MathMex: Search Engine for Math Definitions

Shea Durgin, James Gore, and Behrooz Mansouri(✉)

University of Southern Maine, Portland, Maine, USA
{shea.durgin,james.gore,behrooz.mansouri}@maine.edu

Abstract. This paper introduces MathMex, an open-source search engine for math definitions. With MathMex, users can search for definitions of mathematical concepts extracted from a variety of data sources and types including text, images, and videos. Definitions are extracted using a fine-tuned SciBERT classifier, and the search is done with a fine-tuned Sentence-BERT model. MathMex interface provides means of issuing a text, formula, and combined queries and logging features.

Keywords: Math Search · Formula Search · Definition Extraction

1 Introduction

Math information retrieval refers to information retrieval where information needs are regarding math. When users issue math queries such as "Pythagorean theorem", they may be looking for its definition, examples, proof, or applications. This paper demonstrates MathMex,[1] named after Memex [2], the first hypothetical information retrieval system, carrying the idea of associative links in information. While this version primarily focuses on mathematical definitions, the ultimate goal of the MathMex project is to connect various pieces of information related to mathematical concepts, including definitions, proofs, and applications, in subsequent stages.

Our collection consists of math definitions from Wikipedia, community question-answering websites (both text and image), YouTube videos, and arXiv papers. To extract these definitions, a fine-tuned SciBERT [1] model is used to determine whether a sentence contains a definition. After extracting definitions, the semantic vector representations of definitions are extracted, using a fine-tuned Sentence-BERT [9] model. The vectors are then loaded in OpenSearch, where dense vector retrieval is performed by approximate k-NN search.

MathMex is the first math search engine, focusing on searches for specified information related to a mathematical concept. It is also the first search engine to provide a means of searching for various sources including text, images, and videos. Previous work on math search includes systems such as MathDeck [3]

S. Durgin and J. Gore—These authors contributed equally to this work
[1] https://www.mathmex.com.

N. Goharian et al. (Eds.): ECIR 2024, LNCS 14612, pp. 194–199, 2024.
https://doi.org/10.1007/978-3-031-56069-9_17

that focuses on different query editors for math formula query, and Approach0 [13] that provides means of searching over the Math Stack Exchange platform. Compared to these works, MathMex focuses on definitions and considers a wide variety of documents as the collection, indexing ∼5.8M math definitions.

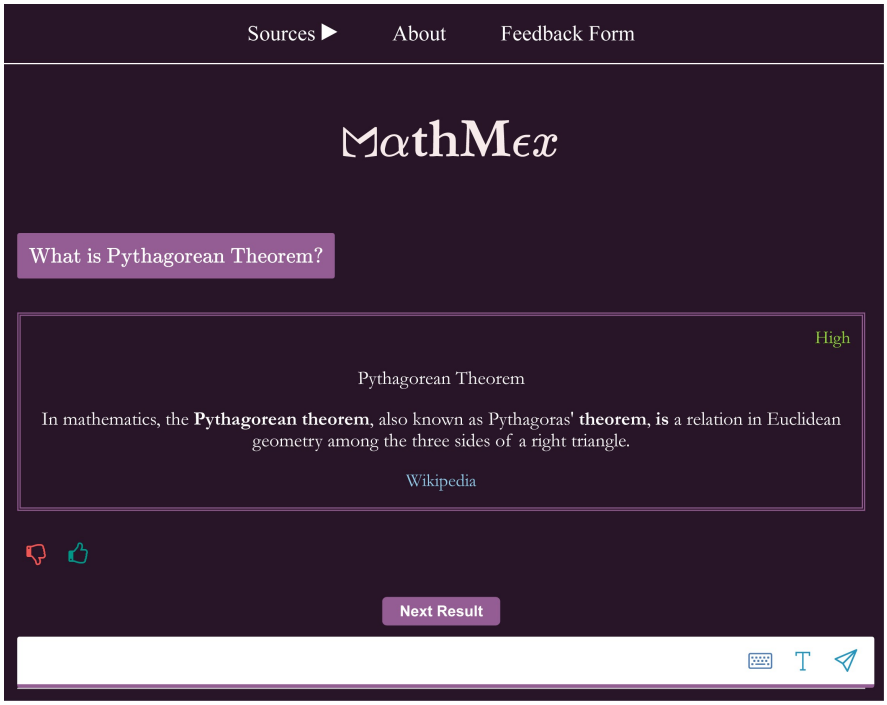

Fig. 1. MathMex Search Interface. The input query is "What is Pythagorean Theorem?"; the most relevant result from Wikipedia is returned.

2 MathMex Demo

This section introduces MathMex and its main features.

A. Collection. MathMex collection consists of the following sources:

- *ArXiv*: Papers from 2019 to July 2023. These papers are processed using their HTML format from ar5iv.[2]
- *Math CQ&A Websites*: Math Stack Exchange, MathOverFlow, and Mathematica Stack Exchange, using their July 2023 snapshot from the Internet Archive.
- *Wikipedia*: Math Wikipedia pages using NTCIR-12 [11] collection.
- *YouTube*: Math-related videos manually extracted from math courses.
- *Math-related Images*: Images from questions in CQ&A websites.

[2] https://ar5iv.labs.arxiv.org/.

B. Interface Features. MathMex utilizes various information sources. Users can filter the sources they wish to search through by employing the "Source" drop-down menu (see Fig. 1).

Using MathLive web component,[3] MathMex users can issue queries using text, LaTeX (for math), or both. This is achieved by switching the input mode of the search bar by pressing the tab or clicking the second button in the bar with T for text and $f(x)$ for math (see Fig. 2). Whenever the user pastes a text, the pasted characters will either be text or math based on the current mode.

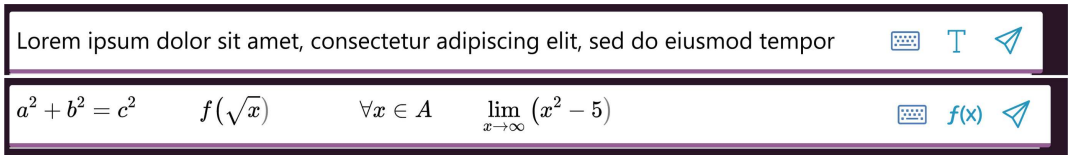

Fig. 2. MathMex Text and Formula Keyboards.

MathMex provides both text and math autocompletion features. For text mode, the suggested queries for autocompletion are determined after 3 characters are inserted (except for the first word that needs the first 5 characters). We use the bucket aggregation feature of OpenSearch on the MathMex collection to find the most common word that completes our last token. It first searches for the preceding word and if that does not return 5 unique results, it searches just using the incomplete token. The system then returns the top-5 ordered list of the potential completions of the current word of the user's query.

For the math mode, we rely on the MathLive library auto-complete feature. In this mode, after the user inserts '\', the suggested formula(e) will appear, with a live update after inserting the characters. For instance, when the user inserts '\si', queries such as sin, sigma, and sim are suggested for autocompletion.

C. Logging. Query logs are a valuable resource to study users' behavior through a search session [5]. MathMex offers explicit feedback buttons for each returned result (see Fig. 1). MathMex keeps a record of the user's activities by logging their feedback, issued query, returned definition, DateTime, and Session ID. A Session ID is assigned to a user after connecting to MathMex by assigning a UUID (universally unique ID). We plan to make these logs available to researchers every other month.

D. Indexing and Search. To extract definitions, we fine-tuned a SciBERT model [1] using data from the DEFT corpus [10], which is specifically designed for definition extraction from English text. One subtask involves determining whether a sentence contains a definition. For fine-tuning, we considered this as a binary classification task, determining whether a sentence contains a definition.

After fine-tuning the model, we utilized the spaCy[4] library to extract sentences from each document in the MathMex collection. For the text sources, each

[3] https://github.com/arnog/mathlive.
[4] https://spacy.io/.

sentence in the text was passed to the model. In the case of images, we focused on the question titles and implemented a filtering process using the CLIP [8] (Contrastive Language-Image Pre-training) model (*clip-ViT-L-14*). Images that fell below a 25% similarity threshold with their corresponding titles were removed. When it comes to YouTube videos, we considered their transcripts. Each sentence was passed through the fine-tuned SciBERT model, and if the classification result was positive with a score of ≥ 0.9, we categorized it as a definition and included it in the index.

MathMex indexing and search are done using a Sentence-BERT [9] model. We examined the bi-encoder model *'all-mpnet-base-v2'* search results, both with and without fine-tuning on ARQMath-3 test collection [7], and measured their effectiveness on the top-5 results. For fine-tuning, we used the data from ARQMAth-1 [12] and -2 [6]. ARQMath test collection provides a set of math questions, each accompanied by associated answers labeled with relevance scores 0, 1, 2, and 3.

To fine-tune the Sentence-BERT model, with ARQMath data, we considered all answers with a relevance score of 0 as non-relevant, and those with a score higher than 0, as relevant. For optimization, we utilized a cosine similarity loss function, with 10 epochs, and a batch size of 16. As presented in Table 1, the fine-tuned model provided better effectiveness across all measures. Therefore, the MathMex search model is fine-tuned using ARQMath-1, -2, and -3 test collections with a 95:5 training validation split, using the same parameters as used in our initial experiment.

Table 1. Model Results on ARQMath-3

Model	MRR@5	P@5	NDCG@5
all-mpnet-base-v2	0.77	0.62	0.41
Fine-Tuned all-mpnet-base-v2	**0.81**	**0.63**	**0.45**

For indexing, each definition is passed to the fine-tuned Sentence-BERT model, and its semantic vector representation is loaded in OpenSearch. For retrieval, the input query's vector is generated (using the same model). This vector is then compared to the definition vectors through an approximate k-NN search using *nmslib* and *Faiss* [4]. The user's chosen sources are individually searched, and the top-5 results from each source are then fused together to form the final search results. The results are fused by selecting the highest-ranked results from each source until the top-5 results are filled.

E. Search Results. MathMex provides the most relevant definition for an input query, with the next top-4 relevant results being presented, upon clicking on "Next Result" button. Users are provided with the relevance level of each retrieved instance: high, medium, or low. We evaluated the relevance levels using the cosine similarity score, where values greater than or equal to 80% are considered high, values between 60% and 80% are deemed medium, and values less than 60% are categorized as low. For each returned item, a link to the source is

available, and for YouTube videos, users are directed to specific seconds of the video.

3 Conclusion

In this paper, we demonstrated MathMex, an open-source search engine for math definitions. While users can search over a text collection, MathMex stands out as the first solution to address the challenge of searching for mathematical content within image and video resources. Moreover, it offers the flexibility of accommodating input queries in both text and mathematical modalities, empowering users to express their specific information needs. For future work, we plan to extend MathMex's capabilities to encompass a broader spectrum of information related to mathematical concepts, including proofs and practical applications.

References

1. Beltagy, I., Lo, K., Cohan, A.: SciBERT: a pretrained language model for scientific text. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (2019)
2. BUSH, V.: As we may think. Atlantic Monthly (1945)
3. Diaz, Y., Nishizawa, G., Mansouri, B., Davila, K., Zanibbi, R.: The mathdeck formula editor: interactive formula entry combining LaTeX, structure editing, and search. In: Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems
4. Johnson, J., Douze, M., Jegou, H.: Billion-scale similarity search with GPUs. IEEE Trans. Big Data **7**(3), 535–547 (2019)
5. Mansouri, B., Zanibbi, R., Oard, D.: Characterizing searches for mathematical concepts. In: 2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL) (2019)
6. Mansouri, B., Zanibbi, R., Oard, D., Agarwal, A.: Overview of ARQMath-2 (2021): second CLEF lab on answer retrieval for questions on math. In: Experimental IR Meets Multilinguality, Multimodality, and Interaction: 12th International Conference of the CLEF Association, CLEF 2021 (2021)
7. Mansouri, B., Novotny, V., Agarwal, A., Oard, D., Zanibbi, R.: Overview of ARQMath-3 (2022): third CLEF lab on answer retrieval for questions on math. In: International Conference of the Cross-Language Evaluation Forum for European Languages (2022)
8. Radford, A., et al.:Others learning transferable visual models from natural language supervision.In: International Conference On Machine Learning (2021)
9. Reimers, N., Gurevych, I.: Sentence-BERT: sentence embeddings using siamese BERT-networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (2019)
10. Spala, S., Miller, N., Yang, Y., Dernoncourt, F., Dockhorn, C.: DEFT: a corpus for definition extraction in free-and semi-structured text. In: Proceedings of the 13th Linguistic Annotation Workshop (2019)
11. Zanibbi, R., Aizawa, A., Kohlhase, M., Ounis, I., Topic, G., Davila, K.: NTCIR-12 MathIR task overview. In: NTCIR (2016)

12. Zanibbi, R., Oard, D., Agarwal, A., Mansouri, B.: Overview of ARQMath 2020: CLEF lab on answer retrieval for questions on math. In: Experimental IR Meets Multilinguality, Multimodality, and Interaction: 11th International Conference of the CLEF Association, CLEF 2020 (2020)
13. Zhong, W., Zanibbi, R.: Structural similarity search for formulas using leaf-root paths in operator subtrees. In: Advances in Information Retrieval: 41st European Conference on IR Research (2019)