NTCIR-10 Math Pilot Task Overview

Akiko Aizawa
National Institute of Informatics
aizawa@nii.ac.jp

Michael Kohlhase
Jacobs University Bremen
m.kohlhase@jacobs-university.de

Iadh Ounis
University of Glasgow
Iadh.Ounis@glasgow.ac.uk

ABSTRACT

This paper presents an overview of a new pilot task, the NTCIR Math Task, which is specifically dedicated to information access to mathematical content. In particular, the paper summarizes the subtasks addressed at the NTCIR Math Task as well as the main approaches deployed by the participating groups.

Team Name

MATH

Subtasks

Math Retrieval (English)
Math Understanding (English)

Keywords

mathematical information retrieval, MathML, query language, evaluation, MIR, IR

1. INTRODUCTION

Mathematical formulae are important means for dissemination and communication of scientific information. They are not only used for numerical calculation but also for clarifying definitions or disambiguating explanations that are written in natural language.

Despite the importance of Math is written documents, most of the contemporary retrieval systems do not support users' access to mathematical formulae in target documents. The major obstacles for the research are the lack of readily available large-scale datasets with structured mathematical formulae, carefully designed tasks, and established evaluation methods. Motivated by the current situation, the NTCIR Math Task (see [NTM13]) aims at the establishment of new challenges in this area by providing a shared dataset and a common evaluation platform to researchers in related fields.

The NTCIR-10 Workshop took place from 03/2012 to 06/2013. During that period, participants were encouraged to initially join the *dry run* and after that, the *formal run*. In the dry run, initial datasets were distributed to the participants to facilitate the development and tuning of their retrieval systems. The initial datasets were also carefully investigated to improve the task design and the evaluation measures. Feedback from the participants was also considered during this period. In the formal run, datasets of larger scale, as well as the topics for the evaluation were released. The participants were requested to submit their formal results in the form of *runs*. These runs were then pooled and reviewed by mathematician panels, who acted as assessors supported by the organizers. Finally, the topics, the performance results, and the evaluation tools are made available to the research community for future use.

In NTCIR-10, the NTCIR-Math Task was organized as two independent subtasks: the first is the Math Retrieval Subtask, and the second is the Math Understanding Subtask. Each subtask is described briefly in the following sections.

2. PARTICIPATION

Sixteen groups registered to the NTCIR-10 Math Pilot Task and six groups submitted their results. KWARC and MCAT are the organizers' groups.

Table 1: NTCIR-10 Math Pilot Task Participants.

Group ID	Organization
BRKLY	University of California, USA
FSE	Technische Universität Berlin, Germany
KWARC	Jacobs University, Germany
MCAT	National Institute of Informatics, Japan
MIRMU	Masaryk University, Czech Republic
NAK	Keio University, Japan

Each group could submit up to four runs. Table 2 shows the number of runs submitted for each subtask category. All the six participating teams contributed to the Math Retrieval Subtask (MR) and one team made their submission to the Math Understanding (MU) Subtask. Math Retrieval Subtask has three query types: Formula Search (FS), Full-Text Search (FT), and Open Information Retrieval (OIR). The numbers of the runs for these categories were also shown in Table 2.

3. MATH RETRIEVAL SUBTASK

In the following, we describe in details the Math Retrieval Subtask, its corresponding search scenarios, as well as the topic development phase. We also briefly describe the used assessment procedure, and summarize the main reported results.

Table 2: Number of runs for each subtask category.

Group ID	Subtasks			
	MIR/FS	MIR/FT	MIR/OIR	MU
BRKLY	4	1*	–	–
FSE	1	1	–	–
KWARC	1	–	–	–
MCAT	1	2	–	4
MIRMU	4	1*	1*	–
NAK	1	–	–	–
Total	12	3(2*)	0 (1*)	4

∗ Reported only document URIs without formula IDs and were not included in the relevance judgment pool.

3.1 Task Design

3.1.1 Math Retrieval Subtask

The Math Retrieval Subtask is designed as a question-and-answer task over a set of 100.000 documents from mathematics, physics, and computer science. The Subtask has challenges in three different search scenarios:

Formula Search (automated) Given a list of formula queries (formulae with query variables that act as wildcards), search the formula database of the used dataset.

Full-Text Search (automated) Search the document collection using combinations of keywords and formula queries.

Open Information Retrieval (semi-automated) Search the document collection using free textual queries.

The queries from the three challenges were numbered with "FS-1" to "FS-22", "FT-1" to "FT-15", and "OMIR-1" to "OMIR-19" respectively. Each participant could submit up to 100 results per query and run (there could be up to four runs, specified by ⟪RunTag⟫).

The document collection of the Math Retrieval Subtask were obtained from the arXMLiv Project [SKG+10, arXb], which converts LaTeX sources from the Cornell ePrint arXiv [ArXa] into XML for processing and analysis. Documents are in the XHTML format, the embedded formulae in *presentation MathML* (an XML format that concentrates on layout of formulase), *content MathML* (an XML format for the functional structure – the operator tree – of a formula), and the original LaTeX source; see Figure 1[1].

The size of the NTCIR-10 Math dataset is ca. 63 GiB, it contains 35.5 million formulae with 297 million subformulae (the targets for formula search). The distribution of depths[2] and sizes[3] of these formulae is given on the right.

	min	mean	max
depth	0	1.94	29
size	1	11.35	2625

For the simple arithmetic expression $\left(\frac{p-2}{p-1}\right)^{p-1}$ (represented as \left(\frac{p-2}{p-1}\right)^{p-1} in LaTeX) the MathML is given in Figure 2. Let us briefly contrast presentation/content markup in this example: on the left we see XML elements like <msup>, which specify that the second child should be laid out as the upper index of the first and <mfrac> for fractions. On the right, we have XML elements like <apply> for function application and <divide/> for the division operator.

```
<math id="fid1">
  <semantics>
    ⟪Presentation Markup⟫
    <annotation-xml id="fid2"
        encoding="MathML-Content">
      ⟪Content Markup⟫
    </annotation-xml>
    <annotation encoding="application/x-tex">
      ⟪LaTeX⟫
    </annotation>
  </semantics>
</math>
```

Figure 1: XML encoding for MathML Formulae

presentation	content
<msup> <mfenced> <mfrac> <mrow> <mi>p</mi> <mo>−</mo> <mn>2</mn> </mrow> <mrow> <mi>p</mi> <mo>−</mo> <mn>1</mn> </mrow> </mfrac> </mfenced> <mrow> <mi>p</mi> <mo>−</mo> <mn>1</mn> </mrow> </msup>	<apply> <exp/> <apply> <divide/> <apply> <minus/> <ci>p</ci> <cn>2</cn> </apply> <apply> <minus/> <ci>p</ci> <cn>1</cn> </apply> </apply> <apply> <minus/> <ci>p</ci> <cn>1</cn> </apply> </apply>

Figure 2: Presentation/Content MathML for 1

3.2 Topic Development

Formula queries are encoded just as normal formulae in presentation and content MathML, but may also contain named query variables that act as wildcards. A query variable with name foo is represented by the XML element <mws:qvar name="foo"/>; we write it as ?foo in LaTeX and presented formulae. $\frac{?f(?v+?d)-?f(?v)}{?d}$ is a typical example for a formula query[4]; here ?f, ?v, and ?i are query variables.

[1]Here and in the following we disregard XML namespaces for legibility.

[2]The depth of a formula is the depth (longest path) of the content MathML tree (single cis have depth 0).

[3]The size of a formula is the number of nodes in the content MathML tree.

[4]This was query FS-05 in the NTCIR 10 Math Challenge.

This formula query matches the definition

$$g'(cx) = \lim_{h \to 0} \frac{g(cx+h) - g(cx)}{h} \quad (1)$$

since we can substitute g for ?f, cx for ?v, and h for ?i to obtain the subformula $\frac{g(cx+h)-g(cx)}{h}$ of (1). Note that depending on whether we express the query in content or presentation MathML we may obtain different results: presentation MathML distinguishes the variants $\frac{n}{d}$, $n : d$ and n/d of a fraction, while content MathML only sees them as applications of the division function to n and d.

For the NTCIR-10 Math Task, we collected 22 formula queries, 15 full text queries, and 19 open queries from mathematicians who had been briefed on the query format. These 56 queries were distributed to the participants in a document and as special XML files; see [Koh12] for details.

3.3 Pooling and Assessment

Six participating groups returned results either in `trec_eval` form or in a special XML format specified in [Koh12] that allowed additional information (justifications like substitutions) to be submitted. As not all participating teams used the extended format – and the task submission policy did not mandate it – all submitted results were first converted into a `trec_eval` result file format[5] and fed into the math-specific extension SEPIA system [SEP] provided by NII for evaluation.

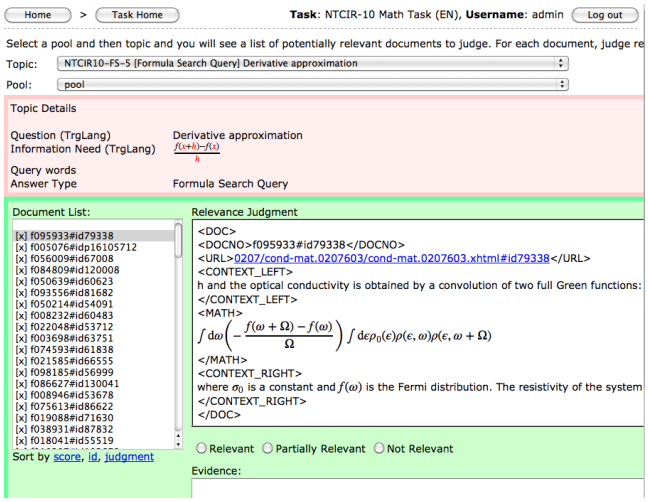

Figure 3: Evaluation Screen in SEPIA

The `trec_eval` format contains one six-tuple of identifiers for each hit:

⟪QueryID⟫ ⟪Q0⟫ ⟪FormulaID⟫ ⟪rank⟫ ⟪score⟫ ⟪Runtag⟫

which is displayed to evaluators as shown on the screen in Figure 3. The central datum is the formula identifier ⟪FormulaID⟫, which consists of the document URI and a formula identifier – i.e. the value of the id attribute on the <**math**> element in the MathML representation of the formula; fid1 in Figure 1. Evaluators[6] judged relevance of the hit to the query (displayed in the upper light red box; query variables are displayed in red) by comparing it to the formula that constituted the hit (displayed in the center of Figure 3) and its document context (above and below the formula).

For each formula, the assessors were asked to select either `relevant` (R), `partially-relevant` (PR), or `not-relevant` (N). Each formula was assessed by one or two assessors. Relevance was judged according to Table 4 and assigned a relevance label according to the perceived relevance of the hit.

Note that relevance is assessed not on document basis but on formula basis in the Math Retrieval Subtask. In particular, it became painfully clear during the evaluation process that results without formula identifier – i.e. results that only consists of the document URI – were almost impossible to judge as the documents contain around 100 formulae each. Therefore the organizers decided to disregard all such results, even though this decision invalidated all results in the Open Information Retrieval category (queries OMIR-1 to OMIR-19; see Table 2). Furthermore, the evaluation process revealed that query FS-17 was ill-formed and had to be taken out of evaluation (see Table 3).

Query type	Distributed	Evaluated
Formula Search	22	21
Full Text Search	15	15
Open Search	19	0

Table 3: Total number of topics.

Table 4: Relevance score assignment.

Score	Assessed by one judge	Assessed by two judges	Overall Judgment
4	R	R/R	Relevant
3	–	R/PR	Relevant
2	PR	PR/PR, R/N	Partially Relevant
1	–	PR/N	Partially Relevant
0	N	N/N	Not relevant

Table 5 and Table 6 summarize the distribution of relevance level for each topics. Also, the total number of formulae judged by the assessors after pooling, and the total number of distinctive hits per a query are also included in these tables. The last column, uniq ratio, is a fraction of formulae that were supported by only a single run. Since the ratio was relatively high, we have not conducted "uniques" contribution test for our task.

Although the task allows up to 100 hits per query and run, the sizes of the ranking lists varied much between the runs. Based on this, we selected formulae for the assessment

[5]This process required manual intervention by the organizers, since not all submissions fully complied with the submission format.

[6]In our case mathematicians from Zentralblatt Math and mathematics students from Jacobs University. It is crucial to understand that (in contrast to other information retrieval tasks), mathematical information retrieval can only be judged by evaluators trained in mathematics. However, at least for formula queries, special familiarity with the domain of the documents (the math discussed in them) was not considered crucial – otherwise an evaluation process would have been much much more difficult to organize.

as evenly as possible from all the runs based on the ranking orders in the individual submitted files: The current top-ranked formulae were taken from all the ranking lists, and added to the pool if they were not found. This process was repeated until the total size of the pool becomes equal or greater than 100.

Table 5: Relevance judgment statistics (Formula Search).

Query	Relevance score					Total	Total	Uniq
ID	4	3	2	1	0	judged	hit	ratio
FS-1	0	1	1	30	69	101	155	0.30
FS-2	0	0	1	1	102	104	453	0.25
FS-3	10	3	12	10	66	101	284	0.33
FS-4	8	6	17	19	52	102	278	0.56
FS-5	38	0	25	0	38	101	274	0.34
FS-6	0	0	25	0	77	102	261	0.53
FS-7	10	0	27	0	68	105	382	0.46
FS-8	45	0	6	0	50	101	993	0.77
FS-9	0	0	40	0	63	103	361	0.58
FS-10	0	0	13	0	87	100	281	0.49
FS-11	0	0	42	0	58	100	161	0.29
FS-12	0	0	26	0	74	100	135	0.26
FS-13	2	0	0	0	98	100	245	0.49
FS-14	1	0	34	0	65	100	231	0.40
FS-15	3	0	0	0	98	101	304	0.23
FS-16	19	0	2	0	81	102	357	0.38
FS-18	44	0	32	0	28	104	610	0.58
FS-19	0	0	24	0	76	100	195	0.29
FS-20	32	0	27	0	41	100	100	0.00
FS-21	27	0	12	0	61	100	178	0.31
FS-22	0	0	72	0	29	101	128	0.22
Total	239	10	438	60	1,381	2,128	6,496	0.45

Table 6: Relevance judgment statistics (Full-Text Search).

Query	Relevance score					Total	Total	Uniq
ID	4	3	2	1	0	judged	hit	ratio
FT-1	50	0	4	0	46	100	129	0.22
FT-2	2	0	26	0	72	100	218	0.94
FT-3	16	0	1	0	8	25	25	0.00
FT-4	0	0	40	0	60	100	101	0.01
FT-5	0	0	56	0	44	100	130	0.23
FT-6	0	0	39	0	61	100	130	0.23
FT-7	5	0	30	0	65	100	129	0.22
FT-8	21	1	28	3	47	100	100	0.00
FT-9	21	0	37	0	42	100	152	0.49
FT-10	10	0	1	0	89	100	130	0.23
FT-11	0	0	7	0	93	100	100	0.00
FT-12	0	0	20	0	80	100	100	0.00
FT-13	0	0	11	0	89	100	100	0.00
FT-14	7	0	10	0	83	100	130	0.23
FT-15	33	0	26	0	41	100	100	0.00
Total	165	1	336	3	920	1,425	1,774	0.26

To check that our assessment guidelines were sufficiently clear to the assessors, multiple assessments were collected for three topics, namely FS-1, FS-3, and FS-4. Indeed, for these topics, at least 2 assessors have been asked to assess the documents in the pool. While the inter-assessor agreement on these three topics, as measured by Cohen's Kappa, was in general moderate, the number of used topics for multiple assessments is such that we cannot confidently conclude on the difficulty of the notion of relevance in Math information retrieval. We leave a more thorough analysis of the inter-assessor agreement to a future work.

3.4 Outline of the Systems

In the following, we briefly describe the salient features of the approaches deployed by the participating groups in NTCIR-10. These descriptions were contributed by the participating groups. Further details about the deployed approaches could be found in the cited papers below.

3.4.1 BRKLY (UC Berkeley; see [LRG13])

The UC Berkeley team combined a standard keyword content information retrieval method with bitmap indexing of math operators identified as separate XML tags in the MathML structure. This approach resulted in a poor ranking of candidate documents from which to extract possible formulae. A single run was made using a hand-crafted query matched against document titles. Math formulae were extracted as a post processing step on the top 100 ranked documents. The post processing step was insufficiently parameterized and thus potentially relevant formulae were simply not found. The BRKLY group's approach demonstrates that math search is both qualitatively and quantitatively different from standard content-based information access.

3.4.2 FSE (TU Berlin; see [SLM13])

The FSE team presented an alternative approach to math search, that is primary intended as a research tool. Instead of relying on indexes that are costly to build, maintain and adapt, the FSE team proposed to employ a distributed data processing system that accesses data in a non-index format. While this system is not suitable for answering single ad-hoc queries from end-users, it is very effective for answering batches of queries that a system developer may wish to evaluate. Different approaches to query processing can thus be assessed simply by changing to source code and re-running the program. Consequently, the FSE approach allows for short prototype/test cycles.

3.4.3 KWARC (Jacobs University; see [KP13])

MathWebSearch is a web service that provides low-latency answers to unification queries over content MathML expressions. The standardized format makes MathWebSearch applicable to a wide range of querying tasks – all, where formulae can be transformed into content MathML. The low-latency makes MathWebSearch well-suited as a back-end for interactive applications, e.g. web-base formula search engines or editing support services. Unification queries form the basis of an expressive, query language with well-defined semantics. As substitution instances of the original query, MathWebSearch results are highly significant, if the encoding of data set and search query are adequate – i.e. do not forget or spuriously introduce salient semantic features.

3.4.4 MCAT, (NII; see [TKNA13])

To compensate for the ambiguity of MathML representation, the MCAT group primarily set flexibility as their design goal, with emphasis on recall. To this end, they implemented an indexing scheme for mathematical expressions within an Apache Solr (Lucene) database. According to this scheme, they encoded mathematical expressions as a series of factors

reflecting both the Presentation MathML tree structure as well as specific symbols they use. Search is performed as matching between the factors of the query and the factors in the database, and the results are ranked according to the number of matched factors, modified by Lucene's length normalization and TF/IDF scoring algorithm. Such an indexing scheme, even though rather simple, satisfies the flexibility requirements: even if the structure is slightly different, even if variables are renamed, the MCAT system can still find the result, albeit with a lower score. However, since even a single matched symbol is sufficient for inclusion into the search results, it is hard to specify a cut-off, which results in a long search tail, and consequently low precision score.

3.4.5 MIRMU (Masaryk University; see [LSM13])

The Masaryk University MIRMU team used a similarity search based on enhanced full text search utilizing attested effective techniques and implementations. The variability of used Math Indexer and Searcher (MIaS) system in terms of the math query notation was tested by submitting multiple runs with four query notations provided. The analysis of the evaluation results showed that the system performs best using TEX queries that are translated to combined Presentation-Content MathML.

3.4.6 NAK (Keio University; see [HS13])

The NAK team proposed two new indices, which hold structure information of math expressions in order to build a partial match retrieval system for math formulae. The first one is an inverted index constructed from paths to the root node from each node, which see a formula as an expression tree. The other index is a table that stores the parent node and the text string for each node in the expression trees. In the NTCIR-10 math task, the number of nodes was about 291 million and the number of path types in the inverted index was about 9 million. The experimental results showed that the search time grows linearly to the number of retrieved documents. Concretely, the search time ranges from 10 milliseconds to 1.2 seconds; the simpler formulae tending to need more search time.

3.5 Evaluation Results

As an initial attempt to evaluate mathematical information retrieval performance, we report the result based on the following four basic measures.

- MAP : Mean average precision.
- P-5 : Precision at rank 5.
- P-10 : Precision at rank 10.
- P-hit : Precision for all the returned search results.

`MAP`, `P-5`, and `P-10` were calculated using `trec_eval`, a standard IR evaluation tool. `P-10` was selected because all the top-10 ranked results were evaluated by human assessors based on our pooling policy. Also, about 85% of the returned results has at least 10 hits.

While `MAP`, `P-5`, and `P-10` are the average precision of all the queries in the task, `P-hit` is the average precision of only the answered queries in the task. `P-hit` was specifically introduced in our task in order to deal with the difference of policies of participating search systems. Since some systems return only the results with high confidence instead of a fixed size of ranking list, a traditional `trec_eval` does not seem to be appropriate to investigate the performance of such systems. Developing appropriate measures for mathematical search is one of the central issues in this pilot task and needs to be further discussed in the future.

Tables 7 and 8 summarize the performance results averaged over all the queries for Formula Search and Full-Text Search. The last row, `P-hit count`, shows the numbers of the relevant and the submitted hits for all the queries.

Table 8: Summary of the retrieval performance (Full-Text Search).

Relevant			
	FSE	MCAT.org	MCAT.mod
MAP avg	0.020	0.249	0.297
P-5 avg	0.053	0.307	0.320
P-10 avg	0.060	0.273	0.293
P-hit avg (P-hit count)	0.078 (19/244)	0.102 (146/1425)	0.103 (147/1425)

Partially relevant			
	FSE	MCAT.org	MCAT.mod
MAP avg	0.042	0.511	0.534
P-5 avg	0.147	0.613	0.680
P-10 avg	0.107	0.620	0.660
P-hit avg (P-hit count)	0.221 (54/244)	0.307 (438/1425)	0.309 (440/1425)

3.6 Discussion

The evaluation results reveal two particular aspects of the Math Retrieval Subtask: Firstly, the very large majority of submitted hits were judged irrelevant to the respective query. In particular, without explicit measures to deal with mathematical formulae, it is almost impossible to score relevant hits in formula queries. We suspect that this is also why 10 participating teams did not submit results.

Secondly, the systems that did get non-trivial results, can be divided in two groups: *i)* "exact-search" systems that only report hits if they find an exact match (they do not usually report a meaningful ⟪score⟫, usually 1, and *ii)* "similarity-search" systems that try to find partial matches and self-rate confidence values and invest into clever ranking strategies. The first category performs better when disregarding unanswered queries, in particular using the `P-hit` measure. The second category performs better when all queries were taken into account, in particular using the `MAP` measure. In particular, Tables 7 and 8 which tabulate the results for these two measures are the closest we can come to a ranking of search systems.

This overall outcome meets our prior expectations – even in the limited field of six participating groups, we have at least four kinds of systems with differing performance profiles:

i) *math-agnostic IR systems* (BRKLY), which had a great trouble dealing with formulae

ii) *similarity-search MIR systems* (MIRMU, MCAT) that return large sets of "similar" formulae scored by "closeness". Query variables are treated by making them "similar" to any subtree.

iii) *matching/unification-based MIR systems* (KWARC, NAK) that specifically return exact instances of the query, only that query variables may be replaced by arbitrary formulae.

iv) *batch MIR processors* (FSE) that do not use a search index and can therefore flexibly "program" queries.

The NTCIR-10 Math task evaluated these systems on a set

Table 7: Summary of the retrieval performance (Formula Search).

Relevant						
	BRKLY.R1	BRKLY.R2	BRKLY.R3	BRKLY.R4	FSE	KWARC
MAP avg	0.024	0.000	0.000	0.024	0.088	0.086
P-5 avg	0.010	0.000	0.000	0.019	0.210	0.162
P-10 avg	0.005	0.000	0.000	0.010	0.148	0.152
P-hit avg (P-hit count)	0.004 (3/815)	0.000 (0/898)	0.001 (1/911)	0.062 (2/32)	0.102 (38/373)	0.187 (78/417)

Relevant (continued)						
	MCAT.org	MIRMU.run1	MIRMU.run2	MIRMU.run3	MIRMU.run4	NAK
MAP avg	0.162	0.060	0.112	0.112	0.127	0.083
P-5 avg	0.219	0.133	0.229	0.229	0.276	0.162
P-10 avg	0.229	0.105	0.191	0.191	0.219	0.148
P-hit avg (P-hit count)	0.065 (137/2099)	0.109 (64/589)	0.185 (92/496)	0.185 (92/496)	0.123 (96/778)	0.091 (103/1127)

Partially relevant						
	BRKLY.R1	BRKLY.R2	BRKLY.R3	BRKLY.R4	FSE	KWARC
MAP avg	0.029	0.001	0.002	0.024	0.130	0.144
P-5 avg	0.076	0.010	0.010	0.019	0.343	0.314
P-10 avg	0.048	0.005	0.010	0.010	0.295	0.286
P-hit avg (P-hit count)	0.016 (13/815)	0.003 (3/898)	0.004 (4/911)	0.062 (2/32)	0.284 (106/373)	0.333 (139/417)

Partially relevant (continued)						
	MCAT.org	MIRMU.run1	MIRMU.run2	MIRMU.run3	MIRMU.run4	NAK
MAP avg	0.379	0.066	0.081	0.081	0.100	0.104
P-5 avg	0.476	0.181	0.267	0.267	0.343	0.257
P-10 avg	0.500	0.143	0.214	0.214	0.267	0.257
P-hit avg (P-hit count)	0.220 (462/2099)	0.148 (87/589)	0.232 (115/496)	0.232 (115/496)	0.161 (125/778)	0.138 (156/1127)

of standard information retrieval tasks and pinpointed their performance profiles.

4. MATH UNDERSTANDING SUBTASK

In the following, we describe the Math Understanding Subtask, the second main subtask of the NTCIR-10 Math Pilot Task.

4.1 Task Design

The goal of the Math Understanding Subtask is to extract natural language descriptions of mathematical formulae in a document for their semantic interpretation.

The dataset contains 10 manually annotated papers used in a dry run, and an additional 35 papers used in the formal run. All the mathematical formulae in the dataset are expressed using MathML Parallel Markup that contains both Presentation and Content MathML Markups. The Presentation Markup was extracted from the XML+MathML files provided by arXMLiv Project [arXb] for the Math Retrieval Subtask. In addition, the Content Markup was newly generated for each formula by human experts. Therefore, the dataset can be also used as a reference dataset for the transformation from a presentation level (LaTeX or MathML Presentation Markup) to semantic level (MathML Content Markup) representations.

4.2 Data Annotation

A description is obtained from a continuous text region or concatenation of some discontinuous text regions. In addition, shorter descriptions may be obtained from longer ones. For instance, in the text "$log(x)$ is a function that computes the natural logarithm of the value x", the complete description of "$log(x)$" is "a function that computes the natural logarithm of the value x". Moreover, the shorter descriptions "a function" and "a function that computes the natural logarithm" can be obtained from the previous one. This subtask defines two types of possible descriptions of mathematical expressions, namely the full description (contains the complete type) and the short description (contains the short type). Participants may extract any type of descriptions in their submission.

The training and test set consist of 35 (including 10 from the dry run) and 10 annotated papers selected from ArXiv.org dataset, respectively. The inter-annotator agreement is conducted on the five papers taken from the dataset. There are three measurements to test the reliability of the annotation: F1-score, Cohen's kappa, and Krippendorff's alpha. To compute the F1-score, the position of the annotated descriptions from two annotators is matched. There are two matching scenarios, namely strict matching and soft matching, which are described in the Evaluation section below. The computation of Krippendorff's alpha used interval-based difference function for binary data. The results of the agreement are depicted in Table 9.

Table 9: Inter-annotator agreement

	F1-Score	kappa	alpha
Full Descriptions	strict: 0.8670 soft: 0.9701	0.8993	0.7630
Full and Short Descriptions	strict: 0.9014 soft: 0.9683	N/A	N/A

4.3 Evaluation Measure

The evaluation is done by matching the position of the extracted descriptions against the positions of gold-standard descriptions. Two matching scenarios, namely strict matching and soft matching, which were used in the evaluation are described as follows.

- The extracted description will pass the strict matching evaluation if its position, consisting of start index and length, is the same as the position of the gold-standard description.
- The extracted description will pass the soft matching evaluation if its position contains, is contained in, or overlaps with the position of the gold-standard description.

The evaluation metrics, precision, recall, and F1-score, are defined as follows.

$$Precision = \frac{\#\text{correct\_detections}}{\#\text{detections}}$$

$$Recall = \frac{\#\text{detected\_descriptions\_in\_test\_data}}{\#\text{all\_descriptions\_in\_test\_data}}$$

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.4 Outline of the System

The MCAT team was the only team who attempted the Math Understanding Subtask. The description of their deployed approach is provided below.

4.4.1 MCAT (NII; see [TKNA13])

In the initial step of the MCAT approach, all noun phrases are extracted and considered as description candidate. Each of these noun phrases is then paired with a mathematical expression that appears in the same sentence. Subsequently, an SVM-based model is trained using features that are extracted from each of these pairs. There are two runs for each full description extraction and short description extraction, i.e., one run using a model that includes the feature of apposition, and one run using a model that does not, making up four runs in total. The result showed that the feature of apposition is good for predicting short descriptions, but not for full descriptions. Furthermore, the MCAT team concluded that the extraction of more advanced features is required to improve the current performance.

4.5 Evaluation Results

Only one team participated in this subtask submitting four runs. The evaluation of the submission is shown in Table 10 and 11. The MCAT team's four submissions consist of two that contain full descriptions, and two with short descriptions.

Table 10: Strict matching evaluation

Run ID	Precision	Recall	F-1
MCAT_full_1	61.94	37.03	46.35
MCAT_full_2	61.92	37.33	46.58
MCAT_short_1	68.24	40.42	50.77
MCAT_short_2	67.67	40.22	50.45

Table 11: Soft matching evaluation

Run ID	Precision	Recall	F-1
MCAT_full_1	86.48	47.41	61.24
MCAT_full_2	87.25	48.30	62.18
MCAT_short_1	81.68	42.81	56.18
MCAT_short_2	81.24	42.61	55.90

4.6 Discussion

The evaluation results showed that the precision of the description extraction is sufficiently high with the soft matching criteria. On the other hand, the recall still needs further improvement, particularly with the strict matching criteria. Regardless of its simplicity, the description extraction technique can be widely applied to mathematical information access applications. For example, the keywords contained in the extracted descriptions can be used for math formula search, and the extracted descriptions can be also used to assist in the users' understanding of mathematical content. Evaluating the effect of the description extraction in these applications is one of the major future challenges.

5. CONCLUSION

This was the first time a task dedicated to Math information retrieval (IR) was run as part of an international IR evaluation forum. A new test collection of 100.000 documents from mathematics, physics and computer science was created, and two main subtasks have been addressed, namely the Math Retrieval and Math Understanding subtasks. The participants results suggest that Math information retrieval is very challenging, requiring further enhanced approaches and evaluation methodologies. However, the results of the NTCIR-10 Math Pilot Task also show that a great deal of work has been done by the participating groups to devise reasonable baselines for the Math Retrieval task. Indeed, the Pilot Task has been very successful in facilitating the formation of a pluri-disciplinary community of researchers interested in the challenging problems underlying Math IR. It is hoped that by learning from this Math Pilot Task, both the organizers and the participating groups could further define the tasks, the topic development and assessment procedures, and the required baselines. It is our intention as organizers to continue the Math Pilot Task, focusing in particular on the Math Retrieval subtask so as to create a more robust research infrastructure for the development and evaluation of Math IR approaches.

Ultimately, the success of MIR systems will be determined by how well they are able to accomodate user needs in terms of the adequacy of the query language, the trade-off between query language expressivity/flexibility and answer latency on the one hand and learnability on the other hand. Similarly, the result ranking and monetization strategies for MIR are still a largely uncharted territory; we hope that this and future NTCIR Math tasks can help to make progress on this front.

Acknowledgments

The work reported here here has been partially supported by the Leibniz association under grant SAW-2012-FIZ_KA-2, Japan Society for the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research (B) under grant 24300062, and Research Center for Knowledge Media and Content Science,

NII.

We are indebted Deyan Ginev for generating the dataset for the Math Retrieval Subtask and to Cevahir Demirkiran, Helena Mihaljevicz-Brand and Wolfram Sperber from Zentralblatt Math for providing formula search queries and helping with the evaluation task, Giovanni Yoko Kristianto for generating the dataset for Math Understanding Subtask, and Goran Topić for administrating a backend system for the entire task. We gratefully acknowledge the contribution of Dr. Sato from Triax in extending the SEPIA system to deal with the specifics of the Math Task. Finally, the outlines of Systems in Sections 3.4 and 4.4 were supplied by the authors of the respective papers.

6. REFERENCES

[ArXa] `arxiv.org` e-Print archive. web page at `http://www.arxiv.org`. seen November 2012.

[arXb] `arXMLiv`: Translating the Math Archives to XML+MathML. web page at `http://trac.kwarc.info/arXMLiv/`. seen April 2010.

[HS13] Hiroya Hagino and Hiroaki Saito. Partial-match retrieval with structure-reflected indices at the ntcir-10 math task. In NTCIR-10 [NTC13]. this volume.

[Koh12] Michael Kohlhase. Topics for the ntcir-10 math task; math retrieval subtask. Technical report, NTCIR, 2012.

[KP13] Michael Kohlhase and Corneliu-Claudiu Prodescu. Mathwebsearch at ntcir-10. In NTCIR-10 [NTC13]. this volume.

[LRG13] Ray R. Larson, Chloe Reynolds, and Fredric Gey. The abject failure of keyword ir for mathematics search: Berkeley at ntcir-10 math. In NTCIR-10 [NTC13]. this volume.

[LSM13] Martin Líška, Petr Sojka, and MichalRůžička. Mirmu at the ntcir-10 math task: Similarity search for mathematics. In NTCIR-10 [NTC13]. this volume.

[NTC13] *NTCIR Workshop 10 Meeting*, Tokyo, Japan, 2013. this volume.

[NTM13] NTICR Pilot Task: Math Task. `http://ntcir-math.nii.ac.jp/`, 2013.

[SEP] sepia: Standard evaluation package for information access systems. `https://code.google.com/p/sepia/`.

[SKG+10] Heinrich Stamerjohanns, Michael Kohlhase, Deyan Ginev, Catalin David, and Bruce Miller. Transforming large collections of scientific publications to XML. *Mathematics in Computer Science*, 3(3):299–307, 2010.

[SLM13] Moritz Schubotz, Marcus Leich, and Volker Markl. Querying large collections of mathematical publications. In NTCIR-10 [NTC13]. this volume.

[TKNA13] Goran Topic, Giovanni Yoko Kristianto, Minh-Quoc Nghiem, and Akiko Aizawa. The mcat math retrieval system for ntcir-10 math track. In NTCIR-10 [NTC13]. this volume.