Which One Is Better: Presentation-Based or Content-Based Math Search?

Minh-Quoc Nghiem[1,4], Giovanni Yoko Kristianto[2],
Goran Topić[3], and Akiko Aizawa[2,3]

[1] Ho Chi Minh City University of Science, Vietnam
[2] The University of Tokyo, Japan
[3] National Institute of Informatics, Japan
[4] The Graduate University for Advanced Studies, Japan
{nqminh,giovanni,goran_topic,aizawa}@nii.ac.jp

Abstract. Mathematical content is a valuable information source and retrieving this content has become an important issue. This paper compares two searching strategies for math expressions: presentation-based and content-based approaches. Presentation-based search uses state-of-the-art math search system while content-based search uses semantic enrichment of math expressions to convert math expressions into their content forms and searching is done using these content-based expressions. By considering the meaning of math expressions, the quality of search system is improved over presentation-based systems.

Keywords: Math Retrieval, Content-based Math Search, MathML.

1 Introduction

The issue of retrieving mathematical content has received considerable critical attention [1]. Mathematical content is a valuable information source for many users and is increasingly available on the Web. Retrieving this content is becoming more and more important.

Conventional search engines, however, do not provide a direct search mechanism for mathematical expressions. Although these search engines are useful to search for mathematical content, these search engines treat mathematical expressions as keywords and fail to recognize the special mathematical symbols and constructs. As such, mathematical content retrieval remains an open issue.

Some recent studies have proposed mathematical retrieval systems based on the structural similarity of mathematical expressions [2–7]. However, in these studies, the semantics of mathematical expressions is still not considered. Because mathematical expressions follow highly abstract and also rewritable representations, structural similarity alone is insufficient as a metric for semantic similarity.

Other studies [8–13] have addressed semantic similarity of mathematical formulae, but this required content-based mathematical formats such as content MathML [14] and OpenMath [15]. Because almost all mathematical content available on the Web is presentation-based, these studies used two freely available toolkits, SnuggleTeX [16] and LaTeXML [17], for semantic enrichment of

S.M. Watt et al. (Eds.): CICM 2014, LNAI 8543, pp. 200–212, 2014.

mathematical expressions. However, much uncertainty remains about the relation between the performance of mathematical search system and the performance of the semantic enrichment component.

Based on the observation that mathematical expressions have meanings hidden in their representation, the primary goal of this paper is making use of mathematical expressions' semantics for mathematical search. To accomplish this problem of retrieving semantically similar mathematical expressions, we use the results of state-of-the-art semantic enrichment methods. This paper seeks the answers to two questions.

- What is the contribution of semantic enrichment of mathematical expressions to content-based mathematical search systems?
- Which one is better: presentation-based or content-based mathematical search?

To implement a *mathematical search system*, various challenges must be overcome. First, in contrast to text which is linear, mathematical expressions are hierarchical: operators have different priorities, and expressions can be nested. The similarity between two mathematical expressions is decided first by their structure and then by the symbols they contain [18, 19]. Therefore, current text retrieval techniques cannot be applied to mathematical expressions because they only consider whether an object includes certain words. Second, mathematical expressions have their own meanings. These meanings can be encoded using special markup languages such as Content MathML or OpenMath. A few existing mathematical search systems also make use of this information. Such markup, however, is rarely used to publish mathematical knowledge related to the Web [18]. As a result, we were only able to use presentation-based markup, such as Presentation MathML or $\TeX$, for mathematical expressions.

This paper presents an approach to a *content-based mathematical search system* that uses the information from *semantic enrichment of mathematical expressions* system. To address the challenges described above, the proposed approach is described below. First, the approach used Presentation MathML markup, a widely used markup for mathematical expressions. This makes our approach more likely to be applicable in practice. Second, a *semantic enrichment of mathematical expressions* system is used to convert mathematical expressions to Content MathML. By getting the underlying semantic meanings of mathematical expressions, a *mathematical search system* is expected to yield better results.

The remainder of this paper is organized as follows. Section 2 provides a brief overview of the background and related work. Section 3 presents our method. Section 4 describes the experimental setup and results. Section 5 concludes the paper and points to avenues for future work.

2 Mathematical Search System

As the demand for mathematical searching increases, several mathematical retrieval systems have come into use [20]. Most systems use the conventional text search techniques to develop a new mathematical search system [2, 3]. Some systems use specific format for mathematical content and queries [4–7, 11]. Based on the markup schema they use, current mathematical search systems are divisible into presentation-based and content-based systems. Presentation-based systems deal with the presentation form whereas content-based systems deal with the meanings of mathematical formulae.

2.1 Presentation-Based Systems

Springer LaTeXSearch. Springer offers a free service, Springer LaTeX Search [3], to search for LaTeX code within scientific publications. It enables users to locate and view equations containing specific LaTeX code, or equations containing LaTeX code that is similar to another LaTeX string. A similar search in Springer LaTeX Search ranks the results by measuring the number of changes between a query and the retrieved formulae. Each result contains the entire LaTeX string, a converted image of the equation, and information about and links to its source.

MathDeX. MathDeX (formerly MathFind [21]) is a math-aware search engine under development by Design Science. This work extends the capabilities of existing text search engines to search mathematical content. The system analyzes expressions in MathML and decomposes the mathematical expression into a sequence of text-encoded math fragments. Queries are also converted to sequences of text and the search is performed as a normal text search.

Digital Library of Mathematical Functions. The Digital Library of Mathematical functions (DLMF) project at NIST is a mathematical database available on the Web [2, 22]. Two approaches are used for searching for mathematical formulae in DLMF. The first approach converts all mathematical content to a standard format. The second approach exploits the ranking and hit-description methods. These approaches enable simultaneous searching for normal text as well as mathematical content.

In the first approach [4], they propose a textual language, Textualization, Serialization and Normalization (TexSN). TeXSN is defined to normalize non-textual content of mathematical content to standard forms. User queries are also converted to the TexSN language before processing. Then, a search is performed to find the mathematical expressions that match the query exactly. As a result, similar mathematical formulae are not retrieved.

In the second approach [23], the search system treats each mathematical expression as a document containing a set of mathematical terms. The cited paper introduces new relevance ranking metrics and hit-description generation techniques. It is reported that the new relevance metrics are far superior to the conventional tf-idf metric. The new hit-descriptions are also more query-relevant and representative of the hit targets than conventional methods.

Other notable math search systems include Math Indexer and Searcher [24], EgoMath [25], and ActiveMath [26].

2.2 Content-Based Systems

Wolfram Function. The Wolfram Functions Site [8] is the world's largest collection of mathematical formulae accessible on the Web. Currently the site has 14 function categories containing more than three hundred thousand mathematical formulae. This site allows users to search for mathematical formulae from its database. The Wolfram Functions Site proposes similarity search methods based on MathML. However, content-based search is only available with a number of predefined constants, operations, and function names.

MathWebSearch. The MathWebSearch system [10, 12] is a content-based search engine for mathematical formulae. It uses a term indexing technique derived from an automated theorem proving to index Content MathML formulae. The system first converts all mathematical formulae to Content MathML markup and uses substitution-tree indexing to build the index. The authors claim that search times are fast and unchanged by the increase in index size.

MathGO! [9] proposed a mathematical search system called the MathGO! Search System. The approach used conventional search systems using regular expressions to generate keywords. For better retrieval, the system clustered mathematical formula content using K-Som, K-Means, and AHC. They did experiments on a collection of 500 mathematical documents and achieved around 70–100 percent precision.

MathDA. Yokoi and Aizawa [11] proposed a similarity search method for mathematical expressions that is adapted specifically to the tree structures expressed by MathML. They introduced a similarity measure based on Subpath Set and proposed a MathML conversion that is apt for it. Their experiment results showed that the proposed scheme can provide a flexible system for searching for mathematical expressions on the Web. However, the similarity calculation is the bottleneck of the search when the database size increases. Another shortcoming of this approach is that the system only recognizes symbols and does not perceive the actual values or strings assigned to them.

System of Nguyen et al. [13] proposed a math-aware search engine that can handle both textual keywords and mathematical expressions. They used Finite State Machine model for feature extraction, and representation framework captures the semantics of mathematical expressions. For ranking, they used the passive–aggressive on-line learning binary classifier. Evaluation was done using 31,288 mathematical questions and answers downloaded from Math Overflow [27]. Experimental results showed that their proposed approach can perform better than baseline methods by 9%.

3 Methods

The framework of our system is shown in Fig. 1. First, the system collects mathematical expressions from the web. Then the mathematical expressions are converted to Content MathML using the *semantic enrichment of mathematical expressions* system of Nghiem et. al [28]. Indexing and ranking the mathematical expressions are done using Apache Solr system [29] following the method described in Topić et. al [30]. When a user submits a query, the system also converts the query to Content MathML. Then the system returns a ranked list of mathematical expressions corresponding to the user's queries.

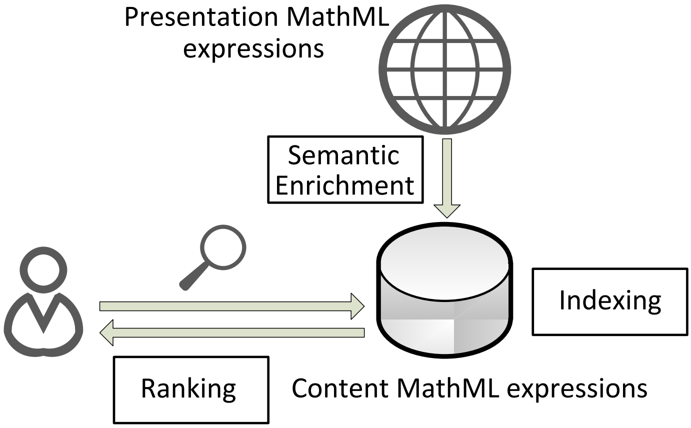

Fig. 1. System Framework

3.1 Data Collection

Performance analysis of a mathematical search system is not an easy task because few standard benchmark datasets exist, unlike other more common information retrieval tasks. Mathematical search systems normally build their own mathematical search dataset for evaluation by crawling and downloading mathematical content from the web. Direct comparison of the proposed approach with other systems is also hard because they are either unavailable or inaccessible.

Recently, simpler and more rapid tests of mathematical search system have been developed. The NTCIR-10 Math Pilot Task [1] was the initial attempt to develop a common workbench for mathematical expressions search. Currently, the NTCIR-10 dataset contains 100,000 papers and 35,000,000 mathematical expressions from ArXiv [31] which includes Content MathML markup. The task was completed as an initial pilot task showing a clear interest in the mathematical search. However, the Content MathML markup expressions are generated automatically using the LaTeXML toolkits. Therefore, this dataset is unsuitable to serve as the gold standard for the research described in the present paper.

As Wolfram Functions Site [8] is the only website that provides high-quality Content MathML markup for every expression, data for the search system was collected from this site. The Wolfram Functions Site data have numerous attractive features, including both Presentation and Content MathML markups, and category for each mathematical expression. In the experiment, the performance of *semantic enrichment of mathematical expressions* component will be compared directly with the system performance obtained using correct Content MathML expressions on Wolfram Functions Site data.

3.2 Semantic Enrichment of Mathematical Expressions

The mathematical expressions were preprocessed according to the procedure described in Nghiem et. al [28]. Given a set of training mathematical expressions in MathML parallel markup, rules of two types are extracted: segmentation rules and translation rules. These rules are then used to convert mathematical expressions from their presentation to their content form. Translation rules are used to translate (sub)trees of Presentation MathML markup to (sub)trees of Content MathML markup. Segmentation rules are used to combine and reorder the (sub)trees to form a complete tree.

After using mathematical expression enrichment system to convert the expressions into content MathML, we use these converted expressions for indexing. The conversion is not a perfect conversion, so there are terms that could not be converted. The queries submitted to the search system are also processed using the same conversion procedure.

3.3 Indexing

The indexing step was prepared by adapting the procedure used by Topić et. al [30]. This procedure used *pq*-gram-like indexing for Presentation MathML expressions. We modified it for use with Content MathML expressions. There are three fields used to encode the structure and contents of a mathematical expression: `opaths`, `upaths`, and `sisters`. Each expression is transformed into a sequence of keywords across several fields. `opaths` (ordered paths) field gathers the XML expression tree in vertical paths with preserved ordering. `upaths` (unordered paths) works the same as `opaths` without the ordering information. `sisters` lists the sister nodes in each subtree. Figure 2 presents an example of the terms used in the index of the expression $\sin(\frac{\pi}{8})$:$< apply >< sin/ >< apply >< times/ >< pi/ >< apply >< power/ >< cntype = “integer” > 8 < /cn >< cntype = “integer” > -1 < /cn >< /apply >< /apply >< /apply >$.

3.4 Searching

In the mathematical search system, users can input mathematical expressions using presentation MathML as a query. The search system then uses the *semantic enrichment of mathematical expressions* module to convert the input expressions to Content MathML. Figure 3 presents an example of the terms used in the query of the expression $\sin(\frac{\pi}{8})$. Matching is then performed using eDisMax, the default query parser of Apache Solr. Ranking is also done using the default modified TF/IDF scores and length normalization of Apache Solr.

```
opaths:
    1#1#apply 1#1#1#sin 1#1#2#apply 1#1#2#1#times 1#1#2#2#pi
    1#1#2#3#apply 1#1#2#3#1#power 1#1#2#3#2#cn#8 1#1#2#3#3#cn#-1
opaths:
    1#apply 1#1#sin 1#2#apply 1#2#1#times 1#2#2#pi 1#2#3#apply
    1#2#3#1#power 1#2#3#2#cn#8 1#2#3#3#cn#-1
opaths:
    apply 1#sin 2#apply 2#1#times 2#2#pi 2#3#apply 2#3#1#power 2#3#2#cn#8
    2#3#3#cn#-1
opaths: sin
opaths: times
opaths: pi
opaths: apply 1#power 2#cn#8 3#cn#-1
opaths: power
opaths: cn#8
opaths: cn#-1
upaths:
    ##apply ###sin ###apply ####times ####pi ####apply #####power
    #####cn#8 #####cn#-1
upaths:
    #apply ##sin ##apply ###times ###pi ###apply ####power ####cn#8
    ####cn#-1
upaths:
    apply #sin #apply ##times ##pi ##apply ###power ###cn#8 ###cn#-1
upaths: sin
upaths: apply #times #pi #apply ##power ##cn#8 ##cn#-1
upaths: times
upaths: pi
upaths: apply #power #cn#8 #cn#-1
upaths: power
upaths: cn#8
upaths: cn#-1
sisters: power cn#8 cn#-1
sisters: times pi apply
sisters: sin apply
sisters: apply
```

Fig. 2. Index terms of the expression sin($\frac{\pi}{8}$)

4 Experimental Results

4.1 Evaluation Setup

We collected mathematical expressions for evaluation from the Wolfram Function Site. At the time collected, there were more than 300,000 mathematical expressions on this site. After collection, we filtered out long expressions containing more than 20 leaf nodes to speed up the semantic enrichment because the processing time increases exponentially with the length of the expressions.

```
opaths:
    1#1#apply 1#1#1#sin 1#1#2#apply 1#1#2#1#times 1#1#2#2#pi
    1#1#2#3#apply 1#1#2#3#1#power 1#1#2#3#2#cn#8 1#1#2#3#3#cn#-1
upaths:
    ##apply ###sin ###apply ####times ####pi ####apply #####power
    #####cn#8 #####cn#-1
upaths:
    #apply ##sin ##apply ###times ###pi ###apply ####power ####cn#8
    ####cn#-1
sisters: power cn#8 cn#-1
sisters: times pi apply
sisters: sin apply
sisters: apply
```

Fig. 3. Query terms of the expression $\sin(\frac{\pi}{8})$

The number of mathematical expressions after filtering is approximately 20,000. Presumably, this number is adequate for evaluating the mathematical search system.

Evaluation was done by comparing three systems:

- Presentation-based search with Presentation MathML (PMathML): indexing and searching are based on the Presentation MathML expressions.
- Content-based search with semantic enrichment (SE): indexing and searching are based on the Content MathML expressions. The Content MathML expressions are extracted automatically using semantic enrichment module.
- Content-based search with correct Content MathML (CMathML): indexing and searching are based on the Content MathML expressions. The Content MathML expressions are those from the Wolfram Function Site.

We used the same data to train the semantic enrichment module by 10-fold cross validation method. The data is divided into 10 folds. The semantic enrichment result of each fold was done by using the other 9 folds as training data.

4.2 Evaluation Methodology

We used "Precision at 10" and "normalized Discounted Cumulative Gain" metrics to evaluate the results. In a large-scale search scenario, users are interested in reading the first page or the first three pages of the returned results. "Precision at 10" (P@10) has the advantage of not requiring the full set of relevant mathematical expressions, but its salient disadvantage is that it fails to incorporate consideration of the positions of the relevant expressions among the top k. In a ranked retrieval context, normalized Discounted Cumulative Gain (nDCG) as given by Equation 1 is a preferred metric because it incorporates the order of the retrieved expressions. In Equation 1, Discounted Cumulative Gain (DCG) can be calculated using the Equation 2, where rel_i is the graded relevance of the result at position i. Ideal DCG (IDCG) is calculable using the same equation, but IDCG uses the ideal result list which was sorted by relevance.

$$\mathrm{nDCG_p} = \frac{DCG_p}{IDCG_p} \tag{1}$$

$$\mathrm{DCG_p} = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2(i)} \tag{2}$$

For performance analysis of the mathematical search system, we manually created 15 information needs (queries) and used them as input queries of our mathematical search system. The queries are created based on NTCIR queries with minor modification. Therefore, the search system always gets at least one exact match. Table 1 shows the queries we used. The top 10 results of each query were marked manually as relevant ($rel = 1$), non-relevant ($rel = 0$), or partially relevant ($rel = 0.5$). The system then calculates P@10 and an nDCG value based on the manually marked results.

Table 1. Queries

No.	Query
1	$\int_0^\infty x\,\mathrm{d}x$
2	$x^2 + y^2$
3	$\int_0^\infty e^{-x^2}\,\mathrm{d}x$
4	$arcsin(x)$
5	k^2
6	$\frac{\cosh ez+\sinh ez}{e}$
7	$\mathcal{R}_z\Psi^\nu(z), \tilde{\infty}$
8	$\int \frac{a^{d+bz}}{z}\mathrm{d}z$
9	$\lim_{\nu\to\infty} \frac{L_{\alpha+\nu}}{L_\nu}$
10	$\mathcal{BP}_z\mathfrak{B}_\nu^\mu(z)$
11	$\nu \in \mathbb{N}$
12	$\Psi^\nu(z)$
13	$\log(z+1)$
14	$H_n(z)$
15	$\frac{1}{\pi}\int_0^\pi(\cos tn - z\sin t)\mathrm{d}t$

4.3 Experimental Results

Comparisons among the three systems were made using P@10 and nDCG scores. Table 2 and figure 5 show the P@10 and nDCG scores obtained from the search. Figure 4 depicts the top 10 precision of the search system. The x axis shows the k number, which ranges from 1 to 10. The y axis shows the precision score. The precision score decreased, while k increased, which indicates that the higher results are more relevant than lower results.

Table 2. nDCG and P@10 scores of the search systems

Method	nDCG	P@10
PMathML	0.941	0.707
CMathML	0.962	0.747
SE	0.951	0.710

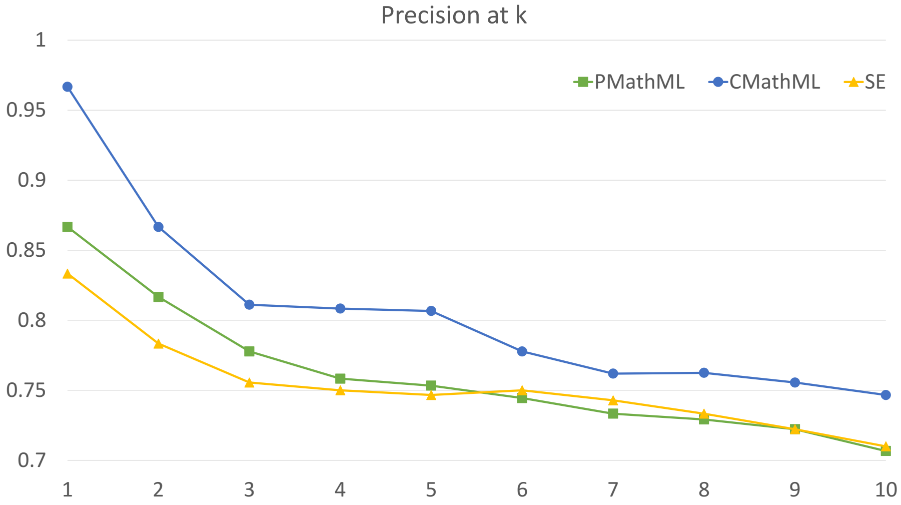

Fig. 4. Top 10 precision of the search system

In the experiment, a strong relation between *semantic enrichment of mathematical expressions* and *content-based mathematical search system* was found. As shown in Nghiem et. al [28], the error rate of *semantic enrichment of mathematical expressions* module is around 29 percent. With current performance, using this module for the mathematical search system still improves the search performance. The system gained 1 percent in nDCG score and 0.3 percent in P@10 score compared to the Presentation MathML-based system. Overall, the system using perfect Content MathML yielded the highest results. In direct comparison using nDCG scores, the system using semantic enrichment is superior to the Presentation MathML-based system, although not by much. Out of 15 queries, the semantic enrichment system showed better results than Presentation MathML-based system in 7 queries, especially when the mathematical symbols contain specific meanings, e.g. Poly-Gamma function (query 10), Hermite-H function (query 14). In case the function has specific meaning but there is no ambiguity representing the function, e.g. Legendre-Q function (query 12), both systems give similar results. Presentation MathML system, however, produced better results than semantic enrichment systems in 5 queries when dealing with elementary functions (query 2, 8, 15), logarithm (query 13), and trigonometric functions (query 6) because of its simpler representation using Presentation

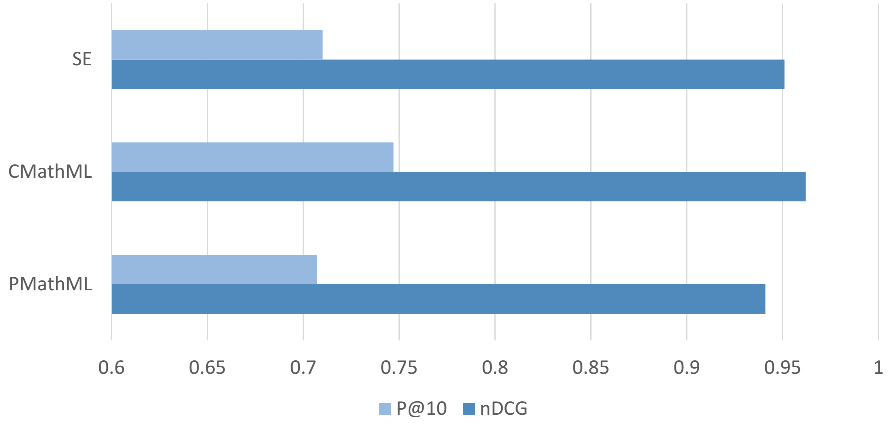

Fig. 5. Comparison of different systems

MathML. One exception is the case of query 4, when there is more than one way to represent an expression with a specific meaning, e.g. sin^{-1} and $arcsin$, Presentation MathML system gives unstable results.

This finding, while preliminary, suggests that we can choose either search strategy depending on the situation. We can use Presentation MathML system for elementary functions or when there is no ambiguity in the Presentation MathML expression. Otherwise, we can use a Content MathML system while dealing with functions that contain specific meanings. Another situation in which we can use a Content MathML system is when there are many ways to present an expression using Presentation MathML markup.

The average time for searching for a mathematical expression is less than one second on our Xeon 32 core 2.1 GHz 32 GB RAM server. The indexing time, however, took around one hour for 20,000 mathematical expressions. Because of the unavailability of standard corpora to evaluate content-based mathematical search systems, the evaluation at this time is quite subjective and limited. Although this study only uses 20,000 mathematical expressions for the evaluation, the preliminary experimentally obtained results indicated that the semantic enrichment approach showed promise for content-based mathematical expression search.

5 Conclusion

By using semantic information obtained from semantic enrichment of mathematical expressions system, the content-based mathematical search system has shown promising results. The experimental results confirm that semantic information is helpful to the mathematical search. Depending on the situation, we can choose to use either presentation-based or content-based strategy for searching. However, this is only a first step; many important issues remain for future studies. Considerably more work will need to be done using a larger collection of queries. In addition, there are many other valuable features that are worth considering besides the semantic markup of an expression, such as the description of the formula and its variables.

Acknowledgments. This work was supported by JSPS KAKENHI Grant Numbers 2430062, 25245084.

References

1. Aizawa, A., Kohlhase, M., Ounis, I.: NTCIR-10 Math pilot task overview. In: National Institute of Informatics Testbeds and Community for Information access Research 10 (NTCIR-10), pp. 654–661 (2013)
2. National Institute of Standards and Technology: Digital library of mathematical functions, `http://dlmf.nist.gov` (visited on March 01, 2014)
3. Springer: Springer LaTeX Search, `http://www.latexsearch.com/` (visited on March 01, 2014)
4. Youssef, A.S.: Information search and retrieval of mathematical contents: Issues and methods. In: The ISCA 14th International Conference on Intelligent and Adaptive Systems and Software Engineering, pp. 100–105 (2005)
5. Altamimi, M.E., Youssef, A.S.: A math query language with an expanded set of wildcards. Mathematics in Computer Science 2, 305–331 (2008)
6. Youssef, A.S., Altamimi, M.E.: An extensive math query language. In: SEDE, pp. 57–63 (2007)
7. Miner, R., Munavalli, R.: An approach to mathematical search through query formulation and data normalization. In: Kauers, M., Kerber, M., Miner, R., Windsteiger, W. (eds.) MKM/CALCULEMUS 2007. LNCS (LNAI), vol. 4573, pp. 342–355. Springer, Heidelberg (2007)
8. Wolfram: The Wolfram Functions Site, `http://functions.wolfram.com/` (visited on March 01, 2014)
9. Adeel, M., Cheung, H.S., Khiyal, S.H.: Math go! prototype of a content based mathematical formula search engine. Journal of Theoretical and Applied Information Technology 4(10), 1002–1012 (2008)
10. Kohlhase, M., Sucan, I.: A search engine for mathematical formulae. In: Calmet, J., Ida, T., Wang, D. (eds.) AISC 2006. LNCS (LNAI), vol. 4120, pp. 241–253. Springer, Heidelberg (2006)
11. Yokoi, K., Aizawa, A.: An approach to similarity search for mathematical expressions using mathml. In: 2nd Workshop Towards a Digital Mathematics Library, DML 2009, pp. 27–35 (2009)
12. Kohlhase, M., Prodescu, C.C.: Mathwebsearch at NTCIR-10. In: National Institute of Informatics Testbeds and Community for Information access Research 10 (NTCIR-10), pp. 675–679 (2013)
13. Nguyen, T.T., Chang, K., Hui, S.C.: A math-aware search engine for math question answering system. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM 2012), pp. 724–733 (2012)
14. Ausbrooks, R., Buswell, S., Carlisle, D., Chavchanidze, G., Dalmas, S., Devitt, S., Diaz, A., Dooley, S., Hunter, R., Ion, P., et al.: Mathematical markup language (MathML) version 3.0. W3C recommendation. World Wide Web Consortium (2010)

15. Buswell, S., Caprotti, O., Carlisle, D.P., Dewar, M.C., Gaetano, M., Kohlhase, M.: The openmath standard. Technical report, version 2.0. The Open Math Society (2004)
16. McKain, D.: SnuggleTeX version 1.2.2, `http://www2.ph.ed.ac.uk/snuggletex/` (visited on March 01, 2014)
17. Miller, B.R.: LaTeXML a LaTeX to XML converter, `http://dlmf.nist.gov/LaTeXML/` (visited on March 01, 2014)
18. Kamali, S., Tompa, F.W.: Improving mathematics retrieval. In: 2nd Workshop Towards a Digital Mathematics Library, pp. 37–48 (2009)
19. Kamali, S., Tompa, F.W.: Structural similarity search for mathematics retrieval. In: Carette, J., Aspinall, D., Lange, C., Sojka, P., Windsteiger, W. (eds.) CICM 2013. LNCS (LNAI), vol. 7961, pp. 246–262. Springer, Heidelberg (2013)
20. Zanibbi, R., Blostein, D.: Recognition and retrieval of mathematical expressions. IJDAR 15, 331–357 (2012)
21. Munavalli, R., Miner, R.: Mathfind: a math-aware search engine. In: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 735. ACM (2006)
22. Miller, B.R., Youssef, A.S.: Technical aspects of the digital library of mathematical functions. Annals of Mathematics and Artificial Intelligence 38, 121–136 (2003)
23. Youssef, A.S.: Methods of relevance ranking and hit-content generation in math search. In: Kauers, M., Kerber, M., Miner, R., Windsteiger, W. (eds.) MKM/Calculemus 2007. LNCS (LNAI), vol. 4573, pp. 393–406. Springer, Heidelberg (2007)
24. Sojka, P., Líška, M.: The Art of Mathematics Retrieval. In: Proceedings of the ACM Conference on Document Engineering, DocEng 2011, Mountain View, CA, pp. 57–60. Association of Computing Machinery (2011)
25. Mišutka, J., Galamboš, L.: System description: EgoMath2 as a tool for mathematical searching on wikipedia.org. In: Davenport, J.H., Farmer, W.M., Urban, J., Rabe, F. (eds.) Calculemus/MKM 2011. LNCS, vol. 6824, pp. 307–309. Springer, Heidelberg (2011)
26. Siekmann, J.: Activemath, `http://www.activemath.org/eu/` (visited on March 01, 2014)
27. MathOverflow: Math overflow, `http://mathoverflow.net/` (visited on March 01, 2014)
28. Nghiem, M.Q., Kristianto, G.Y., Aizawa, A.: Using mathml parallel markup corpora for semantic enrichment of mathematical expressions. Journal of the Institute of Electronics, Information and Communication Engineers E96-D(8), 1707–1715 (2013)
29. Apache: Apache solr, `http://lucene.apache.org/solr/` (visited on March 01, 2014)
30. Topic, G., Kristianto, G.Y., Nghiem, M.Q., Aizawa, A.: The MCAT math retrieval system for NTCIR-10 Math track. In: National Institute of Informatics Testbeds and Community for Information access Research 10 (NTCIR-10), pp. 680–685 (2013)
31. Cornell University Library: arxiv, `http://arxiv.org/` (visited on March 01, 2014)