AN EFFICIENT SIMILARITY BASED SEARCH ENGINE FOR MATHEMATICAL CONTENT IN LaTeX MARKUP

by

Wei Zhong

A thesis submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Master of Science in Electrical and Computer Engineering

Spring 2015

AN EFFICIENT SIMILARITY BASED SEARCH ENGINE FOR MATHEMATICAL CONTENT IN LaTeX MARKUP

by

Wei Zhong

Approved: _____

Fouad E. Kiamilev, Ph.D.
Professor in charge of thesis on behalf of the Advisory Committee

Approved: _____

Kenneth E. Barner, Ph.D.
Chair of the Department of Electrical and Computer Engineering

Approved: _____

Babatunde A. Ogunnaike, Ph.D.
Dean of the College of Engineering

Approved: _____

James G. Richards, Ph.D.
Vice Provost for Graduate and Professional Education

ACKNOWLEDGMENTS

Thank you to my family with their support from every perspective through out my graduate academic education. Thank you to my advisor Hui Fang who offers me the opportunity to develop my idea further and supports me in many other ways. I am also grateful to all InfoLab members for their kind help. And thanks to those not previously mentioned, who have influenced me or helped along the way.

TABLE OF CONTENTS

ABSTRACT

In this paper, we have addressed the problems of searching content in mathematical language, particularly measuring the similarity degree (in terms of structural and semantical) between mathematical expressions, summarized some general properties from mathematical semantics, that a search engine should be aware of. To better deal with these problems in an efficient way, we propose some ideas including: (1) A list of grammar rules to parse mathematical content (particularly in LaTeX markup) into a tree representation in order to preserve as much as information from mathematical expressions; (2) An index approach to break down the tree representation into what we call branch words to enable fast search in a similar fashion with inverted index, with parallelism potential; (3) A search method to capture some level of query-document subgraph isomorphism, combined with two pruning methods to both speed search and improve effectiveness. We also build our own proof-of-concept prototype search engine to demonstrate these ideas, and thus are able to present some evaluation results through this paper.

Chapter 1

BACKGROUND

Apart from general text content, structured information is also widely contained by digital document. Among these, a lot of mathematical content (including documents on Internet), are represented using markups like LaTeX , MathML [1] or OpenMath [2], which is in a rich structural way. Information Retrieval on those structured data in mathematics language is not that well-studied or exhaustively covered by mainstream IR research, compared to that with general text. Thus it can be challenging yet very helpful given the contribution and importance of mathematics to our science.

However, the structured sense of mathematical language, as well as many its semantic properties (see section 1.2), makes general text retrieval models deficient to provide good search results. Through this paper, we have made our efforts to tackle some of these problems. Some of the ideas used in this paper deals with "tree structured" data in a general way, have the potential to be applied by other fields of structured data retrieval besides that from mathematical language.

1.1 Math IR Domains

Mathematical information involves a wide spectrum of topics, we are, of cause, not focusing on every aspects in mathematical information retrieval. It is good to clarify our concentration in this paper here, by first listing a set of concentrations that a mathematical information retrieval topics may be classified into, and define our target field of study.

---

[1] http://www.w3.org/Math/

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

Listed here, are considered four possible concentrations of topic for mathematical information retrieval:

1. Boolean or Similarity Search
2. Math Detection and Recognition
3. Evaluation, Derivation and Calculation
4. Other topics

The first one is doing mathematical information retrieval by searching, and finding the most relevant context of documents that match the query, very similar to the most common ways that other general text search engines will do, by boolean or similarity search. The only difference is, the query may contain mathematical expressions. Instances (examples online are SearchOnMath [3], Uniquation [4] and Tangent [5]) of such search engine can be useful in many ways, for example, student may utilize it to know which identity can be applied to a formulae in order to give a proof of that formulae. This is the area where we focus in this paper. Specifically, we are proposing a series of methods for similarity search of math content. And our method is using query only in LaTeX markup (some math-aware search engines [6] support queries in mathematical formulae and normal text together), and return documents ordered by score which indicates the similarity degree.

Digital mathematical content document can also be in an image format (e.g. generated by a handwritten query), thus to retrieve these information involves detection or recognition. Inspired by the advances from deep learning, we may foresee a large potential to be explored on topics related to this.

---

[3] http://searchonmath.com

[4] http://uniquation.com/en

[5] http://saskatoon.cs.rit.edu/tangent/random

[6] WolframAlpha: https://www.wolframalpha.com/ and Zentralblatt math from MathWebSearch: http://search.mathweb.org/zbl/

Because the nature of mathematical language, a query (e.g. an algebra expression) can be evaluated and potentially derived into an alternate form, or calculated. The result value of evaluation or derived form may also be considered being relevant to that query. These potentially require a system to handle symbolic or value calculation, or even a good knowledge of derivation rules implied by different mathematical expression (e.g. computational engine *Symbolab* [7] and WolframAlpha).

Besides the first three concentrations, there are many other topics. Knowledge mining, for example, will need deeper level of understanding on math content. A typical goal of this topic is to give a solution or answer based on information retrieved from math content. e.g. "Find an article related to the *Four Color Theorem*" [1].

These concentrations somehow overlap in some cases, for example, some derivation can be used to better assess the similarity between math formulae, e.g. $\frac{a+b}{c}$ and $\frac{a}{c}+\frac{b}{c}$ should be considered as relevant. Or, mathematical knowledge being used be know the same meaning (thus high similarity) between $\binom{n}{1}$ and C_n^1. Therefore even boolean or similarity search possibly involves certain level of understanding of mathematics. In terms of similarity, however, we only address the measurement for structural and symbol differences in this paper, without considering further topics lured from measuring math content similarity, such as evaluation, derivation or knowledge inference.

As supplementary, [2] gives a comprehensive review on mathematical IR researches and covers many topics across different domains.

1.2 Issues in Measuring Similarity

Unlike general text content, mathematical language, by its nature, has many differences from other textual documents, there are a number of new problems in measuring mathematical expression similarity. Among these, we select and focus on those regarding to structural similarity and symbolic differences between expressions.

[7] Symbolab Web Search: http://www.symbolab.com

At the same time trying to respect the semantical information inferred from structure or symbols in mathematical expressions. But even without caring about the possible derivations and high level knowledge inference, there are still many new problems.

Firstly, differences of symbols, structure and possible semantic rules in mathematics should be captured, and not one by one, but in an cooperative manner to measure similarity. To illustrate this point, we know that only respecting symbolic information is of course not sufficient in mathematical language. e.g. $ax+(b+c)$ in most cases is not equivalent to $(a+b)x+c$ (although they have the same set of symbols). And the order of tokens in math expression can be commutative in some cases but not always. For example, commutative property in math makes $a+b=b+c$ for addition operation, but on the other hand $\frac{a}{b}$ is most likely not equivalent to $\frac{b}{a}$. These make many general text search methods (e.g. *bag of words* model, *tf-idf* weighting) inadequate. Moreover, symbols can be used interchangeably to represent the same meaning, e.g. $a^2+b^2=c^2$ and $x^2+y^2=z^2$. However, interchangeability comes with some constrains to maintain the same semantical meaning, that is, changes of symbols in expression preserve more syntactic similarity when changes are made by substitution. e.g. For query $x(1+x)$, expression $a(1+a)$ are considered more relevant than $a(1+b)$.

Secondly, how we evaluate structural similarity between expressions is a question. A complete query may structurally be a part of a document, or only some parts of a query match somewhere in a document expression. In cases when a set of matches occur within some measure of "distance", we may consider them to contribute similarity as a whole, but when matches occur "far away" for a query expression, then under the semantic implication of mathematics, they probably will not contribute the similarity degree in any way. We need metres to score these similarity under certain criteria and set up standard and rules for relevance assessments.

Lastly, trying to capture semantic information from expressions will help measure similarity but introduce ambiguity. Apart from the cases covered in [3], semantic incorrect written markups, which is somehow common in many online documents, e.g. writing "sin" in LaTeX markup instead of macro "\sin", will make it difficult to tell

whether it is a product of three symbols or a *sine* function, thus need to disambiguate. And depending on what level of semantical meaning we want to capture, ambiguity cases can be different. Consider $f(2x+1)$, if we want to know if f is a function rather than a variable, the only possibility is looking for implicit contexts, but we can nevertheless always think of it as a product without losing the possibility to search similar expression like $f(1+2y)$, the same way goes reciprocal a^{-1} and inverse function f^{-1}. Most often, even if no semantic ambiguity occurs, efforts are needed to capture some semantical meanings. e.g. In $\int f(x)\frac{\mathrm{d}x}{\sin x}$ and $\sin 2\pi$, it is not easy to figure out, without a little knowledge on integral or trigonometric function, that integral is applied to $\frac{f(x)}{\sin x}$ and the scope applied by sine function is 2π, if we want to capture the subordinative relationship information.

1.3 Related Work

Boolean or similarity search for mathematical content is not a new topic, conference in this topic is getting increasingly research attention and the proposed systems have progressed considerably [4]. And a variety of approaches have already emerged in a early timeline [5]. But there are a limited major ideas, from different angle, to deal with mathematical structured data. [6, 7, 8] use the same way to classify them into text-based and tree-based (structure-based). Here we follow the same classification and give a recap and an overview on their core ideas.

1.3.1 Text-based methods

Many researchers are utilizing existing models to deal with mathematical search, and use texted-based approaches to capture structural information on top of matured text search engine and tools (such as *Apache Lucene*).

DLMF project from NIST [9] uses "flattening process" to convert math to textualized terms, and normalize them into *sorted parse tree normal form* which creates a unique form for all possible orders of nodes (e.g. in a associative or commutative operator). Then further introduces serialization and scoping to stack terms [10], trying

to capture structure information by using text-IR based systems that supports phrase search. Similar idea is also used by [11], expressions are also augmented for various possible representations, but variables are also replaced and normalized, but they are using postfix notation, allows to search subexpressions without knowing the operator between them. MIaS system [12, 13, 14], like the methods above, are also trying to reorder commutative operations and normalize variable and constance into unified symbols, doing augmentation in a similar fashion.

Augmentation comes with a trade-off between storage demand for combination of both symbols (e.g. a and b) and unified items (*id, const*) in different levels, in order to capture both symbolic information and structure information. Thus implies complex expressions with many commutative operators will cost a lot of storage space, the benefits of capturing expression variances will be overshadowed.

Although named as structured-based approach, [15] is using *longest common subsequence* algorithm to capture structure information (in a unified *preprocessed string* and a *level string*). The method takes $O(n^2)$ complexity for comparing a pair of formulae, and no index method is proposed. Therefore is not feasible to efficiently apply to a large collection.

The Mathdex search engine [16], from another perspective, uses query likelihood approach [17] to estimate how likely the document will generate the query expression by a n-gram from root expression to sub-expression and tokens.

Math GO! [18] is anther system advances some transitional method to better search math content. It tries to find all the symbols and map formula pattern to pattern name keywords (like *matrix* or *root*), and proposes to replace term frequency by co-occurrence of a term with other terms.

1.3.2 Structure-based methods

What text-based methods share in common is they are converting math language symbols to bags of searchable words, the intrinsic defect when using a bag of words to replace structured information will make conversion process lose considerable

information or very inefficient. In order to cope with the disadvantages from text-based approach, structure-based methods generally generate intermediate tree-variance structure, and use these information to index or search.

Unification algorithm

Whelp [19] and MathWebSearch (MWS) directed by Kohlhase [20, 21, 22], derived from *automatic theorem proving* and unification theory, is in a boolean search category. The system of MWS uses *term indexing* [23] in a *substitution tree* index to to minimize access time and storage. Because the subexpression is not easy to search using substitution tree, WMS indexes all sub-terms, but the increased index size remains manageable [20]. However, their index relies on RAM memory, even scaling can accommodate nearly entire arXiv site (72% paper on arXiv), the RAM usage will be 170 GB [22], which already needs a considerable hardware resources.

Leaf-root path

[24] uses leaf to root XML path in a MathML object to represent math formula. When efficiency is considered, it only indexes the first and deepest path (to indicate how a formula is started and presumably the most characteristic part of a formula); when user wants to obtain the perfect-match result, it indexes all the MathML object leaf-root path. The boolean search is performed by giving all the paths match with those of the search query. [25] further develops an incorporation of previous method with breath-first search, to add sibling nodes information into index and have achieved better effectiveness.

Very similar idea is proposed by [26] and used in [27]. The authors of latter transform MathML to an "apply free" markup from which the leaf-root path are extracted. Leaf-root path is also used to evaluate similarity between MathML formula.

Symbol layout tree

A *symbol layout tree* [28] or *presentation tree* [6] describes geometric layouts of symbols in a formula. WikiMirs [6] uses two templates to parse LaTeX markup with two

typical operator terms: explicit ones ("\frac", "\sqrt", etc.) and implicit ones ("+", "÷", etc.) to form a presentation tree, then extracts original terms and generalized terms from normalized presentation tree, to provide the flexibility of both fuzzy and exact search. Term level, and df-idf idea of factors are used in scoring.

[29] [7] [30] [6]

Other structure-based methods

A novel indexing scheme and lookup algorithm is proposed by [31], its index has hash signature for each subtree because they have observed a lot of common subtree structure occur in math formula collection. This idea will result in a slower index growth. Their lookup algorithm supports wildcards, and performs a boolean match test. Although their lookup time is generally below 700ms, but the index size where query lookup time is tested is unclear, but presumably no greater than 70,000 expressions. By constructing a DOM tree, [32] extracts semantic keywords, structure description to indicate subordinative relationship in a string format. The similarity is calculated using normalized tf-idf vector (trained by clustering algorithms) by dot product. Although the final index is generated from text, promising results have been achieved. Tree edit distance is adopted by [33], it tries to overcome the bad time complexity of original algorithm by summarizing and using a structure-preserving compromised edit distance algorithm using heavy path. Although the result shows query processing time is long but it is reduced to average 0.8 seconds by applying with an early termination algorithm along with a distance cache [34].

1.3.3 Other related work

There are a number of articles trying use image to assess similarity. [35] compares their image-based approach using connected component-based feature vector with a proposed text-based method, reported precision@k values are low, but the potential for this method to be combined with shape representations or other features will potentially improve it and make it valuable for measuring similarity for image

mathematical expressions. [36] uses X-Y tree to cuts the page in vertical and horizontal directions alternatively, in order to retrieve math symbols from images, then use sub-image matching is performed, this method is intuitive, yet too expensive for regular document with markup language.

A lattice-based approach [37] build formal concept based on selected feature sets of each formula. The ranking is calculated by the distances from query in the lattice map when the query is inserted.

==========

However, another shortcoming is, they usually fail to achieve the desired property for preserving same similarity when changes are made by substitution (see section 1.2).

we doubt that users are likely to query, for example a+b c want- ing to find documents only with occurrences of variable c.

Our system Cowpie [8]

MathML vs LaTeX

distributed indexing to quickly search massive

Further more, a query may be specified with wildcards and thus will match any document with an expression substitution to that wildcard.

The definition of similarity between two mathematical expressions is a key concept that significantly affects a mathematical information retrieval system, but a formal definition of similarity is missing in the literature. Here we formally define the similarity between two mathematical expressions

1.4 Basic Definitions

For the second issue addressed in section 1.2, specifically, to assess the structural similarity, [38] gives some formal definitions, e.g. quantified score function and a n-similarity relation to address two similar expressions if they meet a threshold similarity score.

---

[8] demo page: infolab.ece.udel.edu:8912/cowpie/

Contact author: clock126@126.com or http://www.eecis.udel.edu/~zhongwei

Chapter 2

METHODOLOGY

REFERENCES

[1] Topics for the ntcir-10 math task full-text search queries. http://ntcir-math.nii.ac.jp/wp-content/blogs.dir/13/files/2014/02/NTCIR10-math-topics.pdf. Accessed: 2015-03-31.

[2] Richard Zanibbi and Dorothea Blostein. Recognition and retrieval of mathematical expressions. *International Journal on Document Analysis and Recognition (IJDAR)*, 15(4):331–357, 2012.

[3] Richard J, Fateman, and Eylon Caspi. Parsing tex into mathematics. *SIGSAM Bulletin (ACM Special Interest Group on Symbolic and Algebraic Manipulation)*, 1999.

[4] Akiko Aizawa, Michael Kohlhase, and Iadh Ounis. Ntcir-11 math-2 task overview. *The 11th NTCIR Conference*, 2014.

[5] Jozef Misutka. Mathematical search engine. Master's thesis, Charles University in Prague, May 2013.

[6] Xuan Hu, Liangcai Gao, Xiaoyan Lin, Zhi Tang, Xiaofan Lin, and Josef B. Baker. Wikimirs: A mathematical information retrieval system for wikipedia. *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries. Pages 11-20*, 2013.

[7] David Stalnaker and Richard Zanibbi. Math expression retrieval using an inverted index over symbol pairs in math expressions: The tangent math search engine at ntcir 2014. *Proc. SPIE 9402, Document Recognition and Retrieval XXII, 940207*, 2015.

[8] Qun Zhang and Abdou Youssef. An approach to math-similarity search. *Intelligent Computer Mathematics. International Conference, CICM*, 2014.

[9] Miller B. and Youssef A. Technical aspects of the digital library of mathematical functions. *Annals of Mathematics and Artificial Intelligence 38(1-3), 121136*, 2003.

[10] Youssef A. Information search and retrieval of mathematical contents: Issues and methods. *The ISCA 14th Intl Conf. on Intelligent and Adaptive Systems and Software Engineering (IASSE 2005)*, 2005.

[11] Jozef Miutka and Leo Galambo. Extending full text search engine for mathematical content. *Towards Digital Mathematics Library.*, 2008.

[12] Petr Sojka and Martin Lka. Indexing and searching mathematics in digital libraries. *Intelligent Computer Mathematics*, 6824:228–243, 2011.

[13] Petr Sojka and Martin Lka. The art of mathematics retrieval. *ACM Conference on Document Engineering, DocEng 2011*, 2011.

[14] Martin Lka. Evaluation of mathematics retrieval. Master's thesis, Masarykova University, 2013.

[15] P. Pavan Kumar, Arun Agarwal, and Chakravarthy Bhagvati. A structure based approach for mathematical expression retrieval. *Multi-disciplinary Trends in Artificial Intelligence*, 7694:23–34, 2012.

[16] Robert Miner and Rajesh Munavalli. *An Approach to Mathematical Search Through Query Formulation and Data Normalization*. Springer Berlin Heidelberg, 2007.

[17] Christopher D. Manning, Prabhakar Paghavan, and Hinrich Schutze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[18] Muhammad Adeel, Hui Siu Cheung, and Ar Hayat Khiyal. Math go! prototype of a content based mathematical formula search engine, 2008.

[19] Andrea Asperti, Ferruccio Guidi, Claudio Sacerdoti Coen, Enrico Tassi, and Stefano Zacchiroli. A content based mathematical search engine: whelp. In *In: Post-proceedings of the Types 2004 International Conference, Vol. 3839 of LNCS*, pages 17–32. Springer-Verlag, 2004.

[20] Michael Kohlhase and Ioan A. Sucan. A search engine for mathematical formulae. In *Proc. of Artificial Intelligence and Symbolic Computation, number 4120 in LNAI*, pages 241–253. Springer, 2006.

[21] Michael Kohlhase. Mathwebsearch 0.4 a semantic search engine for mathematics.

[22] Michael Kohlhase. Mathwebsearch 0.5: Scaling an open formula search engine.

[23] Peter Graf. *Term Indexing*. Springer Verlag, 1996.

[24] Yoshinori Hijikata, Hideki Hashimoto, and Shogo Nishida. An investigation of index formats for the search of mathml objects. In *Web Intelligence/IAT Workshops*, pages 244–248. IEEE, 2007.

[25] Yoshinori Hijikata, Hideki Hashimoto, and Shogo Nishida. Search mathematical formulas by mathematical formulas. *Human Interface and the Management of Information. Designing Information, Symposium on Human Interface*, pages 404–411, 2009.

[26] Hiroshi Ichikawa, Taiichi Hashimoto, Takenobu Tokunaga, and Hozumi Tanaka. New methods of retrieve sentences based on syntactic similarity. *IPSJ SIG Technical Reports, DBS-136, FI-79*, pages 39–46, 2005.

[27] Yokoi Keisuke and Aizawa Akiko. An approach to similarity search for mathematical expressions using mathml. *Towards a Digital Mathematics Library. Grand Bend, Ontario, Canada*, pages 27–35, 2009.

[28] Thomas Schellenberg, Bo Yuan, and Richard Zanibbi. Layout-based substitution tree indexing and retrieval for mathematical expressions. *Proc. SPIE 8297, Document Recognition and Retrieval XIX, 82970I*, 2012.

[29] David Stalnaker. Math expression retrieval using symbol pairs in layout trees. Master's thesis, Rochester Institute of Technology, 2013.

[30] David Stalnaker and Richard Zanibbi. Math expression retrieval using an inverted index over symbol pairs. *Proc. SPIE 9402, Document Recognition and Retrieval XXII, 940207*, 2015.

[31] Shahab Kamali and Frank Wm. Tompa. A new mathematics retrieval system. *CIKM*, 2010.

[32] Kai Ma, Siu Cheung Hui, and Kuiyu Chang. Feature extraction and clustering-based retrieval for mathematical formulas. In *Software Engineering and Data Mining (SEDM), 2010 2nd International Conference on*, pages 372–377, June 2010.

[33] Cyril Laitang, Mohand Boughanem, and Karen Pinel-Sauvagnat. Xml information retrieval through tree edit distance and structural summaries. In *Information Retrieval Technology*, volume 7097 of *Lecture Notes in Computer Science*, pages 73–83. Springer Berlin Heidelberg, 2011.

[34] Shahab Kamali and FrankWm. Tompa. Structural similarity search for mathematics retrieval. In *Intelligent Computer Mathematics*, volume 7961 of *Lecture Notes in Computer Science*, pages 246–262. Springer Berlin Heidelberg, 2013.

[35] Richard Zanibbi and Bo Yuan. Keyword and image-based retrieval for mathematical expressions. *Multi-disciplinary Trends in Artificial Intelligence. 6th International Workshop, MIWAI 2012.*, pages 23–34, 2011.

[36] Li Yu and Richard Zanibbi. Math spotting: Retrieving math in technical documents using handwritten query images. *Document Analysis and Recognition (ICDAR)*, pages 446 – 451, 2009.

[37] T. Nguyen, S. Hui, and K. Chang. A lattice-based approach for mathematical search using formal concept analysis. *Expert Systems with Applications*, 2012.

[38] Kamali Shahab and Tompa Frank Wm. Improving mathematics retrieval. *Towards a Digital Mathematics Library. Grand Bend, Ontario, Canada*, pages 37–48, 2009.

Appendix A

TITLE OF APPENDIX A

This is the information for the first appendix, Appendix A. Copy the base file, appA.tex, for each additional appendix needed such as appB.tex, appC.tex, etc. Modify the main base file to include each additional appendix file.

If there is only one appendix, then modify the main file to only use app.tex instead of appA.tex.

Appendix B

TITLE OF APPENDIX B