Cross Encoding as Augmentation: Towards Effective Educational Text Classification

Hyun Seung Lee[*] [1,2] **Seungtaek Choi**[*] [†] [1]
Yunsung Lee[1] **Hyeongdon Moon**[1] **Shinhyeok Oh**[1]
Myeongho Jeong[1] **Hyojun Go**[1] **Christian Wallraven**[†] [2]
[1]Riiid AI Research
[2]Department of Artificial Intelligence, Korea University
{hyunseung.lee, seungtaek.choi}@riiid.co,
{hslrock, wallraven}@korea.ac.kr

Abstract

Text classification in education, usually called *auto-tagging*, is the automated process of assigning relevant tags to educational content, such as questions and textbooks. However, auto-tagging suffers from a data scarcity problem, which stems from two major challenges: 1) it possesses a large tag space and 2) it is multi-label. Though a retrieval approach is reportedly good at low-resource scenarios, there have been fewer efforts to directly address the data scarcity problem. To mitigate these issues, here we propose a novel retrieval approach CEAA that provides effective learning in educational text classification. Our main contributions are as follows: 1) we leverage transfer learning from question-answering datasets, and 2) we propose a simple but effective data augmentation method introducing cross-encoder style texts to a bi-encoder architecture for more efficient inference. An extensive set of experiments shows that our proposed method is effective in multi-label scenarios and low-resource tags compared to state-of-the-art models.

1 Introduction

Due to the overwhelming amount of educational content available, students and teachers often struggle to find what to learn and what to teach. Auto-tagging, or text classification in education, enables efficient curation of content by automatically assigning relevant tags to educational materials, which aids in both students' understanding and teachers' planning (Goel et al., 2022).

However, applying auto-tagging for real-world education is challenging due to **data scarcity**. This is because auto-tagging has a potentially very large label space, ranging from subject topics to knowledge components (KC) (Zhang et al., 2015; Koedinger et al., 2012; Mohania et al., 2021; Viswanathan et al., 2022). The resulting data scarcity decreases performance on rare labels during training (Chalkidis et al., 2020; Lu et al., 2020; Snell et al., 2017; Choi et al., 2022).

In this paper, we aim to solve the data scarcity problem by formulating the task as a retrieval problem following a recent proposal (Viswanathan et al., 2022). This can utilize a language model's ability to understand the tag text, such that even for an unseen tag, the models would be able to capture the relationship between the terms in the input content and labels. However, performance in the auto-tagging context still critically depends on the amount of training data.

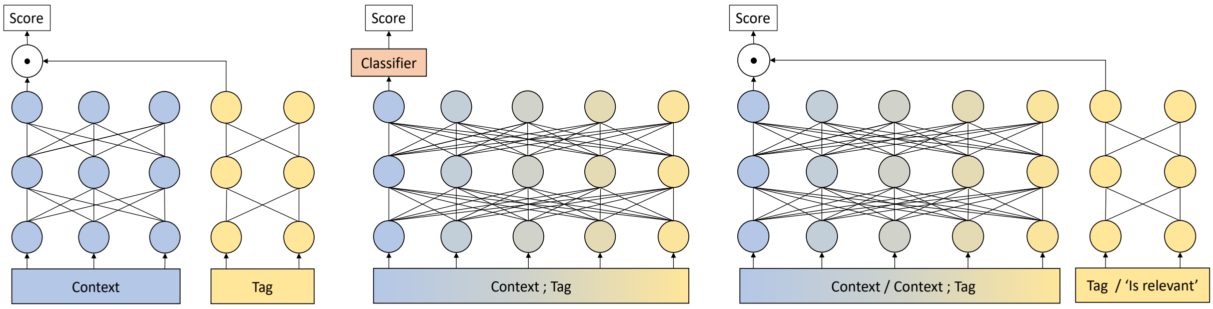
To this end, we first propose to leverage the knowledge of language models that are fine-tuned on large question-answering datasets. Our intuition is that question of finding an answer in a passage can be a direct (or indirect) summary of the passage (Nogueira et al., 2019b), which can serve as an efficient proxy of the gold tag for educational content. The large question-answering datasets thus become a better prior for the tag spaces. Specifically, we adopt a recent bi-encoder architecture, called DPR (Karpukhin et al., 2020)[1], for transfer learning, which performs BERT encoding over the input and candidate label separately and measures the similarity between the final representations. To the best of our knowledge, our work is the first to leverage transfer learning from QA models for text classification tasks.

As a further innovation, we introduce a novel data augmentation method for training a bi-encoder architecture, named CEAA, which adds the cross-encoder *view* of the input-label pair in the bi-encoder architecture, as shown in Figure 1. By capturing the full interaction between input and labels already during training time, the models can be further optimized to take advantage of token-

[*] Equal Contribution.
[†] Corresponding authors.

[1] DPR model is trained on 307k training questions, which is much larger than 7k questions in ARC dataset (Xu et al., 2019) we used in experiments.

Findings of the Association for Computational Linguistics: ACL 2023, pages 2184–2195
July 9-14, 2023 ©2023 Association for Computational Linguistics



(a) Bi-Encoder (b) Cross-Encoder (c) CEAA

Figure 1: Comparative Illustration of Encoding Methods. CEAA is done for the bi-encoder to process input in which context and tag are given together, computing full token-level interactions between context and tag.

level interactions that are missing in traditional bi-encoder training. At the same time, the computational efficiency of the bi-encoder is maintained, which makes CEAA able to tackle large label spaces as opposed to existing solutions based on cross-encoder architectures (Urbanek et al., 2019; Wolf et al., 2019; Vig and Ramea, 2019). Experiments show that CEAA provides significant boosts to performance on most metrics for three different datasets when compared to state-of-the-art models. We also demonstrate the efficacy of the method in multi-label settings with constraints of training only with a single label per context.

2 Related Work

Text classification in the education domain is reportedly difficult as the tags (or, labels) are hierarchical (Xu et al., 2019; Goel et al., 2022; Mohania et al., 2021), grow flexibly, and can be multi-labeled (Medini et al., 2019; Dekel and Shamir, 2010). Though retrieval-based methods were effective for such long-tailed and multi-label datasets (Zhang et al., 2022; Chang et al., 2019), they relied on vanilla BERT (Devlin et al., 2018) models, leaving room for improvement, for which we leverage question-answering fine-tuned retrieval models (Karpukhin et al., 2020).

Recently, (Viswanathan et al., 2022) proposed TagRec++ using a bi-encoder framework similar to ours, with an introduction of an additional cross-attention block. However, this architecture loses the efficiency of the bi-encoder architecture in the large taxonomy space for the education domain. Unlike TagRec++, our distinction is that we leverage the cross-attention only in training time via input augmentation.

3 Approach

3.1 Problem formulation

In this paper, we address the text classification task, which aims to associate an input text with its corresponding class label, as a retrieval problem. Formally, given a context c and tag candidates $\mathcal{T}$, the goal of the retrieval model is to find the correct (or, relevant) tag $t \in \mathcal{T}$, where its relevance score with the context $s(c, t)$ is the highest among the $\mathcal{T}$ or higher than a threshold. For this purpose, our focus is to better train the scoring function $s(c, t)$ to be optimized against the given relevance score between the context c and candidate tag t.

3.2 Bi-Encoder

In this paper, we use a bi-encoder as a base architecture for the retrieval task, as it is widely used for its fast inference (Karpukhin et al., 2020). Specifically, the bi-encoder consists of two encoders, E_C, and E_T, which generate embedding for the context c and the tag t. The similarity between the context and tag is measured using the dot-product of their vectors:

$$s_{\text{BE}}(c, t) = E_C(c) \cdot E_T(t)^\top \tag{1}$$

Both encoders are based on the BERT architecture (Devlin et al., 2018), specifically *"bert-base-uncased"* provided by HuggingFace (Wolf et al., 2020), that is optimized with the training objective of predicting randomly-masked tokens within a sentence. We use the last layer's hidden layer of the classification token is used as context and tag embeddings.

For training the bi-encoder, we follow the in-batch negative training in (Karpukhin et al., 2020). Gold tags from other contexts inside the batch are treated as negative tags. As tags are often multi-labeled, we use *binary cross-entropy loss*:

$$\mathcal{L} = -\frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{N}(y_{i,j}\log(s(c_i, t_j)) + (1 - y_{i,j})\log(1 - s(c_i, t_j))) \tag{2}$$

where $s(c_i, t_j)$ scores the similarity between context c_i and tag t_j, and $y_{i,j}$ is 1 if they are relevant and 0 otherwise. We will denote this model variant as a bi-encoder (BERT) below.

3.3 Cross-Encoding As Augmentation

The cross-encoder (Nogueira and Cho, 2019) is another method in information retrieval tasks in which a single BERT model receives two inputs joined by a special separator token as follows:

$$s_{\text{CE}}(c, t) = F(E([c; t])), \tag{3}$$

where F is a neural function that takes the representation of the given sequence.

Cross-encoders perform better than bi-encoders as they directly compute cross-attention over context and tag along the layers (Urbanek et al., 2019; Wolf et al., 2019; Vig and Ramea, 2019). However, relying on this approach is impractical in our scenario as it requires processing every existing tag for a context during inference time. As a result, this method is typically used for *re-ranking* (Nogueira et al., 2019a; Qu et al., 2021; Ren et al., 2021).

As shown in Figure 1, we adopt an augmentation method that enables the bi-encoder framework to mimic cross-encoder's representation learning. Compared to other knowledge distillation methods (Qu et al., 2021; Ren et al., 2021; Thakur et al., 2020), our approach does not require an additional cross-encoder network for training. Furthermore, as such cross-encoding is introduced as an augmentation strategy, it doesn't require additional memory or architecture modifications, while improving the test performance.

Specifically, for a context c, we randomly sample one of the tags in the original batch. We extend the batch in our training by introducing a context-tag concatenated input $[c; t]$ which has "*is relevant*" as a gold tag. Our bi-encoder must be able to classify relevance when an input includes both context and tag with the following score function:

$$s_{\text{CEAA}}(c, t) = E_C([c; t]) \cdot E_T(\text{“is relevant”})^{\top} \tag{4}$$

Since we use the augmentation method via input editing without an extra teacher cross-encoder model for distillation, we call this model Cross Encoding As Augmentation (CEAA).

3.4 Transfer Learning

To overcome the data scarcity in auto-tagging tasks, we introduce bi-encoder (DPR) models that distill knowledge from large question-answering datasets. We argue that the training objective of question answering is similar to the context and tag matching in the auto-tagging task, as a question is a short text that identifies the core of a given context. Therefore, while the previous works have relied on vanilla BERT, here we explore whether pertaining on question-answering tasks would improve the performance in the auto-tagging tasks. Specifically, we replace the naive BERT encoders with DPR (Karpukhin et al., 2020), which is further optimized with the Natural Question dataset (Lee et al., 2019; Kwiatkowski et al., 2019) to solve open-domain question-answering tasks of matching the representations of document and question. To match the overall length of the texts, we use *"dpr-ctx_encoder-single-nq-base"* and *"dpr-question_encoder-single-nq-base"* for context and tag encoders respectively.

4 Experiments

4.1 Experimental Setup

We conduct experiments on the following datasets: ARC (Xu et al., 2019), QC-Science (Mohania et al., 2021), and EURLEX57K (Chalkidis et al., 2019). Details of datasets, metrics, and training details are in Appendix.

For comparison, in addition to simple baselines, we employ some state-of-the-art methods including BERT (prototype) (Snell et al., 2017), TagRec (Mohania et al., 2021), TagRec++ (Viswanathan et al., 2022), and Poly-encoder (Humeau et al., 2019). For ablations, built on the bi-encoder (BERT) method, we present three variants: Bi-encoder (BERT) + CEAA, Bi-encoder (DPR), and Bi-encoder (DPR) + CEAA, where the comparisons between the variants could highlight the contribution of transfer learning and CEAA.

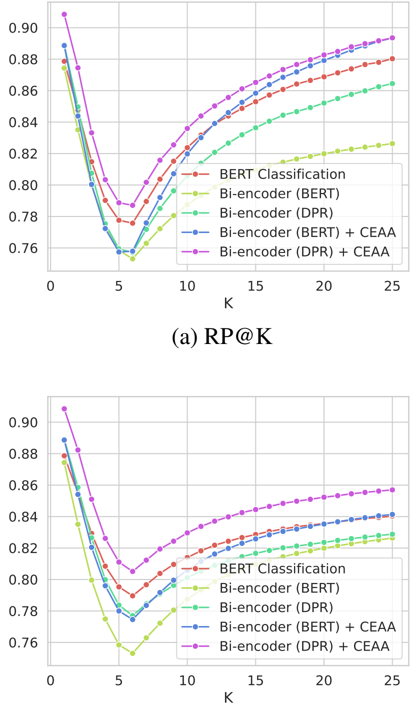
4.2 Results and Analysis

Overall Accuracy: The main objective of this work is to improve the bi-encoder models for the purpose of better text classification in two aspects:

Methods	ARC			QC-Science			EURLEX57K	
	R@1	R@3	R@5	R@1	R@3	R@5	RP@5	nDCG@5
BM25	0.14	0.28	0.34	0.13	0.23	0.27	0.15	0.15
BERT (prototype)	0.35	0.54	0.64	0.54	0.75	0.83	-	-
TagRec	0.36	0.55	0.65	0.54	0.78	0.86	-	-
TagRec++	0.49	0.71	0.78	0.65	0.85	0.90	-	-
BERT (classification)	0.53	0.72	0.79	0.68	**0.87**	**0.91**	0.78	0.80
Poly-encoder-16	0.40	0.65	0.75	0.50	0.75	0.83	0.22	0.23
Poly-encoder-360	0.44	0.68	0.78	0.64	0.85	0.90	0.54	0.54
Bi-encoder (BERT)	0.51	0.71	0.77	0.67	0.85	0.90	0.74	0.76
Bi-encoder (BERT) + CEAA	0.50	0.72	**0.80**	0.68	0.86	0.90	0.76	0.78
Bi-encoder (DPR)	0.54	0.73	**0.80**	0.69	**0.87**	0.90	0.76	0.78
Bi-encoder (DPR) + CEAA	**0.56**	**0.74**	**0.80**	**0.70**	0.86	0.90	**0.79**	**0.81**

Table 1: Results of experiments on ARC, QC-Science, and EURLEX57K dataset. We mainly compared Bi-encoder with Bi-encoder + CEAA where each encoder is pretrained with different training objectives, BERT and DPR.

transfer learning and CEAA. Regarding the effect of using two different pretrained models, the results from Table 1 show that models trained on DPR achieve higher performance than models from BERT. Specifically, Bi-encoder (DPR) outperforms the Bi-encoder (BERT) for ARC (0.54 > 0.51 in R@1) and QC-Science (0.69 > 0.67 in R@1). The performance of the EURLEX57K datasets in both RP@5 and nDCG@5 increases by 0.02. Applying our augmentation method to the Bi-encoder (both vanilla BERT and QA-finetuned BERT) improves the performance by 0.06, 0.02, and 0.03 points in ARC, QC-Science, and EURLEX57k, respectively. Additionally, the Bi-encoder (DPR) + CEAA demonstrates the highest overall performance in most cases (except for R@3 and R@5 of the QC-Science dataset where differences were small). For example, compared to TagRec++, which is the current state-of-the-art model on the datasets, we observed that our best model improves on TagRec++ by 0.05 points in R@1[2]. Figure 2 further demonstrates the change in RP@K and nDCG@K across a varying range of values for K on EURLEX57K, where CEAA shows consistently better performance. Notably, the gap from Bi-encoder (BERT) increases as K increases for both metrics.



(a) RP@K

(b) nDCG@K

Figure 2: Comparison of models on EURLEX57K with two different metrics.

Multi-label Generalization: To further highlight differences between single-label and multi-label settings, the two best models, Bi-encoder (DPR) and Bi-encoder (DPR) + CEAA, were trained with a modified single-labeled EURLEX57K dataset, where we sampled only a single tag from the multi-label space. When the models are evaluated on the original multi-label dataset, as a context in the EURLEX57K dataset has ≥ 5 gold tags on average, it is important to achieve high nDCG@K performance on $K \geq 5$. The results are presented in Figure 3. We observe that the models show comparable performance with values of 0.65, 0.70, and 0.73 for Bi-encoder (DPR), Bi-encoder (DPR) + CEAA and BERT classification, respectively at $K = 1$. Though the classification model performs slightly better than CEAA at low K values, performance significantly degrades for $K \geq 5$. Overall, the cross-encoder augmentation helped the model to better find related tags at the top rank. From

[2] We discuss Poly-encoder's low performance in Appendix B.1.

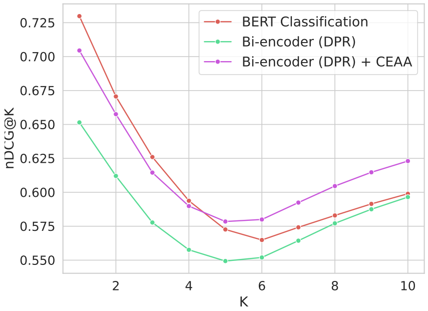
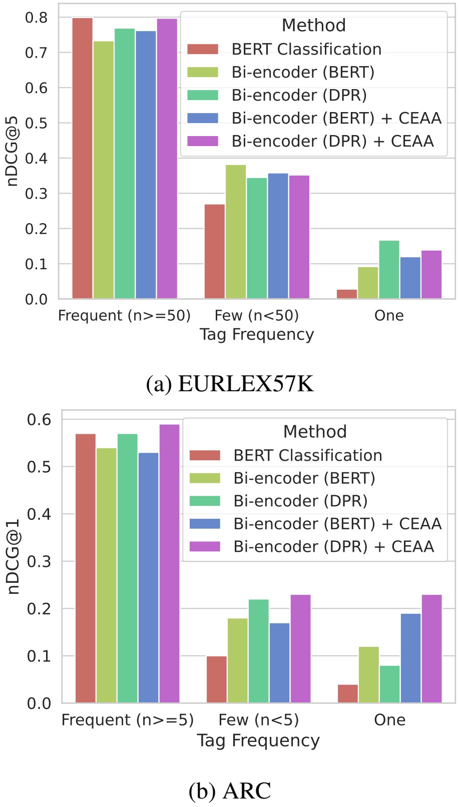


Figure 3: Multi-label evaluation. All models are trained on the single-label version of EURLEX57K but evaluated as multi-label.

these results, we argue that evaluating against the single-labeled dataset may not be an appropriate testing tool for comparing the auto-tagging models, as BERT classification was considered the best at first, even though it is poorly working on multi-label scenarios. This problem is critical as multi-label issues are prevalent in education.

Specifically, we manually checked failure cases of both Bi-encoder (DPR) and Bi-encoder (DPR) + CEAA at top 1, to qualitatively examine which one is better at ranking the relevant tags. The results in Appendix B.2 show that Bi-encoder (DPR) + CEAA is able to retrieve better candidates than the Bi-encoder (DPR) more often. An interesting example is, given the context [“The sector in which employees have more job security is an organized sector”], where the gold tag is one related to the economy, the Bi-encoder (DPR) + CEAA returns a tag [“human resources”], which is sufficiently relevant but not labeled one. From these results, we once again confirm that the multi-label problem is severe in the auto-tagging tasks and that our model yields sufficiently significant results beyond the reported performance.

Data Efficiency: To identify the effectiveness of augmentation with low-resource labels, we measured nDCG@5 on the splits of labels based on their occurrence in training data. EURLEX57 considered the labels that occurred more than 50 times in the training set as frequent and few otherwise. We set the ARC dataset’s threshold to 5. Figure 4 shows that both CEAA and transfer learning contribute to better performance for the frequent labels. Further, we observe that the retrieval methods are more effective for the rarely occurring tags than standard classification methods. Notably, in ARC of a smaller dataset than EURLEX57K (5K < 45K), the combination of CEAA and transfer learning, CEAA (DPR), achieves the best performance.



(a) EURLEX57K

(b) ARC

Figure 4: Analysis on data efficiency. We report nDCG on a varying number of training labels on EURLEX57K and ARC.

5 Conclusion

In this paper, we discuss the problem of ‘*auto-tagging*’ with regard to data scarcity due to its large label space - an issue that is critical in the education domain, but also for other domains with a multi-label structure such as jurisdictional or clinical contexts. We propose two innovations to address this problem: First, exploiting the knowledge of language models trained on large question-answering datasets. Second, applying a novel augmentation for bi-encoder architecture inspired by cross-encoders to better capture the full interaction between inputs and labels while maintaining the bi-encoder’s efficiency. A set of experiments demonstrated the effectiveness of our approach, especially in the multi-label setting. Future research will explore re-ranking scenarios in which the bi-encoder trained with our cross-encoding augmentation (CEAA) is re-used to effectively re-rank the tags with cross-encoding mechanism as in (Nogueira and Cho, 2019).

6 Limitations

6.1 Limited Size of Language Models

Due to the recent successes of generative large language models as zero-shot (or, few-shot) text classifiers (Radford et al., 2019; Brown et al., 2020), one may ask about the practicality of our methods. Even when disregarding computational efficiency[3], we argue that applying such large language models for XMC problems is not trivial, as it is challenging to constrain the label space appropriately. For example, even when the tag candidates we wanted for a task were `entailment`, `neutral`, and `contradiction`), the generative model will output tags outside this range such as `hamburger` (Raffel et al., 2020). In-context learning (Min et al., 2022) may alleviate this concern, but in the context of the large label spaces of our application, the token limits of standard language models will be exceeded.

6.2 Lack of Knowledge-level Auto-tagging

Though we pursue text classification tasks in the education domain, the classes usually represent only superficial information, such as chapter titles, which neglects the deeper relationships between educational contents like precondition between knowledge. For example, to solve a quadratic problem mathematical problem, an ability to solve the first-order problem is required. However, the available texts have only the last superficial tags. These concerns were not considered when creating these public datasets. Instructor-driven labeling would be an effective and practical solution for knowledge-level auto-tagging.

6.3 Inefficiency of Tag Encoder

One may argue that the performance of one BERT system is good enough to cast doubt on using two BERTs for the bi-encoder. In this context, experiments showed additional efficiency of our approach for low-frequency tags. Nonetheless, the current tag encoder could be made much more efficient using a smaller number of layers in BERT which will be explored in the future.

7 Ethical Considerations

Incorrect or hidden decision processes of the AI tagging model could result in the wrong learning path. The system would therefore need to be subject to human monitoring for occasional supervision. At the same time, the potential benefits of properly-tagged content will be large for both the learner's learning experience and the teacher's labeling cost as the model can narrow down full tag space to the top-K candidates.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Ilias Chalkidis, Manos Fergadiotis, Sotiris Kotitsas, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. An empirical study on large-scale multi-label text classification including few and zero-shot labels. *arXiv preprint arXiv:2010.01653*.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, and Ion Androutsopoulos. 2019. Large-scale multi-label text classification on eu legislation. *arXiv preprint arXiv:1906.02192*.

Wei-Cheng Chang, Hsiang-Fu Yu, Kai Zhong, Yiming Yang, and Inderjit Dhillon. 2019. X-bert: extreme multi-label text classification with bert. *arXiv preprint arXiv:1905.02331*.

Seungtaek Choi, Myeongho Jeong, Hojae Han, and Seung-won Hwang. 2022. C2l: Causally contrastive learning for robust text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10526–10534.

Ofer Dekel and Ohad Shamir. 2010. Multiclass-multilabel classification with more classes than examples. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 137–144. JMLR Workshop and Conference Proceedings.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Vasu Goel, Dhruv Sahnan, V Venktesh, Gaurav Sharma, Deep Dwivedi, and Mukesh Mohania. 2022. K-12bert: Bert for k-12 education. In *International Conference on Artificial Intelligence in Education*, pages 595–598. Springer.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2019. Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring. *arXiv preprint arXiv:1905.01969*.

[3] Nevertheless, we believe that actionable language models should keep efficiency as one of their core criteria.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Kenneth R Koedinger, Albert T Corbett, and Charles Perfetti. 2012. The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive science*, 36(5):757–798.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. *arXiv preprint arXiv:1906.00300*.

Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: An easy-to-use python toolkit to support replicable ir research with sparse and dense representations. *arXiv preprint arXiv:2102.10073*.

Jueqing Lu, Lan Du, Ming Liu, and Joanna Dipnall. 2020. Multi-label few/zero-shot learning with knowledge aggregated from multiple label graphs. *arXiv preprint arXiv:2010.07459*.

Christopher D Manning. 2008. *Introduction to information retrieval.* Syngress Publishing,.

Tharun Kumar Reddy Medini, Qixuan Huang, Yiqiu Wang, Vijai Mohan, and Anshumali Shrivastava. 2019. Extreme classification in log memory using count-min sketch: A case study of amazon search with 50m products. *Advances in Neural Information Processing Systems*, 32.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*.

Mukesh Mohania, Vikram Goyal, et al. 2021. Tagrec: Automated tagging of questions with hierarchical learning taxonomy. *arXiv preprint arXiv:2107.10649*.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with bert. *arXiv preprint arXiv:1901.04085*.

Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019a. Multi-stage document ranking with bert. *arXiv preprint arXiv:1910.14424*.

Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019b. Document expansion by query prediction. *arXiv preprint arXiv:1904.08375*.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, Qiaoqiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021. Rocketqav2: A joint training method for dense passage retrieval and passage re-ranking. *arXiv preprint arXiv:2110.07367*.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30.

Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2020. Augmented sbert: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks. *arXiv preprint arXiv:2010.08240*.

Jack Urbanek, Angela Fan, Siddharth Karamcheti, Saachi Jain, Samuel Humeau, Emily Dinan, Tim Rocktäschel, Douwe Kiela, Arthur Szlam, and Jason Weston. 2019. Learning to speak and act in a fantasy text adventure game. *arXiv preprint arXiv:1903.03094*.

Jesse Vig and Kalai Ramea. 2019. Comparison of transfer-learning approaches for response selection in multi-turn conversations. In *Workshop on DSTC7*.

Venktesh Viswanathan, Mukesh Mohania, and Vikram Goyal. 2022. Tagrec++: Hierarchical label aware attention network for question categorization. *arXiv preprint arXiv:2208.05152*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149*.

Dongfang Xu, Peter Jansen, Jaycie Martin, Zhengnan Xie, Vikas Yadav, Harish Tayyar Madabushi, Oyvind Tafjord, and Peter Clark. 2019. Multi-class hierarchical question classification for multiple choice science exams. *arXiv preprint arXiv:1908.05441*.

Ruohong Zhang, Yau-Shian Wang, Yiming Yang, Donghan Yu, Tom Vu, and Likun Lei. 2022. Long-tailed extreme multi-label text classification with generated pseudo label descriptions. *arXiv preprint arXiv:2204.00958*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

A Experimental Setup

A.1 Data Statistics

ARC (Xu et al., 2019): This dataset consists of 7,775 multiple-choice questions and answer pairs from the science domain. Each data is paired with classification taxonomy. The taxonomy is constructed to categorize questions into coarse to fine chapters in a science exam. There are a total of 420 unique labels. The dataset is split in train, validation, and test by 5,597, 778, and 1400 samples.

QC-Science (Mohania et al., 2021): this larger dataset consists of 47,832 question-answer pairs also in the science domain with 312 unique tags. Each tags are hierarchical labels in the form of subject, chapter, and topic. The train, validation, and test sets consist of 40,895, 2,153, and 4,784 samples.

EURLEX57K (Chalkidis et al., 2019): The dataset contains 57,000 English legislative documents from EUR-LEX with a split of 45,000, 6,000, and 6,000. Every document is tagged with multi-label concepts from European Vocabulary. The average number of tags per document is 5, totaling 4,271 tags. Additionally, the dataset divides the tags into frequent (746), few (3,362), and zero (163), based on whether they appeared more than 50, fewer than 50, but at least once, or never, respectively.

A.2 Details on Evaluation Metric

In this section, we explain the metric used in the paper. First, recall@K($R@K$) is calculated as follows:

$$R@K = \frac{1}{N}\sum_{n=1}^{N}\frac{S_t(K)}{R_n} \tag{5}$$

where N is the number of samples to test, R_n is the number of true tags for a sample n, and $S_t(K)$ is the number of true tags within the top-K results. For evaluation on multi-label dataset we used R-Precision@K ($RP@K$) (Chalkidis et al., 2019):

$$RP@K = \frac{1}{N}\sum_{n=1}^{N}\frac{S_t(K)}{min(R_n, K)} \tag{6}$$

RP@K divides the number of true positives within K by the minimum value between K and R_n, resulting in a more fair and informative comparison in a multi-label setting.

nDCG@K (Manning, 2008) is another metric commonly used in such tasks. The difference between RP@K and nDCG@K is the latter includes the ranking quality by accounting for the location of the relevant tags within the top-K retrieved tags as follows:

$$nDCG@K = \frac{1}{N}\sum_{n=1}^{N} Z_{K_n}\sum_{k=1}^{K}\frac{Rel(n,k)}{log_2(1+k)} \tag{7}$$

where $Rel(n,k)$ is the relevance score given by the dataset between a retrieved tag k of a sample n. The value can be different if the tags' relevant score is uniquely given by the dataset. Without extra information, it is always one if relevant and zero otherwise. Z_{K_n} is a normalizing constant that is output of DCG@K when the optimal top-K were retrieved as true tags.

A.3 Hyperparmeter Setting

The architecture we used can handle a maximum of 512 tokens. Therefore, to concatenate tag tokens with context tokens, we set the maximum context token to 490 and truncate if the context is longer. The remaining space is used for tag token concatenation. For every dataset, we used 20 contexts inside a batch. The number of unique tags inside a batch can vary with multi-label settings. During cross-encoder augmentation, we sampled five negative tags for each context to be joined together and one positive tag. We used Adam optimizer with a learning rate of 1e-5. For inference, we used the Pyserini framework to index the entire tag set embeddings (Lin et al., 2021).

B Additional Results and Comments

B.1 Comments on Poly-encoder

In this section, we discuss the low performance of Poly-encoder (Humeau et al., 2019) in our main results. To be more specific, poly-encoder-16 and 360 were found to be performing below TagRec++. The value 16, and 360 is the number of vectors to represent a context. We think the low performance could be due to a potential implementation issue of the poly-encoder into the classification task. The performance could differ if we had used 16 or 360 vectors to represent the tag rather than a context. For our future work, we also aim to investigate this change.

B.2 Extra Qualitative Result

Table 2 shows the samples we used to find the potential of CEAA method in multi-label tasks. The shown results were randomly picked.

Context	A good conductor of heat is a steel ruler.
Ground Truth	**science » heat**
Bi-Encoder	science » fun with magnets
Bi-Encoder + CEAA	science » sorting materials into groups
Context	The operating system which allows two or more users to run programs at the same time is multi-user.
Ground Truth	computer science[c++] » computer overview
Bi-Encoder	computer science » introduction to computer
Bi-Encoder + CEAA	**computer science[c++]»working with operating system**
Context	The radiation which will deflect in electric field is cathode rays
Ground Truth	**physics » physics : part - ii » dual nature of radiation and matter**
Bi-Encoder	physics » physics : part - ii»atoms
Bi-Encoder + CEAA	**physics » physics: part - I » electric charges and fields**
Context	What do we call the resources that helps in production process? Factors of Production
Ground Truth	social science » economics » the story of village palampur
Bi-Encoder	social science » geography : resource and development»resources
Bi-Encoder + CEAA	**social science » economics » people as resource**
Context	The Civil Law to protect women against domestic violence was passed in 2006.
Ground Truth	**social science»civics : social and political life»judiciary**
Bi-Encoder	social science » civics : social and political life » understanding laws
Bi-Encoder + CEAA	**social science » civics : social and political life - ii » women change the world**
Context	In the mid 18 th century, major portion of eastern India was under the control of the British.
Ground Truth	**social science » eighteenth-century poltical formations » the later mughals and the emergence of new states**
Bi-Encoder	**social science » history : our pasts - ii » eighteenth-century political formations**
Bi-Encoder + CEAA	social science » history : india and the contemporary world - i » peasant and farmers
Context	Spirogyra is called so because chloroplasts are spiral.
Ground Truth	science
Bi-Encoder	**science»cell structure and functions**
Bi-Encoder + CEAA	science » life processes
Context	The element having electronic configuration 2,8,4 is silicon.
Ground Truth	**science » periodic classification of elements**
Bi-Encoder	chemistry » chemistry : part I » the solid state
Bi-Encoder + CEAA	science » structure of the atom
Context	The clouds are actually tiny droplets of water.
Ground Truth	**science » water**
Bi-Encoder	**science » air around us**
Bi-Encoder + CEAA	**social science » geography : our environment»air**

Table 2: Extra result of sampled QC-Science to show the strength of CEAA method in multi-label tasks.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Yes, we discuss the limitation in Sec 6, about the limited size of the model, knowledge level auto tagging and inefficient of the tag encoder

☑ A2. Did you discuss any potential risks of your work?
Yes, we discuss the ethical consideration in Sec 7. We talk about potential impact in the education domain, wrong or efficient learning paths for learners, and helping instructors' labeling process.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Yes, we include the abstract and section 1 as an introduction to summarize the main claim.

☑ A4. Have you used AI writing assistants when working on this paper?
To be honest, we initially used AI writing assistant as a "suggestion" for a better way of organizing statements in the abstract. However, soon we found the output of AI writing assistants ("ChatGPT") either includes wrong information or had feeling we, ourselves can already determine which part is from the assistant because it felt like a fixed template style. Therefore, after that, we neglected using the assistant but used "Grammarly" as checking simple grammatical errors throughout the writing.

B ☑ Did you use or create scientific artifacts?

Yes, we used Huggingface discussed in Section 3. We also used data QC-Science, ARC, and EURLEX57K as well as results from TagRec in Section 4m but not sure whether these facts are considered as scientific artificats.

☑ B1. Did you cite the creators of artifacts you used?
We added data, result citations in Section 2, 3, and 4, which links to references

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

□ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

□ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Not applicable. Left blank.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

C ☒ Did you run computational experiments?

Left blank.

□ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Not applicable. Left blank.

□ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Not applicable. Left blank.

□ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Not applicable. Left blank.

□ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Not applicable. Left blank.

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Yes and No, in section 4.2, we discuss one of the qualitative results we obtained after the model training. However, we only used this as a hint for us to investigate the effectiveness of the model in a multi-label setting. We are not sure, but we think this question is more focused on using a human annotator's result for quantitave performance statement.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.