Diverse Semantics Representation is King

MIRMU and MSM at ARQMath 2022

Martin Geletka[1], Vojtěch Kalivoda[1], Michal Štefánik[1], Marek Toma[1] and Petr Sojka[1]

[1]*Faculty of Informatics, Masaryk University, Botanická 68a, 602 00 Brno, Czech Republic*

Abstract

We report on the systems that the Math Information Retrieval group at Masaryk University (MIRMU) and the team of Faculty of Informatics students (MSM) prepared for task 1 (find answers) of the ARQMATH lab at the CLEF conference. To study the effects of different system settings and hyperparameters, we have prototyped several diverse math-aware information retrieval (MIR) systems: both "old" inverted index-based ones and new neural ones. By ensembling the results of the "weak" individual systems into committees, we report on entailments, benefits, and drawbacks of system ensembling. We evaluated the proposed individual systems and ensembles, considering their diversity, hyperparameters, and representations used, and classified their approaches. Our prototypes have helped to understand the challenging problems of question-answering in the STEM domain: the key lies in the proper representation of document semantics. Our reproducible evaluation Python library `pv211-utils` allows to reproduce and further advance MIR re-search.

Keywords

Information retrieval, question answering, math representations, math-aware information retrieval, word embeddings, ensembling, voting, reranking, data fusion, diversity, transformers

Content is king. Bill Gates
Properly fused diverse content is king, and context is queen. Petr Sojka

1. Introduction

Math Information Retrieval (MIR) and math-aware representation of meaning of scientific documents have been researched at MIR laboratory at Masaryk University for decades, as nicely summarized by Novotný [1] in his dissertation. As in the previous year [2], we formed two teams, MIRMU and MSM. Under the MIRMU team, we submitted five different versions of the deep neural information system, which tries to overcome the performance of TF-IDF likes systems. Under the MSM group, we submitted different versions of the student information systems and their ensemble with the best variant from the MIRMU submission. Finally, we report that an ensemble of all fine-tuned individual systems' by reciprocal rank fusion performed best.

Our ARQMATH reports [3, 4] showed promising directions stemming from the enormous capacity of neural languages models, their ensembling [5], their different training sets, hyper-parametrization, input preprocessing, and math tokenization.

CLEF 2022 – Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy
456576@mail.muni.cz (M. Geletka); 527350@mail.muni.cz (V. Kalivoda); stefanik.m@mail.muni.cz (M. Štefánik); 485275@mail.muni.cz (M. Toma); sojka@fi.muni.cz (P. Sojka)
0000-0001-6325-978X (M. Geletka); 0000-0003-1766-5538 (M. Štefánik); 0000-0002-5768-4007 (P. Sojka)
© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

Our systems were mainly developed as part of the Information Retrieval course PV211 and will allow reproducible research using Python package `PV211-utils` [6].

In Section 2 we describe resources and methods used to train and develop our systems. Section 3 describes systems and strategies used to prepare our runs. We report and evaluate our results in Section 4 on page 6. We are summing up our conclusions with Section 5, drawing possible research plans based on computed metrics and availability of collected systems, ensembling techniques, and ground truth datasets.

2. Datasets and Methods

This section describes the math representations ingested by our information retrieval systems, the corpora used for training the models that power our systems, and the relevance judgments we used for parameter optimization, model selection, and performance estimation.

2.1. Math Input Representations

We used the most straightforward math representation for all our submitted systems: LaTeX.

In one submitted run of MIRMU group, we studied the effect of the LaTeX encoding of math compared to the sole text to see how the presence of the math representation affects the resulting score.

2.2. Datasets and Methods

We described our datasets, ensembling methods, and evaluation measures in detail in our previous reports [5] and [2, Section 2]. We also used datasets from ARQMATH 2021 and ARQMATH 2022 [7].

3. Systems Description

The following sections describe nine systems students have developed as part of their studies. Their diversity brings different ways to represent the meaning of math content and how to pick and rerank the answers for given topics. Table 1 on page 7 summarizes ten submitted runs by both MU teams.

3.1. Retriever + ReRanker System

Our systems submitted under the MIRMU group consist of the following parts applied sequentially after one another, inspired by RE^{3}QA architecture [8].

- **Indexer** – to assign each document a dense vector representation;
- **Retriever** – to compute the dense vector representation of the input query and to compute the cosine similarity between query and each document representation and sorting all documents by this similarity;

- **ReRanker** – to rerank top-k most relevant documents from Retrieving part. We achieved the best results with tiered reranking, taking multiple non-overlapping slices from top-k results and reranks each slice separately. This part is computationally expensive; therefore, we cannot do it on the whole dataset in practice.

For implementing our system, we used the Sentence Transformer library, which is the Python framework for state-of-the-art sentence, text, and image embeddings. [9]

3.1.1. Implementation and Hyper-parameters

For the implementation of the Base model, we used the BiEncoder model with the pre-trained `all-MiniLM-L12-v2` model [10]. For the ReRanking phase, we fine-tuned the CrossEncoder model with the pre-trained `roberta-large` model [11]. Identically, we experimented with math-specific CrossEncoder model, MathBERTa[1] [12], that extends generic RoBERTa with math-specific tokenization and fine-tunes the extended model using ArXiv collection.

We used Sentence Transformers library[2] for fine-tuning of all our models. The architecture of both models and their differences are depicted in Figure 1. The Base model fine-tuned only the ReRanker model, as the fine-tuning of the generic Retriever pre-trained for a retrieval on a vast and heterogeneous datasets did not bring measurable benefits of quality. We submitted fine-tuned version of the ReRanker as the alternative run.

3.1.2. ReRanker fine-tuning

The ReRanker was fine-tuned using the BCE with logits loss and the AdamW optimizer. The training set consisted of relevance judgments, a train+validation set from ARQMath 2021, extended by additional samples. The additional positive samples, with the label equal to 1, were pairs of a question and an answer from the same thread, where the answer received at least 50 upvotes. The additional negative samples, with the label equal to 0, were the most similar documents given by Retriever but with no overlapping Math Stack Exchange tags. Because finding similar documents for all questions would not be feasible, we decided to generate ten negative samples for ten random questions at the start of each epoch.

The model was trained on all training queries and the same number of additional samples every epoch. The ratio of positive and negative samples was 0.5. The training process was stopped as there was no decrease in loss over the test set from ARQMath 2021.

3.1.3. Description of Individual Runs

For the MIRMU team, we submitted different versions of the described system to see the effect of the different components of the system. The variation we used are:

- finetuning / not finetuning of the Retriever model
- using ReRanker model pretrained on the Math texts (RoBERTa vs MathBERTa[3]) [12]

[1] https://huggingface.co/witiko/mathberta
[2] https://www.sbert.net/docs/pretrained_models.html
[3] https://huggingface.co/witiko/mathberta

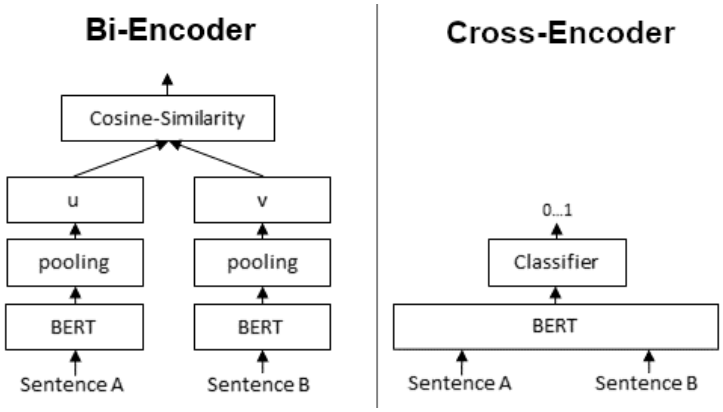

Figure 1: Bi-Encoder vs Cross-Encoder model scheme (image taken from www.sbert.net/examples/applications/cross-encoder/README.html)

- using different representation of the input (text vs text + LaTeX)

We submitted the Base model as the primary as we couldn't outperform its performance. The other variants we submitted as the alternative runs are:

- **Base** – model described in the previous section;
- **Trained only on text** – system using only the text representation of documents and queries;
- **MathBERTa ReRanker** – system using MathBERTa model as the ReRanker;
- **Retriever fine-tuned** – Retriever model adjusted by the relevance judgements collected over ARQMath2020 and ARQMath 2021, using Negatives Ranking loss [13, 5];
- **MathBERTa ReRanker + Retriever fine-tuned** – system with MathBERTa ReRanker and fine-tuned Retriever model.

To quantify the effect of the individual changes in the MIRMU system, only one parameter was changed at time for given run, and all other hyperparameters were left the same.

3.2. TF-IDF

As one of the students' baseline systems, we used the TF-IDF model implementation available in the Gensim library [14].

In the preprocessing phase, we removed extreme values (below 8 absolute term frequency and higher than 0.7 relative document frequency), where we chose the hyperparameters experimentally on train subset. Then we removed punctuation, repeating whitespaces. Lastly, we tokenized the document by splitting on whitespace and stemmed individual tokens using the snowball stemmer available in snowball_py [4] library.

We used SMART (System for the Mechanical Analysis and Retrieval of Text) $l\ n\ c$ TF-IDF weighting variant [15]. We used *l*ogarithm for term frequency weighting, *n*o document weighting, and *c*osine document length normalization.

[4] https://github.com/shibukawa/snowball_py

3.3. BM25

BM25$^+$ is an improvement over BM25 introduced by Lv and Zhai [16]. Together with other alternatives, such as BM25-L, BM25-adapt, and BM25-T, this improvement surpasses the basic BM25 algorithm on TREC collections. [17] BM25$^+$ estimates the relevance of a document d for a query q by formula (1).

$$\mathrm{BM25}^+(d,q) = \sum_{t \in q} \log\left(\frac{N+1}{\mathrm{df}_t}\right) \cdot \left(\frac{(k_1+1)\cdot \mathrm{tf}_{t,d}}{k_1 \cdot \left((1-b)+b\left(\frac{L_\mathrm{d}}{L_\mathrm{avg}}\right)\right) + \mathrm{t}_\mathrm{fd}} + \delta\right), \tag{1}$$

where k_1, b, and δ are hyperparameters, N is the number of documents in the collection, df_t is the number of documents containing the term t, $\mathrm{tf}_{t,d}$ is frequency of term t in document d, L_d the length of document d in words, and L_avg is the expected length of a document in words.

We represented our answers as a concatenation of its body and title, body, and tags of its parent question. In the next preprocessing stage, we firstly removed punctuation, repeating whitespaces, and English stopwords. Then we transformed the text into lowercase and stemmed it using Porter stemmer. Lastly, we tokenized the text by splitting it on whitespace.

We used the implementation of BM25$^+$ in the `rank_bm25` Python library [18] with its default hyperparameters.

3.4. BM25 + TF-IDF ensemble

As the example of the most straightforward possible ensemble system of two unsupervised systems, we used the ensembling of TF-IDF and BM25 models. We constructed the ensemble as the simple sum of the given scores from the individual systems. The BM25 is configured as described in Section 3.3. The TF-IDF system uses the same preprocessing as BM25 system and TF-IDF implementation available in the Gensim library with SMART $l\,t\,n$ TF-IDF weighting variant, which corresponds to *l*ogarithmic term frequency weighting, zero-corrected IDF, and *n*o document normalization.

3.5. COMPUBERT

We also submitted the COMPUBERT model prepared and submitted by the MIRMU team last year. The model and hyper-parameters can be found in last year's report [2, Section 3.4].

3.6. Ensemble Systems

This section describes the used ensemble systems. With ensembling and voting techniques, we can combine the strengths of different systems to produce more accurate results. Historically, there is a long tradition of boosting, [19, 20], ensembling [21], data fusion [22] and voting approaches [23, 24] in the information retrieval research.

As we believe that our systems could agree on a small portion of the most relevant documents, reflecting different 'points of view' on the search problem. Depending on dozens of parameters, each individual system will miss the great majority of relevant documents. With ensembling and voting techniques, we can combine the strengths of different systems to produce more accurate

results. [2]. All our ensemble algorithms are agnostic to the scoring functions of individual systems and only use the ranks of the results.

3.6.1. IBC

The IBC is ensemble technique we introduced in ARQMATH 2020 in our paper [5]. The ensemble combines SERPs from the individual systems by Median Inverse Rank, which is equal to $(1000 - M)/1000$, where M is Median Rank of individual systems. For detail explanation of the IBC ensemble algorithm we refer the reader to our ARQMATH 2020 paper. [5]

3.6.2. RRF

We used reciprocal rank fusion (RRF) [25] to construct the ensemble model from the previously described systems. The ensemble, given ranks from all individual systems, sorts the documents by Formula (2).

$$\text{RRF}^{+}(d) = \sum_{r \in R} \frac{1}{k + r(d)} \tag{2}$$

where R is set of rankings and $r(d)$ is ranking of document d. The hyper-parameter k parameterizes the ensemble. We used default value of $k = 60$, suggested in [25].

3.6.3. RBC

As the RBC model we refer to trained regression model, which predicts the gain of train judgements from the ranks of the individual.

For the performance estimation of RBC, we produced a result list by taking the 1,000 answers with the highest predicted gain for each topic in the test subset.

3.6.4. WIBC

WIBC is weighted variant of the IBC algorithm. Instead of electing the candidate with the highest median rating, WIBC elects the candidate with the highest weighted median rating. Instead of breaking ties by selecting a random rating out of a uniform distribution of all ratings, we select a random rating out of a weighted uniform distribution.

4. Evaluation and Results

To compare the submitted systems, we evaluate their performance on the topics from previous ARQMATH competitions.

4.1. Submitted runs

For each system, we report the resulted scores as in Table 2 in similar form as in the overview ARQMATH 2022 paper by Mansouri et al. [7].

Performance drop in MIRMU 2 run compared to other runs indicate that training and adjusting the system on math is a must.

Table 1
Run submitted by the MSM and MIRMU teams. First run by each team was submitted as primary

Team	Run	official nick	System description
MSM	1	Ensemble_RRF_auto-both-P_primary	RRF Ensemble
MSM	2	TF-IDF-auto-both-A	TF-IDF
MSM	3	BM25_system-auto-both-A	BM25
MSM	4	BM25_TfIdf_system-auto-both-A	TF-IDF + BM25
MSM	5	CompuBERT22-auto-both-A	CompuBERT
MIRMU	1	MiniLM+RoBERTa-auto-both-P	Base
MIRMU	2	MiniLM+RoBERTa-auto-text-A	Trained only on text
MIRMU	3	MiniLM+MathRoBERTa-auto-both-A	MathBERTa ReRanker
MIRMU	4	MiniLM_tuned+RoBERTa-auto-both-A	Retriever fine-tuned
MIRMU	5	MiniLM_tuned+MathRoBERTa-auto-both-A	MathBERTa ReRanker + Retriever fine-tuned

4.2. Runs with enhanced systems and ensembles

We further fine-tuned our systems benefiting from more ground truth data and experience we got by previous evaluations.

- **MathBERTa ReRanker** – MathBerta ReRanker fine-tuned on altered preprocessing;
- **BM25-based system** – BM25-based system MSM 3 described in Section 3.3, where we optimized hyperparameters k_1, b, and δ using grid search. The values we found to yield the best NDCG′ on our training set are $k_1 = 1.8$, $b = 0.75$, and $\delta = 1$.
- **Improved Base** – Base system with reduced preprocessing, more precisely trained ReRanker and improved tiered reranking with slices on indexes: 3, 7, 12, 16, 20, 50, 100, 125.
- **Retriever only** – A system with removed reranking phase. Document ranking is based only on cosine similarity on embeddings obtained from Retriever.

We report the results of extended experiments with ensembles in Table 3 on the next page. RRF ensembles deliver best results by far margin. The more diverse systems one combines, the more the quality metric as NDCG′ monotonously grows. As the math information systems have to cope with really complex problems, it is really hard to build one complex system that is capable of learning all the complex stuff: disambiguation of overloaded math symbols, structured ambiguous notation, deduction and long causal dependencies.

Table 2
ARQMATH 2022 competition results of the runs submitted by the MSM team (1 ensemble and 4 diverse systems) and MIRMU team (5 variants of neural-based systems)

System	2020			2021			2022		
	nDCG′	MAP′@10	P′@10	nDCG′	MAP′@10	P′@10	nDCG′	MAP′@10	P′@10
MSM runs									
MSM 1: RRF– Ensemble of all 4 MSM + Best MIRMU	**0.422**	**0.172**	**0.197**	**0.381**	**0.119**	**0.152**	**0.511**	**0.159**	**0.244**
MSM 2: TF-IDF	0.238	0.074	0.117	0.169	0.040	0.076	0.284	0.065	0.082
MSM 3: BM25	0.332	0.123	0.168	0.285	0.082	0.116	0.401	0.124	0.196
MSM 4: TF-IDF + BM25	0.332	0.123	0.168	0.286	0.083	0.116	0.401	0.124	0.196
MSM 5: COMPUBERT	0.115	0.038	0.099	0.098	0.030	0.090	0.132	0.025	0.060
MIRMU runs									
MIRMU 1: Base	0.466	0.246	**0.339**	**0.487**	**0.233**	**0.316**	**0.505**	**0.186**	0.270
MIRMU 2: Trained only on text	0.298	0.124	0.201	0.277	0.104	0.180	0.354	0.109	0.161
MIRMU 3: MathBERTa ReRanker	**0.470**	**0.250**	0.338	0.484	0.227	0.310	0.503	0.183	**0.277**
MIRMU 4: Retriever fine-tuned	0.466	0.246	**0.339**	**0.487**	**0.233**	**0.316**	0.478	0.167	0.247
MIRMU 5: MathBERTa ReRanker + Retriever fine-tuned	**0.470**	0.248	0.335	0.472	0.221	0.309	0.500	0.180	0.265

Table 3
Results of another (not submitted) runs of fine-tuned systems and ensembles.

System	2020			2021			2022		
	nDCG′	MAP′@10	P′@10	nDCG′	MAP′@10	P′@10	nDCG′	MAP′@10	P′@10
Best systems' runs prepared *ex post*									
FINE 1: MathBERTa ReRanker	0.465	0.243	0.342	0.480	0.222	**0.308**	0.510	0.191	0.275
FINE 2: BM25-based system best	0.334	0.124	0.169	0.288	0.083	0.114	0.402	0.123	0.196
FINE 3: Improved Base	**0.468**	**0.249**	**0.351**	**0.487**	**0.229**	0.304	**0.514**	**0.194**	0.275
FINE 4: Retriever only	0.462	0.241	0.334	0.479	0.221	0.301	0.507	0.186	**0.278**
Ensembles									
ENS 1: RRF 60 of 4 FINE systems	**0.493**	0.253	0.333	**0.493**	0.217	**0.306**	0.551	0.207	0.313
ENS 2: IBC of all	0.401	0.197	0.295	0.400	0.170	0.244	0.473	0.177	0.287
ENS 3: IBC of all 4 MSM	0.324	0.114	0.148	0.285	0.079	0.114	0.407	0.122	0.190
ENS 4: IBC of all 5 MIRMU	0.468	0.247	0.339	0.485	0.229	0.317	0.504	0.188	0.270
ENS 5: IBC of all 4 MSM +MIRMU 1	0.354	0.136	0.181	0.326	0.109	0.156	0.511	0.159	0.245
ENS 6: IBC of MSM 4 and MIRMU 1	0.459	0.200	0.258	0.326	0.109	0.156	0.543	0.197	0.292
ENS 7: RRF 60 of all	0.480	0.237	0.314	0.471	0.195	0.290	0.570	0.209	**0.329**
ENS 8: RRF 180 of all	0.486	0.237	0.314	0.467	0.186	0.261	**0.576**	**0.214**	0.309
ENS 9: RRF 60 of all 4 MSM	0.328	0.125	0.169	0.277	0.078	0.118	0.406	0.114	0.210
ENS 10: RRF 60 of all 5 MIRMU	0.465	0.244	0.323	0.473	0.215	0.294	0.521	0.191	0.280
ENS 11: RRF 60 of all 4 MSM and MIRMU 1	0.422	0.172	0.197	0.381	0.119	0.152	0.511	0.159	0.245
ENS 12: RRF 60 of MSM 4 + MIRMU 1	0.465	0.211	0.268	0.455	0.177	0.231	0.544	0.198	0.303
ENS 13: RBC of all	0.476	0.217	0.267	0.442	0.164	0.190	N/A	N/A	N/A
ENS 14: RBC of all 4 MSM	0.312	0.115	0.116	0.274	0.074	0.107	N/A	N/A	N/A
ENS 15: RBC of all MIRMU	0.468	0.247	0.339	0.423	0.165	0.211	N/A	N/A	N/A
ENS 16: RBC of all 4 MSM and MIRMU 1	0.475	0.220	0.273	0.453	0.171	0.204	N/A	N/A	N/A
ENS 17: RBC of MSM 4 and MIRMU 1	0.474	0.220	0.286	0.468	0.193	0.245	N/A	N/A	N/A
ENS 18: WIBC of all	0.466	0.246	0.339	0.487	0.234	0.316	N/A	N/A	N/A
ENS 19: WIBC of all 4 MSM	0.332	0.123	0.168	0.285	0.082	0.116	N/A	N/A	N/A
ENS 20: WIBC of all MIRMU	0.466	0.246	0.339	0.487	0.233	0.316	N/A	N/A	N/A
ENS 21: WIBC of all 4 MSM and MIRMU 1	0.488	**0.274**	**0.350**	0.285	0.082	0.113	N/A	N/A	N/A
ENS 22: WIBC of MSM 4 and MIRMU 1	0.466	0.246	0.339	0.487	**0.233**	0.316	N/A	N/A	N/A

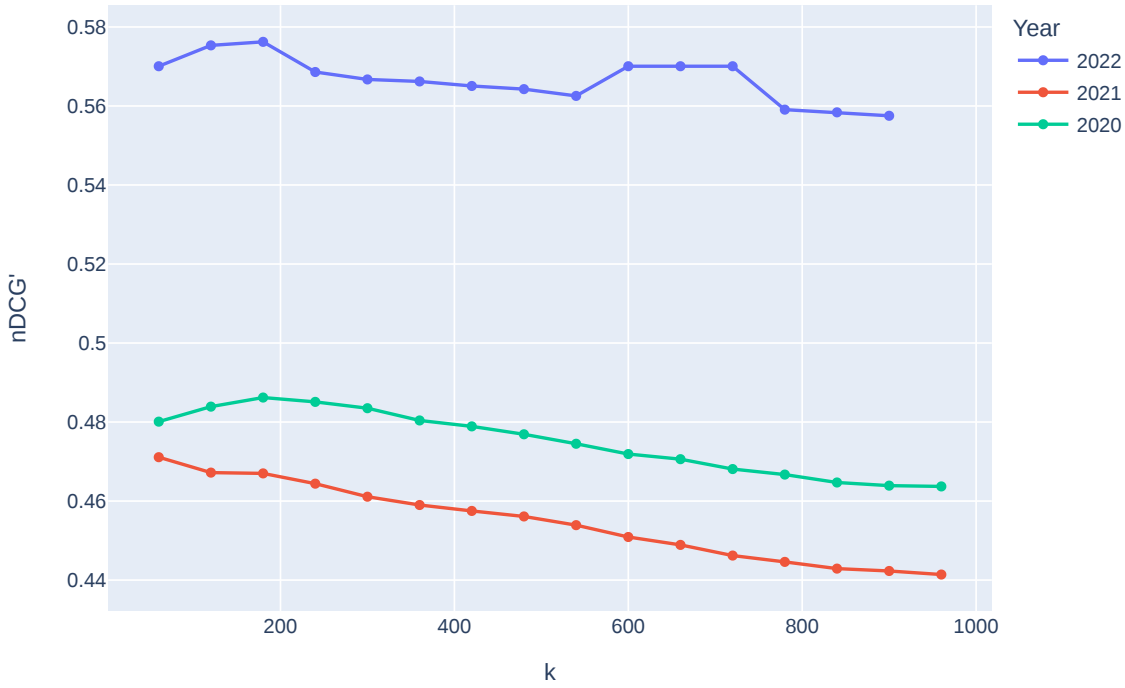

Figure 2: Performance of RRF ensemble of all 9 submitted individual systems, depending on hyperparameter k. The best $k = 180$ reported for ARQMATH 2022 is $nDCG' = 0.576$

The Figure 2 shows the dependence of RRF quality on the parameter k. Instead of suggested $k = 60$ from the original paper, the best performance is with $k = 180$. We hypotetize that the more diverse the primary systems are, the higher optimal parameter k scores.

It remains to be studied to which extend the performance depends on participating individual systems. Also, how the performance change with different hyperparameters and initial setting and choices of pre-trained models, diversity and initial setup of individual systems.

For reproducibility, we are going to publish our notebooks, ensemble implementations and models in our lab and course repository [6].

"You must cultivate activities that you love. You must discover work that you do, not for its utility, but for itself, whether it succeeds or not, whether you are praised for it or not, whether you are loved and rewarded for it or not, whether people know about it and are grateful to you for it or not." Anthony de Mello

5. Conclusion & Future Work

We have developed nine MIR systems with as diverse approaches and variants as possible. We have evaluated them on available ARQMATH data from last three years. We have studied the ways how they could be ensembled to gain better performance. We have reported our findings:

a) math-aware representation with deep models started to outperform flat token-level based systems b) ensembling done with expertise and insight about merits of individual systems matter.

In the future, we plan to further enhance our neural models and ensembling algorithms by several means:

- evaluation of different ensembling strategies based on individual systems' types and hyperparameter settings of neural systems;
- evaluation of initial setting and hyperparameters of neural Retriever and ReRanker systems;
- deep systems' adaptation to math specifics by using Adapt$\mathcal{O}$r library [26];
- study robustness and out-of-domain performance of MIR systems.

Acknowledgments

We thank all PV211 course students and former members of MIR group for their contributions. We thank the two anonymous reviewers for their insightful comments. We extend our gratitude to the ARQMATH 2022 organizers for keeping the research of math information retrieval aflame.

This work has been partly supported by the Ministry of Education of CR within the LINDAT-CLARIAH-CZ project LM2018101.

References

[1] V. Novotný, Interpretable Representations for Fast and Accurate Retrieval of Mathematical Information [online], Dissertation, Masaryk University, Faculty of Informatics, Brno, 2022 [cit. 2022-05-26]. URL: https://is.muni.cz/th/o4thd/Revidovana_verze_po_obhajobe_disertace.pdf, supervisor: Petr Sojka.

[2] V. Novotný, M. Štefánik, D. Lupták, M. Geletka, P. Zelina, P. Sojka, Ensembling Ten Math Information Retrieval Systems: MIRMU and MSM at ARQMath 2021, in: Proceedings of the Working Notes of CLEF 2021 – Conference and Labs of the Evaluation Forum, volume 2696, CEUR-WS, Bucharest, Romania, 2021, pp. 82–106. URL: http://ceur-ws.org/Vol-2936/paper-06.pdf.

[3] A. Reusch, M. Thiele, W. Lehner, TU_DBS in the ARQMath Lab 2021, CLEF, in: Proceedings of the Working Notes of CLEF 2021 – Conference and Labs of the Evaluation Forum, volume 2696, CEUR-WS, Bucharest, Romania, 2021, pp. 107–124. URL: http://ceur-ws.org/Vol-2936/paper-07.pdf.

[4] S. Rohatgi, J. Wu, C. L. Giles, Ranked List Fusion and Re-ranking with Pre-trained Transformers for ARQMath Lab, in: Proceedings of the Working Notes of CLEF 2021 – Conference and Labs of the Evaluation Forum, volume 2696, CEUR-WS, Bucharest, Romania, 2021, pp. 125–132. URL: http://ceur-ws.org/Vol-2936/paper-08.pdf.

[5] V. Novotný, P. Sojka, M. Štefánik, D. Lupták, Three is Better than One, in: CEUR Workshop Proceedings: ARQMath task at CLEF conference, volume 2696, CEUR-WS, Thessaloniki, Greece, 2020, pp. 1–30. URL: http://ceur-ws.org/Vol-2696/paper_235.pdf.

[6] M. Štefánik, V. Novotný, M. Geletka, V. Kalivoda, M. Toma, D. Lupták, P. Sojka, PV211 Utils, 2022. URL: https://github.com/MIR-MU/pv211-utils/.
[7] B. Mansouri, V. Novotný, A. Agarwal, D. W. Oard, R. Zanibbi, Overview of ARQMath-3 (2022): Third CLEF lab on Answer Retrieval for Questions on Math (Working Notes Version), in: G. Faggioli, N. Ferro, A. Hanbury, M. Potthast (Eds.), Working Notes of CLEF 2022 – Conference and Labs of the Evaluation Forum, CEUR-WS, 2022.
[8] M. Hu, Y. Peng, Z. Huang, D. Li, Retrieve, Read, Rerank: Towards End-to-End Multi-Document Reading Comprehension, in: Proceedings of the 57th Annual Meeting of the ACL, 2019, pp. 2285–2295. doi:10.18653/v1/P19-1221.
[9] N. Reimers, I. Gurevych, Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, in: Proceedings of the 2019 Conference on Empirical Methods in NLP and the 9th International Joint Conference on NLP (EMNLP-IJCNLP), ACL, Hong Kong, China, 2019, pp. 3982–3992. doi:10.18653/v1/D19-1410.
[10] W. Wang, H. Bao, S. Huang, L. Dong, F. Wei, MiniLMv2: Multi-head self-attention relation distillation for compressing pretrained transformers, in: Findings of the ACL: ACL-IJCNLP 2021, ACL, 2021, pp. 2140–2151. URL: https://aclanthology.org/2021.findings-acl.188. doi:10.18653/v1/2021.findings-acl.188.
[11] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A Robustly Optimized BERT Pretraining Approach, ArXiv abs/1907.11692 (2019). URL: https://openreview.net/forum?id=SyxS0T4tvS.
[12] V. Novotný, M. Štefánik, Combining Sparse and Dense Information Retrieval, in: Proceedings of the Working Notes of CLEF 2022, CEUR-WS, 2022. To appear.
[13] M. Henderson, R. Al-Rfou, B. Strope, Y.-H. Sung, L. Lukács, R. Guo, S. Kumar, B. Miklos, R. Kurzweil, Efficient Natural Language Response Suggestion for Smart Reply, ArXiv abs/1705.00652 (2017). doi:10.48550/ARXIV.1705.00652.
[14] R. Řehůřek, P. Sojka, Software Framework for Topic Modelling with Large Corpora, in: Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks, ELRA, Valletta, Malta, 2010, pp. 45–50. doi:10.13140/2.1.2393.1847.
[15] G. Salton, C. Buckley, Term-weighting approaches in automatic text retrieval, Information Processing & Management 24 (1988) 513–523. doi:10.1016/0306-4573(88)90021-0.
[16] Y. Lv, C. Zhai, A Log-Logistic Model-Based Interpretation of TF Normalization of BM25, in: R. Baeza-Yates, A. P. de Vries, H. Zaragoza, B. B. Cambazoglu, V. Murdock, R. Lempel, F. Silvestri (Eds.), Advances in Information Retrieval, Springer, Berlin, Heidelberg, 2012, pp. 244–255. doi:10.1007/978-3-642-28997-2_21.
[17] A. Trotman, A. Puurula, B. Burgess, Improvements to BM25 and Language Models Examined, in: Proceedings of the 2014 Australasian Document Computing Symposium, ADCS '14, ACM, New York, NY, USA, 2014, pp. 58–65. doi:10.1145/2682862.2682863.
[18] D. Brown, S. Jain, V. Novotný, nlp4whp, dorianbrown/rank_bm25:, 2022. doi:10.5281/zenodo.6106156.
[19] A. Gulin, I. Kuralenok, D. Pavlov, Winning The Transfer Learning Track of Yahoo!'s Learning To Rank Challenge with YetiRank, in: O. Chapelle, Y. Chang, T.-Y. Liu (Eds.), Proceedings of the Learning to Rank Challenge, volume 14 of *Proceedings of Machine Learning Research*, PMLR, Haifa, Israel, 2011, pp. 63–76. URL: http://proceedings.mlr.press/v14/gulin11a.html.

[20] Q. Wu, C. J. C. Burges, K. M. Svore, J. Gao, Adapting boosting for information retrieval measures, Information Retrieval 13 (2010) 254–270. doi:10.1007/s10791-009-9112-1.

[21] Y. Wang, I.-C. Choi, H. Liu, Generalized Ensemble Model for Document Ranking in Information Retrieval, Computer Science and Information Systems 14 (2017) 123––151. doi:10.2298/csis160229042w.

[22] R. Nuray, F. Can, Automatic Ranking of Information Retrieval Systems Using Data Fusion, Information Processing and Management 42 (2006) 595–614. doi:10.1016/j.ipm.2005.03.023.

[23] M. Mosbah, B. Boucheham, Majority Voting Re-ranking Algorithm for Content Based-Image Retrieval, in: E. Garoufallou, R. J. Hartley, P. Gaitanou (Eds.), Metadata and Semantics Research, Springer International Publishing, Cham, 2015, pp. 121–131.

[24] A. T. Albaham, N. Salim, Quality Biased Thread Retrieval Using the Voting Model, in: Proceedings of the 18th Australasian Document Computing Symposium, ADCS '13, ACM, New York, NY, USA, 2013, pp. 97–100. doi:10.1145/2537734.2537752.

[25] G. V. Cormack, C. L. A. Clarke, S. Buettcher, Reciprocal Rank Fusion Outperforms Condorcet and Individual Rank Learning Methods, in: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '09, ACM, New York, NY, USA, 2009, pp. 758–759. doi:10.1145/1571941.1572114.

[26] M. Štefánik, V. Novotný, N. Groverová, P. Sojka, Adapt$\mathcal{O}$r: Objective-Centric Adaptation Framework for Language Models, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL, Dublin, Ireland, 2022, pp. 261–269. URL: https://aclanthology.org/2022.acl-demo.26.