QUALIBETA at the NTCIR-11 Math 2 Task: An Attempt to Query Math Collections

José María González Pinto, Simon Barthel, and Wolf-Tilo Balke
IFIS TU Braunschweig
Mühlenpfordstrasse 23 38106 Braunschweig, Germany
{pinto, barthel, balke}@ifis.cs.tu-bs.de

Abstract

This project introduces our first attempt to mathematical retrieval of formulae from a large collection for the NTCIR-11 Math 2 task. Our approach combined a feature-extracted sequence mechanism of the formulae and a sentence level representation of the text describing the formulae to model the collection. The feature-extracted sequences used were: the category of the formulae, the sets of identifiers, constants, and operators. This representation with the text surrounding the formulae were indexed in Elastic Search for query processing. Even though our information extraction model results are below the average's participants and our expectations, the experience will help us to improve our work in several directions.

Team Name

IFISB_QUALIBETA

General Terms Information Retrieval

Keywords Mathematical Formula Retrieval, MathML Indexing, Information Systems.

1. Introduction

Today we can find numerous documents with highly specialized knowledge encoded in different objects such as formulae. Current state of the art search engines are deficient for querying such document collections due to the elusive meaning inside the formulae. After all, what does a formula mean? How to represent it to give relevant results to a given information need? Moreover, is the formula by itself sufficient for a system to be able to query large collections and return relevant results to the user?. Indeed, querying for formulae represents an interesting challenge and has become in recent years an interesting domain for the information retrieval community. A few efforts from industry and academia trying to solve this problem exist. One commercial product example is Symbolab[1] (Scientific Equation Search). Symbolab provides both free text search and equation search. Symbolab contains resources from Wikipedia[2], Khan Academy[3] videos, dictionaries, online books and some other resources. However, its domain is more general and it cannot answer highly specialized queries such as those presented for the NTCIR-11 task. Another example that deals with LaTeX search is the Springer LaTeX [4] search engine which uses similarity algorithms to compare LaTeX strings; a query can obtain results even when users write the equations with some variations. From the research community we can find different approaches. We will only mention a couple of them relevant to our context. One the most interesting references is Math Indexer and Searcher[4] which is part of the search engine capabilities in the European Digital Mathematics Library (EuDML)[5]. Another approach from the research community is MatWebSearch[5] developed by the KWARC group at Jacobs University and which is being used by one of the main resources for mathematics: Zentralblatt [6]. As their authors mention their approach is a content-based search engine that indexes MathML using term indexing, a tecnique derived from automated theorem proving. The system processes documents containing math formulae encoded in MathML format. All of these previous attempts have reported plausible results although a final solution remains to be found. Indeed, one of the issues that was missing in this challenge was a common dataset and a setting in which researchers could form a community to compare their results, share findings and iterate to improve for the benefit of the scientific community. Last year, the NTCIR-10 Math-1 task was organized to evaluate specialized retrieval systems for math formulae. Some attempts can be found at the proceedings summary from last year in [6], including some of the projects we have briefly mentioned.

In the second installment, the NTCIR-11 Math-2 task for retrieving mathematical formulae in scientific documents, the interest focus in a formula based search with keywords using a similar dataset. The Math-2 task presented an opportunity for research and further information about the task and the collection can be found at [10]. The remainder of this paper is organized as follows: Section 2 describes our methodology. Section 3 presents our results in the task. And Section 4 concludes the paper and points to future work.

2. Qualibeta Approach

We modeled the collection as two sets: the formulae and the context of the formulae. For each set we extracted features and used them as index for our query processing mechanism. Our goal was to capture both, the semantics of the formula and its syntactic structure and combine them to find relevant documents. We first describe the representation of the formulae and then their context.

2.1 Formula Representation

The collection consisted of many sections per document. Each section is a paragraph and contains both text and formulae in MathML. We favored Content MathML as our main source for parsing the formulae and therefore ignored the layout semantics provided by LaTeX and Presentation MathML. This decision turned out to be

[1] http://www.symbolab.com

[2] http://www.wikipedia.org

[3] https://www.khanacademy.org/

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

[5] https://eudml.org/

[6] https://zbmath.org/formulae/

incorrect, as we will discuss in the results section. After some empirical tests we decided to model each formula as a set of string sequences of features which combined can give certain degree of discrimination for query processing. We considered the following: category of the formula, set of unique identifiers, constants and operators. Some empirical experimentation with the combination of features gave us the insight to structure the queries. The idea behind this simple representation was to have a model that combined with the semantics associated with the formula could achieve high precision. After extracting each of the formula features from the collection, we used the Elastic Search engine [1] to build our index. As a simple example of our approach, consider the formula $x + y$ found in a document section. Our model will extract the following set of features:

- Category: arithmetic
- Sets of identifiers: xy
- Sets of constants: [] (empty set)
- Operators: +
- Sets of unique identifiers: $id0, id1$

We used a combination of the boolean query operators and text capabilities of Elastic Search to query the generated syntactic structure of the formula. The idea was to consider for nearby matches and at the same time generalize the syntax of the formula. For example, querying for the formula outlined above could be interpreted as "any aggregate of two variables". And that type of query could be achieved using a combination of the operators "must", "should", "match" and "match_phrase" available from the query engine and applied to our representation. With this representation we were able to express queries aiming at similar syntax. To continue with our example, a code fragment of a possible query for the above formula would be:

must: {match_phrase: {"uniqueIdentifiers":"id0 id1"}}

should: {match: {"identifiers": "x y"}}

should: {match: {"operators":"+"}}

The query will return all formulae with the following conditions:

1. With at least two identifiers (e.g. "ci" in MathML).
2. Preferring formulae with "x" and "y" sequence of identifiers and with the operator "+"
3. But still retrieving results such as $y + x$

We implemented the query processing in Java using the API provided by Elastic Search engine.

2.2 Context Representation

To represent the context of a formula for each document section of the collection, we extracted three context sentences: before, after and where the formula appears. From each of the sentences we extracted the noun-context and the verb-context. The set of nouns is the "noun-context" feature and the set of verbs is the "action-context" feature of the formula. The goal of the action-context is to capture the role of the formula to study how it is used. For instance, one can query formulae used to "determine" *subspaces* or formulae used to "define" *density energy*. Even though for the NTCIR task we did not use this capability of our system we envision a study of the role of frequently found formulae in large historical collections of scientific documents and this work is one step towards that purpose. The noun-context is used to represent the "label" of the formula in a document. For instance, one can query formulae mentioned as "density energy" and our system will retrieve all the formulae associated with that context. This representation was used for the NTCIR task. All the implementation of this module is in Java and for NLP processing tasks we used the Stanford CoreNLP[9].

2.3 Query processing

Once we have parsed the collection and extracted the features mentioned before, we generated the index which size was 12.3 GB. The machine used for the experiments was a Microsoft Windows 7 Professional Intel Core i5 CPU 3.4 GHz with 16 GB of RAM. Then, to process each query topic automatically we proceeded as follows: first we extracted the keywords and the Content MathML. Afterward, we generated the representation of the formula by getting the features needed for our system to query. Once we had these features we first query by keywords and from the set of documents retrieved, our algorithm matches the structure of the formula that is nearest to the topic.

3. Results

The results from the task in Figure 1 shows the MAP Avg. Relevance among all the participants for the relevant results. As shown in the figure our performance was below average. In particular, in 48% of the topics our results were considered irrelevant and from the others only in one Topic we were the best of all the competitors (Topic 29) with a 0.2569 MAP Avg.; far from what we will consider as good enough for a query need satisfaction. Only in Topic 29 and Topic 6 we were above the average.

In Figure 2 shows the Topics where some results existed from our approach. We plot against the best results and the average results. It is clear from the results that our approach needs further reconsideration to be a contender.

In the second case, the considered "Partially Relevant" results, our performance was again below the mean. In 12% of the topics our results were considered not partially relevant. Figure 3 shows the partially relevant MAP Avg.

Furthermore, in 12 topics we were above the average and in only one topic our performance was the best. In summary, in 6 out of the 50 topics our system could not get any result either relevant or partially relevant. We can see the results in Figure 4 and Figure 5 summarized in this paragraph.

4. Future work

The task gave us the opportunity to evaluate our model and to establish future directions. From our results, it is clear that we need to find a more efficient mechanism to combine both: the formula and its context. First, the context representation needs to go beyond sentence level and perhaps should be at the paragraph level. Indeed, our system was unable to find some of the results which were judged as relevant because of our short context. And second, the model behind the formula representation implores for a redesign. In particular, relying only in Content MathML was inadequate. These two elements were the reasons behind our performance. However, after the analysis of the results we can understand the relevant features of our model for the task.

4.1 Formula Semantics

First of all, we will study the intended meaning of the formulae from the user perspective. The main reason to investigate the user perspective comes from the fact that one of the things that our system handle incorrectly was the presence of variables in the formula (*"qvar"*). This not only suggests to design a better substitution mechanism to represent the MathML source but also to investigate to what extend a user will query with variables and what are the other query "capabilities" that should be considered. For example, would the user find useful to query formulae by the components

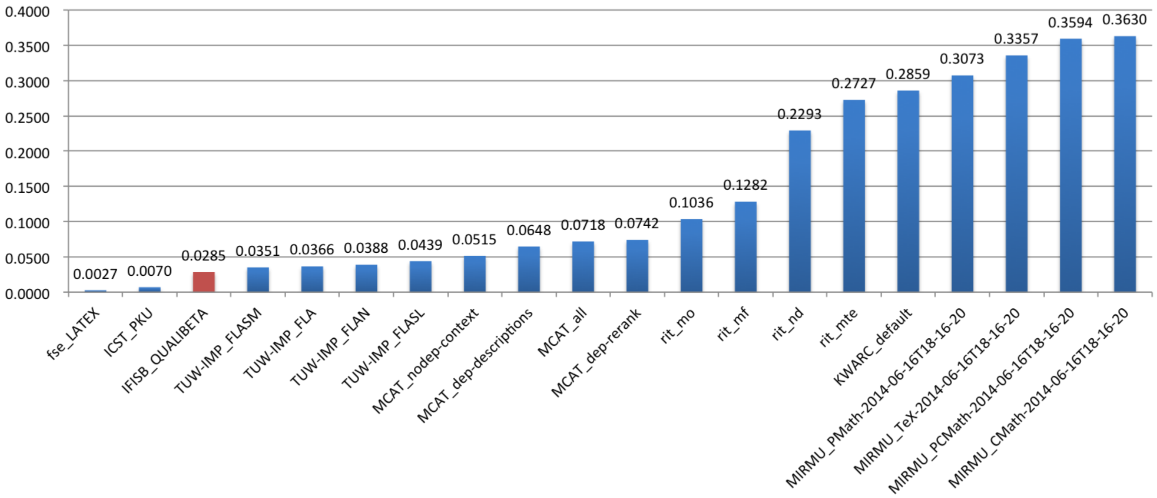

Figure 1: MAP Avg. Relevance >= 3 (Relevant)

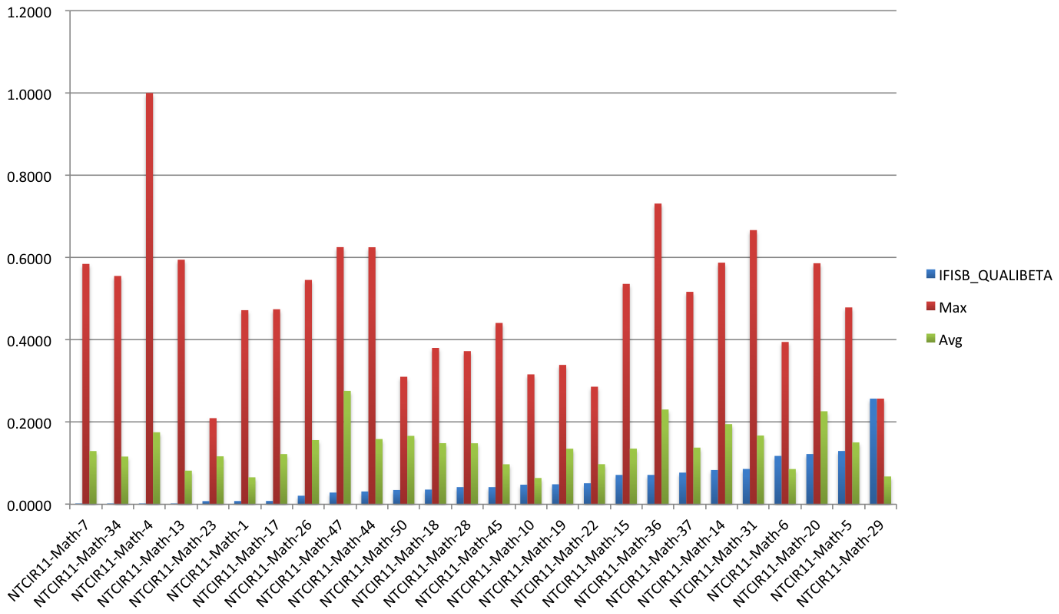

Figure 2: MAP Avg. Qualibeta vs Max and Avg.

of the formula: “a product of partial differential equations”. Understanding the query intent and how a user sees the meaning of a formula is the key to reason about how to model the formulae. As an example that motivates our thinking, consider Topic 2, in which our results were considered completely irrelevant; in our analysis the formula was indeed in our database but the encoded meaning for this particular topic was not captured by our parser. And moreover, our parser builds a representation from Content MathML and this topic yielded two identifiers instead of one -the syntax of the formula was incorrect. Our system therefore was unable to return relevant documents after combining the context and the formula features. One possible solution could be to represent the formula with implicit knowledge and increase the context of the parser beyond sentence level. Yet another noteworthy example is Topic 12 where our approach was unable to return relevant and partially relevant documents. After investigating the top results from the best of the participants in this topic, we found a document paragraph where the formula expression is present and also the four keywords. Certainly, a nice result considering that the formula itself is a “sh-ie algebra” and the keywords did not mention it explicitly but somehow the context was very well captured combined with the structure of the formulae. Thus, one idea to consider is to model “formulae aspects”, borrowing some ideas from previous work in [2] and extending it to model the aspects of the formula that permits a better exploration of the search space. Another issue we will consider is the relationship between the formula and the document itself. We are currently working to integrate topic models to our solution and use a Latent Dirichlet Allocation approach to attach a set of topics

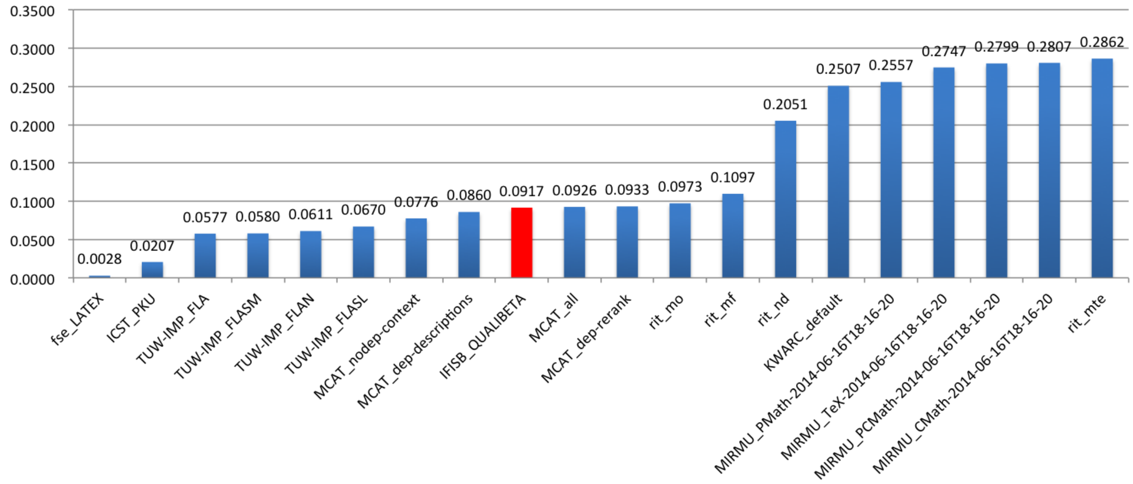

Figure 3: MAP Avg. Relevance >= 1 (Partially Relevant)

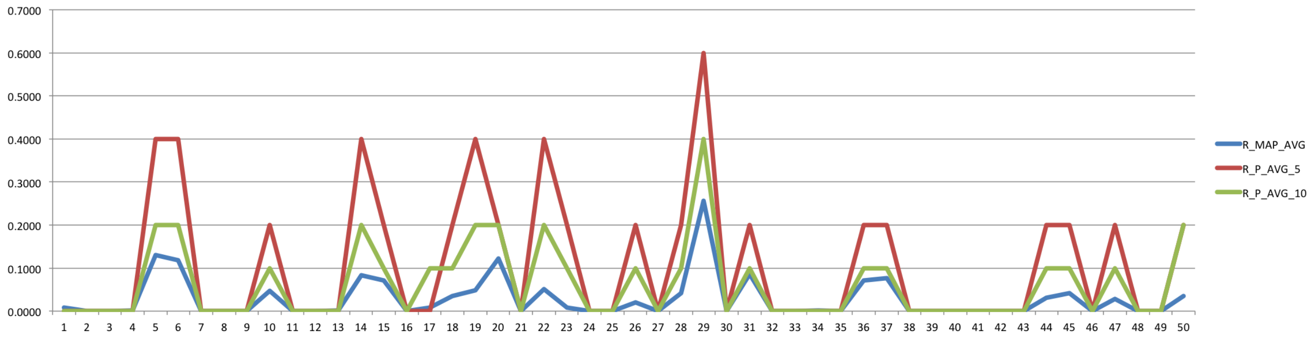

Figure 4: Relevance Level >= 3 (Relevant)

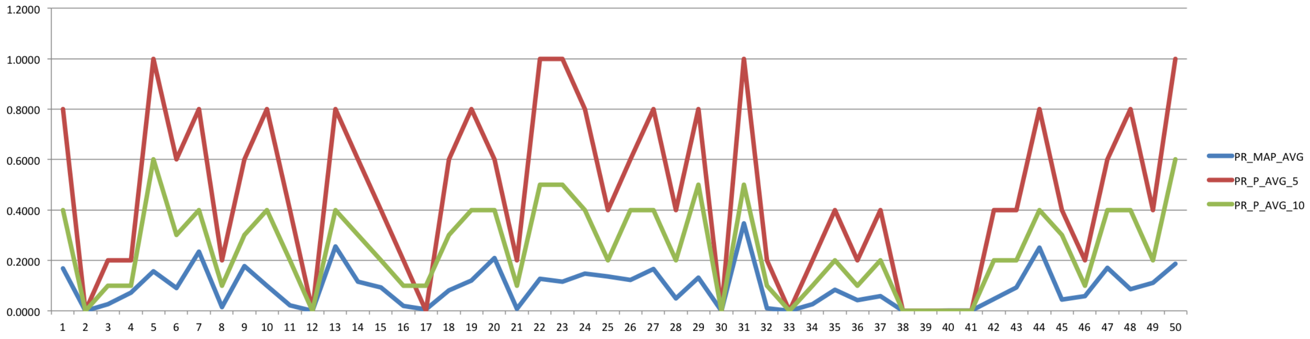

Figure 5: Relevance Level >= 1 (Partially Relevant)

to each formula[7]. The goal with this approach is to give structure to the context of the formula by identifying key topics of its use. Semantic compositionality is another work we are currently exploring to model the semantics of the key sentences where formulae appear[8]. We are confident that a combination of the two approaches may lead us to capture a more meaningful formula context.

4.2 Formula Structure

Another plausible improvement would be to model formulae as high-level objects with certain semantic properties and relations which can be represented as a tree and then apply subtree similarity search extending work such as [3]. Furthermore, we would like to investigate if there is a significant difference when we consider the type of users we can think of as our targets: phd students, professors, and master students. There might be a possibility of modeling the formulae sensitive to the user context and we would like to perform a user study that can give us an idea of how this context looks. One can imagine some usage patterns of the formulae based on the type of user and therefore it could be an interesting path to follow. Another important issue is the syntactic structure of the formulae. For this task we were provided with documents with LaTeX and MathML (Presentation and Content) of each formula. And given the observation that some of the queries were bad written in one of these representations motivates us to develop a model that can make better transformations among the three representations to interpret correctly the Query Topics.

We thank the organizers for this great opportunity to participate and to learn from the task.

After analyzing all the results, we can see that further innovations in the field are needed. We look forward to continue working in this interesting problem and its derivatives.

References

[1] Elastic Search. http://www.elasticsearch.org/

[2] Wang, X., Chakrabarti, D. and Prunera K. Mining Broad Laten Query Aspects from Search Sessions. KDD 2009

[3] Sara Cohen and Neyra Or. A General Algorithm for Subtree Similarity-Search. 30th IEEE International Conference on Data Engineering. 2014

[4] P. Sojka and M. Lška. The Art of Mathematics Retrieval. Proceedings of the ACM Conference on Document Engineering. 2011.

[5] Michael Kohlhase, Bogdan A. Matican, and Corneliu C. Prodescu. MathWebSearch 0.5 -Scaling an open Formula Sarch Engine. Conferences on Intelligent Computer Mathematics (CICM). 2012

[6] Akiko Aizawa, Michael Kohlhase and Iadh Ounis NTCIR-10 Math Pilot Task Overview. Proceedings of the 10th NTCIR Conference 2013

[7] D. M. Blei, A. Y. NG, and M. I. Jordan. Latent dirichlet allocation. Journal of Machine Learning Research. 2003

[8] Tim Van de Cruys, Thierry Poibeau, Anna Korhonen: A Tensor-based Factorization Model of Semantic Compositionality. Proceedings of HLT-NAACL 2013.

[9] Manning, Christopher D., Surdeanu, Mihai, Bauer, John, Finkel, Jenny, Bethard, Steven J., and McClosky, David. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55-60

[10] Akiko Aizawa, Michael Kohlhase and Iadh Ounis NTCIR-11 Math Pilot Task Overview. Proceedings of the 11th NTCIR Conference 2014