CROSS-JEM: Accurate and Efficient Cross-encoders for Short-text Ranking Tasks

Anonymous ACL submission

Abstract

Ranking a set of items based on their relevance to a given query is a core problem in search and recommendation. Transformer-based ranking models are the state-of-the-art approaches for such tasks, but they score each query-item independently, ignoring the joint context of other relevant items. This leads to sub-optimal ranking accuracy and high computational costs. We propose Cross-encoders with Joint Efficient Modeling (CROSS-JEM), a novel ranking approach that enables transformer-based models to jointly score multiple items for a query, maximizing parameter utilization. CROSS-JEM leverages (a) redundancies and token overlaps to jointly score multiple items (short-text phrases in search and recommendations), and (b) a novel training objective that models ranking probabilities. CROSS-JEM achieves state-of-the-art accuracy on publicly available ranking benchmarks with over 4x-lower ranking latency compared to the baselines.

1 Introduction

We consider the problem of ranking that arises in search and recommendation pipelines, wherein the goal is to rank a set of items based on their relevance to a given query. Our work is in the context of two-stage *retrieve-then-rank* pipelines in modern recommendation systems consisting of retrieval and ranking stages (Liu et al., 2017; Zhao et al., 2024; Lin et al., 2021; Fan et al., 2022). In this work, we focus on the ranking of short-text items, given a black-box retriever, which appear in a myriad of recommendation systems applications. In designing the ranking model, two key axes are the model architecture, and the choice of the loss function. The key performance metrics for such systems are accuracy and inference latency. Existing state-of-the-art ranking approaches use *encoders* with attention layers to encode query-item pairs and *classifiers* to score them (Nogueira and Cho, 2020; Nogueira et al., 2019; Zhou et al., 2023). Recent works have proposed using sequence-to-sequence models with encoder-decoder or decoder-only architectures (Nogueira et al., 2020; Zhuang et al., 2023; Zhang et al., 2024). However, all of these models are *pointwise*, scoring query-item pairs in isolation, ignoring the list context, and producing scores that may neither reflect the optimal order nor be calibrated for sorting (Qin et al., 2024a). Pointwise transformer models are also computationally expensive and impractical for real-time ranking.

Another line of research is along listwise loss functions (Gao et al., 2021; Zhuang et al., 2023; Cao et al., 2007), and aims to improve ranking accuracy by optimizing training objective for the whole list of items, not just query-item pairs. Yet, architecturally, they score items independently, ignoring inter-item dependencies and query context. Some recent works use pre-trained LLMs for listwise ranking (Sun et al., 2023; Pradeep et al., 2023; Qin et al., 2024b; Zhang et al., 2024). However, these models have a huge parameter count (running into a few billions), limiting their scalability and efficiency. **In this work, we bridge this gap by proposing a ranking model that works at the list level, explicitly models inter-item interactions, and achieves superior latency-accuracy tradeoff, making it deployable in real-time scenarios.**

2 Method

CROSS-JEM learns to rank items for a query by exploiting two insights: a) listwise modeling captures item-item interactions better than pointwise methods; b) items in the candidate set have a high token overlap. Hence, given query $\boldsymbol{q}$ and item set $\mathbb{K}_{\boldsymbol{q}}$, the core idea is to form the union set of tokens $\mathbb{T}_{U_q}$ of all items in $\mathbb{K}_{\boldsymbol{q}}$. CROSS-JEM uses a transformer based encoder to map a sequence of tokens to a sequence of token level contextual embeddings. Existing state-of-the-art approaches model

Table 1: All baselines and our method, CROSS-JEM are fine-tuned on the corresponding datasets, except for the large pre-trained models (indicated with asterisk (*)), which are used as is without any further fine-tuning.

Method		Parameters	SODQ		MS MARCO-Titles	
			MAP@5	MAP@10	MRR@5	MRR@10
Sparse Models	BM25	–	32.80	39.26	23.71	24.57
Early-interaction	monoBERT (Nogueira and Cho, 2020)	66M	46.79	48.04	30.89	32.47
Late-interaction	ColBERT (Khattab and Zaharia, 2020)	109M	36.10	37.68	30.25	32.00
Dual Encoders	DPR (Karpukhin et al., 2020)	66M	47.32	48.48	28.78	30.87
	ANCE (Xiong et al., 2021)	66M	48.31	49.41	28.48	30.53
	INSTRUCTOR* (Su et al., 2023)	335M	49.47	50.81	28.84	30.55
Seq2Seq	RankT5-6L	74M	49.50	50.75	30.73	32.52
	RankT5-base* (Zhuang et al., 2023)	223M	45.66	49.47	27.87	29.75
Ranking LLMs	RankingGPT-7B* (Zhang et al., 2024)	7B	47.64	50.62	28.66	30.47
Ours	**CROSS-JEM-6L**	**66M**	**52.40**	**53.05**	**33.82**	**35.45**

ranking in a pointwise approach. Thus, contextual embeddings are obtained for each query-item pair individually, leading to N encoder passes for the N items in $\mathbb{K}_{\boldsymbol{q}}$ and a high computational cost. CROSS-JEM embeds all *unique* tokens in item set $\mathbb{K}_{\boldsymbol{q}}$ in single encoder pass by inferring over the combined set of query tokens $\mathbb{T}_{\boldsymbol{q}}$ and the union of all item tokens, $\mathbb{T}_{U_{\boldsymbol{q}}}$. Since the number of tokens in the item union set is significantly smaller than the sum of the number of tokens in $\mathbb{K}_{\boldsymbol{q}}$, CROSS-JEM enables highly efficient computation of contextual embeddings. Next, a pooled representation for each pair $(\boldsymbol{q}, \boldsymbol{k}_j)$ is computed as the mean of the contextual embeddings of tokens in $\mathbb{T}_{\boldsymbol{q}}$ combined with the intersection of $\mathbb{T}_{U_{\boldsymbol{q}}}$ and $\mathbb{T}_{\boldsymbol{k}_j}$. A linear classifier $\boldsymbol{w} \in \mathbb{R}^d$ computes the relevance score associated with each pair $(\boldsymbol{q}, \boldsymbol{k}_j)$. The pooled representations for all $\boldsymbol{k}_j \in \mathbb{K}_{\boldsymbol{q}}$ are batched together ($\boldsymbol{e}^{\boldsymbol{qk}} \in \mathbb{R}^{N\times d}$) allowing for the computation of all logits $[\boldsymbol{f}_{\boldsymbol{q}}]_j = \langle \boldsymbol{w}, \boldsymbol{e}^{\boldsymbol{qk}_j} \rangle$ in a single shot.

CROSS-JEM is trained with a novel listwise objective proposed in this work, called **Ranking Probability Loss (RPL)**, that models the joint ranking probabilities of items rather than their pointwise relevance. Different from existing listwise losses such as ListNet (Cao et al., 2007), RPL factors in the availability of all logits $[\boldsymbol{f}_{\boldsymbol{q}_i}]$. Given the scores $\boldsymbol{f}_{\boldsymbol{q}_i}$ and the ground-truth $\boldsymbol{y}_i$ for a query $\boldsymbol{q}_i$, and an item $\boldsymbol{k}_j$, RPL penalizes ranking $\boldsymbol{k}_j$ above any item $\boldsymbol{k}_k$ with higher ground-truth score. Formally, RPL ($\mathcal{L}^{RPL}$) is:

$$\sum_{i=1}^{|\mathbb{Q}_{tr}|} \sum_{j=1}^{N} \Big(\sum_{k\in\mathbb{L}_j} [\boldsymbol{y}_i]_k \Big) \log\Big(\mathrm{SM}\Big(\sum_{k\in\mathbb{L}_j} [\boldsymbol{f}_{\boldsymbol{q}_i}]_k \Big)\Big), \quad (1)$$

where SM denotes the SoftMax operator and $\mathbb{L}_j$ is defined as $\mathbb{L}_j = \{k \in \{1, N\} : [\boldsymbol{y}_i]_k < [\boldsymbol{y}_i]_j\}$.

3 Experiments

Experimental Setup: We use **Stack Overflow Duplicate Questions** (Liu et al., 2018) **(SODQ)** and a short-text version of **MS MARCO** (Dai and Callan, 2020), where we only keep webpage titles (**MS MARCO-Titles**) to align it with short-text ranking applications. We use the mean average precision (MAP) and mean reciprocal rank (MRR) for evaluation. MAP is a generalization of MRR when there are multiple positive items per query, *i.e.*, on MS MARCO, $\mathrm{MAP@}K = \mathrm{MRR@}K$, $\forall K$.

Results: Table 1 shows that CROSS-JEM outperforms cross-encoders and dual encoders by up to 3%, and sparse models (such as BM25) by 20% in terms of accuracy, demonstrating the effectiveness of its listwise ranking. We also report that CROSS-JEM, which uses a 6-layer BERT as the base encoder, has the same number of parameters as monoBERT, but can support over 4× lower latency than monoBERT (9.8 ms vs 41.3 ms) for scoring 700 items per query on A100 GPUs.

4 Conclusions and Future Scope

CROSS-JEM is the first joint ranking approach that can effectively model listwise ranking in both the model architecture and training objective with real-time latency constraints. Overcoming the limitations of pointwise approaches, it establishes a new state-of-the-art with significantly lower computational costs on publicly available ranking benchmarks. The scope of this work is on ranking short texts, a common requirement in both industrial sponsored search applications and academic benchmarks. CROSS-JEM opens up new directions for designing accurate ranking architectures and algorithms, accounting for task-specific constraints.

References

Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, ICML '07, pages 129–136, New York, NY, USA. ACM.

Zhuyun Dai and Jamie Callan. 2020. Context-aware document term weighting for ad-hoc search. In *Proceedings of The Web Conference 2020*, pages 1897–1907.

Y. Fan, X. Xie, Y. Cai, J. Chen, X. Ma, X. Li, R. Zhang, J. Guo, et al. 2022. Pre-training methods in information retrieval. *Foundations and Trends® in Information Retrieval*, 16(3):178–317.

L. Gao, Z. Dai, and J. Callan. 2021. Rethink training of BERT rerankers in multi-stage retrieval pipeline. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021*, page 280–286, Berlin, Heidelberg. Springer-Verlag.

V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-T. Yih. 2020. Dense passage retrieval for open-domain question answering. In *EMNLP*.

O. Khattab and M. Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48.

J. Lin, R. Nogueira, and Yates A. 2021. Pretrained Transformers for Text Ranking: BERT and Beyond. In *NAACL*.

S. Liu, F. Xiao, W. Ou, and L. Si. 2017. Cascade ranking for operational e-commerce search. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1557–1565.

X. Liu, C. Wang, Y. Leng, and C. X. Zhai. 2018. LinkSO: a dataset for learning to retrieve similar question answer pairs on software development forums. In *4th ACM SIGSOFT International Workshop on NLP for Software Engineering*, pages 2–5.

R. Nogueira and K. Cho. 2020. Passage re-ranking with BERT. *Preprint*, arXiv:1901.04085.

R. Nogueira, Z. Jiang, R. Pradeep, and J. Lin. 2020. Document ranking with a pretrained sequence-to-sequence model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 708–718, Online.

R. Nogueira, W. Yang, K. Cho, and J. Lin. 2019. Multi-stage document ranking with BERT. *arXiv preprint arXiv:1910.14424*.

R. Pradeep, S. Sharifymoghaddam, and J. Lin. 2023. RankZephyr: Effective and robust zero-shot listwise reranking is a breeze! *arXiv preprint arXiv:2312.02724*.

Z. Qin, R. Jagerman, K. Hui, H. Zhuang, J. Wu, L. Yan, J. Shen, T. Liu, J. Liu, D. Metzler, X. Wang, and M. Bendersky. 2024a. Large language models are effective text rankers with pairwise ranking prompting. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1504–1518, Mexico City, Mexico.

Z. Qin, R. Jagerman, K. Hui, H. Zhuang, J. Wu, L. Yan, J. Shen, T. Liu, J. Liu, D. Metzler, et al. 2024b. Large language models are effective text rankers with pairwise ranking prompting. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1504–1518.

H. Su, W. Shi, J. Kasai, Y. Wang, Y. Hu, M. Ostendorf, W.-T. Yih, N. A. Smith, L. Zettlemoyer, and T. Yu. 2023. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121, Toronto, Canada.

W. Sun, L. Yan, X. Ma, S. Wang, P. Ren, Z. Chen, D. Yin, and Z. Ren. 2023. Is ChatGPT good at search? investigating large language models as re-ranking agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14918–14937.

L. Xiong, C. Xiong, Y. Li, K.-F. Tang, J. Liu, P. Bennett, J. Ahmed, and A. Overwijk. 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *ICLR*.

L. Zhang, Y. Zhang, D. Long, P. Xie, M. Zhang, and M. Zhang. 2024. A two-stage adaptation of large language models for text ranking. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 11880–11891, Bangkok, Thailand.

W. X. Zhao, J. Liu, R. Ren, and J.-R. Wen. 2024. Dense text retrieval based on pretrained language models: A survey. *ACM Trans. Inf. Syst.*, 42(4).

Y. Zhou, T. Shen, X. Geng, C. Tao, C. Xu, G. Long, B. Jiao, and D. Jiang. 2023. Towards robust ranker for text retrieval. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5387–5401, Toronto, Canada.

H. Zhuang, Z. Qin, R. Jagerman, K. Hui, J. Ma, J. Lu, J. Ni, X. Wang, and M. Bendersky. 2023. RankT5: Fine-Tuning T5 for Text Ranking with Ranking Losses. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 2308–2313, New York, NY, USA.