GATERERANKER: LISTWISE RERANKING WITH JOINT RANKING AND ADAPTIVE SELECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reranking is a crucial component in modern multi-stage retrieval and retrieval-augmented generation systems, where it is widely used to refine the ordering of candidate documents before downstream processing. In practical pipelines, however, the reranked list is usually followed by a selection step, such as fixed top-K truncation or global score thresholding, to decide which documents should be retained. Such post-hoc selection strategies are simple and effective, but they rely on static decision rules and may be brittle across queries, domains, and candidate sets. In this work, we propose **GateReranker**, a listwise LLM reranker that unifies document ranking and adaptive selection within a single model. GateReranker introduces a dedicated gate token to learn a list-conditioned cutoff, allowing the model to retain documents whose relevance scores exceed the gate score while simultaneously producing their relative ordering. Experiments on BEIR show that GateReranker achieves the best retained-set F1 and the highest nDCG compared with strong open-source rerankers paired with fixed top-K truncation and global score thresholding. These results suggest that adaptive selection can be naturally integrated into reranking, providing an effective in-model alternative to static post-hoc filtering.

1 INTRODUCTION

Multi-stage retrieval has become a standard paradigm in modern search, question answering, and retrieval-augmented generation (RAG) systems (Guo et al., 2022; Lewis et al., 2020). In such pipelines, a first-stage retriever efficiently retrieves a set of candidate documents, and a reranker is then applied to refine their ordering with a more expressive relevance model (Zhao et al., 2024). The quality of reranking is therefore crucial: it directly affects which evidence is exposed to downstream components and can substantially influence both retrieval effectiveness and generation quality.

While rerankers are typically designed to produce better document orderings, practical systems often require more than a ranked list. Before the reranked results are passed to a reader, generator, or other downstream module, the system must decide which documents should be retained (Posokhov et al., 2025). This retained-set selection step is important in real applications. Retaining too many irrelevant documents may introduce noise, increase inference cost, and degrade downstream reasoning, whereas retaining too few documents may discard useful evidence and hurt recall (Yoran et al., 2024; Yu et al., 2024; Xu et al., 2025). Moreover, the amount of useful evidence is query dependent, making the appropriate retrieval depth variable across instances (Taguchi et al., 2025).

In existing retrieval pipelines, this selection step is usually handled by simple post-hoc rules, such as fixed top-K truncation or global score thresholding (Meng et al., 2025; Posokhov et al., 2025). These strategies are easy to implement and often effective, but they rely on static operating points that are shared across queries or datasets. As a result, a fixed top-K policy may over-select documents for queries with few relevant candidates and under-select documents for queries with many relevant candidates. Similarly, a global threshold can be sensitive to score calibration and may not transfer robustly across domains or rerankers. These observations suggest that retained-set selection should ideally be conditioned on the query and the entire candidate list, rather than determined solely by a manually specified global rule.

In this work, we study how to integrate adaptive selection into listwise reranking. Our key intuition is that, when an LLM is given a query together with a numbered list of candidate documents and is

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

asked to output the most relevant document identifier, or a special token if no document is relevant, the first-step output distribution can be used as a shared space for both ranking and selection. Ideally, if documents 1 and 2 are relevant while document 3 is not, and document 1 is more relevant than document 2, the model should assign the highest logit to the identifier of document 1, a lower but still relevant logit to document 2, a gate logit between relevant and non-relevant documents, and a lower logit to document 3. In other words, the gate token can be interpreted as a learnable boundary in the same output space as document identifiers.

Based on this intuition, we propose **GateReranker**, a listwise LLM reranker that unifies document ranking and adaptive selection within a single model. GateReranker associates each candidate document with an identifier token and introduces a dedicated gate token. At the first prediction step, the logits of document identifier tokens are used as document relevance scores, while the logit of the gate token serves as an adaptive cutoff. Documents whose scores exceed the gate score are retained and then sorted by their relevance scores, producing a ranked subset in a single forward pass.

To train this behavior, GateReranker jointly optimizes ranking and gating objectives. The ranking objectives encourage relevant documents to score above non-relevant ones, while the gating objectives explicitly train the gate score to lie between relevant and non-relevant candidates. In this way, the model learns both a structured ordering over documents and an instance-dependent decision boundary for retained-set selection.

We evaluate GateReranker on BEIR, a heterogeneous benchmark spanning diverse English retrieval tasks. Experiments show that GateReranker achieves the best retained-set F1 and the highest nDCG compared with strong open-source rerankers equipped with fixed top-K truncation and global score thresholding. These results indicate that learning an instance-dependent selection boundary within the reranker can improve the final retained set while preserving high-quality document ordering.

Our contributions are summarized as follows:

- We formulate retained-set selection as an adaptive selection problem within listwise reranking, emphasizing that practical retrieval systems require not only document ordering but also query-dependent document retention.
- We propose GateReranker, a listwise LLM reranker that introduces a gate token to learn a list-conditioned cutoff jointly with document relevance scores.
- We show on BEIR that GateReranker improves retained-set F1 and nDCG over strong post-hoc selection baselines, and that the gains persist on unseen tasks and when comparing adaptive gating against fixed top-K truncation of the same document logits.

2 Related Work

Reranking is a central component of multi-stage retrieval systems, where a more expressive model refines the output of a first-stage retriever. Early neural rerankers based on pretrained transformers, such as BERT cross-encoders (Nogueira & Cho, 2020), established strong effectiveness by directly modeling query–document interactions. Subsequent sequence-to-sequence rerankers, including monoT5 (Nogueira et al., 2020), showed that ranking can be framed as token prediction, while later models such as RankT5 (Zhuang et al., 2022) further improved ranking quality with stronger ranking-oriented objectives. Despite these architectural differences, the primary goal of these methods remains the same: producing better relative document ordering. In most practical systems, the final retained set is still determined by an external policy such as top-K truncation or score thresholding.

Listwise reranking methods model candidate documents jointly, allowing the model to reason over cross-document interactions that are unavailable to point-wise or pairwise approaches. This is particularly attractive in the LLM era, where prompting or seq2seq decoding can exploit richer context over an entire candidate set. Representative recent examples include RankGPT-style permutation generation (Sun et al., 2024), ListT5 (Yoon et al., 2024), efficient listwise reranking variants such as FIRST (Reddy et al., 2024), and self-calibrated listwise reranking methods such as SCaLR (Ren et al., 2024). These methods show that listwise reasoning can improve ranking quality and, in some cases, robustness and efficiency. However, their outputs are still primarily used as rankings or per-document relevance estimates, while the final retained set is typically decided by a separate truncation or

thresholding rule. In contrast, our method extends listwise reranking to jointly model document ordering and adaptive selection within the reranker itself.

In practical retrieval systems, the reranked list is often further processed using fixed top-K truncation, global score thresholds, or related post-hoc rules in order to control the precision–recall trade-off. A growing body of work has therefore examined score calibration and ranked-list truncation, with the goal of making retrieval scores more comparable across queries and supporting more transferable thresholding policies (Posokhov et al., 2025; Meng et al., 2024). These studies are closely related to our motivation because they recognize that a single static operating point can be brittle across settings. However, they remain fundamentally post-hoc: the reranker first produces document scores, and a separate calibration or truncation policy is then applied to those scores. Our approach differs in that it internalizes this decision boundary within the reranker itself. Rather than learning a post-hoc policy over reranker scores, GateReranker learns a list-conditioned cutoff jointly with document ranking in a single model, so that ranking and retained-set selection are produced together.

3 METHODS

As illustrated in Fig. 1, GateReranker jointly performs ranking and adaptive selection from a single output distribution.

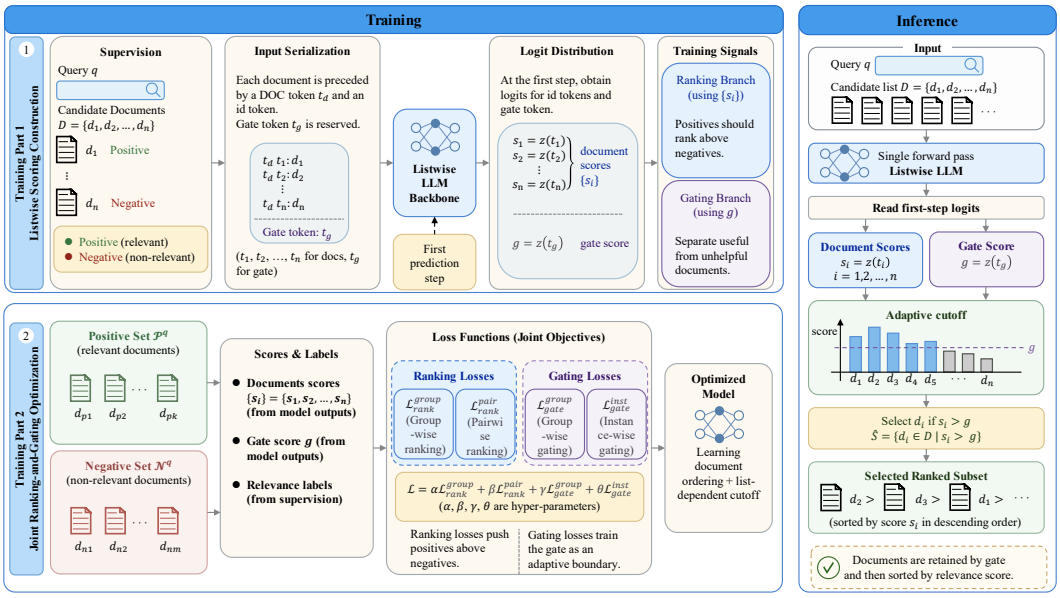


Figure 1: Overview of GateReranker. Given a query q and a list of candidate documents $\{d_1, d_2, \dots, d_n\}$, the listwise LLM produces a single logit distribution over document identifier tokens $\{t_1, t_2, \dots, t_n\}$ and a gate token t_g. The logits of identifier tokens are used as document relevance scores $\{s_1, s_2, \dots, s_n\}$ in the scoring branch, while the logit of the gate token gives the gate score g in the gating branch. Documents whose scores satisfy $s_i > g$ are retained, yielding the final ranked subset $\hat{\mathcal{S}}$.

3.1 PRELIMINARIES

We consider a standard multi-stage retrieval setting. Given a query q, a first-stage retriever returns a list of candidate documents $\mathcal{D} = \{d_1, d_2, \dots, d_n\}$, where n denotes the number of retrieved candidates. The goal of reranking is to produce a high-quality ranked subset of $\mathcal{D}$ for downstream use.

We adopt a listwise LLM reranking formulation, in which the query and the full candidate list are encoded jointly within a single inference instance. To expose document-level decisions directly in the model output space, each candidate document d_i is associated with a unique identifier token t_i (e.g.,

"1", "2", . . ., "n"). In the serialized input, each candidate is further preceded by a dedicated document marker token, denoted as t_d, which makes the correspondence between documents and identifier tokens explicit. In addition, we introduce a special gate token t_g, which is used to represent the adaptive selection boundary. Let $\mathcal{T} = \{t_1, t_2, \ldots, t_n, t_g\}$ denote the set consisting of all document identifier tokens and the gate token. Details of the prompt template are provided in Appendix D.

Given the input constructed from q and $\mathcal{D}$, the model produces a vocabulary logit distribution at the first prediction step. We use the logits corresponding to tokens in $\mathcal{T}$ as the scoring signals for reranking and selection. Specifically, the relevance score of document d_i is defined as

$$s_i = z(t_i), \quad i = 1, 2, \ldots, n, \tag{1}$$

where $z(t_i)$ denotes the logit assigned to the identifier token t_i. That is, the score of each document is read directly from the logit of its corresponding identifier token in the output distribution. For example, if document d_1 is represented by token "1", then its relevance score is given by the logit of token "1". Similarly, we define the gate score as

$$g = z(t_g), \tag{2}$$

where $z(t_g)$ is the logit of the gate token.

This formulation allows the model to produce, within a single output distribution, both document-level relevance scores and an adaptive gating signal. The selected subset is then defined by

$$\hat{\mathcal{S}} = \{d_i \in \mathcal{D} \mid s_i > g\}, \tag{3}$$

and the retained documents are ranked according to their relevance scores $\{s_i\}$. Consequently, the output of the reranker is not merely an ordered list over all candidates, but a ranked subset whose size may vary across queries and candidate lists.

Under this view, reranking is formulated as a joint ranking-and-selection problem. Unlike conventional pipelines that first compute document scores and then apply a manually specified post-hoc rule such as fixed top-K truncation or global score thresholding, our formulation represents the selection boundary explicitly in the model output space through the gate token. This perspective serves as the basis for the proposed GateReranker.

3.2 Learning Joint Ranking and Gating

Given a query q and its candidate list $\mathcal{D} = \{d_1, \ldots, d_n\}$, let $y_i \in \{0, 1\}$ denote the relevance label of candidate document d_i, where $y_i = 1$ indicates that d_i is relevant to q. We denote the corresponding sets of positive and negative candidates by $\mathcal{P}^q = \{j \mid y_j = 1\}$ and $\mathcal{N}^q = \{i \mid y_i = 0\}$. To jointly learn document ordering and an adaptive selection boundary, we optimize two types of objectives: ranking objectives, which improve the separation between positive and negative candidates, and gating objectives, which train the gate to act as a meaningful decision boundary.

We first introduce a group-level ranking objective:

$$\mathcal{L}_{\text{group}}^{\text{rank}} = \mathbb{E}_q \left[\exp(\mathbb{E}_{i \in \mathcal{N}^q}[s_i] - \mathbb{E}_{j \in \mathcal{P}^q}[s_j] + 2m) \right], \tag{4}$$

where $m > 0$ is a margin hyperparameter. This term encourages the average score of positive candidates to exceed that of negative candidates by at least $2m$, thereby improving coarse-grained separation at the query level.

To further improve fine-grained ordering, we add a pairwise ranking objective:

$$\mathcal{L}_{\text{pair}}^{\text{rank}} = \mathbb{E}_q \left[\mathbb{E}_{i \in \mathcal{N}^q,\, j \in \mathcal{P}^q} \left[\exp(s_i - s_j + 2m) \right] \right]. \tag{5}$$

Unlike $\mathcal{L}_{\text{group}}^{\text{rank}}$, which operates on average scores, this term directly penalizes positive–negative pairs whose relative ordering violates the desired margin.

We next train the gate as an adaptive decision boundary. At the group level, we define

$$\begin{aligned} \mathcal{L}_{\text{group}}^{\text{gate}} = \mathbb{E}_q[&\exp(g - \mathbb{E}_{j \in \mathcal{P}^q}[s_j] + m) \\ &+ \exp(\mathbb{E}_{i \in \mathcal{N}^q}[s_i] - g + m)]. \end{aligned} \tag{6}$$

This objective encourages the gate score to lie between the positive and negative groups, such that the average positive score remains above the gate and the average negative score remains below it, both with margin m.

To enforce the same property at the individual-document level, we further introduce

$$\begin{aligned}\mathcal{L}_{\text{inst}}^{\text{gate}} =& \mathbb{E}_q[\mathbb{E}_{j\in\mathcal{P}^q}\left[\exp(g - s_j + m)\right] \\ &+ \mathbb{E}_{i\in\mathcal{N}^q}\left[\exp(s_i - g + m)\right]].\end{aligned} \tag{7}$$

This term explicitly encourages each positive candidate to score above the gate and each negative candidate to score below it, both with margin m.

The overall training objective is defined as

$$\mathcal{L} = \alpha\mathcal{L}_{\text{group}}^{\text{rank}} + \beta\mathcal{L}_{\text{pair}}^{\text{rank}} + \gamma\mathcal{L}_{\text{group}}^{\text{gate}} + \theta\mathcal{L}_{\text{inst}}^{\text{gate}}, \tag{8}$$

where α, β, γ and θ are hyperparameters satisfying $\alpha + \beta + \gamma + \theta = 1$. The two ranking losses encourage correct document ordering at both the group and pairwise levels, while the two gating losses train the gate to serve as a list-dependent decision boundary for adaptive selection.

4 Evaluation

4.1 Experimental Setup

In our implementation, GateReranker is built on top of Llama-3.2-3B (Grattafiori et al., 2024). We instantiate the gate token as `<|reserved_special_token_0|>` and the document marker token as `<|reserved_special_token_1|>`, both taken from the model's reserved special-token space. Built on a long-context backbone, GateReranker supports reranking up to 150 candidate documents within a single 131,072-token context window. Additional training details are provided in Appendix A.

We use BEIR (Thakur et al., 2021), a heterogeneous benchmark covering diverse English retrieval tasks, for both supervised training and evaluation. For training, we use the official BEIR training splits of DBPedia, FEVER, FiQA2018, HotpotQA, NFCorpus, QuoraRetrieval, and SciFact, comprising 204,092 training queries in total. We refer to these seven datasets as the *seen* training tasks. For evaluation, we use the BEIR test sets of the evaluated tasks, comprising 33,234 test queries in total. Among them, TREC-COVID, NQ, ArguAna, Touche-2020, SCIDOCS, and Climate-FEVER have no training split used by GateReranker; we treat them as *unseen* tasks for analyzing task-level generalization. We directly use the provided query-positive-negative supervision without additional hard-negative mining or custom candidate reconstruction. In result tables, we abbreviate datasets as TC (TREC-COVID), NFC (NFCorpus), NQ (Natural Questions), HQA (HotpotQA), FQA (FiQA2018), AA (ArguAna), TCH (Touche-2020), DBP (DBPedia), SD (SCIDOCS), FVR (FEVER), CFV (Climate-FEVER), SF (SciFact), and QRA (QuoraRetrieval).

For evaluation, we use BM25 as the first-stage retriever and keep the top 150 retrieved documents as candidates for reranking. All methods are evaluated on the same candidate pools to ensure a fair comparison. GateReranker processes the 150 candidate documents in a single prediction step and predicts an adaptive gate score for each query-candidate list.

We compare against strong open-source rerankers, including BGE-Reranker-v2-m3 (Chen et al., 2024), Qwen3-Reranker (Zhang et al., 2025), and Jina-Reranker-v3 (Wang et al., 2025), using their official released checkpoints without additional task-specific finetuning. All datasets, model backbones, and baseline rerankers used in this work are publicly available, and we use them in accordance with their respective licenses, terms of use, and intended research purposes. Table 1 summarizes the parameter sizes of GateReranker and the compared rerankers. Since these baselines produce document-level relevance scores but do not directly predict an adaptive selection boundary, we evaluate them with two post-hoc selection strategies: fixed top-K truncation and global score thresholding. For Jina-Reranker-v3, which supports at most 64 documents per listwise input, we split each 150-document candidate pool into three batches and apply the model to each batch independently. We then merge the batch outputs using the returned document scores before applying fixed top-K truncation or global score thresholding. This follows a practical batching strategy under the model's input-size constraint.

Model	Parameters
BGE-Reranker-v2-m3	0.5B
Qwen3-Reranker-0.6B	0.6B
Qwen3-Reranker-4B	4B
Qwen3-Reranker-8B	8B
Jina-Reranker-v3	0.6B
GateReranker	3B

Table 1: Parameter sizes of GateReranker and the compared open-source rerankers.

For fixed top-K truncation, we sweep K over $\{1,3,5,10,20,100,150\}$ and report the best-F1 operating point on the test benchmark, together with the corresponding nDCG.

For global score thresholding, we randomly sample 1,000 examples from the training set as a threshold-selection subset. For each reranker, we sweep score thresholds on this subset and choose the threshold that maximizes F1. The selected threshold is then fixed and applied to the full evaluation set.

We report F1 to evaluate the quality of the final retained document set, and nDCG to measure ranking quality. For GateReranker, the predicted set consists of documents whose relevance scores exceed the predicted gate score. For fixed top-K baselines, the predicted set consists of the reranked top-K documents. For global-threshold baselines, the predicted set consists of documents whose reranker scores exceed the selected threshold. In all cases, nDCG is computed using the document ordering induced by the corresponding reranker scores.

4.2 Retained-Set Quality

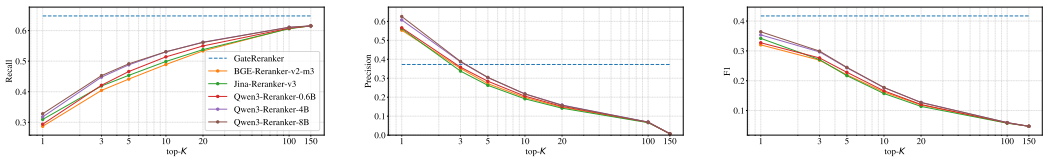
We first evaluate the quality of the final retained document set using F1. In retained-set selection, precision measures how many retained documents are truly relevant, while recall measures how many relevant documents are successfully retained. F1 balances these two aspects, penalizing both over-selection of irrelevant documents and under-selection of relevant ones. It is therefore well aligned with the document selection requirements in downstream retrieval and RAG pipelines, where the system must preserve useful evidence while avoiding unnecessary noise.

				Seen							Unseen					
Models	Avg.	Avg_S	Avg_U	NFC	HQA	FQA	DBP	FVR	SF	QRA	TC	NQ	AA	TCH	SD	CFV
BGE-Reranker-v2-m3 ($K=1$)	0.32	0.45	0.17	0.08	0.61	0.24	0.12	0.77	0.61	**0.73**	0.00	**0.44**	0.25	0.07	0.06	0.18
BGE-Reranker-v2-m3 ($\tau=3.36$)	0.23	0.33	0.11	0.05	0.66	0.12	0.15	0.48	0.26	0.62	0.07	0.30	0.03	0.22	0.02	0.03
Qwen3-Reranker-0.6B ($K=1$)	0.33	0.45	0.19	0.09	0.59	0.22	0.12	0.76	0.63	0.71	0.01	0.35	0.45	0.05	0.08	0.20
Qwen3-Reranker-0.6B ($\tau=0.99$)	0.29	0.39	0.17	0.06	0.60	0.22	0.18	0.62	0.50	0.56	0.07	0.28	0.23	0.16	0.10	0.15
Qwen3-Reranker-4B ($K=1$)	0.35	0.46	0.23	0.09	0.60	0.27	0.13	0.79	0.66	0.68	0.00	0.43	0.57	0.06	0.09	0.22
Qwen3-Reranker-4B ($\tau=0.99$)	0.28	0.37	0.18	0.03	0.65	0.28	0.14	0.62	0.51	0.36	0.02	0.36	0.24	0.13	0.11	0.24
Qwen3-Reranker-8B ($K=1$)	0.36	0.47	0.25	0.09	0.59	0.29	0.13	**0.82**	0.66	0.68	0.00	0.42	**0.65**	0.08	0.09	0.23
Qwen3-Reranker-8B ($\tau=0.92$)	0.29	0.36	0.20	0.05	0.59	0.30	0.14	0.70	0.53	0.23	0.03	0.36	0.36	0.14	0.09	0.24
Jina-Reranker-v3 ($K=1$)	0.34	0.46	0.21	0.09	0.60	0.26	0.12	0.80	0.62	0.70	0.00	**0.44**	0.51	0.04	0.08	0.19
Jina-Reranker-v3 ($\tau=0.28$)	0.32	0.43	0.20	0.06	0.64	0.27	0.16	0.63	0.56	0.68	0.04	0.36	0.42	0.13	0.07	0.20
GateReranker	**0.42**	**0.54**	**0.27**	**0.35**	**0.76**	**0.32**	**0.38**	0.65	**0.67**	0.68	**0.55**	0.18	0.21	**0.25**	**0.18**	**0.25**

Table 2: F1 results on English BEIR reranking tasks, grouped into *seen* and *unseen* datasets (Sec. 4.1). Avg_S and Avg_U are means over the seven seen and six unseen tasks, respectively; Avg. is the mean over all 13 tasks. Dataset abbreviations are defined in Sec. 4.1. For each open-source reranker, we report the fixed top-K result that achieves the best overall average F1, with the selected K shown in parentheses. We also report global score-thresholding results, where the threshold τ is selected by maximizing F1 on 1,000 randomly sampled training examples and then applied to the evaluation set. GateReranker predicts an adaptive selection boundary without fixed-K truncation or global thresholding, and achieves the best Avg., Avg_S, and Avg_U.

Table 2 reports F1 results on the BEIR reranking tasks, with columns grouped into seen and unseen datasets. GateReranker achieves the highest overall average F1 of 0.42 among all compared systems. This represents a 16.7% relative improvement over the best fixed top-K baseline, Qwen3-Reranker-8B with $K=1$ (0.36), and a 31.3% relative improvement over the best global-threshold baseline, Jina-Reranker-v3 with $\tau=0.28$ (0.32). The same ranking holds on both splits: GateReranker obtains

the best Avg_S (0.54) and Avg_U (0.27), indicating that the gains are not confined to the training tasks. These results demonstrate that the learned gate yields a stronger retained-set selection policy than both static rank-based truncation and dataset-level score thresholding.



(a) Recall@K. (b) Precision@K. (c) F1@K.

Figure 2: Precision–recall trade-off under fixed top-K truncation on BEIR. For each open-source reranker, we vary K over $\{1,3,5,10,20,100,150\}$ and report Recall@K, Precision@K, and F1@K. Increasing K generally improves recall but reduces precision, leading to a dataset-dependent trade-off for retained-set selection. GateReranker predicts an adaptive selection boundary without specifying K, achieving a stronger F1 balance than fixed top-K truncation.

Figure 2 further illustrates why a fixed top-K rule is difficult to choose. As K increases, fixed truncation generally retains more relevant documents and therefore improves recall, but it also introduces more irrelevant documents and reduces precision. As a result, the best F1 is achieved only at a particular operating point, and this operating point may vary across datasets. In contrast, GateReranker does not require a manually specified K; it predicts a query-dependent gate score and directly selects a retained set, leading to a better precision–recall balance.

At the task level, GateReranker obtains the best F1 on 9 out of 13 BEIR tasks, including TREC-COVID, NFCorpus, HotpotQA, FiQA2018, Touche-2020, DBPedia, SCIDOCS, Climate-FEVER, and SciFact. This indicates that the learned gate is effective across diverse retrieval scenarios. In contrast, fixed top-K baselines apply the same retention depth to all queries, while global thresholding applies a single score threshold across the dataset. GateReranker instead predicts a list-dependent boundary for each query-candidate list, allowing it to better adapt to variations in candidate quality and the number of relevant documents.

Complete F1 results across all evaluated fixed top-K truncation depths are provided in Appendix E.1.

4.3 Ranking Quality

We next evaluate ranking quality using nDCG. While retained-set F1 measures whether the selected document set balances precision and recall, nDCG evaluates whether relevant documents are placed near the top of the ranked output. This metric is important for verifying that adaptive selection does not compromise the core ranking ability of the reranker.

				Seen							Unseen					
Models	Avg.	Avg_S	Avg_U	NFC	HQA	FQA	DBP	FVR	SF	QRA	TC	NQ	AA	TCH	SD	CFV
BGE-Reranker-v2-m3 ($K=1$)	0.54	0.65	0.42	0.43	0.91	0.41	0.55	0.80	0.63	0.81	0.86	0.48	0.25	0.45	0.18	0.30
BGE-Reranker-v2-m3 ($\tau=3.36$)	0.37	0.46	0.26	0.17	0.83	0.19	0.35	0.57	0.28	0.85	0.66	0.44	0.03	0.34	0.04	0.05
Qwen3-Reranker-0.6B ($K=1$)	0.55	0.64	0.44	0.49	0.88	0.36	0.55	0.80	0.65	0.78	**0.91**	0.38	0.44	0.29	0.25	0.34
Qwen3-Reranker-0.6B ($\tau=0.99$)	0.47	0.56	0.37	0.21	0.79	0.33	0.40	0.82	0.55	0.79	0.85	0.47	0.26	0.25	0.18	0.22
Qwen3-Reranker-4B ($K=1$)	0.59	0.67	0.50	0.48	0.90	0.43	0.59	0.84	0.68	0.75	**0.91**	0.46	0.56	0.42	0.28	0.39
Qwen3-Reranker-4B ($\tau=0.99$)	0.43	0.51	0.34	0.11	0.82	0.40	0.42	0.86	0.53	0.41	0.49	0.47	0.26	0.28	0.20	0.35
Qwen3-Reranker-8B ($K=1$)	0.61	0.67	**0.54**	0.49	0.88	0.47	0.58	0.86	0.68	0.75	0.90	0.45	**0.65**	0.55	0.26	0.40
Qwen3-Reranker-8B ($\tau=0.92$)	0.46	0.49	0.43	0.16	0.76	0.43	0.38	0.86	0.58	0.26	0.76	0.50	0.45	0.34	0.17	0.34
Jina-Reranker-v3 ($K=1$)	0.55	0.65	0.43	0.44	0.89	0.42	0.54	0.84	0.64	0.77	0.82	0.48	0.51	0.22	0.25	0.32
Jina-Reranker-v3 ($\tau=0.28$)	0.52	0.59	0.44	0.24	0.82	0.37	0.47	0.85	0.58	0.81	0.83	**0.56**	0.58	0.22	0.18	0.29
GateReranker	**0.65**	**0.76**	**0.54**	**0.60**	**0.94**	**0.54**	**0.71**	**0.88**	**0.75**	**0.88**	0.89	0.52	0.43	**0.57**	**0.37**	**0.43**

Table 3: nDCG results on English BEIR reranking tasks under the same selection settings as Table 2, with the same seen/unseen grouping. Dataset abbreviations follow Sec. 4.1. GateReranker predicts an adaptive selection boundary without fixed-K truncation or global thresholding, and achieves the best overall Avg. and Avg_S; on Avg_U it ties the strongest fixed top-K baseline (Qwen3-Reranker-8B, $K=1$).

Table 3 reports nDCG results on the BEIR reranking tasks under the same selection settings as Table 2, using the same seen/unseen grouping. GateReranker achieves the best overall average nDCG of 0.65,

Selection	Avg. nDCG	Avg. F1
$K = 1$	0.38	0.18
$K = 3$	0.46	0.24
$K = 5$	0.49	0.23
$K = 10$	0.51	0.20
$K = 20$	0.53	0.18
$K = 100$	0.59	0.14
$K = 150$	0.62	0.12
Adaptive gate	0.54	**0.27**

Table 4: Fixed top-K truncation versus adaptive gating on the same GateReranker document logits, averaged over the six unseen BEIR tasks. The adaptive gate yields the best F1; larger K improves nDCG mainly by retaining more of the ranked list.

outperforming the best fixed top-K baseline, Qwen3-Reranker-8B with $K = 1$ (0.61), and the best global-threshold baseline, Jina-Reranker-v3 with $\tau = 0.28$ (0.52). On the seen split it also leads with Avg_S=0.76, while on the unseen split its Avg_U=0.54 matches Qwen3-Reranker-8B ($K = 1$).

At the task level, GateReranker achieves the best nDCG on 10 out of 13 BEIR tasks, including NFCorpus, HotpotQA, FiQA2018, Touche-2020, DBPedia, SCIDOCS, FEVER, Climate-FEVER, SciFact, and QuoraRetrieval. These results indicate that the proposed gate mechanism not only improves retained-set selection, but also preserves strong document ordering. In particular, the gains in both F1 and nDCG suggest that GateReranker does not simply retain more documents to increase recall; instead, it learns a score structure that supports both adaptive selection and high-quality ranking.

Complete nDCG results across all evaluated fixed top-K truncation depths are provided in Appendix E.2.

4.4 Isolating Adaptive Selection

The main comparisons above pit GateReranker against open-source rerankers whose scores and selection rules both differ. To isolate the contribution of adaptive selection at inference time, we keep the fully trained GateReranker fixed and evaluate it on the six unseen tasks while ignoring the gate token. We then apply fixed top-K truncation to the same document logits $\{s_i\}$, sweeping $K \in \{1, 3, 5, 10, 20, 100, 150\}$.

Table 4 reports the results. The best fixed-K F1 is 0.24 at $K = 3$, whereas the adaptive gate achieves 0.27, an 11.3% relative improvement. Under the same best-F1 comparison, adaptive gating also improves nDCG from 0.46 to 0.54. Because this comparison uses the same trained model and the same document logits, it directly isolates the benefit of gate-based selection. We note that nDCG continues to rise as K increases (e.g., 0.62 at $K = 150$), which is expected when the metric is computed over a longer retained prefix of an already ranked list; this does not imply a better precision–recall trade-off for retained-set selection, where F1 remains the primary criterion. Appendix B further shows that the predicted retained-set size varies substantially across queries and unseen tasks, rather than collapsing to a nearly constant depth.

4.5 Ablation and Robustness Analysis

Table 5 reports ablation results and input-order robustness analysis. Since GateReranker is a listwise reranker, we evaluate each variant under three random permutations of the candidate list while keeping the candidate pool fixed. We report the mean (μ) and standard deviation (σ) of F1 and nDCG across the three shuffled evaluations on BEIR.

The full model achieves the best F1 and nDCG with near-zero variance across shuffled inputs, showing that GateReranker is robust to input ordering. This is important in the listwise setting, where candidate documents are processed jointly and the model could otherwise be sensitive to positional changes.

	F1		nDCG	
Variant	μ	σ	μ	σ
GateReranker	0.42	0.00	0.65	0.00
w/o doc token	0.14	0.15	0.28	0.21
w/o gate loss (adaptive)	0.08	0.00	0.40	0.14
w/o gate loss ($K = 3$)	0.27	0.00	0.53	0.00
w/o gate loss (τ)	0.29	0.00	0.35	0.00
w/o rank loss	0.14	0.16	0.28	0.23

Table 5: Ablation and input-order robustness analysis on BEIR. We report the mean (μ) and standard deviation (σ) of F1 and nDCG over three evaluations with randomly shuffled candidate order. Unless noted otherwise, variants use adaptive gating ($s_i > g$). For the gating-loss ablation, we additionally report fixed top-K truncation ($K = 3$) and global score thresholding.

All ablations lead to clear degradation relative to the full model. Removing the document marker token substantially reduces performance, indicating that explicit document boundary markers help align candidate passages with their identifier tokens in the serialized input. Excluding the ranking objectives also hurts performance and increases variance, suggesting that ranking supervision complements gating by improving score structure and making the model more robust to changes in candidate order.

For the gating-loss ablation, applying the untrained gate as an adaptive cutoff collapses retained-set F1 to 0.08, showing that g is not a meaningful decision boundary without gating supervision. Replacing adaptive gating with fixed top-K truncation ($K = 3$) or a train-selected global threshold recovers substantially stronger scores (F1 0.27 / 0.29 and nDCG 0.53 / 0.35), but both remain well below the full model (F1 0.42, nDCG 0.65). Thus, gating losses are necessary for reliable in-model selection, and they also improve the document-score structure used by post-hoc truncation.

To better understand how training shapes the selection boundary, we additionally inspect raw-logit distributions before and after training (Appendix C). The ridgeline plots show that joint ranking-and-gating training improves positive–negative separation and makes the predicted gate a clearer decision boundary, consistent with the ablation results above.

Overall, the results show that the gains of GateReranker come from the joint effect of structured input design, adaptive boundary learning, and ranking supervision, while maintaining strong robustness to input ordering.

5 Conclusion

We presented **GateReranker**, a listwise LLM reranker that jointly models document ranking and retained-set selection. By introducing a dedicated gate token, GateReranker learns an instance-dependent selection boundary in the same output space as document identifier tokens. This allows the model to directly produce an adaptive ranked subset from first-step logits, rather than relying on fixed top-K truncation or global score thresholding as a separate post-hoc step.

Experiments on BEIR show that GateReranker achieves the best retained-set F1 and the highest nDCG compared with strong open-source rerankers equipped with post-hoc selection strategies. These results demonstrate that adaptive selection can be effectively integrated into reranking without sacrificing ranking quality. More broadly, our findings suggest that document retention should be treated as a first-class prediction target alongside document ordering in future reranking models.

Limitations

This work evaluates GateReranker under a supervised BEIR reranking setting with candidate lists limited to the top 150 documents retrieved by BM25. Future work may examine larger candidate pools, additional retrieval benchmarks, and zero-shot transfer settings.

GateReranker is currently instantiated with a single LLM backbone and a specific reserved-token design for document markers and the gate token. Further studies could investigate how the ranking-and-gating mechanism transfers across model families, model scales, and tokenizer designs.

AI USE STATEMENT

We used generative AI tools to assist with language editing and polishing of the manuscript text. We did not use generative AI tools to generate experimental results, mathematical claims, proofs, or core methodological designs. All AI-assisted text was reviewed and verified by the authors, who take full responsibility for the final content of this work.

ETHICS STATEMENT

All experiments are conducted on publicly available retrieval benchmarks without intentional inclusion of sensitive or personally identifiable information. Because the method affects which documents are retained for downstream systems, deployment decisions should consider potential impacts on information exposure and fairness.

REPRODUCIBILITY STATEMENT

Training hyperparameters are reported in Appendix A. Prompt templates are provided in Appendix D, and complete top-K sweeps appear in Appendix E. Datasets and baseline checkpoints used in this work are publicly available. We will release code and configuration files to facilitate reproduction of the reported results.

REFERENCES

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie

Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz,

Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Jiafeng Guo, Yinqiong Cai, Yixing Fan, Fei Sun, Ruqing Zhang, and Xueqi Cheng. Semantic models for the first-stage retrieval: A comprehensive review. *ACM Transactions on Information Systems (TOIS)*, 40(4):1–42, 2022.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Chuan Meng, Negar Arabzadeh, Arian Askari, Mohammad Aliannejadi, and Maarten de Rijke. Ranked list truncation for large language model-based re-ranking. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 2024, pp. 141–151. ACM, July 2024. doi: 10.1145/3626772.3657864. URL http://dx.doi.org/10.1145/3626772.3657864.

Siyuan Meng, Junming Liu, Yirong Chen, Song Mao, Pinlong Cai, Guohang Yan, Botian Shi, and Ding Wang. From ranking to selection: A simple but efficient dynamic passage selector for retrieval augmented generation. *arXiv preprint arXiv:2508.09497*, 2025.

Rodrigo Nogueira and Kyunghyun Cho. Passage re-ranking with bert, 2020. URL https://arxiv.org/abs/1901.04085.

Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin. Document ranking with a pretrained sequence-to-sequence model, 2020. URL https://arxiv.org/abs/2003.06713.

Pavel Posokhov, Sergei Masliukhin, Skrylnikov Stepan, Danil Tirskikh, and Olesia Makhnytkina. Relevance scores calibration for ranked list truncation via tmp adapter. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 7728–7734, 2025.

Revanth Gangi Reddy, JaeHyeok Doo, Yifei Xu, Md Arafat Sultan, Deevya Swain, Avirup Sil, and Heng Ji. First: Faster improved listwise reranking with single token decoding, 2024. URL https://arxiv.org/abs/2406.15657.

Ruiyang Ren, Yuhao Wang, Kun Zhou, Wayne Xin Zhao, Wenjie Wang, Jing Liu, Ji-Rong Wen, and Tat-Seng Chua. Self-calibrated listwise reranking with large language models, 2024. URL https://arxiv.org/abs/2411.04602.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. Is chatgpt good at search? investigating large language models as re-ranking agents, 2024. URL https://arxiv.org/abs/2304.09542.

Chihiro Taguchi, Seiji Maekawa, and Nikita Bhutani. Efficient context selection for long-context qa: No tuning, no iteration, just adaptive-k. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 20116–20141, 2025.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=wCu6T5xFjeJ.

Feng Wang, Yuqing Li, and Han Xiao. jina-reranker-v3: Last but not late interaction for document reranking, 2025. URL https://arxiv.org/abs/2509.25085.

Yifan Xu, Vipul Gupta, Rohit Aggarwal, Varsha Mahadevan, and Bhaskar Krishnamachari. Cluster-based adaptive retrieval: Dynamic context selection for rag applications. *arXiv preprint arXiv:2511.14769*, 2025.

Soyoung Yoon, Eunbi Choi, Jiyeon Kim, Hyeongu Yun, Yireun Kim, and Seung won Hwang. Listt5: Listwise reranking with fusion-in-decoder improves zero-shot retrieval, 2024. URL https://arxiv.org/abs/2402.15838.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. Making retrieval-augmented language models robust to irrelevant context. In *International Conference on Learning Representations*, volume 2024, pp. 29862–29883, 2024.

Tan Yu, Anbang Xu, and Rama Akkiraju. In defense of rag in the era of long-context language models. *arXiv preprint arXiv:2409.01666*, 2024.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. Dense text retrieval based on pretrained language models: A survey. *ACM Transactions on Information Systems*, 42(4):1–60, 2024.

Honglei Zhuang, Zhen Qin, Rolf Jagerman, Kai Hui, Ji Ma, Jing Lu, Jianmo Ni, Xuanhui Wang, and Michael Bendersky. Rankt5: Fine-tuning t5 for text ranking with ranking losses, 2022. URL https://arxiv.org/abs/2210.10634.

A Training Hyperparameters

Table 6 lists the training hyperparameters used for GateReranker.

Backbone	Llama-3.2-3B
GPUs	8 × NVIDIA H20
Precision	bf16
Learning rate	1×10^{-5}
Optimizer	AdamW
Weight decay	0.01
Warmup ratio	0.1
Epochs	1
Per-device batch size	1
Gradient accumulation steps	8
Margin m	0.5
α	0.4
β	0.3
γ	0.2
θ	0.1
Max length	131072
Max train chunks	150
Max eval chunks	150
Max chunk tokens	512

Table 6: Training hyperparameters of GateReranker.

B Retained-Set Size Variation

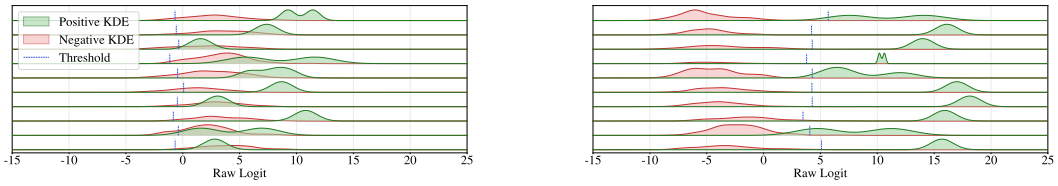
To check whether adaptive gating collapses to a nearly constant retention depth, Table 7 reports the mean and standard deviation of GateReranker's retained-set size on the six unseen BEIR tasks. The means differ substantially across datasets (from about 4–5 documents on Climate-FEVER and ArguAna to about 44 on TREC-COVID and Touche-2020), and the within-dataset standard deviations are large relative to these means. This indicates that the predicted retained-set size varies across queries rather than behaving like a fixed top-K rule.

Dataset	Retained size (mean $\pm$ std)
TREC-COVID	44.14 ± 22.09
Touche-2020	44.67 ± 24.39
SCIDOCS	10.52 ± 7.21
NQ	11.95 ± 8.44
Climate-FEVER	4.40 ± 2.85
ArguAna	4.92 ± 2.94

Table 7: Mean and standard deviation of GateReranker's retained-set size on the six unseen BEIR tasks.

C Score Separation Before and After Training

To visualize how training reshapes the selection boundary, we sample 10 test queries from CMedQAv2-reranking and plot the distributions of raw document logits before and after training (Figure 3). This analysis is complementary to the BEIR ablations in the main text: it illustrates class separability at the logit level, rather than providing an additional benchmark comparison.



(a) Before training.

(b) After training.

Figure 3: Raw-logit ridge plots on CMedQAv2-reranking before and after training. Green and red curves show positive and negative logit distributions, and the dashed blue line marks the predicted threshold. Training improves score separation and yields a clearer decision boundary.

Initially, the scores of positive and negative instances overlap considerably, resulting in poor separation by the threshold. After training, positive logits shift toward higher values while negative logits remain low, creating a clear margin between the two distributions. This observation aligns with the ablation results: the joint ranking-and-gating objective enhances class separability, rendering the gate score a more robust decision boundary for retained-set selection.

D Prompt Templates

GateReranker adopts a chat-style prompt format consisting of a system message and a user message. The system prompt defines the model role and output constraints, while the user prompt provides the query and the serialized candidate passages. During both training and inference, we use a consistent prompt structure so that the model produces its decision at the first prediction step.

Each candidate document is serialized with an explicit document marker token and an identifier token, as described in the main text. Concretely, the candidate list is formatted as $\{\texttt{DOC_TOKEN}\}\ t_1$: $\text{document}_1, \backslash\text{n} \ldots, \backslash\text{n}\{\texttt{DOC_TOKEN}\}\ t_n$: document_n, where $\{\texttt{DOC_TOKEN}\}$ denotes the document marker token and the numeric identifier is used to align each document with its corresponding output token. We additionally reserve a gate token, denoted as $\{\texttt{GATE_TOKEN}\}$, to represent the no-relevant-passage option in the prompt and to provide the gate score used for adaptive selection.

Although the prompt elicits a single-choice distribution, the training objectives reshape the full identifier-token logit distribution so that multiple relevant documents can score above the gate.

The user prompt and system prompt used in our experiments are shown below.

User prompt.

```
## Question
{query}

## Candidate Passages
{chunks}

## Instruction
Please choose the single most relevant passage from the candidates above.
If none of the passages are relevant, reply with `{GATE_TOKEN}`.

The number of the most relevant passage is:
```

System prompt.

```
# Role
You are a strict relevance ranking judge.

# Task
Given a question and several text passages, select the most relevant passage.

# Rules
1. You must choose one option from the candidates, or choose {GATE_TOKEN}.
2. Do not output any explanation, reasoning, or extra text.

# Selection Criteria
- Choose the number of the most relevant passage.
- If none of the passages are relevant, choose {GATE_TOKEN}.

# Output Format
The passage number or {GATE_TOKEN}
```

In the actual input serialization, {query} and {chunks} are replaced by the query text and the serialized candidate passages, respectively. In our implementation, the identifier token t_i is instantiated as the numeric index of the corresponding document, i.e., $t_i = i$.

Although the prompt is phrased as selecting a single most relevant passage or the special gate token, GateReranker does not use the generated token itself as the final prediction. Instead, we read the logits of all identifier tokens together with the gate token at the first prediction step, and use them as document relevance scores and the gate score, respectively. Under this formulation, the prompt serves to elicit a clear ranking-oriented preference distribution over the candidate passages, while the final retained set is determined by the joint scoring-and-gating rule described in the main text. This design preserves the structured decision semantics of the prompt while avoiding the instability of unconstrained decoding.

E Additional Fixed Top-K Results

In the main experiments, we compare GateReranker with open-source rerankers under their best fixed top-K operating points and global score thresholds. To provide a more complete view of how fixed truncation depth affects performance, this appendix reports results across all evaluated top-K values on BEIR. We sweep K over {1,3,5,10,20,100,150} for each open-source reranker, while GateReranker directly predicts an adaptive retained set without requiring a fixed truncation depth.

E.1 Retained-Set F1 under Fixed Top-K Truncation

Table 8 reports retained-set F1 across different fixed top-K truncation settings. The results show that the optimal truncation depth varies across datasets and rerankers. Although many baselines achieve

their best average F1 at $K = 1$, larger values of K can be beneficial for individual datasets such as Touche-2020, where more relevant documents may need to be retained. This confirms that a single fixed truncation depth is brittle in heterogeneous retrieval settings.

GateReranker achieves the best average F1 without selecting a fixed K. Instead of applying the same retention depth to every query, it predicts a list-dependent gate score and retains documents whose relevance scores exceed that boundary. These results support the main finding that adaptive selection provides a stronger retained-set selection mechanism than static top-K truncation.

Models	Top-K	Avg.	TC	NFC	NQ	HQA	FQA	AA	TCH	DBP	SD	FVR	CFV	SF	QRA
BGE-Reranker-v2-m3	1	0.32	0.00	0.08	**0.44**	0.61	0.24	0.25	0.07	0.12	0.06	0.77	0.18	0.61	**0.73**
	3	0.27	0.01	0.12	0.39	0.58	0.25	0.23	0.19	0.18	0.11	0.44	0.20	0.38	0.50
	5	0.22	0.02	0.13	0.25	0.43	0.21	0.18	0.19	0.20	0.12	0.30	0.18	0.27	0.37
	10	0.16	0.04	0.13	0.15	0.25	0.15	0.12	0.24	0.22	0.12	0.17	0.13	0.16	0.22
	20	0.12	0.07	0.11	0.08	0.14	0.10	0.08	0.26	0.22	0.09	0.09	0.08	0.09	0.12
	100	0.06	0.19	0.07	0.02	0.03	0.03	0.02	0.17	0.12	0.03	0.02	0.02	0.02	0.03
	150	0.05	0.19	0.06	0.01	0.02	0.02	0.01	0.13	0.09	0.02	0.01	0.01	0.01	0.02
Qwen3-Reranker-0.6B	1	0.33	0.01	0.09	0.35	0.59	0.22	0.45	0.05	0.12	0.08	0.76	0.20	0.63	0.71
	3	0.28	0.02	0.13	0.31	0.56	0.24	0.37	0.11	0.18	0.14	0.44	0.22	0.39	0.48
	5	0.23	0.02	0.14	0.24	0.42	0.21	0.29	0.17	0.20	0.15	0.30	0.19	0.28	0.36
	10	0.17	0.04	0.14	0.15	0.25	0.16	0.17	0.20	0.22	0.14	0.17	0.14	0.17	0.22
	20	0.12	0.08	0.12	0.08	0.14	0.10	0.09	0.23	0.22	0.11	0.09	0.08	0.09	0.12
	100	0.06	0.19	0.07	0.02	0.03	0.03	0.02	0.16	0.12	0.04	0.02	0.02	0.02	0.03
	150	0.05	0.19	0.06	0.01	0.02	0.02	0.01	0.13	0.09	0.02	0.01	0.01	0.01	0.02
Qwen3-Reranker-4B	1	0.35	0.00	0.09	0.43	0.60	0.27	0.57	0.06	0.13	0.09	0.79	0.22	0.66	0.68
	3	0.30	0.01	0.13	0.34	0.58	0.28	0.42	0.15	0.19	0.16	0.45	0.24	0.41	0.48
	5	0.24	0.02	0.14	0.25	0.43	0.24	0.30	0.21	0.21	**0.18**	0.31	0.21	0.29	0.36
	10	0.18	0.04	0.14	0.15	0.25	0.17	0.17	0.25	0.24	0.16	0.17	0.15	0.17	0.22
	20	0.13	0.08	0.13	0.08	0.14	0.11	0.09	0.27	0.24	0.12	0.09	0.09	0.09	0.12
	100	0.06	0.19	0.07	0.02	0.03	0.03	0.02	0.17	0.12	0.04	0.02	0.02	0.02	0.03
	150	0.05	0.19	0.06	0.01	0.02	0.02	0.01	0.13	0.09	0.02	0.01	0.01	0.01	0.02
Qwen3-Reranker-8B	1	0.36	0.00	0.09	0.42	0.59	0.29	**0.65**	0.08	0.13	0.09	**0.82**	0.23	0.66	0.68
	3	0.30	0.01	0.13	0.34	0.57	0.29	0.44	0.16	0.18	0.16	0.45	0.24	0.42	0.48
	5	0.24	0.02	0.14	0.25	0.42	0.25	0.31	0.23	0.20	0.17	0.31	0.21	0.30	0.36
	10	0.18	0.04	0.14	0.15	0.25	0.17	0.17	**0.28**	0.23	0.16	0.17	0.15	0.17	0.22
	20	0.13	0.08	0.13	0.08	0.14	0.10	0.09	**0.28**	0.23	0.12	0.09	0.09	0.09	0.12
	100	0.06	0.19	0.07	0.02	0.03	0.03	0.02	0.17	0.12	0.04	0.02	0.02	0.02	0.03
	150	0.05	0.19	0.06	0.01	0.02	0.02	0.01	0.13	0.09	0.02	0.01	0.01	0.01	0.02
Jina-Reranker-v3	1	0.34	0.00	0.09	**0.44**	0.60	0.26	0.51	0.04	0.12	0.08	0.80	0.19	0.62	0.70
	3	0.27	0.01	0.12	0.33	0.56	0.23	0.38	0.08	0.16	0.14	0.45	0.20	0.38	0.47
	5	0.22	0.02	0.12	0.25	0.42	0.20	0.29	0.12	0.18	0.14	0.31	0.17	0.26	0.35
	10	0.16	0.04	0.12	0.15	0.25	0.15	0.17	0.17	0.19	0.13	0.17	0.13	0.16	0.22
	20	0.11	0.07	0.11	0.08	0.14	0.10	0.09	0.21	0.19	0.10	0.09	0.08	0.09	0.12
	100	0.06	0.18	0.07	0.02	0.03	0.03	0.02	0.16	0.12	0.04	0.02	0.02	0.02	0.03
	150	0.05	0.19	0.06	0.01	0.02	0.02	0.01	0.13	0.09	0.02	0.01	0.01	0.01	0.02
GateReranker	-	**0.42**	**0.55**	**0.35**	0.18	**0.76**	**0.32**	0.21	0.25	**0.38**	**0.18**	0.65	**0.25**	**0.67**	0.68

Table 8: F1 results on BEIR across different fixed top-K truncation settings. GateReranker directly predicts an adaptive selection boundary instead of relying on a fixed truncation depth, and achieves the best average F1 overall.

E.2 Ranking Quality under Fixed Top-K Truncation

Table 9 reports nDCG across the same fixed top-K truncation settings. Compared with F1, nDCG focuses on the quality of the induced document ordering by rewarding relevant documents placed near the top of the ranked output. The results show that increasing K can sometimes improve nDCG for certain baselines and datasets, but the best truncation depth remains inconsistent across tasks.

GateReranker obtains the best average nDCG overall, indicating that adaptive selection does not come at the cost of weaker ranking quality. Together with the F1 results in Table 8, these results show that GateReranker improves retained-set selection while preserving strong document ordering.

Models	Top-K	Avg.	TC	NFC	NQ	HQA	FQA	AA	TCH	DBP	SD	FVR	CFV	SF	QRA
BGE-Reranker-v2-m3	1	0.54	0.86	0.43	0.48	0.91	0.41	0.25	0.45	0.55	0.18	0.80	0.30	0.63	0.81
	3	0.53	0.83	0.39	0.56	0.76	0.36	0.37	0.42	0.46	0.15	0.82	0.26	0.69	0.85
	5	0.53	0.80	0.36	0.58	0.77	0.37	0.40	0.38	0.44	0.14	0.83	0.27	0.70	0.87
	10	0.54	0.77	0.33	0.60	0.78	0.38	0.45	0.35	0.41	0.17	0.83	0.30	0.72	**0.88**
	20	0.54	0.73	0.30	0.60	0.78	0.40	0.48	0.36	0.40	0.19	0.83	0.31	0.73	**0.88**
	100	0.54	0.51	0.28	**0.61**	0.78	0.41	0.51	0.48	0.41	0.23	0.83	0.31	0.74	**0.88**
	150	0.52	0.21	0.27	**0.61**	0.78	0.41	0.51	0.50	0.41	0.24	0.83	0.31	0.74	**0.88**
Qwen3-Reranker-0.6B	1	0.55	**0.91**	0.49	0.38	0.88	0.36	0.44	0.29	0.55	0.25	0.80	0.34	0.65	0.78
	3	0.55	0.90	0.43	0.48	0.73	0.34	0.62	0.30	0.45	0.20	0.82	0.28	0.71	0.83
	5	0.55	0.88	0.40	0.51	0.75	0.35	0.67	0.31	0.43	0.17	0.83	0.29	0.73	0.85
	10	0.56	0.85	0.36	0.54	0.76	0.38	0.69	0.28	0.41	0.21	0.83	0.31	0.75	0.86
	20	0.56	0.79	0.33	0.55	0.76	0.40	0.70	0.30	0.40	0.23	0.83	0.32	0.76	0.87
	100	0.55	0.54	0.29	0.55	0.76	0.41	0.70	0.43	0.41	0.27	0.83	0.33	0.76	0.87
	150	0.53	0.22	0.28	0.55	0.76	0.41	0.70	0.46	0.41	0.27	0.83	0.33	0.76	0.87
Qwen3-Reranker-4B	1	0.59	**0.91**	0.48	0.46	0.90	0.44	0.56	0.42	0.59	0.28	0.84	0.39	0.68	0.75
	3	0.59	0.90	0.43	0.55	0.76	0.40	0.72	0.42	0.50	0.23	0.85	0.32	0.74	0.82
	5	0.59	0.88	0.40	0.58	0.77	0.41	0.75	0.41	0.47	0.20	0.85	0.33	0.75	0.83
	10	0.59	0.85	0.36	0.59	0.77	0.43	0.77	0.37	0.44	0.24	0.85	0.34	0.77	0.85
	20	0.59	0.80	0.33	0.60	0.78	0.44	0.77	0.38	0.43	0.27	0.85	0.35	0.78	0.85
	100	0.58	0.54	0.29	0.60	0.78	0.45	0.77	0.50	0.43	0.29	0.85	0.35	0.78	0.85
	150	0.55	0.22	0.28	0.60	0.78	0.45	0.77	0.51	0.43	0.29	0.85	0.35	0.78	0.85
Qwen3-Reranker-8B	1	0.61	0.90	0.49	0.45	0.88	0.47	0.65	0.55	0.58	0.26	0.86	0.40	0.68	0.75
	3	0.60	0.89	0.44	0.55	0.75	0.43	0.78	0.48	0.48	0.22	0.85	0.32	0.75	0.82
	5	0.60	0.88	0.40	0.57	0.76	0.44	0.80	0.46	0.45	0.19	0.86	0.33	0.77	0.84
	10	0.59	0.83	0.37	0.59	0.77	0.45	0.81	0.41	0.43	0.23	0.86	0.35	0.78	0.85
	20	0.59	0.78	0.33	0.60	0.77	0.46	**0.82**	0.40	0.42	0.26	0.86	0.35	0.78	0.86
	100	0.58	0.53	0.30	0.60	0.77	0.46	**0.82**	0.52	0.42	0.28	0.86	0.35	**0.79**	0.86
	150	0.56	0.22	0.29	0.60	0.77	0.46	**0.82**	0.53	0.42	0.28	0.86	0.35	**0.79**	0.86
Jina-Reranker-v3	1	0.55	0.82	0.44	0.48	0.89	0.42	0.51	0.22	0.54	0.25	0.84	0.32	0.64	0.77
	3	0.54	0.82	0.39	0.55	0.73	0.35	0.66	0.21	0.44	0.19	0.84	0.26	0.69	0.82
	5	0.54	0.82	0.35	0.58	0.75	0.36	0.70	0.22	0.41	0.16	0.84	0.27	0.70	0.84
	10	0.54	0.79	0.32	0.59	0.76	0.38	0.72	0.23	0.38	0.19	0.84	0.29	0.72	0.85
	20	0.55	0.74	0.29	0.60	0.76	0.40	0.73	0.26	0.36	0.22	0.85	0.30	0.73	0.86
	100	0.55	0.51	0.27	0.60	0.77	0.42	0.73	0.40	0.39	0.26	0.85	0.31	0.74	0.86
	150	0.53	0.21	0.27	0.60	0.77	0.42	0.73	0.43	0.39	0.26	0.85	0.31	0.74	0.86
GateReranker	-	**0.65**	0.89	**0.60**	0.52	**0.94**	**0.54**	0.43	**0.57**	**0.71**	**0.37**	**0.88**	**0.43**	0.75	**0.88**

Table 9: nDCG results on BEIR across different fixed top-K truncation settings. GateReranker directly predicts an adaptive selection boundary instead of relying on a fixed truncation depth, and achieves the best average nDCG overall.