*, *year* and *content*.

A bibliographic reference can be represented as a sequence of tokens with unknown labels. Each token can be described by a vector of features calculated from its content. The sequence of tokens can be treated as a sequence of hidden states in a Hidden Markov Model. The vectors of features are messages emitted in every state. We use Viterbi algorithm to calculate the most probable sequence of states (token labels) based on emitted messages (feature vectors).

As in the case of zone classifier, all probability information needed by Viterbi algorithm is obtained from a training set. Training set consists of bibliographic references with tagged tokens. Initial and transition probability can be calculated directly from the training set, emission probability is calculated based on the decision tree constructed from training elements' feature vectors.

Our parser uses 48 features to describe citation's tokens. Some of the features measure relative number of particular character type, e.g. digits or uppercase letters. Other features check whether the token is a particular character (a comma, a dot, a quote, etc.) or a particular word ("and", "http", "vol", etc.). We also use features that are based on dictionaries built from the training set, e.g. a dictionary of cities or words commonly occuring in the journal title.

Both training and test sets were obtained from digital collections NUMDAM and CEDRAM. The training set we used consisted of 2,318 bibliographic references. The tests we performed on 2,532 references gave the accuracy rate 85.8% of correctly identified fragments of bibliographic references.

Apart from Hidden Markov Models, there are other supervised learning approaches available, e.g. Conditional Random Fields or template matching using regular expressions. As in the case of the zone classifier, we chose Hidden Markov Model approach due to its simplicity, flexibility and good quality results.

2.6 Metadata extraction

In the final step the metadata is extracted based on labelled zones, document structure and content. The final step has not been implemented yet.

3 Current Status and Future Work

So far we have implemented and tested most of the metadata extraction steps described in the previous section. Our implementation of character extraction is based on iText library, for page segmentation we currently use X-Y cut algorithm and zone classifier is based on a Hidden Markov Model. We have also implemented bibliographic references processing: references extractor uses simple character frequency heuristic and parser is based on a Hidden Markov Model.

However, the metadata extraction process still needs some work. In the near future we plan to replace current implementation of the page segmenter

with a version based on Docstrum algorithm. We are also going to implement a better bibliographic reference extractor, that makes use not only of the text content, but also of the document's tree structure. Also the final step of the process, the metadata extraction, is still to be implemented.

So far for testing page segmentation and zone classification implementations we have been using documents obtained from MARG repository. Unfortunately, MARG repository has its drawbacks: it contains only first pages of the documents and only a subset of all zones is included in the document's structure. In the future we plan to semi-manually construct a better test set. We hope that it will make our implementations and test results more reliable.

4 Summary

We have proposed a workflow for metadata extraction which is especially suitable for retro-born-digital documents, while still applicable in the case of full digitisation from bitmap images. We have reported our current implementation and testing efforts and stated future work.

Acknowledgements. This work is partly financed by the EuDML project, which is in turn partly financed by the European Union through its Competitiveness and Innovation Programme (Information and Communications Technologies Policy Support Programme, "Open access to scientific information", Grant Agreement No. 250503). We would also like to thank the anonymous reviewers for their insightful comments.

References

1. iText, `http://itextpdf.com/`
2. MARG, `http://marg.nlm.nih.gov/`
3. PDFBox, `http://pdfbox.apache.org/`
4. Automating the production of bibliographic records for MEDLINE. Tech. rep. (2001).
5. Cui, B., Chen, X.: An improved hidden Markov model for literature metadata extraction. Advanced Intelligent Computing Theories and Applications. pp. 205–212 (2010).
6. Hetzner, E.: A simple method for citation metadata extraction using Hidden Markov Models. In: JCDL '08: Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries. pp. 280–284. ACM, New York, NY, USA (2008).
7. Marinai, S.: Metadata Extraction from PDF Papers for Digital Library Ingest. 10th International Conference on Document Analysis and Recognition. pp. 251–255 (2009).
8. Nagy, G., Seth, S., Viswanathan, M.: A prototype document image analysis system for technical journals. Computer 25(7), 10–22 (1992).
9. O'Gorman, L.: The document spectrum for page layout analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(11), 1162–1173 (1993).
10. Sojka, P.: An Experience with Building Digital Open Access Repository DML-CZ. In: Proceedings of CASLIN 2009. pp. 74–78 (2009).
11. Sutton, C., McCallum, A.: An Introduction to Conditional Random Fields for Relational Learning (2006).

The EuDML Metadata Schema
Version 1.0

Thierry Bouche[1], Claude Goutorbe[1], Jean-Paul Jorda[2], and Michael Jost[3]

[1] Cellule Mathdoc (UMS 5638)
Université Joseph-Fourier, (Grenoble 1)
B.P. 74, 384 02 Saint-Martin d'Hères, France
thierry.bouche@ujf-grenoble.fr, claude.goutorbe@ujf-grenoble.fr
[2] EDP Sciences
17, avenue du Hoggar
B.P. 112, 919 40 Les Ulis cedex A, France
jean-paul.jorda@edpsciences.org
[3] FIZ Karlsruhe, Zentralblatt MATH
Franklinstr. 11, D-105 87 Berlin, Germany
jo@zentralblatt-math.org

Abstract. After an extensive study of the metadata policy of each of its content partners, the EuDML project evaluated many different strategies and existing schemas that could store every detail faithfully, and yet reserve room for the enhancements foreseen in the project's work plan. The framework provided by the so-called NLM Journal Archiving and Interchange Tag Suite was selected as best readily available approximation of our needs. Some modifications of it have been endorsed by the project, defining the first version of our interchange schema for heavily math-based content.

Keywords: EuDML project, metadata schema, XML, interoperability

1 Introduction

1.1 The EuDML project

The EuDML project aims to design and build a collaborative digital library service that will collate the mathematical content brought by 11 of its partners and make it accessible from a single platform, tightly integrated with relevant infrastructures such as Zentralblatt MATH. As such, it is the first attempt toward a large-scale international implementation of a Digital Mathematics Library (DML), and is expected to pave the way towards a truly inclusive and global DML. In this direction, we will try to accommodate new associated partners and to interoperate with relevant infrastructures in the fields of scientific information. Interoperability needs published and documented standards, which is one of the tasks undertaken by EuDML's third work package.

Petr Sojka, Thierry Bouche (editors): DML 2011, Towards a Digital Mathematics Library, pp. 45–61.
 ISBN 978-80-210-5542-1

1.2 Why a EuDML schema?

A public well-specified EuDML schema is needed:

1. Because content providers need to know *which metadata* they should expose to EuDML harvesters, which details and granularity is *required* (obligatory metadata), *appreciated* (fundamental metadata), and which further enhancements are expected to *provide added value* to their cooperation with the EuDML project (supplemental metadata).
   Thanks to the specification, they can see what information is wanted by EuDML. They can expose their holdings directly encoded in this way, or they can expose their "richest" format which contains all the relevant information to the extent possible.
2. Because the search engine has to know *where to look for*, e.g., an author when a user searches for an author. (Search for "Hilbert" as author must produce quite different results than searching for "Hilbert" as free text or within the whole record of a given item, think of "Hilbert space", "Hilbert transform", which can appear as key words, in titles, cited titles, etc.). The schema serves as a pivot norm for various provider formats and schemas.
3. Because the search engine has to know *what to display* when showing search result lists (Author, Title, bibliographic source, link to full text...) as well as *how to display* complex structures (multilingual information, reference lists, mathematical formulae...). It has thus to know *how they are encoded* in order to present them in the best shape for a given user in a given environment.
4. Because metadata enhancers toolsets have to work on a *defined basis* so that they know what they start from and where they store their results.
   Some examples: reference citation matching, duplicate detection and records merging or metadata enrichment from various sources. More specifically to our corpus: a metadata enhancer should be able to scan an existing metadata record to find, for instance, a reference to a formula, generate a new format for this formula (e.g., by OCR or translation to MathML from LaTeX code), and to store the resulting object in parallel to the pre-existing one(s).
5. Finally, EuDML must be able to *export* its content in a predictable, reusable way, for safety backups, interoperability, and to enable content providers to retrieve the EuDML-enhanced metadata for their collection of items, in order to improve their local collections to a higher level of quality. They need to know how this new information will be encoded so that they can use it. This cannot be done with their internal format, as many added-value elements will be beyond the scope of such format.

1.3 A metadata model for EuDML

This paper presents the first specification of the EuDML metadata schema, which is already used by the current prototype of the system.

Its main goals are to:

– provide details on the structure, granularity, and encodings that should be supported by content partners (see § 2);

- provide incentive to more content providers to contribute their best metadata to the EuDML central metadata repository using adequate schemas and interoperability devices (see § 2.5 and § 3.5);
- present the NLM Journal Archiving and Interchange Tag Suite (JATS) as the general frame adopted to encode and exchange the EuDML metadata and list the changes needed in order to support all content types contributed to EuDML (see § 3);
- introduce a set of best practices to ensure perfect understanding of tagging practice among EuDML partners (see § 4);
- outline directions for improvements (see § 5).

2 Methodology & Definitions

This section describes the principles, methods and notions that are used to define the EuDML metadata schema in the next section.

2.1 Scope of this work

Metadata is usually defined as "data about the data", so in order to target our work on metadata, it is important to make explicit what is the central data we expect to describe with our metadata.

EuDML being the digital metaphor of a mathematics oriented professional library, important concepts that will necessarily be handled in the system, and thus need some internal metadata schema, are: publication (publication containers such as journals or books, as well as individual contributions aka items), person (contributors, and users aka patrons), legal person (person's affiliations, publishers, etc.), user community, user annotation.

However, given the nature of the EuDML central repository, which will be assembled by aggregating content from a number of partner's catalogues, we are lead to single out the individual mathematical works in the library as our main relevant data, and hence to focus on a metadata schema designed to bear all relevant information that can be gathered, consolidated or generated for each integrated full text.

We thus consider publication containers, persons and their affiliations, and publishers, as peripheral information attached to some full text (yet supporting the ability to link to an authority list of such). We also discard all registered users information as well as their possible annotations in this iteration of our work, for these are considered private concepts to the EuDML system, thus inappropriate in a static, exportable representation of the library's content metadata.

2.2 The EuDML item

The central object in EuDML, used as the unit of delivery and thus as the pivot for the metadata schema, is an *item*.

EuDML defined relevant logical units that can be delivered in the context of EuDML in the following way:

"An item is a self-contained mathematical text which has been scientifically validated and formally published."

We warn readers from the library community that this definition is incompatible with that used in FRBR [11] terminology, where it would be rather called *manifestation*.

Loosely put, an item is the kind of mathematical content that would be reviewed in *Zentralblatt MATH* or *Mathematical Reviews* , so the relevance of this concept is quite consensual in the mathematical community. It is the object an unambiguous citation in a scientific article would point to, thus the importance of this concept for a reference library.

Two different editions of a book would be different items, but not a new print; two digitisations of the same article would be related to a single item, but the reprinting of that article in its author's collected works would yield a new item (hopefully related to the previous one). We must be very cautious with mathematical references that, although the same "ideal work" can be manifested through various channels such as a conference, an abstract, a full paper, or a monograph, it is not possible in subsequent works to refer uniquely to them collectively, as the actual details contained in each manifestation could differ enough to make the reference ambiguous. Even a solid abstract reference such as "the Hahn-Banach theorem" might be stated with quite varying hypothesis and conclusions depending on the context where it is manifested.

As the main focus of the EuDML project is to ease discovery, access, use and exchange of mathematical items, *the EuDML item* is thus the primary entity type described by the schema.

The identification of an item, as a formally published text, essentially requires bibliographic data which describes *where* and *by whom* it was published and depends on the *type* of publication (journal article, book, etc.).

For this version of the schema, we explicitly support the following publication types, which are logical subclasses of the generic "item" class:

- a multivolume work;
- a book, namely
  - a single volume from a multivolume work,
  - a monograph (which might be a doctoral dissertation, a memoir...),
  - an edited book (a book that contains chapters or articles that have been written by different authors and collated by scientific editors, which might be a conference proceedings volume);
- a part of a book such as a chapter, or a contribution in a proceedings volume;
- a journal article.

2.3 Out of scope functionality

The following do not constitute requirements on EuDML services and are thus not in the scope of a EuDML metadata schema:

- Handle material that is not considered as having been persistently and formally published (e.g. preprints, personal web pages...).
- Special provisions for papers not generally accessible online (e.g. on paper only, in house access only, library catalogue...).
- Version control for documents, as EuDML only considers works in published final form.
- Complicated author/contributor structures for documents, as this is of no significance in math publishing. We won't try to record authors' contribution weights, ordered authors' list where either the first or the last name has more significance, etc.
- Description of access embargo periods (moving wall) and other licensing, access barriers, digital rights management issues, since EuDML follows an eventual open access policy and leaves those issues under full control of the respective content (full text) providers.

2.4 Analysis of the EuDML metadata requirements

Metadata exists to support the functionalities expected from the system. In this section we describe the functional aspects of a Digital Mathematics Library (DML) that we intend to provide:

- Uniquely identify an item not only within EuDML, but across the whole mathematical literature.
- Discovery of published items by
  - fielded search on various attributes such as author names, titles, publication year, subject, abstracts, journal title, key words,
  - browsing collections by selecting a starting point such as a given journal name, mathematical classification code, author name,
  - sorting and filtering search or browse results,
  - automated reference matching to help external resources turn their citations into links to EuDML items.
- Retrieving a specific item through a known identifier such as a DOI, URI or other unique identifier.
- Assert the relevance to the user of a given item through the display of attributes such as subject, abstract, language, and citations to and from that item.
- Display and indexing of attributes in multiple languages or transliteration systems.
- Interlinking as a powerful access tool to mathematical resources. Examples of this consist of links to reviews in the major reviewing databases (Jahrbuch, Zentralblatt MATH, MathSciNet), and links to and from citations from subsequent works.
- Linking to other material such as user provided annotations, author identification services.
- Display of mathematical formulae in various formats based on the user's choice or capabilities (e.g. MathML, TEX, graphics, speech synthesis).

Besides the end user oriented functionalities, the schema should also serve as an exchange model.

2.5 Quality insurance on metadata

From the above requirements analysis, we derived the following functional definitions, which help identify more objectively whether a given item's metadata is eligible to support minimal digital library operation, standard full-featured operation, or advanced operation tweaked for EuDML math-specific content.

Obligatory metadata We define obligatory metadata as the bare minimum of metadata information that is requested from EuDML data providers. This is not exactly a functional category but rather a policy requirement.

Obligatory metadata is the required minimum of metadata in order to unambiguously identify and handle a relevant mathematical publication in the scope of EuDML: Item type, authors, original title, bibliographic reference for this publication with enough structure so as to enable collection's browsing, unique identifier, URL of full text.

Fundamental metadata Fundamental metadata is what satisfies the functional requirements for browsing, searching and reference matching over the collections at item level. It enables basic digital library interaction with the EuDML corpus.

The term fundamental was chosen so that it is clear which information is needed to provide the fundamental functionality expected by typical users. It is a qualitative superset of obligatory metadata.

If this information is relevant to the item described, then it must be present in the metadata. If it is absent from provider's original metadata, then our enhancing tool set must provide a solution in order to enable this publication in EuDML.

It contains obligatory metadata (see above) as well as standard optional information (abstract, key words, main language) that should be there, or generated by the project.

Supplemental metadata Beyond fundamental metadata, this is additional metadata that should be stored, generated, and exploited within EuDML.

Supplemental metadata is whatever goes beyond fundamental metadata (e.g. relations to subject ontologies, authority lists, MR/ZM IDs, multilingual, multiscript, bibliographies/references, interlinking, math handling...), yet has relevance to the EuDML's corpus specificities and EuDML system functionalities.

3 The EuDML Metadata Schema

This section is about how EuDML metadata will be encoded and physically appear or be transported in certain given scenarios (such as during metadata harvest from EuDML data providers, or exposition of EuDML metadata

to aggregators, e.g. Europeana, or for a "snapshot" or "dump" of EuDML contents).

As we do not want to reinvent the wheel, a quick survey of existing XML encodings was conducted, paying special attention to the following requirements:

- mathematical formulae should be supported in a variety of formats, including MathML;
- rich text should be allowed where applicable, in other words the encoding used must account for a number of basic formatting elements such as typographical attributes;
- the description of reference lists (bibliographies) should be taken into account, as they are an essential tool for researchers;
- using a recognised and widely deployed standard would be a bonus. However, as we do not expect an existing XML document type definition or schema to be able to describe our data "out of the box", it should be easily customisable.

3.1 Review of evaluated metadata encodings

We evaluated the following schemas which all provide some partial solution to our query:

EULER Euler FP5 project metadata, which was developed for cataloguing (non-digital) resources existing in various European libraries [4];

SWAP Scholarly Works Application Profile in qualified Dublin Core, which essentially provides granularity to describe (with raw text metadata) any digital scholarly work (detailed bibliographic description, eprint versioning, validation status) [3] ;

MODS Metadata Object Description Schema from the Library of Congress, which is pretty much an interchange format for multimedia library catalogues [7];

DML-DC Euclid/NUMDAM/GDZ recommendation on presenting DML metadata in simple Dublin Core, which was an attempt to qualify simple Dublin Core for making metadata interchange more useful between DMLs by URI-like prefixing repeated elements, as well as some best practices recommendations for mathematical expression encoding in titles and abstract [2];

MLAP Mathematical Literature Application Profile for Dublin Core by David Ruddy, which is a relatively strict yet very generic schema for interchanging precise bibliographic records of scholarly works [10]. Besides the fact that mathematicians are eager to exchange this kind of information in order to build larger DMLs and further the interlinking of existing DMLs, this proposal has nothing specific to mathematical content;

JATS NCBI/NLM Journal Archiving Tag Suite, which was created with the primary intent of providing a common format in which publishers and archives could exchange journal content [9].

While Dublin Core metadata is nowadays a central device for wide interoperability, especially for enhancing visibility of heterogeneous collections, it was felt that DC based formats would be useful for exporting EuDML metadata but not for storing the consolidated master with all information and additions foreseen in the project's work plan. In fact, DC is so generic that, among its 15 elements, few are relevant to a digital library project such as EuDML, and a lot of structure has to be added to qualify and organise information we would expect from each of the principal elements. This is what application profiles such as EULER, SWAP, DML-DC and MLAP are aimed at, each of these developed with a specific aspect of literature interchange in mind. MODS is a more constrained framework that can be used, together with METS, in order to describe a precise bibliographic record of a catalogued object, as well as its physical description — no room exists, apart from using relations to external objects conforming to some other format, for encoding parts of an item's textual content like bibliographies. However, none of these provide support for mathematical knowledge encoded as such: the mathematically oriented standards just favour TEX notation as it can be embedded into any XML file as text modulo some escaping.

Inera Inc. provides some introduction to the NLM Journal Archiving Tag Suite (JATS in the following) [5]:

> The NLM Journal Archiving and Interchange DTD Suite, co-authored by Inera Inc., Mulberry Technologies, and NCBI, is the de facto standard full-text DTD for scholarly publishing.
> Since the DTD was first released in April 2003, it has been (for the scholarly publishing world) rapidly adopted. Whereas ISO 12083 has never achieved broad acceptance, the NLM DTD has already been adopted by hundreds of journals (probably north of 500) worldwide. Many small and medium-sized publishers have adopted the NLM DTD, and a number of larger publishers are preparing to deliver content according to the NLM DTD when asked. Most of the major journal publishing compositors and service suppliers are up to speed on the DTD and happy to deliver content tagged with it.
> The NLM DTD has also proven popular with aggregators. It is the "house" DTD of Atypon Systems and the recommended DTD for full-text content at Ingenta and Highwire Press. And, of course, NLM uses it for PubMed Central.
> The NLM DTD has been no less popular with libraries. In a joint press release, the British Library and the Library of Congress announced that they would support the NLM DTD as their archiving standard for electronic content. It has also been adopted by Portico (a major Mellon-funded archive effort).

Complemented by Mulberry Technologies, Inc. [8]:

> The Journal Archiving and Interchange Tag Suite (also called the NLM DTD although it is available in DTD, XSD, and RNG forms) provides a common XML format for preserving the intellectual content of journal articles, independent of the form in which that content was originally delivered. The Tag Suite consists of Tag Sets for Archiving, Publishing, and Authoring journal article content and a Tag Set for Books and book material. The Tag Sets have been widely adopted by archives, libraries, and publishers and are supported by many data conversion vendors and XML tools.

NISO (the National Information Standards Organization) is now working to make the JATS into a NISO standard.

As JATS was already used internally by one of our partners (EDP Sciences), and proved to have room to store faithfully all of the metadata encountered while reviewing the EuDML content to be integrated, and moreover provided standard structures for most of the new elements foreseen in the work plan (full text encoding, native support for MathML and alternative versions of formulae, notably), it was an easy task to select it as best candidate for our purpose. It is a trivial task to derive most DC based metadata from carefully organised JATS files (while the converse would require a JATS application profile in Dublin Core).

In summary, here are some decisive features that highlight NLM JATS as the best available framework to host EuDML metadata:

- It has been adopted as the internal format of one of our partners (EDP Sciences), and is already vastly deployed as an interchange format by many scientific publishers because of their interoperability with PubMed Central, JSTOR, or Portico. Its wide deployment and large user community makes it a good reference model for outer interoperability.
- It is highly customisable and meant for customisation (nevertheless, we decided to keep minimal any deviation from the standard schemas in order to maximise wider interoperability).
- It has room to store any kind of scholarly content up to the full text itself, and to store parallel versions of the same content encoded differently (which is crucial for our enhancing workflow).
- Last but not least, it is MathML-ready (yet allowing storage of alternative representation of the mathematical content).

JATS provides three DTDs that we will adapt for describing our three main content types:

- **The Journal Archiving and Interchange Tag Set** implements article3.dtd for journal articles (cf. `http://dtd.nlm.nih.gov/archiving/`)
- **The NCBI Book Tag Set** implements book3.dtd for books and bookcollection3.dtd for collections of books (cf. `http://dtd.nlm.nih.gov/book/`)

Although JATS can be easily customised to fit any special need [1], we will try to adhere to its readily published DTDs to the largest extent possible, specifying best practices recommendations in order to attain maximum compatibility among EuDML partners, and reliability of exchanged metadata with third parties.

3.2 The EuDML schema, initial version, based on JATS

To assess the suitability of JATS to our needs more objectively, the above analysis was completed by an attempt to transform large samples of available EuDML metadata as contributed by their providers to one of the JATS DTDs.

From this experience, we concluded that JATS needed more work to suit our needs, in two opposite directions:

1. The item types currently supported by JATS published DTDs are: journal article, book, and book collection (which is defined as "a series of books related in some manner"). While the EuDML "first class citizens" are more diversified, cf. the supported item types listed in § 2.2.
   We thus decided to organise all our content in three major *containers*: journal article, single book, and multiple volume books. The two first item types required very minor extensions to existing JATS schemas for article and book (such as allowing a conference description in a book metadata for conference proceedings that are not published in a journal). As the last one doesn't fit perfectly the JATS collection model, we created a new one, called mbook, which has the metadata of a book, but whose content is a list of separate books as in JATS collection.
   These slight deviations from the three standard JATS schemas form the initial version of our EuDML schema specification: see § 3.3.
2. A drawback of JATS versatility is that it doesn't impose strict constraints on metadata encoding, and often allows for different ways to encode the same information. For efficiency of metadata interchange and exploitation in EuDML, we felt that we needed guidelines so as to have a common encoding practice and understanding among all EuDML partners and content providers. The initial version is outlined here, § 4, and available on our web site (see `http://www.eudml.eu/eudml-metadata-specification`). Further revisions will be released periodically based on feedback from other activities within the project.

3.3 EuDML metadata specification v. 1.0

The EuDML metadata schema version 1.0 as defined by deliverable D3.2 [6] is implemented in three DTDs providing the 3 root elements holding XML metadata for three major types of items, namely journal articles, books, and multivolume works. A consequence of this choice is that some book parts (typically individual articles in a proceedings volumes), while being "first class citizens" in our abstract model, are described and exchanged within the whole book they belong to. This is a decision on the formal way used to store and transport our items' metadata, pragmatically rooted in the existing JATS DTDs, and in the fact that bibliographical data cannot be structured the same way for these different items. It is not intended to restrict in any way the items' records are exposed to or navigated by end users.

Journal articles are described with a minimal extension of the Journal Archiving and Interchange Tag Set version 3.0 with root element <article>:
- the @xml:lang attribute is allowed for the <issue-title> element.

Books are described with a minor extension of the Book Tag Set version 3.0 with root element <book>:
- a child <conference> element (as in <article-meta>) is allowed in <book-meta>; this element is needed to describe conference proceedings volumes;

- a child <book-part-id> with attribute @pub-id-type is allowed in <book-part-meta>; this element is used to preserve item-level identifiers, when parts of a book are EuDML items;
- the @pub-id-type attribute to <book-id> and <book-part-id> can have values beyond a restricted list; it is used in particular to identify the authority who assigned the identifier.

Multivolume works are described by a new root element <mbook>. Multivolume works' metadata is identical to <book> metadata, with the addition of references to individual constituents (volumes). The element <book-meta> is replaced by <mbook-meta> with same structure, except:

- a child <mbook-list> element is required in <mbook-meta>. It is a container for individual volumes, as in JATS collection DTD;
- each component volume reference is captured by an <mbook-volume> element (child of <mbook-list>), with the following children:
  - <title>: the title of the volume,
  - any number of <book-id> and <ext-link> elements.

  While the EuDML internal machinery only needs <book-id>s in order to implement the multivolume work/individual book relationship, the <title> and <ext-link> elements should be useful to external applications for display and access purposes. Each individual volume in a multivolume work is encoded with the Book DTD.

3.4 Conversion summary

While developing this work, we converted large sample metadata sets from a number of partners to plain NLM DTD and inspected where conversion was difficult to achieve, when doubts or choices had to be made, when the target structure did not accommodate the source structure, etc. When we faced the necessity to choose between different structures offered by NLM DTD, we took note and started an open discussion within our working group which ended up in a number of best practices recommendations. When we found metadata that could not be faithfully stored in the existing NLM DTDs, we took note of this for further processing. Finally, we took the design decision to adhere as closely as possible to the existing NLM DTDs, implemented the small modifications that were required to faithfully store all encountered item types, and left aside some more modifications, waiting for more feedback from the actual implementation of the project to be realised in the forthcoming months.

The following table summarises the number of item types from various EuDML collections which were successfully converted in order to evaluate our results presented here.

Collection	**EuDML metadata (Schema)**	**Notes**
Gallica-Math	2,081 (**article**)	converted from internal XML with LaTeX

Collection	EuDML metadata (Schema)	Notes
CEDRAM	1,868 (**article**)	converted from internal XML with MathML
NUMDAM	43,944 (**article**)	converted from internal XML
DML-E	6,401 (**article**)	converted from SQL database
EDPS journals	200 (**article**)	slight variation of native EDP schema to obey best practices
ElibM	25,453 (**article**)	converted from internal XML
BulDML	436 (**article**)	converted from DC XML
DML-CZ	26,476 (**article**), 132 (**book**)	converted from internal XML
GDZ Mathematica	53,396 (**article**), 2,298 (**book**), 296 (**mbook**)	converted from METS XML
RusDML	16,486 (**article**)	converted from METS XML
Port. Mat.	1,347 (**article**)	converted from TEL XML
All	**180,814 records**	

In order to get the most out of the contributed metadata, all converted items meet the obligatory requirements, even when the original metadata didn't meet them. For this, we had to heuristically split some unstructured fields into tagged bibliographic references, e.g. We also tagged all LaTeX formulae encountered with NLM superstructure with MathML alternative encoding (see § 4.3). We thus have currently 206,775 tagged formulae in our metadata. They were mostly processed from TeX encoding embedded into a text field from the provider's metadata.

3.5 A note on interoperability

When acquiring metadata from different partners, it was observed that any reasonably structured format is rather easily converted to JATS format. The big drawback with many OAI-PMH servers is that they only serve the mandatory OAI-DC format in such a way that many different metadata elements are stored in the same, repeated Dublin Core element. As a consequence, only heuristics based on order of appearance, or pattern matching on an element's value allows

disambiguating the metadata thus contributed. For instance, <identifier> can be used to transport an ISSN, a textual bibliographic reference, a URL, etc.

Qualified versions of Dublin Core that are modelled on the metadata schema with finest grain available to the content provider allow faithful interchange of metadata. Qualification can be embedded into the value of simple Dublin Core elements as in the DML-DC recommendation or similar qualification using URN-like prefixes, or it can use qualified elements and a documented application profile such as SWAP or MLAP.

As of writing this report, the best scenario for returning EuDML metadata to providers is to use the EuDML schema over OAI-PMH communication channels.

For interoperability and visibility beyond EuDML partners or associated partners, a simple transformation has been developed to represent a subset of EuDML metadata in OAI-DC (compliant with DML-DC) so that general harvesters can manage our metadata. A prototype implementation of this is available to the project partners in a dedicated OAI-PMH server.

4 Best Practices Recommendation for Mapping EuDML Abstract Metadata to the EuDML Schema

A best practices working group for representing EuDML metadata in JATS notation was formed. A set of recommendations has been developed, and has now been tested on all available EuDML items. Complete examples of EuDML XML files obeying these recommendations are available on our web site. We give some examples of the issues tackled below.

The recommendation itself is a work-in-progress, which is available to the project's partners in an internal wiki as a live HTML page they can edit. Its first version has been made accessible in an area of the `www.eudml.eu` website dedicated to developers' resources (`http://www.eudml.eu/eudml-metadata-specification`). Up-to-date documentation is in the process of being made available there for download as well: the specification, the DTDs and possible associated tools.

4.1 Special item types

Proceeding volumes were an interesting case, as they were handled in very different ways by different providers. As many EuDML partners were primarily journal digitisation projects, they had "journalised" proceedings, even when they were published as independent books: modelling a conference series as a journal, each proceedings volume as a journal issue, and each contribution as an article. However, the bibliographic metadata of a conference series publishing its yearly proceedings in a general lecture notes series, e.g., is quite different from that of a journal special issue. Our modifications of JATS standard DTDs are meant to have exactly the same details for each flavour of conference publishing. As soon as the volume holds conference proceedings, the conference details

are placed in the metadata, either in <article-meta> for journal articles, or in <book-meta>. The <conference> element is the same in each case.

To set information about editors (e.g. for a proceedings, when editors replace authors), <contrib> element is used together with @contrib-type attribute set to "editor".

Books, like articles, can have multiple translations of their title in JATS. For some reason, journal issue metadata is somewhat less detailed than for books. In EuDML you can have multiple <issue-title> elements, distinguished by their @xml:lang attributes. In the case of an article published in a special issue, its authors (or editors) should be distinguished from the issue editors: the <contrib> element is used together with @contrib-type attribute set to "issue-editor".

4.2 Identifiers and relations

The item-centric vision of the EuDML imposes to be very careful on identifiers and relations. Each EuDML item *must* have a unique persistent identifier. As it comes from elsewhere, it will also typically come with a number of different identifiers that we should register in order to enable further interoperability.

The item metadata record is something like a linking hub for the given item: it should hold links to various resources attached to the described item, as well as to other related items. Obviously, a relation needs some sort of identifier for the related object as well.

We identified three main classes of relations, for which we recommend to use three different JATS structures:

1. *Primary identifiers* identify an item or a container. They are assigned by the publisher (DOI, PII and specific internal identifier) or by the local DML. They are not necessarily associated with an URL.
   Such identifiers are stored in a dedicated element relative to the item's type (e.g. <article-id>). They must have a @pub-id-type attribute.
2. *Document identifiers* provide links to the different versions of the *content* pertaining to an item or a container on the provider's web site (the PDF version, the full HTML version, etc.).
   These identifiers come in the form of an activable link stored as @xlink:href attribute to the <self-uri> element. The combination of the mime-type and a controlled vocabulary for values allows to predict the nature of the resource the link points to.
3. *External identifiers* are primarily identifiers proceeding from other authorities, such as Zentralblatt MATH, Math. Reviews, CrossRef, which assign IDs to articles, authors, journals or books, or related resources.
   External identifiers must be set using <ext-link>, the value being the identifier itself. An activable link should be stored as @xlink:href attribute when applicable while the @ext-link-type attribute should keep track of the identifying mechanism and authority. Links to related items (including other EuDML items) should use <ext-link> in a similar way.

4.3 Mathematics

Although the MSC reads "Mathematical *subject* classification", and although JATS provides a <subject> element, MSC should be encoded with the <kwd> element inside a <kwd-group> with attribute @kwd-group-type set to the actual scheme: "msc" followed by the year (e.g. "msc2000").

Inline and display mathematical formulae are expressed respectively with <inline-formula> and <disp-formula> elements. Both MathML and (La)TEX version of the same formula can be wrapped up using <alternatives> element. This mechanism will be extended so that other versions (accessible, aural) can be stored in a similar fashion.

It is recommended to attach a unique ID to each formula to ease further processing.

For TEX notation, it is recommended to put compilable code into the <tex-math> element. This means that the switch to math mode should be part of the element's value, and control sequences should be standard (standard meaning: macros defined in plain and LATEX formats, possibly with AMS mathematical extensions). It is important because some environments such as `multline` change the internal grammar of their content while switching to math, and this would be lost. We currently recommend that the content of the <tex-math> element be literal TEX code with two characters (& and <) escaped using standard XML entities (`&` and `<`). Putting a full LATEX source in a CDATA section (as exemplified in JATS documentation) is explicitly disapproved.

For instance

A product of four (p,q)-sections (with $p < q$).

should be encoded the following way.

```
A product of four
 <inline-formula id="d1e4">
   <alternatives>
      <mml:math xmlns:mml="http://www.w3.org/1998/Math/MathML">
         <mml:mrow>
            <mml:mo>(</mml:mo>
            <mml:mi>p</mml:mi>
            <mml:mo>,</mml:mo>
            <mml:mi>q</mml:mi>
            <mml:mo>)</mml:mo>
         </mml:mrow>
      </mml:math>
      <tex-math>$(p,q)$</tex-math>
   </alternatives>
 </inline-formula>-sections (with
 <inline-formula id="d1e20">
   <alternatives>
      <mml:math xmlns:mml="http://www.w3.org/1998/Math/MathML">
         <mml:mrow>
            <mml:mi>p</mml:mi>
```

```
            <mml:mo>&lt;</mml:mo>
            <mml:mi>q</mml:mi>
          </mml:mrow>
        </mml:math>
        <tex-math>$p&lt;q$</tex-math>
    </alternatives>
</inline-formula>).
```

5 Conclusion & Further Work

We have exposed the rationale that led us to base the EuDML schema for descriptive metadata on NLM Journal Archiving and Interchange Tag Suite. We provided the current specification of the schema as a diff to three existing standard JATS DTDs. We also gave some examples of the recommendations we came up with so that design choices allowed by JATS are correctly understood by all partners. This work is assessed by the fact that we could convert all EuDML partner's metadata available to us into this framework.

Now, EuDML is starting to exploit what we have generated so far. A number of tools are developed in order to improve the quantity as well as the quality of the metadata available to the project, they will evolve into various automated workflows. Using these tools, we should be able to get new metadata elements, that may superseed or just add to the items' descriptions. Duplicates, similar, and related items should also be detected. We will thus face the necessity to merge item records from a number of sources, some of them "trusted" (e.g. manual keywords or copy-edited translations), some of them much less so (computed similarity, guessed MSC, automatic translation, OCRed math formulae...). We feel that the current schema is robust enough to store all this information faithfully side by side, while retaining its origin. Indeed, we think that managing this will boil down to adding a number of rules and attribute values to our Best practices. However, we are now expecting feedback from a number of project's activities to assess and refine the work reported here.

Acknowledgements. This work is partly financed by the European Union through its Competitiveness and Innovation Programme (Information and Communications Technologies Policy Support Programme, "Open access to scientific information", Grant Agreement no. 250503).

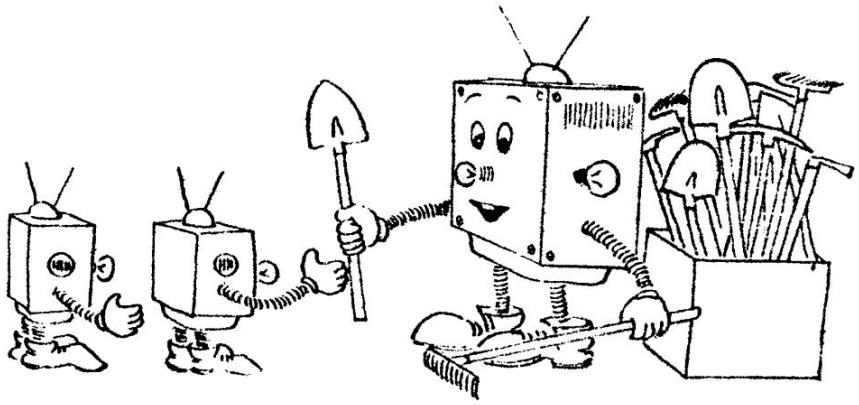
References

1. Jeff Beck, editor. *Proceedings of the Journal Article Tag Suite Conference 2010*, Bethesda (MD), 2010. National Center for Biotechnology Information (USA). Available online at `http://www.ncbi.nlm.nih.gov/books/NBK47086/`.
2. Thierry Bouche, Thomas Fischer, Claude Goutorbe, and David Ruddy. A recommended best practice for unqualified Dublin Core metadata records. Available online at `http://projecteuclid.org/collection/euclid/documents/metadata/dml_dc.html`, 2009.

3. DCMI Eprints Working Group. Scholarly works Dublin Core application profile. Available online at http://dublincore.org/scholarwiki/SWAPDSP, 2006.
4. EULER project. The EULER application profile. Available online at http://www.emis.de/projects/EULER/metadata.html, 2002.
5. INERA Inc. NLM DTD Resources: Introduction. Web page available online at http://www.inera.com/nlmresources.shtml.
6. Michael Jost, Thierry Bouche, Claude Goutorbe, Jean-Paul Jorda, et al. Deliverable D3.2: The EuDML metadata schema, initial version. Technical report, The EuDML project, 2010.
7. Library of Congress. MODS schema. Documentation available online at http://www.loc.gov/standards/mods/mods-outline.html, v. 3.4: 2010.
8. Mulberry Technologies Inc. JATS – The Journal Archiving and Interchange Tag Suite. Web page available online at http://www.mulberrytech.com/JATS/index.html.
9. National Center for Biotechnology Information (NCBI) of the National Library of Medicine (NLM). Journal archiving and interchange tag suite. Documentation available online at http://dtd.nlm.nih.gov/, v. 3.0: 2008.
10. David Ruddy. Developing a metadata exchange format for mathematical literature. In Petr Sojka, editor, *Towards a Digital Mathematics Library*, pages 27–36, Brno, Czech Republic, 2010. Paris, France, July 7–8th 2010, Masaryk University Press. Paper available online at http://www.dml.cz/bitstream/handle/10338.dmlcz/702570/DML_003-2010-1_4.pdf, XML application profile available at http://projecteuclid.org/documents/metadata/mlap/mlap_dsp.xml.
11. The International Federation of Library Associations and Institutions (IFLA). *Functional requirements for bibliographic records*, volume 19 of *UBCIM publications; new series*. K.G. Saur, München, 1998. Current version available online at http://archive.ifla.org/VII/s13/frbr/frbr_current_toc.htm.

Part III

DML Building Technologies



Towards Reverse Engineering of PDF Documents

Josef B. Baker, Alan P. Sexton, and Volker Sorge

School of Computer Science, University of Birmingham
Email: {j.baker|a.p.sexton|v.sorge}@cs.bham.ac.uk
URL: http://www.cs.bham.ac.uk/~{jbb|aps|vxs}

Abstract. We present a progress report on our ongoing project of reverse engineering scientific PDF documents. The aim is to obtain mathematical markup that can be used as source for regenerating a document that resembles the original as closely as possible. This source can then be a basis for further document processing. Our current tool uses specialised PDF extraction together with image analysis to produce near perfect input for parsing mathematical formula. Applying a linear grammar and specific drivers for each output format to this input, we can produce an accurate reproduction of formulae when presented with their coordinates. In this paper we will show how this information can be exploited to discover the locations of both inline and display formulae, and also to perform rudimentary layout analysis of the whole document, identifying structures such as headings and paragraphs.

1 Introduction

Converting PDF files into alternative formats can offer users the ability to do more than just view or print a document. Indeed, there exist a number of software tools that enable their conversion into formats such as ASCII or Word, along with the copy-to-clipboard function available with the majority of PDF viewers. However, all the currently existing tools focus on the extraction of regular text from documents and none are capable of faithfully extracting and translating non-textual components, such as the document's format and styling, mathematical formulae or tables. We are working on a system that allows faithful reverse engineering of entire PDF documents, with a particular emphasis on converting mathematical content into markup languages like LaTeX or MathML.

In previous work we have focused on the reconstruction of mathematical formulae in PDF documents and their parsing into LaTeX and MathML using formal grammars. While this yielded good results and enabled reproduction of formulae very close to the original, the main drawback of the technique was that formulae had to be manually identified and clipped from PDF documents.

In this paper we report on a significant extension to our previous work by automatically identifying formulae through the analysis of symbols, fonts and their spatial relationships within each page. Furthermore, we show how this allows us to extract both text and mathematical content and we demonstrate how this information can be used to perform layout analysis of a page. This

Petr Sojka, Thierry Bouche (editors): DML 2011, Towards a Digital Mathematics Library, pp. 65–75.
 ISBN 978-80-210-5542-1

paper is a progress report on the current state of the overall project. An evaluation of the effectiveness of our approach is presented in [3], which compares our results to those of the Infty mathematical document analysis system [9].

In Sec. 2 we review the key points of our previous approaches to formula recognition from PDF documents as described in [2]. Sec. 3 then explains how this process is extended to perform whole page analysis, including layout analysis and recognition of inline mathematics. We discuss advantages and disadvantages of our approach as well as future improvements in Sec. 4 and conclude in Sec. 5.

2 Previous Work

In [2] we demonstrated an approach to formula recognition from PDF documents, which bypassed the standard but troublesome OCR stage by replacing it with PDF analysis to produce high quality results. Here we summarise this approach.

The PDF specification is very large and complex, covering 9 different versions. Some PDF files even store their contents in a raster image format and contain no more usable information than the images themselves. We therefore focus on a subset of PDF, which encode symbol information in an analysable form and covers a large amount of published scientific and mathematical material.

The files that we can parse use either Type 1 or Type 2 fonts, with their respective font encoding and width objects contained in the PDF file. We also require a valid PDF file, as many have corrupt reference tables, missing or erroneous objects. This generally limits our software to files generated from LaTeX, but not exclusively, as we have also had success working with those generated from `Troff`, `OpenOffice`.

We begin by rendering a PDF file to a TIFF image, from which a user is able to select specific areas of mathematics, an example of which is shown in Fig. 1. For each of these clips, the co-ordinates and dimensions are calculated, along with those of every connected component contained in that area. The information is then saved together with other meta data about the image, such as the name and page number of the file it was selected from. This meta data is then used by a PDF extractor in order to find the correct page within the file to process.

The extractor makes two passes over the file, the first of which is to collate the required content streams, which hold instructions for placing and displaying characters, lines and images, along with a number of other resources, including font dictionaries and character names. The second pass of the file sequentially processes each of these instructions to identify and extract each symbol, its respective font name, size and base point along with all lines and their coordinates.

The information produced by this process, whilst sufficient for the analysis of one dimensional text, is not accurate enough for the recognition of the two

dimensional relationships that occur with mathematical formulae. Therefore the connected components obtained during image analysis are registered to the characters and lines, resulting in precise spatial information.

The next stage is to parse this input, with the ultimate aim of producing a version of the formula in an output such as LaTeX. This is achieved by using a parser based on a linear grammar, a heavily modified version of that described by Anderson [1], in which spatial relationships between symbols in a given formula were analysed. His grammar, whilst very efficient, was quite restrictive and lacked the flexibility to cope with the different styles of typesetting that are common today. Therefore we removed many of the spatial relationship restrictions and extended the grammar to include accents, under bars, over bars, braces and multiline formulae and also to analyse symbol fonts, sizes and alignment.

This analysis produces a string representing the formula, an example of which is shown in Listing 8.1, the linearised version of Fig. 1.

Parse trees are then generated from the linearised string and used as intermediate representations for subsequent translation into mathematical markup. Different types of parse trees are generated, from simple parse trees that hold only the basic information on the relationships between symbols in a formula, to more complex parse trees that incorporated information on font, character sizes, and relative spacing between symbols.

Finally, output specific drivers are used to translate these parse trees into mathematical markup. Two main drivers have been created: One producing LaTeX code that faithfully reproduces the original formula taking spatial information into account and sometimes inserting this information explicitly into the produced code. A second is aimed at generating LaTeX that closely resembles code that could have been written by a human mathematician. Whilst the latter does not necessarily reproduce *exactly* the same formula as in the original document, it has the advantage that its LaTeX lends itself more to a semantic evaluation as well as cleaning up potential layout mistakes introduced by the author. The idea to use more than one driver to implement these goals is primarily to have a clear separation that enables an easy parameterisation of our software tool depending on the target application.

The resultant LaTeX code produced for Fig. 1 is shown in Listing 8.2. Observe that the translation is a straightforward translation into standard LaTeX without assuming any third party packages in the LaTeX environment (e.g., we use array environments as opposed to anything more sophisticated, such as, for example, those provided by some of the `amsmath` packages). Note also that the `cmbxa` prefix command is used to set its argument into a specialised font defined in the preamble of the resulting LaTeX file. While this treatment has the drawback that the produced LaTeX is less intuitive, it has the advantage that we can reproduce any specialist font that is actually used in the document. As a consequence, we can deal not only with documents that have been produced within a standard LaTeX environment, but also with those that have been produced by other

tools or where standard fonts have been replaced — a common practice among publishers.

In order to regain more intuitive LaTeX that is closer to what a user would actually write, one could introduce font mappings to specialist commands, as well as use specialist environments for matrices or multiline formulae, etc. However, this will require a more elaborate level of analysis, which is currently not implemented, but might be added in the future.

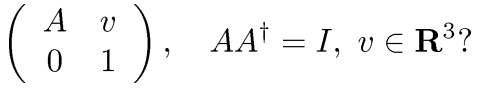


Fig. 1. Clipped image of a formula

```
matrix(<parenleftbigg, CMEX10, 9.963>)(row(col(<A, CMMI10,
    9.963>)col(<v, CMMI10, 9.963>))row(col(<zero, CMR10, 9.963>)
    col(<one, CMR10, 9.963>)))(<parenrightbigg, CMEX10, 9.963>)
    w3 <comma, CMMI10, 9.963> w4 sup <A A, CMMI10, 9.963>)(<
    dagger, CMSY7, 6.974>) w3 <equal, CMR10, 9.963> w2 <I comma,
     CMMI10, 9.963> w4 <v, CMMI10, 9.963> w2 <element, CMSY10,
    9.963> w2 sup(<R, CMBX10, 9.963>)(<three, CMR7, 6.974>) w1 <
    question, CMR10, 9.963>
```

Listing 8.1. Linearised version of clip

```
\[\left(\begin{array}{cc} A & v \\ 0 & 1 \end{array}\right)
    , \quad AA ^{ \dagger } = I , \quad v \in \
    cmbxa{R} ^{ 3 } ? \]
```

Listing 8.2. Output LaTeX code

Further drivers consist of a module producing Presentation MathML, as well as one that generates input for Festival, a speech synthesis tool. Most of the drivers focus upon the reconstruction of mathematics for presentation. However, we have also made some initial steps towards supporting a semantic interpretation of the parsed mathematical content [4], by constructing tools for semantic ground truthing of mathematical documents.

3 Layout Analysis

The main change over our previous work is that we now analyse and translate entire documents automatically rather than just single, manually clipped formulae. As a consequence we need to analyse the layout of each page of the document in order to reproduce it as faithfully as possible. This requires both changes to the extraction process and new drivers to perform the layout analysis. The former is realised by adding a pre-processing step in the extraction

process that identifies single lines on a page. The latter consists of two steps: separating mathematics and regular text in single lines and attempting to reassemble specific print areas from consecutive lines. This information can then be exploited during the translation of extracted content into a final output format.

3.1 Linewise Extraction of PDF Content

In order to extract character information for the whole document, the input PDF file is initially burst into single page PDFs, which are all rendered to TIFF images. For the purpose of connected component to symbol registration, each image and its respective PDF file are then treated as standard clips. From this we attempt the first stage of layout analysis, where we try to identify any columns and lines comprising the page. Projection profile cutting is used for this task though horizontal cuts, i.e. those between lines, are only made if the white space between symbols exceeds a certain threshold. This is found by ordering the connected components by their top y coordinate and calculating the median white space between each pair of sequential components, when the value is greater than zero.

The result of this process is a number of files, each representing a line, containing a list of symbols and their attributes. Each line is then linearised to produce its string representation as discussed in Sec. 2. In addition, we pass the bounding box information for each line, which can be used in the subsequent analysis steps.

Consider the example given in Fig. 2, a page from a freely available book on function theory [7], where the left hand side is an image of the original PDF page (observe that for the example it is not necessary to explicitly read individual characters). This page will be broken down into 26 lines and for each of the identified lines a representation will be computed. For instance the representation for the second line would be of the form

```
894 1057 248 58 <P r o o f period , CMBX10 , 9.963>
```

where the first four integers represent the bounding box information in the form of the x, y coordinate of the line on the page plus height and width of the line. Given this line-by-line information, the main layout analysis proceeds in two steps. First, lines are separated into text lines and display style mathematics, which are then grouped together into paragraphs and further classified.

One interpretation error can be observed in the fourth line of the multi-line math expression in Fig. 2. Here the author has forgotten a closing parenthesis in the superscript expression of the last B. In our current implementation fences are explicitly opened with LaTeX `left` and closed with `right` commands. While our implementation keeps track of matching fences and if necessary adds dummy `left` or `right` commands if there is a mismatch, this is done for the entire formula and not for subgroups inside the formula (e.g., like a superscript). As a consequence the opening parenthesis in the superscript is closed only after the entire expression, which leads to the misinterpretation

Proof.

$$\begin{aligned}\|[\exp(n(B-I))-B^n]x\| &= \|e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}(B^k-B^n)x\|\\ &\le e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}\|(B^k-B^n)x\|\\ &\le e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}\|(B^{|k-n|}-I)x\|\\ &= e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}\|(B-I)(I+B+\cdots+B^{(|k-n|-1)}x\|\\ &\le e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}|k-n|\|(B-I)x\|.\end{aligned}$$

So to prove (11.22) it is enough establish the inequality

$$e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}|k-n|\le\sqrt{n}. \tag{11.23}$$

Consider the space of all sequences $\mathbf{a}=\{a_0,a_1,\dots\}$ with finite norm relative to scalar product

$$(\mathbf{a},\mathbf{b}):=e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}a_k\overline{b_k}.$$

The Cauchy-Schwarz inequality applied to $\mathbf{a}$ with $a_k=|k-n|$ and $\mathbf{b}$ with $b_k\equiv 1$ gives

$$e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}|k-n|\le\sqrt{e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}(k-n)^2}\cdot\sqrt{e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}}.$$

The second square root is one, and we recognize the sum under the first square root as the variance of the Poisson distribution with parameter n, and we know that this variance is n. QED

11.7 Convergence of semigroups.

We are going to be interested in the following type of result. We would like to know that if A_n is a sequence of operators generating equibounded one parameter semi-groups $\exp tA_n$ and $A_n\to A$ where A generates an equibounded semi-group $\exp tA$ then the semi-groups converge, i.e. $\exp tA_n\to\exp tA$. We will prove such a result for the case of contractions. But before we can even formulate the result, we have to deal with the fact that each A_n comes equipped with its own domain of definition, $D(A_n)$. We do not want to make the overly

Spanning Line
Flushleft Line
Multi Line Math
Flushleft Line
Single Equation
Unindented Paragraph
Single Line Math
Unindented Paragraph
Single Line Math
Unindented Paragraph
Flushleft Line
Unindented Paragraph

Proof.

$$\begin{aligned}\|\left[\ \exp\left(n\left(B-I\right)\right)-B^n\right]x\| &= \|e^{-n}\textstyle\sum_{k=0}^{\infty}\frac{n^k}{k!}\left(B^k-B^n\right)x\|\\ &\le e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}\|\left(B^k-B^n\right)x\|\\ &\le e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}\|\left(B^{|k-n|}-I\right)x\|\\ &= e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}\|\left(B-I\right)\left(I+B+\cdots+B^{(|k-n|-1)x}\right\|\\ &\le e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}|k-n|\|\left(B-I\right)x\|.\end{aligned}$$

So to prove (11.22) it is enough establish the inequality

$$e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}|k-n|\le\sqrt{n}. \tag{11.23}$$

Consider the space of all sequences $\mathbf{a}=\{a_0,a_1,...\}$ with finite norm relative to scalar product

$$(\ \mathbf{a}\ ,\ \mathbf{b}\ ):=e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}a_k\overline{b_k}.$$

The Cauchy-Schwarz inequality applied to $\mathbf{a}$ with $a_k=|k-n|$ and $\mathbf{b}$ with $b_k\equiv 1$ gives

$$e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}|k-n|\le\sqrt{e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}\left(k-n\right)^2}\cdot\sqrt{e^{-n}\sum_{k=0}^{\infty}\frac{n^k}{k!}}.$$

The second square root is one, and we recognize the sum under the first square root as the variance of the Poisson distribution with parameter n, and we know that this variance is n. QED

1 1 . 7 Convergence of semigroups.

We are going to be interested in the following type of result. We would like to know that if A_n is a sequence of operators generating equibounded one pa – rameter semi-groups $\exp\ tA_n$ and $A_n\to A$ where A generates an equibounded semi-group $\exp tA$ then the semi-groups converge, i.e. $\exp\ tA_n\to\exp\ tA$. We will prove such a result for the case of contractions. But before we can even formulate the result, we have to deal with the fact that each A_n comes equipped with its own domain of definition, $D\left(A_n\right)$. We do not want to make the overly

Fig. 2. Page 317 from [7]. Original on the left and rendered LaTeX output on the right.

that can be observed in Fig. 2. Having analysed this error, we plan to modify our fence balancing algorithm to respect formula subgroups.

3.2 Analysis of Lines

All linearised lines of a page are then parsed using a LALR parser, resulting in a collection of parse trees. These parse trees are an intermediate representation, one for each line, containing structural information that can be exploited for the next steps in the layout analysis as well as in subsequent translation into output formats.

Each line is analysed separately and classified by whether it is primarily a text line or a math line. The single elements in a line are translated by linearly assembling consecutive words, identifying sequences of mathematical expressions and assembling them into single inline math formulae. A line is then treated as a text line if it

(a) contains only a sequence of words,
(b) if it contains at least two consecutive words and the number of inline math expressions is not larger than the number of words,
(c) contains more than three consecutive words regardless of the number of inline math expressions.

Everything else will be treated as display style mathematics.

In our example in Fig. 2 we get 8 math lines, whereas all others are recognised as text lines, possibly containing inline mathematics.

3.3 Assembling Vertical Areas

In a next step we then combine consecutive lines as much as possible to assemble meaningful multi-line areas. Here we also exploit the bounding box information of each line by comparing it with the overall dimension of the print area of the page. That latter can be easily computed by combining all bounding boxes of all lines. This additional structural information can be further exploited for setting content by the output drivers.

Consecutive display-style math lines are combined into single multi-line math expressions. We thereby distinguish four different types of math expressions:

Single Line Math A single display style math expression. Both previous and next lines (if either exist) have to be text lines.

Multi Line Math A contiguous sequence of display style math lines.

Single Equation A single line display style math expression where a tag has been identified that might function as a label for the formula. Tags are identified if (1) a math line starts within a small threshold of the left margin or ends within a small threshold of the right margin, but not both, and (2) there is a discernible distance between the leftmost (or rightmost) expression and the other expressions of that line. An identified tag can be subsequently exploited by any translation driver.

Multi Line Equation A contiguous sequence of display style math lines where some lines have been identified as equation lines in the above sense.

Similar to math lines we also combine consecutive text lines into paragraphs, where paragraphs are separated if

(a) there is a change of font size,
(b) the vertical space between lines is larger than the arithmetic median of vertical space between all consecutive text lines identified on the page,
(c) the horizontal orientation of lines changes,
(d) if a line has a left indentation or the previous line ends prematurely.

We again distinguish a number of different types of single lines and paragraphs, depending on their spatial relationship to the text margins:

Spanning Line A single line starting at the left margin and ending at the right margin.

Flushleft Line A single line starting at the left margin but ending observably before the right margin.

Flushright Line A single line ending at the right margin but starting observably after the left margin.

Centred Line A line that both starts and ends observably after the left margin and before the right margin, respectively. It does not have to be fully centred around the horizontal centre of the text area.

Indented Paragraph A paragraph of consecutive lines, where the first line is a flushright line, while all other lines are spanning lines, with the exception of the last line, that can be a flushleft line.

Unindented Paragraph A paragraph of consecutive lines, where all lines are spanning lines, with the exception of the last line, that can be a flushleft line.

Centred Paragraph A paragraph consisting of consecutive centred lines. This paragraph can be both ragged left and ragged right.

Observe that for all the above we allow for a certain fuzziness, i.e., a line only has to match within a small threshold of the left or right margin for classification.

For our example in Fig. 2 the result of the layout analysis is given in the middle column. We can see that the topmost line is recognised as a spanning line, simply because it starts and ends with the margins of the text area in spite of the significant white space in the line. Also note that the fifth area is recognised as Single Equation with a right hand tag (11.23). On the other hand the third area is classified as Multi Line Math although its fourth line is within the right margin of the text area and could be considered an Equation. This is due to the fact that there is no rightmost expression that has significant distance to the other expressions.

3.4 Translation into Markup

Once the layout analysis is complete, specific drivers are employed to translate the content into actual markup. Currently we have two drivers, one for MathML

and one for LaTeX markup. The most developed driver is a LaTeX driver, which attempts to set the text components as faithfully as possible according to the classification derived in the layout analysis. For the translation of formulae we make use of the already developed mechanisms described in Sec. 2. In addition the contained font and spacing information is exploited to set characters and words in the correct font and size as well as to include additional space if necessary.

The result of the LaTeX driver for our example is given in the right column of Fig. 2. While the actual output is already close to the original input document, there are still a number of discrepancies. We will therefore use this example when we discuss some of the shortcomings of the translation in the next section.

4 Discussion

We have evaluated our current approach of combined layout analysis and formula translation quantitatively against a small ground truth set of articles similar to the one presented in [8]. These results are presented in [3] together with a comparison to the results of the Infty system [9] on the same data, which uses a conventional OCR approach to extract content and layout. The paper also presents a qualitative comparison of our results with Infty's.

In this section we will now concentrate on a qualitative discussion exclusively for the results of our procedure and in particular point out some of the shortcomings we have identified and that we intend to address in the future.

The advantages of using projection profile cutting to find lines are that of speed and efficiency, and it also works well on many standard layouts including those with multiple columns. However the presence of figures, tables and vertically overlapping, but not touching lines can severely impact its performance. Also, the limits on large equations are sometimes erroneously treated as separate lines. Therefore, a major improvement would be to use a bottom up approach for line finding, employing image blurring, or to take into account more of the information available including the semantics of the page.

In general, the current strict order of first identifying lines and then linearising these separately is not ideal as it precludes some of the advantages of our grammar for linearising multi-line mathematical expressions. As demonstrated in [2], our grammar is capable of recognising and marking up certain alignment points when parsing multiline mathematical expressions. However, when parsing each line separately, these alignments can not be detected. This effect can be seen in our example, where the Multi Line Math expression is not correctly aligned.

Other obvious alignment problems can be observed in the section heading in the example in Fig. 2, which, while being correctly recognised as a Flushleft Line, is nevertheless too spread out. This is a consequence of the current mechanism for detection of white space, which divides it into only five classes, which are computed relative to the average distance between characters in

expressions. While these spacing classes correspond to the spacing design in LaTeX [6], we do not use any absolute thresholds but only relative values. This makes us independent from the actual tool that has produced the PDF as well as from any specialised fonts used by the authors. Obviously, the spacing information can therefore only be seen as a heuristic guide and is not necessarily adhered to by all documents that we consider.

While this system to classify internal space in formulae is deliberately coarse grained in order to aid the assignment of semantic information to components of single mathematical formulae (see [4] for details), in the setting of text lines it has the drawback that all white spaces above the relative threshold are not further distinguished and consequently replaced with the same explicit large space. A more sophisticated treatment of white space, possibly with the explicit representation of distances, might ameliorate this problem.

This could then also lead to a more effective approach to horizontally separate text areas in single lines or in vertically separate paragraphs. For example, the very first line of Fig. 2 is classified exclusively as a Spanning Line. However, it would be desirable to be able to split it at the position of the largest white space in order to enable better recognition of the single components, i.e., running header and page number. This would then also allow assigning certain semantic properties to areas, such as title, section headings and page numbers, as the Infty system does, and which would lead to a more human-like LaTeX translation and therefore to a better modelling of vertical space.

Finally, our current algorithm for deciding whether a line is primarily mathematics containing some embedded text or primarily text including some inline mathematics is rather ad hoc. A more sophisticated mechanism to distinguish mathematics from text and, in particular, display mathematics from inline expressions, such as the techniques proposed in [5], will be explored in the future.

5 Conclusion

In this paper we have presented significant improvements over previous iterations of our software. By automating the location of mathematical formulae, we have removed the most costly component, in terms of operator time, from the system; that of manual clipping. We have also shown how the system can be extended to not only deal with formula recognition, but also full layout analysis. However, this is at an early stage of development and we envisage significant improvements in the future, some of which we have discussed in the previous section. For a full comparison to the Infty system [9], we refer the reader to [3]. These results demonstrate the effectiveness of our approach.

Acknowledgements. This work is supported by the EuDML project, which is in turn partly financed by the European Union through its Competitiveness and Innovation Programme (Information and Communications Technologies Policy Support Programme, "Open access to scientific information", Grant Agreement No. 250503).

References

1. Anderson, R.H.: Syntax-Directed Recognition of Hand-Printed Two-dimensional Mathematics. Ph.D. thesis, Harvard University, Cambridge, MA (1968).
2. Baker, J., Sexton, A., Sorge, V.: A linear grammar approach to mathematical formula recognition from PDF. In: Proceedings of Intelligent Computer Mathematics (2009).
3. Baker, J., Sexton, A.P., Sorge, V., Suzuki, M.: Comparing approaches to mathematical document analysis. In: 11th International Conference on Document Analysis and Recognition (to appear) (2011).
4. Baker, J., Sexton, A., Sorge, V.: Faithful mathematical formula recognition from PDF documents. In: 9th IAPR International Workshop on Document Analysis Systems, Extended Abstracts. pp. 485–492. ACM Press, Boston, USA (2010).
5. Garain, U.: Identification of mathematical expressions in document images. In: Document Analysis and Recognition, International Conference on. pp. 1340–1344. IEEE Computer Society, Los Alamitos, CA, USA (2009).
6. Mittelbach, F., Goossens, M.: The LaTeX Companion. Pearson Education, 2e edn. (2005), TeX spacing table, page 525.
7. Sternberg, S.: Theory of functions of a real variable (2005), `http://www.math.harvard.edu/~shlomo/docs/Real_Variables.pdf`
8. Suzuki, M., Uchida, S., Nomura, A.: A ground-truthed mathematical character and symbol image database. In: Proc. of ICDAR. pp. 675–679. IEEE Computer Society (2005).
9. Suzuki, M.: Infty (2011), `http://www.inftyproject.org`

Web Interface and Collection for Mathematical Retrieval

WebMIaS and MREC

Martin Líška, Petr Sojka, Michal Růžička, and Petr Mravec

Masaryk University, Faculty of Informatics, Botanická 68a, 602 00 Brno, Czech Republic
255768@mail.muni.cz, sojka@fi.muni.cz, mruzicka@mail.muni.cz

Abstract. We demonstrate searching of mathematical expressions in technical digital libraries on a *MREC collection* of 439,423 real scientific documents with more than 158 million mathematical formulae. Our solution — *the WebMIaS system* — allows the retrieval of mathematical expressions written in TEX or MathML. TEX queries are converted on-the-fly into tree representations of Presentation MathML, which is used for indexing. WebMIaS allows complex queries composed of plain text and mathematical formulae, using MIaS (Math Indexer and Searcher), a math aware search engine based on the state-of-the-art system Lucene. MIaS implements proximity math indexing with a subformulae similarity search.

Keywords: math indexing and retrieval, mathematical digital libraries, information systems, information retrieval, mathematical content search, document ranking of mathematical papers, math text mining, WebMIaS, MIaS, Tralics, TEX, UMCL, Lucene

1 Introduction

The gateway to the vast treasures held in digital libraries' content is entered by *searching*. The Google generation is starting to demand a simple Google-like interface to access digital content, even on a global scale. The mainstream technologies and interfaces are developed only for plain text without support for mathematical formulae handling — documents are represented in a bag of words representation, in a simple vector space model.

Scientific and technical documents are full of indexes, exponents, and complex mathematical expressions, even in paper basic metadata, titles and abstracts. Our experience with Google Scholar shows that not handling mathematical expressions in citations causes severe problems. For example the paper by Kováčik and Rákosník [3] appears as more than twenty different papers there[1] mainly because of different and wrong (by different OCR) representation of mathematics in the paper metadata (title).

Although there have been several attempts to solve the mathematics search problem, none of them have, as yet, fulfilled the expectations. For

[1] cf. http://scholar.google.com/scholar?q=Kovacik+Rakosnik

Petr Sojka, Thierry Bouche (editors): DML 2011, Towards a Digital Mathematics Library, pp. 77–84.
 ISBN 978-80-210-5542-1

example, Springer offers LaTeXSearch[2] based just on TEX math string matching, which does not take into account the structural or semantical similarity of mathematical expressions at all.

We have created the web interface WebMIaS for our MIaS (Math Indexer and Searcher) system [6] indexing hundreds of thousands[3] of mathematical documents. We demonstrate a solution built on the state-of-the-art fulltext indexing engine Lucene ™ — we have added 'math-awareness' to it as a plug-in.

To test the system, we have created (Section 2) and indexed (Section 3) the MREC collection of hundreds of thousands mathematical documents. In Section 4 on the facing page we describe necessary transformations needed during querying and indexing (canonicalization of MathML). The WebMIaS web interface is then presented in Section 5 on page 80. The reader finds final remarks in Section 6 on page 83.

2 Mathematical Retrieval Collection MREC

To evaluate our system, we have built a corpus of mathematical texts, called MREC. We downloaded documents from arXMLiv[4] [8], where TEX documents from arXiv.org are transformed into XML documents. For the representation of mathematical formulae, MathML, a W3C standard, is used. The documents used come from different scientific areas (Physics, Mathematics, Computer Science, Quantitative Biology, Quantitative Finance and Statistics).

ArXMLiv[5] sorts transformed documents into several classes, based on the return value of transformation to MathML: successful, complete with errors, incomplete and none. MREC does not contain full arXiv, only documents from conversion classes successful and complete with errors (missing macros) — see Table 1 on the next page. We have collected 439,423 documents in well-formed XHTML, containing mathematical formulae in valid MathML. We hope that this corpus might be used for benchmarking mathematical retrieval, thus we have named it MREC (Mathematical REtrieval Collection) and made it available for this purpose at [4].

In our web interface for math searches we currently use this corpus of real mathematical papers.

3 Math-aware Indexing

We have developed a math aware, full-text based search engine called MIaS (Math Indexer and Searcher). [6] It processes documents containing mathematical notation in Presentation MathML format, however, it filters out all unnecessary presentational elements as well as any other MathML

[2] http://www.latexsearch.com/

[3] LaTeXSearch currently searches only three million formulae.

[4] http://kwarc.info/projects/arXMLiv/

[5] http://arxmliv.kwarc.info/

Table 1. Documents collected from arXMLiv

arXMLiv transformation result class	Quantity
successful (no problem)	65,874
successful (warning)	291,879
complete with errors (missing macros)	81,670
All documents	439,423

notation (Content MathML or other markup). MIaS allows users to search for mathematical formulae as well as the textual content of documents.

Since mathematical expressions are highly structured and have no canonical form, our system pre-processes formulae in several steps to facilitate a greater possibility of matching two equal expressions with different notation and/or non-equal, but similar formulae. With an analogy to natural language searching, MIaS searches not only for whole sentences (whole formulae), but also for single words and phrases (subformulae down to single variables, symbols, constants, etc.). For every formula and its subformulae on the input, MIaS creates several differently generalized representations to allow similarity searching of mathematics. For calculating the relevance of matched expressions to the user's query, MIaS uses a heuristic weighting of indexed terms, which accordingly affects scores of matched documents and thus the order of results. Weights are assigned to the formula according to the complexity of the formula, its level in the input formula tree and level of generalization.

At the end of all these processing methods, formulae are converted from XML nodes to a compacted linear string form which can be handled by the indexing core.

4 System Workflow

The top-level indexing scheme is shown in Figure 1 on page 81. Document and query processing is done separately for plain text terms and mathematical terms. Indexing of mathematics is done by our Presentation MathML tokenizer implemented in Java for Apache Lucene™3.1, and Lucene Solr™ 3.1 taking advantage of open Lucene architecture.

MathML notation in the query and indexed documents is normalized into Canonical MathML [1] to increase precision of the system. For conversion into this normalized MathML format we are using the software library UMCL (Universal Maths Conversion Library). The main purpose of the UMCL toolset is the transcription of the MathML formulae to Braille national codes. Related to our task is also the need for MathML formulae unification. UMCL transformation of the MathML to Canonical MathML is carried out using a set of XSL stylesheets. This transformation was integrated into the WebMIaS system with only the slightest modifications — the UMCL transformation adds attributes in the form of `id="formula:xx"` to every node of the output MathML.

This is not necessary for the WebMIaS purposes as it adds additional 'noise' to the formulae and increased size of the index. Thus, these attributes are not added to the Canonical MathML used by WebMIaS.

Our latest experiments with canonical forms of MathML generated by the UMCL transformation show that it not only increases fairness of similarity ranking, but also helps to match a query against the indexed form of MathML. For example, if the user asked the system for the

$$x^2 + y^2$$

formula using MathML of the form

```
<math xmlns="http://www.w3.org/1998/Math/MathML">
    <msup>
        <mi>x</mi>
        <mn>2</mn>
    </msup>
    <mo>+</mo>
    <msup>
        <mi>y</mi>
        <mn>2</mn>
    </msup>
</math>
```

the system would not be able to find any similar formulae due to omission of the `<mrow>` element in the MathML. Provided that the MathML canonicalization of the query is done prior to the search, the canonical form of the query

```
<math xmlns="http://www.w3.org/1998/Math/MathML">
  <mrow>
    <msup>
        <mi>x</mi><mn>2</mn></msup>
    <mo>+</mo>
    <msup>
        <mi>y</mi><mn>2</mn></msup>
  </mrow>
</math>
```

results in 36,817 hits in MREC 2011.4.

For a user-friendly math-aware information retrieval demonstration, we have built web interface *WebMIaS* (see Figure 2 on page 82).

5 WebMIaS

WebMIaS demonstrates the possibility of querying mathematical content on a large-scale. This has been facilitated by the full indexation of the mathematical corpus MREC. In the user interface (UI) we tried to mimic the simplicity of Google. In addition to the standard textual query terms, mathematics terms (mterms) may appear in the query as well, adding to the document

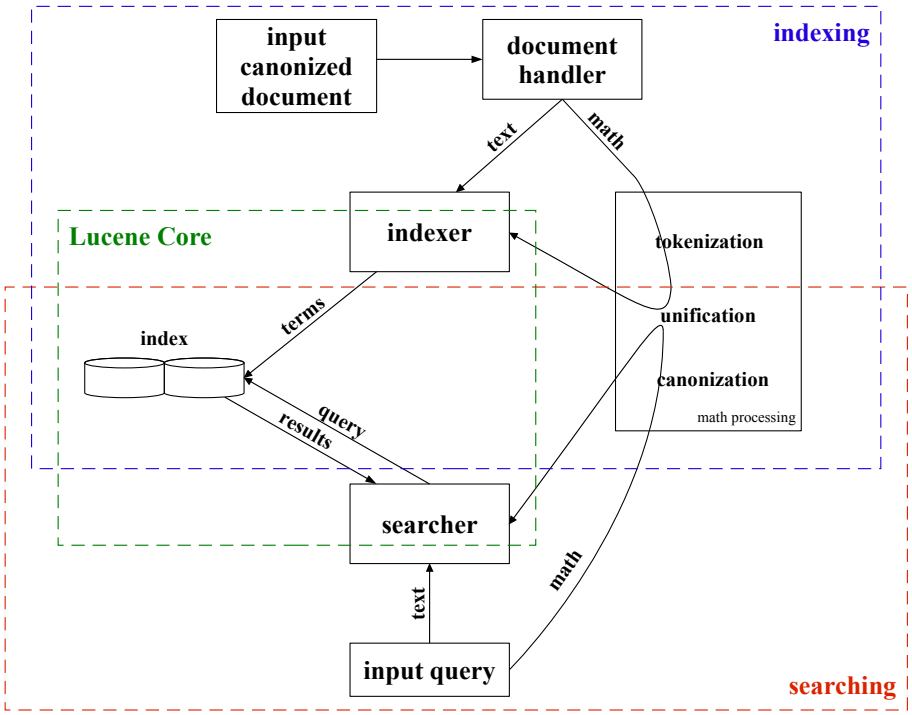


Fig. 1. Scheme of the system workflow

score with the weight depending on the similarity of matched formula to the queried one. Mterm could be either in MathML, or in TEX notation enclosed in two dollar signs. Since most mathematicians are used to using TEX compact notation for mathematical formulae, we have implemented on-the-fly TEX to MathML conversion [7] of queries using Tralics [2] as a library. Furthermore, canonicalization of the both MathML and TEX input queries has been employed to improve querying and to avoid notation flaws restraining proper results retrieval. For the best visual experience of the search results, we incorporated a much requested snippet retrieval and mathematical match highlighting in the hit list for each matched document. This will also help us to evaluate the search results and to be able to tweak the whole indexing and searching process for better results. We additionally incorporated MathJax in the UI for a better rendering and look of displayed mathematical snippets, which will in turn enhance web browser support, since not all of the web browsers have natural MathML rendering capabilities.

As is shown in Table 2 on page 83, the performance of the system scales linearly. This gives feasible response times even for our billions of indexed subformulae. One has to be patient for small formulae, as they score/match in

Fig. 2. WebMIaS web interface

most documents. We also tried to measure the average query time of WebMIaS working over the MREC 2011.4 corpus. We queried the created index with a set of differently complex queries (mixed, non-mixed, more/less complex single/multiple formulae). The resulting average query time was 469 ms.

It is very difficult to evaluate the mathematical search result and verify the soundness of our design. For a given set of queries, there should exist beforehand a complete list of the documents ordered by their relevance to the query with which the actual results can be compared with.

We have applied an empirical approach to the evaluation so far using our WebMIaS demo interface which is publicly available at `http://nlp.fi.muni.`

Table 2. Scalability test results (run on 448 GiB RAM, eight 8-core 64bit processors Intel Xeon™ X7560 2.26 GHz driven machine).

# Docs	Input formulae	Indexed formulae	Indexing run-time [ms]	Indexing CPU time [ms]
10,000	3,406,068	64,008,762	2,145,063	2,102,770
50,000	18,037,842	333,716,261	11,382,709	10,871,500
100,000	36,328,126	670,335,243	23,066,679	21,992,100
200,000	72,030,095	1,326,514,082	46,143,472	44,006,180
300,000	108,786,856	2,005,488,153	71,865,018	66,998,550
350,000	125,974,221	2,318,482,748	83,199,724	77,886,160
439,423	158,106,118	2,910,314,146	104,829,757	97,393,301

`cz/projekty/eudml/mias/`. It currently works on our mathematical corpus MREC version 2011.4 with 158,106,118 input formulae, 2,910,314,146 indexed (sub)formulae.

6 Conclusion

We have demonstrated the fully functioning information retrieval interface, WebMIaS, capable of retrieving both text and math from fulltexts in Presentation MathML. The system scales well and has got the power to be used in several digital libraries.

As our developments were motivated by future deployment in the EuDML[6] project [9], experience with WebMIaS results will be projected and employed in the EuDML UI. Another area of long-term research planned is supporting Content MathML, in a way similar to the current handling of Presentation MathML. The architectural design is suited to it, but as most of the math within EuDML will be in Presentation MathML taken from PDFs, this is not currently a high priority.

Acknowledgements. This work has been in part financed by the European Union through its Competitiveness and Innovation Programme (Information and Communications Technologies Policy Support Programme, "Open access to scientific information", Grant Agreement No. 250503).

References

1. Archambault, D., Moço, V.: Canonical MathML to Simplify Conversion of MathML to Braille Mathematical Notations. In: Miesenberger, K., Klaus, J., Zagler, W., Karshmer, A. (eds.) Computers Helping People with Special Needs, Lecture Notes in Computer Science, vol. 4061, pp. 1191–1198. Springer Berlin / Heidelberg (2006), `http://dx.doi.org/10.1007/11788713_172`

[6] `http://eudml.eu`

2. Grimm, J.: Producing MathML with Tralics. In: Sojka [5], pp. 105–117, http://dml.cz/dmlcz/702579
3. Kováčik, O., Rákosník, J.: On spaces $L^{p(x)}$ and $W^{k,p(x)}$. Czechoslovak Mathematical Journal 41, 592–618 (1991), http://dml.cz/dmlcz/102493
4. MREC — Mathematical REtrieval Collection, http://nlp.fi.muni.cz/projekty/eudml/MREC/index.html
5. Sojka, P. (ed.): Towards a Digital Mathematics Library. Masaryk University, Paris, France (Jul 2010), http://www.fi.muni.cz/~sojka/dml-2010-program.html
6. Sojka, P., Líška, M.: Indexing and Searching Mathematics in Digital Libraries – Architecture, Design and Scalability Issues. In: Davenport, J.H., Farmer, W., Rabe, F., Urban, J. (eds.) Proceedings of CICM Conference 2011 (Calculemus/MKM). Lecture Notes in Artificial Intelligence, LNAI, vol. 6824, pp. 228–243. Springer-Verlag, Berlin, Germany (Jul 2011)
7. Stamerjohanns, H., Ginev, D., David, C., Misev, D., Zamdzhiev, V., Kohlhase, M.: MathML-aware Article Conversion from LaTeX. In: Sojka, P. (ed.) Proceedings of DML 2009. pp. 109–120. Masaryk University, Grand Bend, Ontario, CA (Jul 2009), http://dml.cz/dmlcz/702561
8. Stamerjohanns, H., Kohlhase, M., Ginev, D., David, C., Miller, B.: Transforming Large Collections of Scientific Publications to XML. Mathematics in Computer Science 3, 299–307 (2010), http://dx.doi.org/10.1007/s11786-010-0024-7
9. Sylwestrzak, W., Borbinha, J., Bouche, T., Nowiński, A., Sojka, P.: EuDML—Towards the European Digital Mathematics Library. In: Sojka [5], pp. 11–24, http://dml.cz/dmlcz/702569

Using Discourse Context to Interpret Object-Denoting Mathematical Expressions

Magdalena Wolska[1], Mihai Grigore[2⋆], and Michael Kohlhase[3]

[1] Computational Linguistics and Phonetics, Saarland University
D-660 41 Saarbrücken, Germany
magda@coli.uni-saarland.de
[2] Information Systems Engineering, Goethe University
D-603 23 Frankfurt am Main, Germany
grigore@wiwi.uni-frankfurt.de
[3] Computer Science, Jacobs University Bremen
D-287 59 Bremen, Germany
m.kohlhase@jacobs-university.de

Abstract. We present a method for determining the context-dependent denotation of simple object-denoting mathematical expressions in mathematical documents. Our approach relies on estimating the similarity between the linguistic context within which the given expression occurs and a set of terms from a flat domain taxonomy of mathematical concepts; one of 7 head concepts dominating a set of terms with highest similarity score to the symbol's context is assigned as the symbol's interpretation. The taxonomy we used was constructed semi-automatically by combining structural and lexical information from the Cambridge Mathematics Thesaurus and the Mathematics Subject Classification. The context information taken into account in the statistical similarity calculation includes lexical features of the discourse immediately adjacent to the given expression as well as global discourse. In particular, as part of the latter we include the lexical context of structurally similar expressions throughout the document and that of the symbol's declaration statement if one can be found in the document. Our approach has been evaluated on a gold standard manually annotated by experts, achieving 66% precision.

1 Introduction

Consider the following discourse fragment from [14]:

> ... Let $f : X \to B$ be a surjective morphism and let $\omega_{X/B}$ denote the relatively canonical sheaf of differentials. Let us assume that the generic fibre is smooth of genus g and let us denote by δ the number of singular points in the fibres. We write Λ_n for the determinant of $f_*\omega^n_{X/B}$ and λ_n for the degree of $\boxed{\Lambda_n}$. ...

⋆ This work was performed while the second author was visiting the Computational Linguistics and Phonetics Department of Saarland University as a student of Jacobs University Bremen.

Petr Sojka, Thierry Bouche (editors): DML 2011, Towards a Digital Mathematics Library, pp. 85–101.
 ISBN 978-80-210-5542-1

Even a layperson, without any knowledge whatsoever on the subject matter of the discourse from which the above fragment has been extracted, is capable of inferring the name of the object which the boxed expression, Λ_n, denotes: In the same sentence in which the expression in question occurs she finds a statement "We write Λ_n for the determinant of $f_*\omega^n_{X/B}$" from which she can infer that Λ_n must denote an object called "determinant". She may not know what a determinant is specifically and how to compute it, but she can at least identify the domain term that names the object for which the symbol stands in order to find its meaning, for instance, in a textbook.[4]

The above-quoted fragment exemplifies a fairly typical way in which mathematical discourse is written. While mathematical documents abound in symbols, a large proportion of the symbols used are explicitly introduced in the discourse or stated to denote specific objects. A corpus study on symbol declarations in mathematical writing revealed that around 70% of object-denoting symbolic expressions randomly selected from mathematical scientific papers were explicitly stated to denote objects of specific types [15].

In computational linguistics, the problem of identifying which sense of a polysemous word is meant in a given sentence is known as *word sense disambiguation* (WSD) and has been one of the active research areas since the beginning of interest in word sense disambiguation in the forties.[5] Clearly, an automated text understander, if it is to make inferences about a discourse, must be in a position to discriminate between the meanings of words and correctly recognize the meaning in context. With the increasing interest in automated processing of technical and scientific documents, in particular, with the view to building interactive digital libraries of scientific writing[6] the same holds of automated processing of scientific prose. In particular, in case of exact sciences which make use of symbolic notation, identifying the meaning of not only the linguistic expressions, but also formal expressions is an obvious task and challenge. Interpretation of symbolic mathematical expressions can be a useful source of information in a number of sub-tasks in a mathematical document processing pipeline for digitizing mathematics. For instance, in the task of parsing mathematical notation, i.e. identifying the structure and (compositional) semantics of symbolic expressions, the information about the expressions' interpretation can guide the selection (or weighing) of likely parse candidates. This could be useful in processing LaTeX documents as well as in mathematical OCR, in particular, in handwriting recognition; for instance, in examples such as

[4] Note that for the purposes of the knowledge-poor methods discussed in this paper, it is sufficient to determine that "determinant of $f_*\omega^n_{X/B}$" denotes an object via a domain term ("determinant"). In fact, as a reviewer pointed out, in this particular example, this determinant is a sheaf over a smooth fibration, not a number computed from a matrix, as a non-expert might suspect. This shows that for more knowledge-rich methods, a tight collaboration between authors and linguists is of essence.

[5] For a recent comprehensive overview of the state of the art, see [8].

[6] (See, for instance, EuDML (`http://www.eudml.eu/`) or WDML (`http://www.mathunion.org/WDML/`) for recent efforts in this direction.)

above, in deciding between horizontal adjacency and super-/subscript relation when the super-/subscript is written partly across the centre horizontal line of the expression.

Our previous study on disambiguating symbolic expressions has shown that the local linguistic context, within which mathematical expressions are embedded, provides a good source of information for recognizing a class of objects to which a mathematical expression belongs [5]. However, the approach addressed only those mathematical expressions which are syntactically part of a nominal group and, in particular, are in an apposition relation with an immediately preceding noun phrase; i.e. the expressions addressed came from a linguistic pattern: "... noun_phrase symbolic_math_expression ...", as in the example: "... the inverse function ω_1 ...". Only the immediate left linguistic context of a symbolic was used in the disambiguation process, despite the fact that mathematical texts are known to introduce notations and concepts as they go along.

In this paper we propose a new approach to interpreting mathematical expressions. Our interpretation strategy is inspired by recent computational WSD approaches which use statistical co-occurrence measures to estimate semantic relatedness between lexical contexts. In our case, co-occurrence statistics are computed using on the one hand, both the local discourse within which the expression under analysis is embedded as well as the relevant segments of the entire document (global discourse) and, on the other hand, sets of terms from a lexical resource we built. As in [5], we use a lexical resource of mathematical terms which corresponds to a flat taxonomy of mathematical objects and which provides an association between sets of domain terms which name mathematical objects and names of broader semantic classes of mathematical objects. The taxonomy has been constructed semi-automatically by combining structural and lexical information from the Mathematics Subject Classification and the Cambridge Mathematics Thesaurus. The class names themselves serve as symbolic interpretations of mathematical expressions under analysis.

The class of expressions addressed In this work we attempt to interpret only *simple object-denoting mathematical expressions*. "Simple" refers to the expressions' high-level structure: The terms may be atomic identifiers and super-/sub-scripted atomic identifiers; the expression(s) in the super-/subscripts can be of arbitrary complexity. For instance, Λ_n and $\omega_{X/B}$ are simple expressions, while $f : X \to B$ is not. Throughout this paper, the term "simple mathematical expression(s)" refers to this class of symbols.

Problem statement We formulate the interpretation problem as follows: Given a mathematical document containing a target mathematical expression of the type described above, a simple object-denoting term, can we indicate one (or more) concepts from a predefined set of concepts which corresponds to a coarse-grained denotation of the given expression?

Outline The paper is organized as follows: In Section 2 we introduce the corpus from which the documents we analyze stem and outline the preprocessing steps. In Section 3 we introduce the taxonomy of mathematical objects constructed for this study. In Section 4 we describe the approach to interpreting simple object-denoting mathematical terms: We introduce the similarity measures, the two types of context based on which similarity is computed, and the algorithm itself. In Section 5 we summarize the creation of a gold standard for evaluation, the evaluation measures we used, and the results themselves.

2 The Data and Preprocessing

For the purposes of this work we used 10,000 mathematical documents from the arXMLiv collection, processed by the `LaTeXML` system [10,12]. arXMLiv is subset of arXiv, an archive of electronic preprints of scientific papers in the fields of, among others, mathematics, statistics, physics, and quantitative biology[7]. The documents we used stemmed from the mathematics subset of arXiv.

`LaTeXML` uses three formats for representing mathematical expressions of which two are relevant for this study: In the XMath format mathematical expressions are encoded as a linear sequence of tokens, with the explicit requirement for `LaTeXML` not to generate any semantic parse tree beyond the token level (unless the semantics is explicitly encoded in the LaTeX source). The presentation format, MathML, is a widely used W3C standard for rendering mathematical content on the Web [1].[8] Figure 1 shows the XMath and MathML representations of the expression $\mathcal{D}/\mathcal{D}_0$. The two formats are used to retrieve simple mathematical terms as defined in the Introduction.

2.1 Tokenization and identification of target expressions

Each of the 10,000 documents in the corpus was word- and sentence-tokenized,[9] and the words were stemmed.[10] Then mathematical expressions were normalized by replacing them with unique identifiers and the mappings between the identifiers and the two relevant representations were stored for each mathematical expression. Simple mathematical expressions were identified

[7] `http://www.arxiv.org`

[8] `http://www.w3.org/Math/`

[9] A sentence is understood, in a standard sense, as a grammatical unit consisting of one or more clauses. Sentence-tokenization was performed using a rule-based tokenizer based on a standard set of end-of-sentence punctuation marks and a number domain-specific rules for sentences ending with mathematical expressions which may not end with end-of-sentence punctuation.

[10] We use stemming as a knowledge-poor substitute for lemmatization. This solution has obvious drawbacks, however, context-sensitive lemmatization is out of scope at the time this work is conducted because we do not have access to a large-scale dictionary for mathematical discourse, nor to any standard language processing tools for this domain.

```
<Math mode="inline" tex="{\cal D}/{\ cal D}_{0}" xml:id="S1.p3.m6">
   <XMath>
      <XMTok role="UNKNOWN" font="caligraphic">D</XMTok>
      <XMTok meaning="divide" role="MULOP" style="inline">/</XMTok>
      <XMTok role="UNKNOWN" font="caligraphic">D</XMTok>
      <XMApp role="POSTSUBSCRIPT" scriptpos="2">
         <XMArg rule="Subscript">
            <XMTok meaning="0" role="NUMBER">0</XMTok>
         </XMArg>
      </XMApp>
   </XMath>
</Math>
```

```
<m:math display="inline ">
   <m:mrow>
      <m:mi mathvariant=" script ">D</m:mi>
      <m:mo>/</m:mo>
      <m:msub>
         <m:mi mathvariant=" script ">D</m:mi>
         <m:mn>0</m:mn>
      </m:msub>
   </m:mrow>
</m:math>
```

Fig. 1. XMath (top) and MathML (bottom) representations of the expression $\mathcal{D}/\mathcal{D}_0$

by analyzing the MathML and XMath representations[11] and the results were manually verified for the expressions used for the gold standard.

2.2 Domain term identification

The purpose of identifying mathematical domain terms was two-fold: First, we identify domain terms in the Mathematics Subject Classification while building the lexical resource for interpretation and, second, we use domain terms in the course of identifying symbol declaration statements which are used in the interpretation process.

To identify domain terms, we implemented a modified version of the algorithm presented in [4]. In our implementation, only n-gram counts are used and no linguistic information; in particular, we do not have part of speech (POS) tag information which the authors use to identify noun phrases.[12] We

[11] We omit the algorithm here.

[12] Again, due to the notorious lack of linguistic processing tools for mathematical discourse we opt for a knowledge-poor approach here. We are presently working on building up an annotated corpus of mathematical discourse in order to train a

Table 1. An excerpt from MSC 2010

40-XX	SEQUENCES, SERIES, SUMMABILITY
40 Axx	Convergence and divergence of infinite limiting processes
40 Bxx	Multiple sequences and series
40 Cxx	General summability methods
40 Dxx	Direct theorems on summability
40 Exx	Inversion theorems
40 Fxx	Absolute and strong summability
40 Gxx	Special methods of summability
40 Hxx	Functional analytic methods in summability
40 Jxx	Summability in abstract structures

therefore employed a tailored stop-word list including items which are not closed-class words and not part of classical stop-word lists, but which are also not likely to be part of names of mathematical domain objects. These included, for the most part, common verbs.[13]

The threshold for discarding n-gram candidates was set at five or less occurrences in the corpus (low-frequency n-grams). As in the original algorithm, the remaining n-grams were scored by taking into account their length, frequency, and the number of their nested occurrences in longer n-grams. The score threshold for discarding candidate domain terms was set at 10.[14]

3 A Taxonomy of Mathematical Objects

3.1 The resources

Mathematics Subject Classification The Mathematics Subject Classification[15] (MSC) is a hierarchically organized classification of mathematical domains encompassing over 5,000 sub-areas of mathematics and has been developed with the view to helping retrieval of documents from the AMS Mathematical Reviews Database (MathSciNet)[16] and the Zentralblatt MATH (ZMATH)[17]. Table 1 shows an excerpt from the MSC 2010 representing the first level of the "SEQUENCES, SERIES, SUMMABILITY" class. Each MSC subject class consists of a class code and a high-level class name, and includes a list of mathematical sub-areas subsumed under the given class. The sub-ares, in turn, may also

POS tagger for the domain. Small-scale experiments with tagging using off-the-shelf POS-tagging models yielded, unsurprisingly, highly sub-standard results, therefore we aim at training a dedicated tagger for mathematical discourse.

[13] The extended stop-word list did not, however, include prefixes which do occur in mathematical object names, such as "semi-", "quasi-", "sub-", "pseudo-", etc., be it hyphenated or not.

[14] For the details of the algorithm, please refer to the cited article.

[15] `http://www.ams.org/mathscinet/msc/`

[16] `http://www.ams.org/mathscinet/`

[17] `http://www.zentralblatt-math.org`

```
function ExtractMinimalLengthPaths(MSC, CMT)

MathTerms := findMultiwordTerms(MSC)
removeModifiers(mathTerms)
TopNode := CMT node with no node along "broader"-relation
SetOfPaths := ∅
foreach Term in Mathterms
   if Term occurs in CMT
      MinLengthPath=Dijkstra(TopNode, Term)
      add MinLengthPath to SetOfPaths
return SetOfPaths
```

Fig. 2. Pseudo-code of the minimal length path extraction algorithm

include sub-classes which denote more fine-grained topical distinctions within the given sub-domain. Using the domain term identification algorithm outlined above, we automatically extract mathematical domain terms contained in the names of the MSC classes.

Cambridge Mathematics Thesaurus The University of Cambridge Mathematics Thesaurus[18] (CMT) is part of the Millennium Mathematics Project.[19] The CMT contains 4,583 concepts together with short explanations and thesaurus relations such as "broader/narrower" and "references/referenced by". We exploit the thesaurus' hierarchy by following the "broader/narrower" relations in order to find hypernyms of mathematical terms.

3.2 Building the taxonomy

Automated processing First, in order to obtain a set of concepts, multi-word mathematical terms were extracted from the MSC using a variant of the domain term identification algorithm from [4] (function `findMultiwordTerms` in the pseudo-code in Figure 2). The extracted multi-word terms were simplified to single-word terms by removing their adjectival or noun modifiers using lexical rules (`removeModifiers`). The obtained set consisted of 341 unique mathematical concept names. 170 of these were also found in the CMT and were used in further automated processing.

Next, we used the "broader/narrower" relations from the CMT to traverse the CMT graph in order to retrieve the hypernyms of the extracted MSC terms. For each of the 170 terms, the algorithm first finds the root of the CMT graph (a node without any parent nodes along the "broader" relation) and then looks for the shortest path down to the given term, i.e. we find the minimal sub-graph induced by the set of common higher mathematical concepts. The algorithm is summarized in Figure 2.

[18] `http://thesaurus.maths.org`
[19] `http://mmp.maths.org/`

```
Algebraic object : Set : Semigroup : Monoid
Attribute : Quality : Property : Physical property : Position
Number : Real : Rational : Integer : Divisor
```

Fig. 3. Examples of minimal length paths extracted from the CMT

The obtained minimal length paths serve as a starting point to clustering mathematical concepts under higher-level concepts. Consider, for instance, the extracted minimal length paths corresponding to the concepts **Monoid**, **Position**, and **Divisor** shown in Figure 3. These paths allowed us to further manually classify the concepts as more general object types, e.g. **Monoid** as an **Algebraic object**, **Position** as a **Qualitative attribute**, and **Divisor** as a **Number object**. The manual classification process is summarized below.

Manual processing We manually transformed the obtained minimal length paths into paths of length at most two (i.e. each term/concept has at most one intermediate hypernym/super-concepts) obtaining the following top-level classes of mathematical objects:

1. Algebraic object : General algebraic object,
2. Algebraic object : Mapping or function,
3. Number object,
4. Notational and logical object,
5. Geometric object,
6. Qualitative attribute,
7. Method or Process.

The top-level classes were selected in such way that they are the least ambiguous in terms of classifying a mathematical concept into one of them. We therefore merged two closely related classes: **Algebraic object : Number object** and **Quantitative attribute** because the distinction between them was too fine-grained, obtaining a common class for number concepts, **Number object**. The class **Algebraic object : Mapping or function** is the result of merging **Algebraic object : Mapping** and **Algebraic object : Function**; In the CMT **Function** is subsumed both under **Algebraic object** directly and under **Map** which is also subsumed under **Algebraic object**, resulting in a cycle. In order to avoid ambiguity in interpretation, these two classes were merged.

170 MSC concepts were already subsumed under the above-mentioned classes. We then manually classified the remaining 171 MSC concepts which were not found in the CMT, obtaining a flat taxonomy of mathematical objects. An excerpt of the taxonomy is shown in Table 2.

While the flat structure captures the complexity of the relations between mathematical object types only at a coarse-grained level, this is sufficient for our current purposes for two reasons: First, given the knowledge-poor approach we pursue, we aim at a high-level classification at present, and second, the

Table 2. An excerpt from the taxonomy/the lexical resource

Mathematical object class	Set of subsumed mathematical concepts
Algebraic object: General algebraic object	array, element, field, intersection, group, module, matroid, matrix, ring, category, groupoid, set, domain, neighborhood, pair, range, region, semigroup, monoid, ...
Algebraic object: Mapping or function	code, correspondence, function, functor, intersection, metric, morphism, order, transformation, bundle, functional, mapping, norm, operator, kernel, homomorphism, ...
Number object	number, quaternion, harmonic, dimension, prime, limit, index, exponent, real, error, rational, fraction, integer, divisor, factor, quotient, residue, constant, difference, ...
Notational or logical object	equation, formula, notation, symbol, variable, unknown, index, form, representation, scheme, condition, conjecture, constraint, convention, criterion, hypothesis, lemma, ...
Geometric object	curve, path, trajectory, diagram, figure, polygon, square, graph, network, lattice, tessellation, tiling, polyhedron, torus, space, ...
Qualitative attribute	concentration, position, property, invariant, symmetry, singularity, convexity, complexity, additivity, adjunction, coherence, compactness, computability, connectedness, ...
Method or process	algorithm, inference, calculation, computation, inverse, method, transformation, dilation, reduction, glide, differentiation, integration, measurement, operation, ...

purpose of the taxonomy is to serve as a *lexical resource* with all the 341 MSC terms subsumed under one of the above-mentioned names of mathematical object classes which correspond to high-level common denotations of the sets of terms.[20] Section 4.2 shows how the sets of terms are matched with linguistic contexts in which a symbolic mathematical expression, whose interpretation is to be disambiguated, occurs. The present approach to disambiguation is at its core similar to the one introduced previously in [5], however, the new lexical resource for interpretation is superior by comparison with the one we used earlier in several respects: First, there is a clear relation between the top-level classes and the subsumed concepts: in all the cases the relation is of *is-a* type. Secondly, as mentioned above, the sets of terms themselves are coherent: they cluster terms which are hyponyms of a more general term which all of the member terms denote (at a coarse level of detail). Finally, the lexical resource comprises a smaller number of classes which should remove some spurious ambiguity in selecting a type as an interpretation of a mathematical expression.

[20] Note that EngMath [6], an existing ontology of mathematics, cannot be directly used for this purpose; EngMath is a formal ontology developed with the goal of serving as a machine-readable formal specification. Also the scope of EngMath is focused on mathematics in the engineering domain; it encompasses the following concepts: scalar, vector, and tensor quantities, physical dimensions, units of measure, functions of quantities, and dimensionless quantities (ibid.)

4 Interpreting Simple Object-Denoting Expressions

The process of interpreting mathematical expressions consists of three stages: First, the documents are preprocessed and mathematical expressions which are targets for interpretation, i.e. simple mathematical expressions, are identified (see Section 2). Then for each target mathematical expression, we calculate the similarity between the linguistic context in which it occurs and each set of mathematical terms in the lexical resource. The final interpretation of target expression is assigned using a scoring function.

4.1 Word-to-word similarity

In the disambiguation process we use similarity measures in order to decide which sets of terms from the lexical resource is closest to the lexical context of a target expression. The similarity between two words is calculated as follows:

$$\mathrm{sim}(w_1, w_2) = \begin{cases} \mathrm{Dice}(w_1, w_2) & \text{when } \mathrm{Dice}(w_1, w_2) > \lambda \\ \textit{Co-occurrence-based measure} & \text{otherwise} \end{cases} \quad (1)$$

where $\mathrm{Dice}(w_1, w_2)$ is the Dice's character-based word-to-word similarity:[21]

$$\mathrm{Dice}(w_1, w_2) = \frac{2 * n_{\text{common\_bigrams}}}{n_{\text{bigrams\_w}_1} + n_{\text{bigrams\_w}_2}} \quad (2)$$

and *Co-occurrence-based measure* is one of the following measures of lexical co-occurrence: Pointwise Mutual Information (PMI), Mutual dependency (MD), Pearson's χ^2, and Log-likelihood ratio (LL). All of these are standard corpus-based lexical association measures and have been previously successfully used in various computational linguistics tasks to estimate the relative probability with which words occur in proximity [13,3,2]. Based on experimentation we used $\lambda = 0.7$ as the threshold for using string-based similarity.

4.2 The interpretation algorithm

Before presenting the core interpretation algorithm we make precise what constitutes the components of the linguistic context of a target expression which we take into account in the course of interpretation.

The context of a mathematical expression For each target mathematical expression which we attempt to interpret, we take into account the global and local lexical context $C_C = C_L \cup C_G$ consisting of two sets of domain terms:[22]

C_L is the set of domain terms which occur in the *local context* of a mathematical expression, more specifically, within a window of textual content

[21] Dice accounts for different inflectional variants of words in the lexical resource and in the linguistic context of a mathematical expression.

[22] *Lexical* mathematical domain terms are meant here.

preceding and following the given mathematical expression, i.e. within the immediately preceding and following linguistic context,[23]

C_G is the set of domain terms which occur in the global context of the entire document. More specifically, we consider terms which occur in the *declaration statements* of the given target expression or of other expressions which are structurally similar to the target expression, according to the notion of structural similarity defined in [15].[24]

Each extracted mathematical term from C_L and C_G contributes to the final similarity score proportionally to its importance in the disambiguation process.

Disambiguation To infer the meaning a mathematical expression we use an approach inspired by methods of word sense disambiguation from computational linguistics which use inventories of word senses and measures of semantic similarity to map a word in context to its possible sense(s) from an inventory; see, for instance, [11,9].

Our approach to interpreting mathematical expressions uses the mathematical object classes shown in Table 2 on the left as the the inventory of possible "senses" of symbolic mathematical expressions. In order to identify the class which corresponds to the given use of a target mathematical expression, we map the mathematical terms (w) from the expression's context, C_C (defined above), to the mathematical terms ($term$) subsumed under each class of mathematical objects from the taxonomy ($Class$). To accomplish this, we adapt the approach to estimating the semantic similarity of two text segments T_1 and T_2 proposed in [9]. As estimates of semantic similarity between sets of words, we use the measures presented in Section 4.1. The Context-to-Class similarity is calculated using the following scoring function:

$$Sim(C, Class) = \sum_{w \in C} maxSim(w, Class) \times cw(w), \text{ where} \tag{3}$$

$$maxSim(w, Class) = \max_{term \in Class} [sim(w, term)] \tag{4}$$

[23] In the current implementation we used the window of ± 2 sentences with respect to the sentence within which the target mathematical expression occurs. Paragraph and section boundaries were not considered at this time.

[24] Identification of declaration statements was performed automatically by applying a set of regular expressions to preprocessed documents in which domain terms have been identified (see Section 2). The set was bootstrapped from a small set of seed patterns using the simple variant of the *anchored patterns* approach proposed in [7]. Using the final set of bootstrapped patterns, the algorithm achieved retrieval precision of 89% and recall of 77% on a test set. We do not include the details of the approach here. For a general description of the method, see [7].

Following [15] we consider two simple expressions to be structurally similar if they share the same top-level node in the expression tree and their expression trees have the same structure modulo the structure of the super-/subscript terms. For instance, ω_i and ω_{n-1} are structurally similar according to these criteria. By contrast, P_c^2 and A_n^k are not similar because they differ in the top-node identifier.

```
function findCandidateInterpretations(targetME)

C_G := ∅, C_L := ∅
foreach occurrence of targetME
if occurrence is explicitly declared
   add definiendum to C_G
foreach ME structurally similar to targetME
   foreach occurrence of ME
      if occurrence is explicitly declared
         add definiendum to C_G
select ±2-sentence context window W of targetME
foreach word w in W
   add w to C_L

foreach C in {C_L, C_G}
   foreach Class in Taxonomy
      compute Sim(C, Class)
      update maxSim(C, Class)
   return Class corresponding to maxSim(C, Class)
```

Fig. 4. Pseudo-code of the interpretation algorithm; targetME is a simple mathematical expressions as defined in the Introduction

$Sim(\mathrm{C}, Class)$ is computed for each class of concepts from the taxonomy and represents the similarity score between the context of the given mathematical expression and the domain terms which name concepts from the given class, C is $\mathrm{C_L}$ or $\mathrm{C_G}$ (see above), $sim(w, term)$ is the word-to-word similarity defined by Equation 1, and $\mathrm{cw}(w)$ is a weight (see below). The pseudo-code of the interpretation algorithm is shown in Figure 4.

The weight $\mathrm{cw}(w)$ is computed according to the following criteria:

- For term in $\mathrm{C_L}$ (local context; here: window of ±2-sentences) we consider the distance to the target mathematical expression with the weights decreasing with the distance in words between the term and the target expression,
- For terms in $\mathrm{C_G}$ (global context; declarations) the weights decrease from the first to the last occurrence of the expression in the document. This reflects the fact that in most cases symbols are declared with their first occurrence [15].

The final score for a *Class* as an interpretation of the given mathematical expression is computed as a combination of the local and global context scores:

$$Score(\mathrm{C_C}, Class) = \alpha Sim(\mathrm{C_G}, Class) + (1 - \alpha) Sim(\mathrm{C_L}, Class) \tag{5}$$

where $Sim(C, Class)$ was defined by Equation 3. computed for all classes from the taxonomy. The class with the maximum score is assigned as the interpretation of the given mathematical expression.[25]

5 Evaluation

In order to evaluate the interpretation procedure we created a *gold standard* set of mathematical expressions with interpretations provided by experts. The interpretation algorithm was run on the gold standard and two evaluation measures were computed for different values of the α parameter.

5.1 The gold standard

The evaluation set Mathematical expressions for the gold standard set were selected as follows: A set of 200 mathematical documents was randomly selected from the preprocessed corpus described in Section 2. Then one random simple mathematical expression was picked from each of the selected documents yielding a set of 200 occurrences of different mathematical expressions. The selected mathematical expressions were annotated by experts as described below.

Procedure The data for the gold standard was randomly split into 7 disjoint annotation sets each of which contained from 28 to 30 mathematical expressions. The annotation was performed using a web-interface we created. Mathematical expressions were presented to the annotators together with the entire document. The annotators were asked to assign a type to a symbolic expression highlighted in the document. An excerpt from the annotation instructions is shown in Figure 5. The 7 object types listed in the instructions directly corresponded to the classes from the taxonomy we used as a lexical resource.

Annotators The annotators were recruited on voluntary basis from colleagues with strong mathematical background. We contacted 18 candidate annotators, out of whom 7 responded: five were computer scientists (three post-graduates and two with doctorates), and two were working mathematicians with doctorates in mathematics. Two sets were moreover annotated by the second author of the paper. Four sets were annotated by 2 annotators in order to verify agreement. 7 identified disagreement cases were adjudicated by the authors of the paper.

[25] Note that the lexical similarity-based approach as such is language-independent; it is likely, though, that for a heavily inflected language a different threshold for word-to-word similarity would have to be used. However, because the lexical resource we use is English and because the rules for identifying declaration statements are language-specific, the evaluation is currently limited to English discourse.

Your task is to annotate symbolic mathematical expressions in mathematical documents. For each indicated expression we ask you to provide the information on the type of object the expression denotes in the given context.

For this task we distinguish seven general classes of mathematical objects or concept types [...]. These are:

1. General algebraic objects, such as "array", "element", "field", "intersection", "group", etc.
2. Algebraic objects which denote correspondences, i.e. mappings or functions, such as "correspondence", "function", "functor", "intersection", "metric", "morphism", etc.
...
7. Objects denoting methods or processes, such as "algorithm", "inference", "calculation", "computation", "inverse", "method", etc.

Many mathematical objects could be classified as more than one of the above types. For instance, many algebraic objects could be also classified as geometric objects. A manifold is such an example: it can be viewed as a set on the one hand, i.e. a general algebraic object, or, in geometry, as a mathematical space with a dimension, a geometric property, i.e. a geometric object. In cases of such ambiguities, please annotate the type corresponding to the sense in which the object is used in the given context.

Fig. 5. Excerpt from the annotation instructions

5.2 Evaluation measures

We use precision (P) and mean reciprocal rank (MRR) as evaluation measures. In classification, precision is the proportion of correctly labeled examples out of all labeled examples. MRR is one of the standard measures used in information retrieval for evaluating performance of systems which produce ranked lists of results, for example, ordered lists of documents retrieved in response to a query. It is the inverse of the rank of the expected (best) result. More specifically,

$$P = \frac{tp}{tp + fp} \times 100 \quad \text{and} \quad MRR = \frac{1}{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{N}} \frac{1}{rank_{\mathrm{i}}}$$

where tp is the number of true positive classifications, fp is the number of false positives, N is the number of evaluated instances, and $rank$ is the position of the correct classification in the list of results.

5.3 Results

Figure 6 shows the results plots for both evaluation measures at different values of α (the parameter which weighs the contribution of local vs. global context similarity scores to the overall score). The lines correspond to the different word-to-word similarity measures: Pointwise Mutual Information, Mutual Dependency, χ^2, and Log-Likelihood (see Section 4.1). Solid lines denote precision, dashed lines mean reciprocal rank.

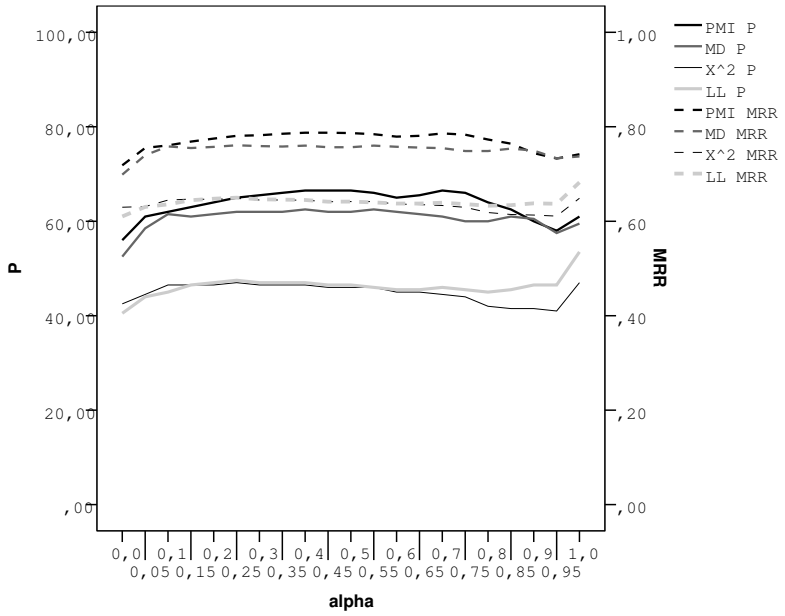


Fig. 6. Precision and mean reciprocal rank plots.

In general PMI and MD yield better results than χ^2 and LL on our data set. The maximum precision and mean reciprocal rank scores are obtained for $\alpha = \{0.40, 0.45, 0.50, 0.70\}$ using PMI as the similarity measure ($P = 66.50\%$, $MRR = 0.79$). The general pattern for PMI and MD appears the same: the combination of local and global context (α mid-range) gives better results than local or global context alone ($\alpha = 0$ and $\alpha = 1$, respectively). While yielding somewhat lower results, the MD plot line appears flatter than the PMI plot on both measures, that is, MD is more stable across the different values of α than PMI. Both χ^2 and LL perform best when relying on the global context alone, that is, when the interpretation is based solely on the explicit symbol declaration.

6 Conclusion

We presented a knowledge-poor method of finding a denotation of simple object-denoting symbolic expressions in mathematical discourse. We have shown that the lexical information from the linguistic context immediately surrounding the expression under analysis as well as the lexical information from the larger document context both contribute to achieving the best interpretation results.

Considering that the presented method relies on only limited linguistic knowledge (co-occurrence statistics over documents preprocessed using stemming and stop-word filtering), the precision results we have obtained encourage further exploration of the approach, in particular, extending it

with more linguistically-informed analysis. We are presently annotating a subset of the corpus used in the experiments described here with parts of speech tags in order to train domain-specific POS-tagging models. We expect several improvements due to POS-tagging, among others, better domain term identification and, consequently, better identification of declaration statements, as well as access to shallow syntactic analysis of the immediate context of mathematical expressions.

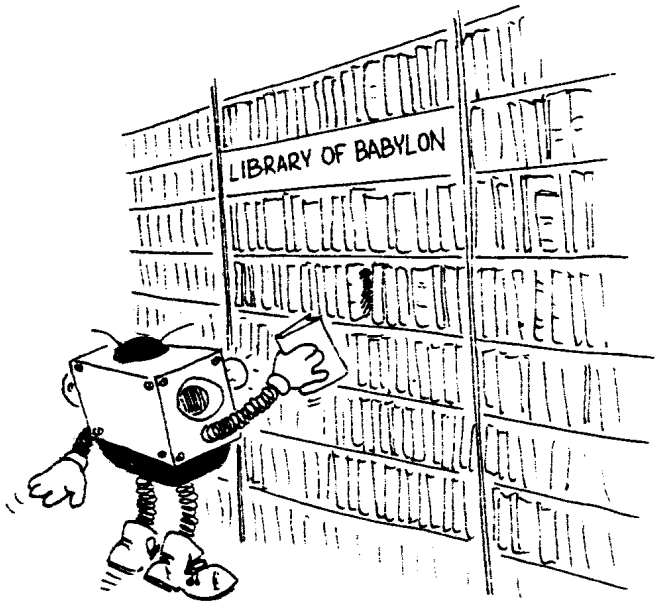
We have also shown a method of constructing a flat taxonomy of mathematical objects which can serve as a lexical resource for corpus similarity-based approaches. Multi-annotator tagging of a subset of a gold standard by two annotators, using the classes from the taxonomy as annotation labels, resulted in only 7 disagreements on 112 instances. In spite of the low disagreement count, there are at least two obvious problems with the evaluation presented here: First, admittedly, the annotation with the taxonomy classes and the evaluation was conducted on a small-scale. We are planning further annotation experiments in order to further validate the suitability of the taxonomy for the mathematical expression interpretation task. Second, from a mathematical perspective, the taxonomy we constructed is disappointingly high-level. However, this is about all we can hope for with *knowledge-poor* methods. As we already remarked in footnote 4, *knowledge-rich* methods will need a tight collaboration between experts and linguists. The former need to supply machine-understandable mathematical domain ontologies (classifications of mathematical objects and relations between them) while the latter need to adapt parsing and semantic analysis algorithms to take advantage of these and also to accomodate the fact that these ontologies are dynamic, i.e., change over the course of a document (or document collection). We conjecture that ontologies needed for document processing tasks are best created by semantically annotating (and thus partially formalizing) the mathematical documents that introduce them — a process that will have to involve linguistic analysis to scale. The knowledge-poor methods presented in this paper can be viewed as a small step in this direction.

Acknowledgments. We are indebted to Deyan Ginev of Jacobs University Bremen for compiling and preparing the corpus used in this work and for the many preprocessing scripts without which it would not have been possible to conduct this study at ease. We would like to thank the annotators who were so kind as to dedicate their time and knowledge to constructing the gold standard we used in the evaluation. We would also like to thank the three anonymous reviewers for their insightful comments and suggestions.

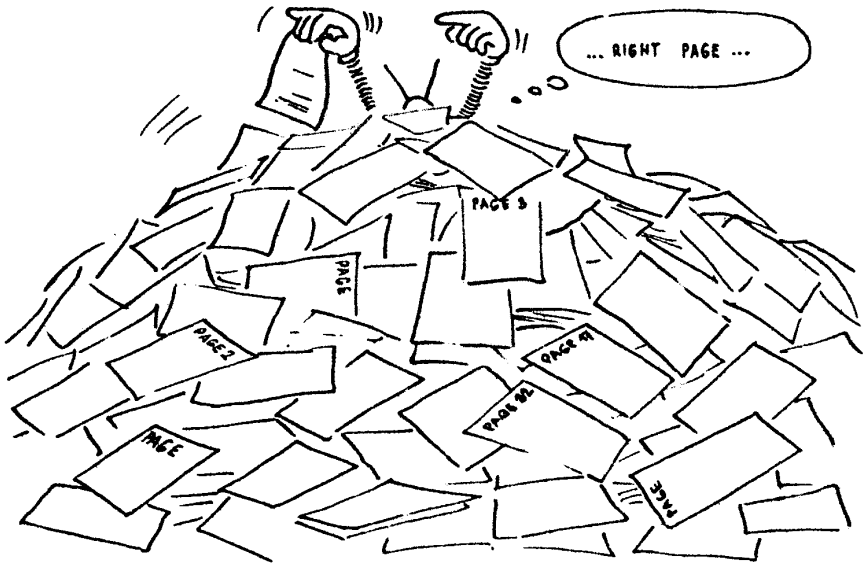
References

1. Ausbrooks, R., Carlisle, S.B.D., Chavchanidze, G., Dalmas, S., Devitt, S., Diaz, A., Dooley, S., Hunter, R., Ion, P., Kohlhase, M., Lazrek, A., Libbrecht, P., Miller, B., Miner, R., Sargent, M., Smith, B., Soiffer, N., Sutor, R., Watt, S.: Mathematical Markup Language (MathML) version 3.0. W3C Working Draft of 24. September 2009, World Wide Web Consortium (2009), `http://www.w3.org/TR/MathML3`.

2. Budiu, R., Royer, C., Pirolli, P.: Modeling information scent: a comparison of LSA, PMI-IR and GLSA similarity measures on common tests and corpora. In: Proceedings of the 8th Conference on Large Scale Semantic Access to Content (RIAO-07). pp. 314–332 (2007).
3. Bullinaria, J., Levy, J.: Extracting semantic representations from word co-occurrence statistics: A computational study. Behavior Research Methods 39(3), 510–526 (2007).
4. Frantzi, K., Ananiadou, S., Mima, H.: Automatic recognition of multi-word terms: the C-value/NC-value method. International Journal on Digital Libraries 3(2), 115–130 (2000).
5. Grigore, M., Wolska, M., Kohlhase, M.: Towards context-based disambiguation of mathematical expressions. In: Selected Papers from the joint conference of ASCM 2009 and MACIS 2009: the 9th Asian Symposium on Computer Mathematics and the 3rd International Conference on Mathematical Aspects of Computer and Information Sciences. pp. 262–271 (2009).
6. Gruber, T., Olsen, G.: An ontology for engineering mathematics. In: Proceedings 4th International Conference on Principles of Knowledge Representation and Reasoning. pp. 258–269 (1994).
7. Kozareva, Z., Riloff, E., Hovy, E.: Semantic class learning from the Web with Hyponym Pattern Linkage Graphs. In: Proceedings of the ACL/HLT-08 Conference. pp. 1048–1056 (2008).
8. McCarthy, D.: Word sense disambiguation: An overview. Language and Linguistics Compass 3(2), 537–558 (2009).
9. Mihalcea, R., Corley, C., Strapparava, C.: Corpus-based and knowledge-based measures of text semantic similarity. In: Proceedings of the 21st National Conference on Artificial Intelligence. pp. 775–780 (2006).
10. Miller, B.: `LaTeXML`: A LaTeX to XML Converter. Web Manual at `http://dlmf.nist.gov/LaTeXML/` (September 2007).
11. Pedersen, T., Banerjee, S., Patwardhan, S.: Maximizing semantic relatedness to perform word sense disambiguation. Research Report 25, University of Minnesota Supercomputing Institute (2005).
12. Stamerjohanns, H., Kohlhase, M., Ginev, D., David, C., Miller, B.: Transforming Large Collections of Scientific Publications to XML. Mathematics in Computer Science 3, 299–307 (2010).
13. Turney, P.D.: Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. In: Proceedings of the 12th European Conference on Machine Learning. pp. 491–502 (2001), `http://cogprints.org/1796/`.
14. Wessler, M.: An algebraic proof of Iitaka's conjecture. Archiv der Mathematik 79, 268–273 (2002), `http://dx.doi.org/10.1007/s00013-002-8313-2`.
15. Wolska, M., Grigore, M.: Symbol declarations in mathematical writing. In: Sojka, P. (ed.) Proceedings of the 3rd Workshop on Digital Mathematics Libraries. pp. 119–127 (2010).

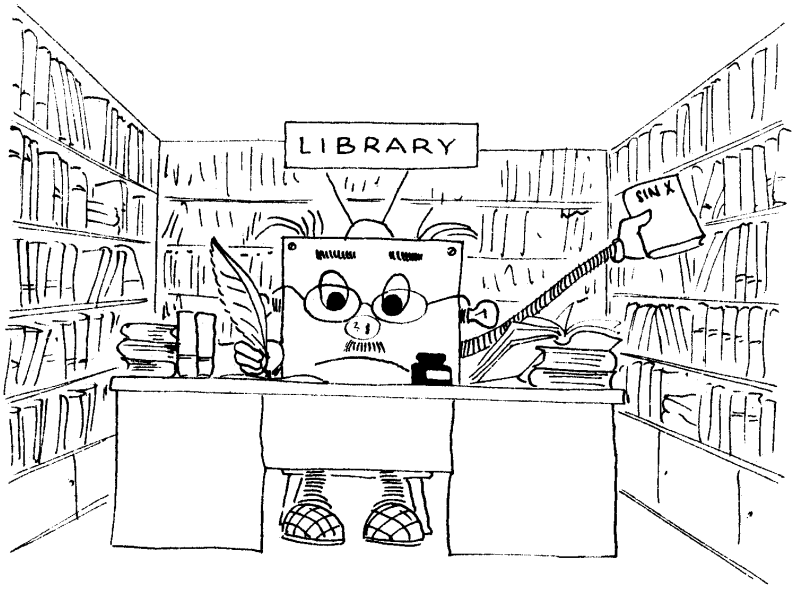

LIBRARY OF BABYLON

Subject Index



Name Index

Author Index


LIBRARY
SIN X

Colophon

The DML 2011 proceedings were produced from the authors' electronic manuscripts. Following the guidelines, the authors prepared their papers using LaTeX markup.

Contributions were edited into the uniform markup of Springer llncs style and custom-written TeX macros, and were processed by the proceedings editor in Brno.

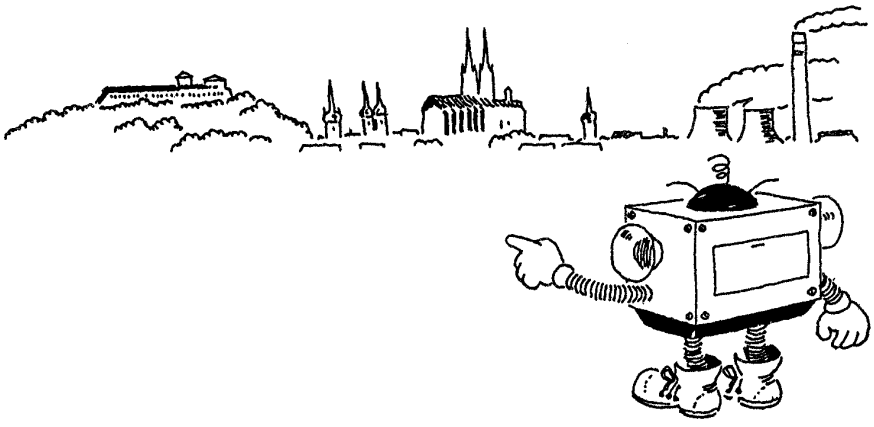
Jiří Rákosník proofread the whole book and pointed out hundreds of spelling and typographical corrections that remained in the submitted author's final versions of papers.

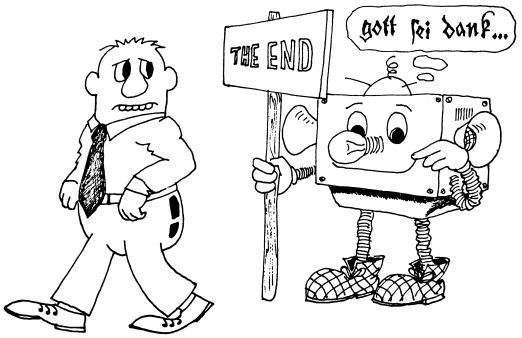
The proceedings was typeset in Palatino by Hermann Zapf and in AMS Euler fonts named after pioneering mathematician Leonhard Euler. The book was typeset using the PdfTeX typesetting system primarily developed by Hàn Thế Thành during his studies in Brno (1990–2001). Microtypographical extensions that PdfTeX implements were used, and the book was composed with the LaTeX macro package in a single PdfTeX run. Generating the hypertext version of the proceedings in PDF was done from the same source files.

The main editing, typesetting and proofreading steps were undertaken at the Natural Language Processing Laboratory of the Faculty of Informatics, Masaryk University, Brno.

The proceedings editor thanks sincerely all the authors for their contributions and everybody who was involved in the book production. Without their hard and diligent work the proceedings would not have been ready on time for the DML 2011 workshop.

Brno, July 11, 2011 Petr Sojka





DML 2011
Towards a Digital Mathematics Library
Bertinoro, Italy
July 20–21, 2011
Proceedings

Petr Sojka, Thierry Bouche (editors)

Published by Masaryk University, Brno in 2011

Typesetting, cover design: Petr Sojka

Illustrations: Jiří Franek

Printing: http://www.tribun.info
Tribun EU s.r.o., Cejl 32, 602 00 Brno
First edition, 2011

ISBN 978-80-210-5542-1