

Explain then Rank: Scale Calibration of Neural Rankers Using Natural Language Explanations from LLMs

Anonymous ACL submission

Abstract

In search settings, calibrating the scores during the ranking process to quantities such as click-through rates or relevance levels enhances a system’s usefulness and trustworthiness for downstream users. While previous research has improved this notion of calibration for low complexity learning-to-rank models, the larger data demands and parameter count specific to modern neural text rankers produce unique obstacles that hamper the efficacy of methods intended for the learning-to-rank setting.

This paper proposes exploiting large language models (LLMs) to provide relevance and uncertainty signals for these neural text rankers to produce scale-calibrated scores through Monte Carlo sampling of natural language explanations (NLEs). Our approach transforms the neural ranking task from ranking textual query-document pairs to ranking corresponding synthesized NLEs. Comprehensive experiments on two popular document ranking datasets show that the NLE-based calibration approach consistently outperforms past calibration methods and LLM-based methods for ranking, calibration, and query performance prediction tasks.

1 Introduction

Neural ranking models act as the core component of many search systems, often producing the final document scores. However, these scores are usually treated as transient information and only the relative orderings are preserved to produce a ranking. While this approach results in well-performing systems with respect to common retrieval metrics, such as nDCG and MAP, it ignores vital information that could be used by end users and downstream applications with real-world impacts, such as fair ranking (Zerveas et al., 2022), ranked list truncation (Bahri et al., 2020), and query performance prediction (Zhou and Croft, 2007; Shtok et al., 2012; Faggioli et al., 2023b).

The common decision to discard the model scores comes from the fact that almost all neural ranking models, even probabilistic approaches, are trained to optimize relative orderings of documents as opposed to their absolute level of relevance. Aligning these ranking scores to a fixed scale is particularly difficult due to the nature of each query requiring different amounts of information to satisfy relevance, and it is much easier to determine if a document is more relevant than another rather than determining whether the information is sufficient for complex queries (Khattab and Zaharia, 2020; Santhanam et al., 2021; Zhuang et al., 2023b; Laitz et al., 2024).

This concept, that output scores should have meaningful real-world interpretations, is known as *calibration*. Having this property ensures that a model’s predictions reliably reflects the “true score.” The most commonly recognized form of this is, where, for example, a classification model with a confidence of $p = 0.47$ should be correct 47% of the time. Building upon this idea of grounded scores, **scale calibration** extends this setting to values that do not have a direct probabilistic interpretation, such as document dwell time (Smucker and Clarke, 2012) and multiple levels of relevance (Yan et al., 2022; Bai et al., 2023) which can exist beyond the $[0,1]$ range.

Recent studies by Yan et al. (2022) and Bai et al. (2023), have highlighted a conflict between the objectives of the scale calibration and the ranking task, where optimizing for one may compromise the other. They propose new training objectives to improve scale calibration in learning-to-ranking (LTR) models, which are low in parameter count due to relying on these LTR datasets containing solely preprocessed features like length and query-term overlap as inputs. Applying these methods to larger neural rankers over raw text remains a challenge. As opposed to LTR datasets, text-based collections with graded relevance judgements re-

Task: to assign a **meaningful** ranking score to the query-document pair.
0 means irrelevant and 3 means perfectly relevant.

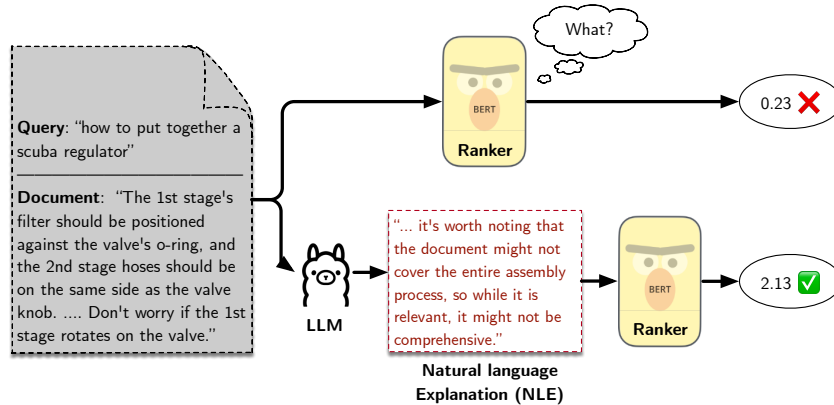


Figure 1: The key idea of this study: Neural ranking models struggle to produce *meaningful* ranking scores when encountering complex query-document pairs. We investigate the integration of natural language explanations as inputs to neural rankers, aiming to simplify the scale-calibrated ranking task for these rankers.

main limited and orders of magnitude smaller. Further, the drastically higher parameter count of neural text rankers (Craswell et al., 2021; Nogueira et al., 2019b, 2020; Zhuang et al., 2023b; Ma et al., 2023a) make them even more data-intensive and reduces the efficacy of listwise loss functions due to hardware constraints and data sparsity.

In light of these challenges, we propose an alternative approach to improve scale calibration and score usability in these large text ranking models that does not rely on the loss functions of past works specific to the LTR settings. We instead ground the candidate query-document pairs via external knowledge, i.e., specifically leveraging the recent advances in generative and autoregressive large language models (LLMs) and their emerging capabilities for general language tasks with minimal supervision (Wei et al., 2022a). By producing natural language explanations (NLEs) of relevance with this additional knowledge, the ranking model will be able to leverage this information and produce better calibrated scores. As current LLMs have stability issues when generating text and can produce varying responses given the same input (Ye and Durrett, 2022; Turpin et al., 2023; Chen et al., 2023), we hypothesize that the confidence and stability of a NLE generated from a zero-shot LLM *explaining* the relevance of a query-document pair correlates well with a calibrated ranking score. We validate this hypothesis by developing a method that aggregates the Monte Carlo (MC) generated NLEs of query-document relevance that serves as input into a traditional neural ranking model.

This work makes the following three contributions:

1. Introduce MC-aggregate NLEs to improve calibration error and downstream usability of document rankings.
2. Significantly enhance the scale calibration of neural text rankers up to 25% and downstream usability while maintaining or improving ranking effectiveness.
3. Conduct a wide set of experiments on two benchmark text ranking datasets to validate performance and properties.

2 Related Work

Scale Calibration of LTR Models. Previous approaches to calibrating ranking models primarily involved post-processing methods (Tagami et al., 2013; Chaudhuri et al., 2017), where a function maps the outputs for the frozen ranking model (Platt, 2000; Zadrozny and Elkan, 2002). Recent studies, such as those by Yan et al. (2022) and Bai et al. (2023), have identified the issue that commonly used pairwise and listwise ranking objectives theoretically lead to uncalibrated scores and advocate for calibrated ranking objectives as a solution. While effective for the LTR setting that uses extracted features and small models, the limitations imposed by large text ranking models makes their effective application non-trivial.

Uncertainty Calibration of Neural Ranking Models. To achieve a notion of calibration, Co-

hen et al. (2021) and Penha and Hauff (2021) suggest leveraging a Bayesian interpretation of neural rankers to capture a distribution of possible scores for each document based on the internal uncertainty of the model. However, these approaches fail to ground the scores on any scale as such uncertainty information only makes sense when comparing documents, and is distinct from the type of calibration discussed in this paper.

Natural Language Explanations. NLEs are an emerging point of research in NLP and IR. These explanations can be presented directly to users to elucidate the rationale behind system decisions, such as in recommendation systems (Li et al., 2020) and search results (Yu et al., 2022). More pertinent to this study, NLEs have been employed as intermediaries to enhance task efficacy (Camburu et al., 2018; Wei et al., 2022b; Ferraretto et al., 2023). Ye and Durrett (2022) use the term overlap ratio between NLEs and their corresponding inputs as a feature to calibrate LLM predictions. Wadhwa et al. (2023) investigated explanation-based rescaling, converting coarse numeric scales into finer gradations. This technique leverages *annotator-written* NLEs and scoring rubrics to prompt LLMs in generating rescaled outputs. Our research diverges by concentrating on NLEs *autonomously* generated by LLMs, which we leverage as inputs for neural rankers to facilitate scale calibration.

Calibration of Large Language Models. Various recent studies have focused on eliciting well-calibrated confidence scores from LLM responses. Some works treat LLMs as black boxes and use various prompting-based techniques to generate confidence scores (Lin et al., 2022; Kadavath et al., 2022). Other works rely on training external models to associate confidence with LLM responses (Kuhn et al., 2022; Kadavath et al., 2022; Mielke et al., 2022). Tian et al. (2023) conducted systematic evaluations over a wide range of methods to obtain confidence scores from LLMs and discovered that LLM-verbalized probabilities are better calibrated than conditional probabilities across several black-box RLHF-tuned LLMs. This framework was later extended to include hybrid methods (i.e., a mix of self-consistency and verbalized confidence) and evaluated the calibration performance on five different types of datasets (Xiong et al., 2023). In Section 4.3, we introduce two methods that take advantage of the calibration capabilities of LLMs to elicit scale calibrated ranking scores.

3 Methodology

In this section, we formalize the definition of scale calibration of neural ranking models, following which we introduce the proposed method of leveraging natural language explanations (NLEs) for obtaining scale calibrated ranking scores.

3.1 Problem Statement and Motivation

In ad-hoc ranking, we define a scoring function ϕ for a given query q and its n associated candidate documents $\{d^q\}_1^n$. This function, denoted as $\phi_\Phi(q, \{d^q\})$ produces a score for each query-document pair under the given retrieval model parameterized by Φ . The ideal parameters of Φ are obtained by optimizing an empirical loss on a query-grouped training dataset $\mathcal{D} = \{(\{d^q\}, \{y^q\}) \mid q \in Q\}$. Here, Q represents the set of training queries, and $\{y^q\}$ is the corresponding set of labels of documents $\{d^q\}$. The empirical loss is defined as:

$$\mathcal{L}(\Phi) = \frac{1}{|Q|} \sum_{q \in Q} l^{\text{rank}}(\{y^q\}, \phi_\Phi(q, \{d^q\})) \quad (1)$$

where l^{rank} is a ranking loss function for an individual query. In the case of neural ranking, the scoring function ϕ is defined by a backbone neural language model. For instance, using a pretrained BERT checkpoint as Φ , concatenating query and each candidate document with a [SEP] token in between as the inputs, and leveraging cross entropy loss or pairwise cross entropy loss as l^{rank} leads to the development of the widely known monoBERT or duoBERT models (Nogueira et al., 2019a), respectively. However, it has been observed that popular pairwise and listwise ranking losses are not scale calibrated due to their translation-invariant property (Yan et al., 2022)¹; adding a constant to all outputs of ϕ does not alter the loss value.

Although calibrated ranking losses works well for LTR models (Yan et al., 2022; Bai et al., 2023), the application of these approaches to neural text rankers is non-trivial. In LTR datasets with graded relevance judgements (Chapelle and Chang, 2011; Qin and Liu, 2013; Dato et al., 2017), $|Q|$ is orders of magnitude larger than that of text based collections, which often only have less than 100 queries. Further, $|\Phi|$ is much larger in neural text ranking models, limiting the use of listwise loss functions

¹Although the cross entropy loss used in monoBERT is scale-calibrated, but it assumes only binary labels; which is something not observed in real-world datasets (e.g., multiple levels of relevance (Craswell et al., 2020; Mao et al., 2019)).

due to GPU VRAM constraints. Addressing the scale calibration problem in neural rankers therefore requires a multifaceted approach that extends beyond the mere application of calibrated ranking loss.

3.2 Scale Calibration via Natural Language Explanations

We propose a novel two-step approach to obtain a scale-calibrated numerical score from the textual query and document: map the query-document to a NLE and then score this NLE using a standard neural ranking model. This overall strategy is grounded in the recent success of LLMs to establish the relevance of query-document pairs (Ferretto et al., 2023) and the demonstrated efficacy of LLM-generated explanations in various reasoning tasks (Wei et al., 2022b). We use an LLM, denoted as $g(\cdot)$, to generate NLEs $\{e^q\}$ for query-document pairs $\{d^q\}$, and then leverage a neural ranker over *only* the NLEs to predict relevance scores. This process can be formally represented as a decomposition of ϕ into:

$$\phi_{\Phi}(q, \{d^q\}) = f_{\Theta}(g_{\Psi}(q, \{d^q\})) \quad (2)$$

$$= f_{\Theta}(\{e^q\}) \quad (3)$$

where Ψ represents the parameters of the LLM, and Θ encapsulates the parameters of the neural ranker. The neural ranker $f(\cdot)$ in this setup is adapted to take the NLEs of the original inputs as its new inputs. Note that when using Eq. 1 to optimize parameters $\Phi = \{\Theta, \Psi\}$, we can optimize the parameters of the LLM and the neural ranker. While it is possible to perform full or partial fine-tuning on the LLM parameters Ψ , for simplicity and considering the limited amount of training data, we choose to freeze Ψ and only optimize parameters of the neural ranker Θ .

3.3 Acquiring NLEs via LLM Prompting

Given the impact of prompting, we investigate two distinct methods for acquiring NLEs with varying characteristics.

3.3.1 Literal Explanation

We present the query-document pair to the LLM and ask for both a relevance prediction (either “relevant” or “non-relevant”) and an accompanying explanation. The exact prompt we use to generate such literal explanations is listed in Appendix A.

A notable limitation of this approach is its susceptibility to inaccuracies stemming from the LLM’s prediction errors. To mitigate this, we employ a strategy involving Monte Carlo (MC) sampling of multiple NLEs for the same input and then forming a single meta NLE via an aggregation function AGGR (detailed in Section 3.4):

$$e^q = \text{AGGR}(\{y_i \sim g_{\Psi}(y|q, d^q)\}) \quad (4)$$

which aims to diminish the influence of erroneous predictions in the preference of the most likely generation.

3.3.2 Conditional Explanation

To address instances where the LLM persistently errs in judging the relevance of an input, we also explore a different explanation generation strategy termed *conditional explanation*. Motivated by the work of sampling multiple reasoning paths to enhance the self-consistency capabilities of LLMs (Wang et al., 2023), this method involves prompting the LLM to generate rationales supporting both the relevance and non-relevance of a given query-document pair. The exact prompt we use to generate conditional explanations is also listed in Appendix A. Both conditional explanations are then used by the neural ranker $f(\cdot)$ to produce a scale calibrated ranking score (Eq. 3).

3.4 Aggregating Multiple NLEs

As previously discussed, relying solely on the most probable output generated by the LLM, particularly in the literal explanation approach, could result in incorrect and overconfident explanations (Ye and Durrett, 2022). To mitigate this issue, we propose sampling multiple generations from the LLMs. This technique has the advantage to introduce additional information, which may either support the initial judgment with varied reasoning or present contrasting viewpoints. We consider both outcomes to be advantageous: in the former scenario, it leads to a more robust and multi-faceted argument supporting a specific prediction; in the latter scenario, the generation of conflicting information by the LLM partially reveals its uncertainty in comprehending and assessing the relevance of the input query-document pair. We hypothesize that this uncertainty is indicative of predicting a more calibrated ranking score.

Combining the information contained in the collection of Monte Carlo sampled NLEs produces a

Algorithm 1: Novelty-based Aggregation

```
1 Inputs:  $x$ : input prompt;  $\mathcal{E}$ : sentence
  splitter;  $\mathcal{S}$  &  $\lambda$ : text similarity function &
  threshold;  $k_l$  &  $k_s$ : sampling budgets;
   $g_\Psi(y|x)$ : conditional output distribution
  defined by LLM.
2 Output: Meta NLE  $e$ .
3 Function AGGR( $x, \mathcal{E}, \mathcal{S}, \lambda, k_l, k_s, g_\Psi$ ):
4    $e \leftarrow \emptyset$ 
5   for  $i \in 1, 2, \dots, k_l$  do
6      $y_i \leftarrow y \sim g_\Psi(y|x)$ 
7     for  $s \in \mathcal{E}(y_i)$  do
8       if  $e \neq \emptyset$  and
           $\max(\{\mathcal{S}(s, e_j); e_j \in e\}) > \lambda$ 
9         then
10          continue
11        else
12           $e \leftarrow e \cup s$ 
13          if  $|e| \geq k_s$  then
14            return  $e$ 
15  return  $e$ 
```

334 **meta NLE.** This meta NLE then serves as a more
335 comprehensive and nuanced representation of the
336 query-document relationship.

337 To create the meta NLE, we adopt an iterative
338 approach (Algorithm 1) to continually adds novel
339 information. This process begins with an initially
340 empty set of sentences (Line 4). If a sentence from
341 the newly generated explanation (Line 6-7) intro-
342 duces novel information – as determined by its
343 maximum text similarity to the existing sentences
344 in the set being at or below a predefined threshold
345 – then this sentence is added to the set (Line 11).
346 This sampling of new explanations continues until
347 we either reach the predefined maximum number
348 of sampling iterations (Line 5) or fulfill the limit
349 for the number of sentences in the meta NLE set
350 (Line 12). This approach is able to latently encode
351 the uncertainty of the LLM, as a high variance set
352 of sampled NLEs will inevitably produce a longer
353 meta NLE.

354 4 Experiments

355 In this section, we assess our proposed methods
356 using two real-world, extensively used document
357 ranking datasets to evaluate the following research
358 questions (RQ):

359 **RQ1:** How do LLM-generated natural language ex-
360 planations impact the calibration and ranking

Metric	TREC	NTCIR
# Q (train/val/test)	97/53/67	48/16/16
Avg. # docs per query	282.7	345.3
Levels of relevance	4	5
Label dist. (low to high)	58/22/14/6	48/23/17/8/3
Avg. query length	8.0	22.0
Avg. doc. length	70.9	493.2

Table 1: Statistics of the TREC and NTCIR datasets.

performance of neural rankers?

361

RQ2: Is there a consistent improvement across dif-
ferent training objectives when using these
explanations?

362

363

364

RQ3: Does the MC sampled aggregate-NLEs from
the LLM yield empirical improvements?

365

366

RQ4: Does the proposed methodology produce
meaningful scores in a real-world setting?

367

368

4.1 Data

369

370 We employ two widely acknowledged datasets in
371 information retrieval research: the TREC Deep
372 Learning Track (**TREC**) (Craswell et al., 2021),
373 covering the period from 2019 to 2022, and the
374 NTCIR-14 WWW-2 (**NTCIR**) (Mao et al., 2019).

375 The choice of these datasets is motivated by their
376 comprehensive multi-level relevance judgments
377 provided by human annotators and the ample vol-
378 ume of labeled documents for each query. Specifi-
379 cally, TREC uses passages from the MS MARCO
380 collection (Bajaj et al., 2018), while NTCIR em-
381 ploys web pages from ClueWeb12 Category-B.²
382 For TREC, we partition the queries from the years
383 2019 and 2020 for training, use the 2021 queries
384 for validation, and the 2022 queries for testing. For
385 NTCIR, the queries are divided into training, val-
386 idation, and testing sets in a 6:2:2 ratio. Table 1
387 shows provides the statistics for these collections.

4.2 Metrics

388

389 Our goal is to devise methods that not only enhance
390 the calibration of neural rankers but also maintain
391 their ranking effectiveness. Accordingly, we evalu-
392 ate both calibration and ranking aspects.

Ranking Metrics: We employ the normalized dis-
counted cumulative gain (**nDCG**), which accounts
for multiple levels of relevance judgment. We re-
port **nDCG** and **nDCG@10**.

393

394

395

396

²<https://lemurproject.org/clueweb12/>

Calibration Metrics: For calibration effectiveness, mean square error (MSE), as well as the expected calibration error (ECE) (Guo et al., 2017), a metric initially developed for *classification* calibration, are commonly used in ranking literature (Yan et al., 2022; Bai et al., 2023; Penha and Hauff, 2021). ECE measures the discrepancy between a model’s predictions and gold labels in a binning approach via

$$ECE = \sum_{m=1}^M \frac{|B_m|}{n} |acc(B_m) - conf(B_m)|, \quad (5)$$

where M is the number of bins B . However, given the highly skewed and unbalanced label distribution in our test set, as indicated in Table 1, MSE and ECE are biased. Opting for scores that are near the most frequent label results in lower errors while having no real world usability. This bias undermines these metrics’ effectiveness in the ranking context, and is further discussed in Appendix C. To achieve a metric that more closely aligns with real world usability for the search task, we propose the adoption of a class-balanced version of ECE, called **CB-ECE**, which allocates equal significance to all candidate scale values. The CB-ECE is computed by initially calculating the ECE for each individual class, and subsequently taking the mean,

$$CB-ECE = \frac{1}{C} \sum_{c=1}^C ECE_c, \quad (6)$$

with ECE_c as the error for each class, c .

Query Performance Prediction (QPP): As the motivation of this work comes from the desire of usable scores, we also evaluate the methods using the QPP task. We evaluate the effectiveness of each approach by using **Pearson’s** ρ and **Kendall’s** τ between the predicted and actual query performance.

4.3 Baselines

Due to the scarcity of relevant work in scale calibrated search, we have developed several methods to establish meaningful baselines in addition to the past relevant work of Yan et al. (2022). For consistency, we use the same neural ranking architecture, when applicable, for all approaches. Additional details are in Appendix B.

No calibration (NC). We use monoBERT (Nogueira and Cho, 2019), a BERT-based ranker that has been previously finetuned with the MS MARCO passage ranking training set.

Post-hoc calibration (PC). We apply Platt scaling (Platt, 2000) adapted for regression calibration to the uncalibrated monoBERT. It is important to note that the parameters of the ranker remain fixed under this approach.

Full calibration (FC). We use the training data to finetune a BERT-based ranker using calibrated softmax loss (Yan et al., 2022), which represents a straight adoption of the approach from LTR to neural ranking. As discussed, due to substantially larger size of the neural ranker and reduced training data, we are forced to only use a small fraction of negatives for each query compared to under the LTR setting.

LLM prompting with rubrics (PR). Inspired by studies on using LLMs for rendering relevance judgments (Zhuang et al., 2023a; Thomas et al., 2023; Faggioli et al., 2023a) and rescaling (Wadhwa et al., 2023), we explore leveraging the scoring rubrics as a contextual guide for LLM prompting. We present each query-document pair, along with the rubrics from each dataset that were initially used to instruct human annotators, to the LLM and take the scalar predictions.

Post-hoc calibrating LLM predictions (PL). Based on prior studies that LLM confidence scores are well-calibrated (Kadavath et al., 2022; Tian et al., 2023), we use the LLM as a zero-shot binary classifier and take the average predictions over 20 samples as the confidence score. We then use Platt scaling (Platt, 2000) to map these scores to the target scale.

Full calibration with NLEs (NLE). We first acquire meta NLEs for candidate query-document pairs and then fine-tune a BERT model³ using the NLEs as inputs with the calibrated softmax loss with the same restrictions as FC. This forms a direct comparison with FC, as they use the same initialization weights, labels and training objectives, but different inputs.

4.4 Downstream Performance: QPP

The ultimate goal of a calibrated model is interpretable scores by end users and algorithms. Therefore we evaluate the scores produced from each approach using the query performance prediction (QPP) task. The goal of QPP, as discussed

³We experimented with using monoBERT to initialize this ranker but found it to perform worse than general-purpose BERT weights.

Collection	TREC					NTCIR					
	Metric	Ranking		Calibration			Ranking		Calibration		
		nDCG(\uparrow)	nDCG@10(\uparrow)	CB-ECE(\downarrow)	ECE(\downarrow)	MSE(\downarrow)	nDCG(\uparrow)	nDCG@10(\uparrow)	CB-ECE(\downarrow)	ECE(\downarrow)	MSE(\downarrow)
NC	0.799	0.494	1.205	0.320	0.773	0.735	0.337	1.757	0.799	1.824	
PC	0.799	0.494	1.141	0.125	0.684	0.735	0.337	1.624	0.457	1.462	
FC (monoBERT)	0.776	0.422	1.093	0.221	0.721	0.696	0.268	1.843	0.709	1.874	
FC (BERT)	0.738	0.327	1.253	0.266	0.785	0.727	0.285	1.756	0.546	1.416	
PR	0.786	0.457	1.000	1.246	2.137	0.728	0.328	1.294	1.194	2.773	
PL	0.790	0.473	1.165	0.145	0.673	0.736	0.364	1.677	0.472	1.540	
NLE (literal)	0.815 [†]	0.529 [†]	0.996	0.067[†]	0.602[†]	0.742	0.340	1.534	0.355 [†]	1.330 [†]	
NLE (conditional)	0.822[†]	0.534[†]	0.862[†]	0.428	0.832	0.720	0.322	1.405	0.257[†]	1.290[†]	

Table 2: Ranking and scale calibration performance on TREC and NTCIR. Note that lower is better with calibration metrics (CB-ECE, ECE and MSE). Statistically significant improvements (t-tests with Bonferroni correction at the 95% confidence level) over all baselines are marked with [†].

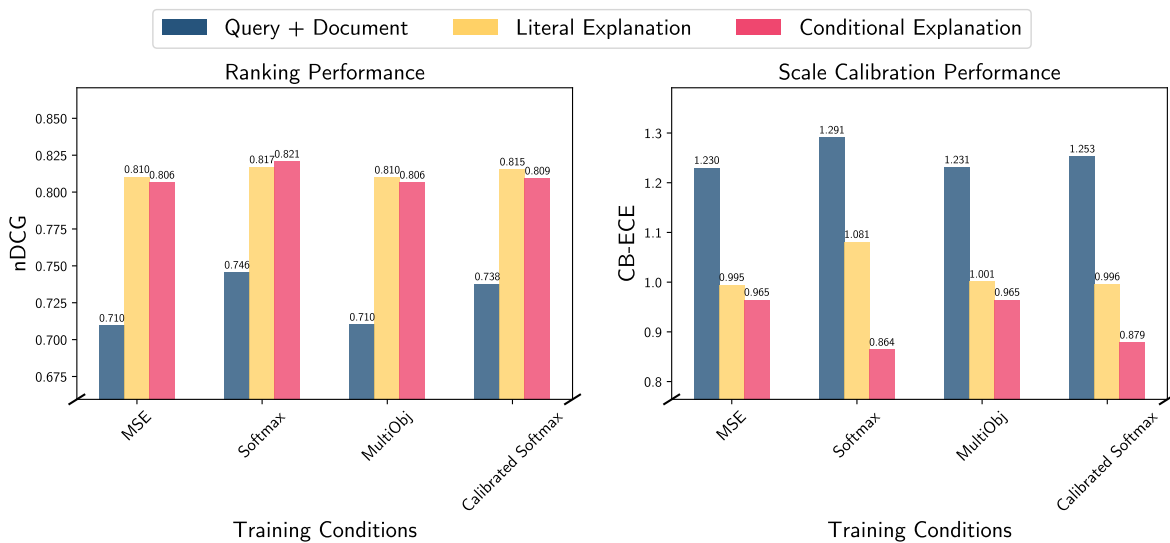


Figure 2: Ranking and scale calibration performance on TREC of full calibration of BERT, taking query + document inputs (FC BERT) and our proposed explanations, using four different optimization objectives. NLE-based approaches consistently yield better ranking (left) and calibration (right) performance.

by Arabzadeh et al. (2024), is to estimate the performance of individual queries such that hard-to-satisfy queries can be better addressed by search systems. We use two popular post-hoc QPP methods—Weighted Information Gain (WIG) (Zhou and Croft, 2007) and Normalized Query Commitment (NQC) (Shtok et al., 2012)—to explore whether the proposed NLE based approaches yield more usable scores. Evaluations are conducted on the test set of TREC, using the top-10 documents to calculate WIG and NQC.

4.5 Reproducibility

We leverage the open-source LLaMA2-13B-Chat model (Touvron et al., 2023) as the LLM and the BERT model (Devlin et al., 2019) as the neural ranker. Implementation details are described in Ap-

pendix D. The anonymized code repository can be found at <https://anonymous.4open.science/r/LLM-NLE-Ranking>.

4.6 Results and Analysis

RQ1: How do LLM-generated natural language explanations impact the calibration and ranking performance of neural rankers? We find in Table 2 that NLE-based methods yield statistically significant improvements for both aspects. For scale calibration, NLEs exhibit lower CB-ECE compared to both post-hoc calibration (PC) and full calibration (FC) across both datasets, with literal NLEs performing better on TREC while conditional NLEs perform slightly better on NTCIR. Regarding ranking, the proposed approach significantly surpasses NC and FC baselines, which

shares the same fine-tuning data and initial weights, with improvements of up to 11.4% in nDCG and 63.3% in nDCG@10 in the TREC dataset. This indicates that LLM-generated NLEs provide valuable insights for document differentiation beyond the scope of calibration.

RQ2: Is there a consistent improvement across different training objectives when using these explanations? Yan et al. (2022) and Bai et al. (2023) advocate for addressing the scale calibration of LTR models through calibrated loss functions. As a significant motivator for this work are the challenges of applying LTR approaches to larger text models, we investigate the efficacy of naively applying these objectives to the neural ranking task. To this end, we experiment with four fine-tuning objectives commonly used in this setting: MSE as a pointwise regression loss, uncalibrated listwise softmax (Softmax), a multi-objective combination of MSE and Softmax (MultiObj) from Bai et al. (2023), and the calibrated listwise softmax (Calibrated Softmax) from Yan et al. (2022). The FC baseline is also trained with each of these objectives. As seen in Fig. 2, the NLE-based approaches consistently surpass the performance of neural models that process raw text queries and documents across all four optimization methods. Further, while these objectives do produce different results, the majority of the improved calibration performance comes from the NLEs themselves and is relatively agnostic of the loss function.

RQ3: Does the MC sampled aggregate-NLEs from the LLM yield empirical improvements?

To further the understanding of the dynamics between LLM-generated NLEs and their impact on scale calibration and ranking, we perform additional experiments on the TREC dataset. We evaluate the decision to use an aggregate NLE rather than the most probable NLE (discussed in Section 3.4) in Table 3. We include an oracle baseline, where we repeatedly sample a NLE until we achieve one that aligns with the true relevance label. If this is not achieved within 20 samples, we default to the most confident NLE even if it’s incorrect. It can be observed from the results that, across both literal and conditional NLE settings, using the proposed aggregate MC method achieves better performance both w.r.t. ranking as well as calibration.

RQ4: Does the proposed methodology produce meaningful scores in a real-world setting? The results from using ranking scores from different

Explanation	Selection	nDCG(↑)	CB-ECE(↓)
Literal	Most Probable	0.789	1.093
	Aggregate MC	0.815[†]	0.996[†]
	Oracle	0.883	0.801
Conditional	Most Probable	0.797	0.895
	Aggregate MC	0.822[†]	0.862[†]

Table 3: The effect of different types of natural language explanations and selection strategies on the ranking and scale calibration performance of neural rankers.

Metric	WIG		NQC	
	P- ρ	K- τ	P- ρ	K- τ
NC	0.176	0.091	0.003	-0.083
PC	0.198	0.113	-0.173	-0.123
FC	-0.001	-0.012	-0.385	-0.242
PR	0.052	0.003	-0.002	-0.025
PL	0.019	0.057	-0.262	-0.101
NLE (lit.)	0.063	0.104	0.075	0.048
NLE (con.)	0.268	0.165	0.183	0.146

Table 4: The effect of scale calibration on post-hoc QPP.

calibration approaches for QPP are reported in Table 4, and we are able to definitively show that the scores produced via NLEs result in substantially more usable scores. Interestingly, directly adapting the calibrated ranking loss of Yan et al. (2022) from LTR to neural ranking (FC) leads to much worse QPP performance compared to no calibration (NC). PC, PR, and PL, though improving calibration metrics, do not consistently improve QPP either. Comparing between the literal and conditional NLE methods, there is a large gap in performance suggesting that including the two extreme perspectives of the conditional NLEs play a critical role. We highlight that while MSE, ECE and CB-ECE are often used in past works for ranking calibration, it leaves out critical information for the ranking task. We hope to investigate more effective metrics in future work.

5 Conclusions

This study tackled the challenge of scale calibration for neural rankers. Our method leverages zero-shot LLMs’ inherent understanding of textual data through NLEs to enhance scale calibration while maintaining or improving ranking performance. We showed that our approach surpasses established baselines in ranking and calibration metrics, demonstrates consistency across various training objectives, and is effective for downstream post-hoc QPP applications.

600 **6 Limitations**

601 This work prototypes the idea of leveraging LLM-
602 generated NLEs of query-document pairs to im-
603 prove the scale calibration of neural ranking mod-
604 els. Like many recent works that incorporate LLMs
605 as an integral part of the document ranking pro-
606 cess (Ma et al., 2023b; Qin et al., 2023; Zhuang
607 et al., 2024), our approach is not yet efficient
608 enough to be directly applied to an online rank-
609 ing system with high latency requirements. Having
610 validated the feasibility of NLEs for calibration,
611 future work could incorporate techniques such as
612 distillation to improve run-time efficiency.

613 Another limitation stems from the unreliability
614 and inconsistency of NLEs from LLMs (Ye and
615 Durrett, 2022). Although we paradoxically lever-
616 age the inconsistency of NLEs to further improve
617 the scale calibration of neural ranking models, the
618 potential effects of addressing unreliability or in-
619 consistency on the calibration task remain unclear.
620 Moreover, while we aim for the scale-calibrated
621 ranking scores to be useful for explainability pur-
622 poses as they correspond to certain criteria, the
623 NLEs themselves are not meant to be, nor should
624 they be directly displayed to users for interpretabil-
625 ity purposes in the current form.

626 A notable performance disparity is observed be-
627 tween our current aggregate MC approach and the
628 oracle explanation setting in Table 3. This gap
629 highlights a significant potential for improvement
630 in NLE-based neural rankers contingent upon bet-
631 ter alignment of LLM judgments with human an-
632 notators on a binary scale. Although we experi-
633 mented with various prompts for generating NLEs
634 in some pilot studies, this research is designed to
635 explore the feasibility of LLM-generated NLEs
636 for a specific important task. Consequently, vari-
637 ous directions that could potentially improve the
638 LLMs in terms of accuracy of predictions and qual-
639 ity of explanations—including but not limited to
640 few-shot prompting, prompt tuning and optimiza-
641 tion, LLM fine-tuning, and leveraging better (even
642 closed-source) LLMs—are beyond the interest and
643 scope of this study.

644 **7 Potential Risks**

645 Although calibrated ranking scores can be lever-
646 aged in certain ways to mitigate bias for fairer
647 document ranking (Heuss et al., 2023), the use of
648 NLEs acts as an information transformation, filter-
649 ing parts of the candidate document while adding

650 the LLM’s own biases. Certain topics, demograph-
651 ics, and perspectives could be impacted and conse-
652 quently change the proposed document rankings.
653 While traditional *ad-hoc* web search is transitioning
654 to retrieval augmented generation, the consequence
655 remains the same. These systems act as informa-
656 tion brokers for the general public, and can easily
657 introduce or reinforce biases into the population
658 based on the retrieved information. With this in
659 mind, we elected to use an off-the-shelf text rank-
660 ing paradigm as the final scorer. This facilitates the
661 drop in use of the many techniques already devel-
662 oped to improve fairness in ranking (Zehlike et al.,
663 2023a,b).

References

665
666
667
668
669

Negar Arabzadeh, Chuan Meng, Mohammad Alian-nejadi, and Ebrahim Bagheri. 2024. [Query Performance Prediction: From Fundamentals to Advanced Techniques](#). In *Advances in Information Retrieval*, pages 381–388, Cham. Springer Nature Switzerland.

670
671
672
673
674
675

Dara Bahri, Yi Tay, Che Zheng, Donald Metzler, and Andrew Tomkins. 2020. [Choppy: Cut Transformer for Ranked List Truncation](#). In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1513–1516, Virtual Event China. ACM.

676
677
678
679
680
681
682
683

Aijun Bai, Rolf Jagerman, Zhen Qin, Le Yan, Pratyush Kar, Bing-Rong Lin, Xuanhui Wang, Michael Bendersky, and Marc Najork. 2023. [Regression Compatible Listwise Objectives for Calibrated Ranking with Binary Relevance](#). In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 4502–4508, Birmingham United Kingdom. ACM.

684
685
686
687
688
689
690

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2018. [MS MARCO: A Human Generated MACHine Reading COMprehension Dataset](#). *arXiv preprint*. ArXiv:1611.09268 [cs].

691
692
693
694
695

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. [e-SNLI: Natural Language Inference with Natural Language Explanations](#). In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

696
697
698
699

Olivier Chapelle and Yi Chang. 2011. [Yahoo! Learning to Rank Challenge Overview](#). In *Proceedings of the Learning to Rank Challenge*, pages 1–24. PMLR. ISSN: 1938-7228.

700
701
702
703
704

Sougata Chaudhuri, Abraham Bagherjeiran, and James Liu. 2017. [Ranking and Calibrating Click-Attributed Purchases in Performance Display Advertising](#). In *Proceedings of the ADKDD'17*, pages 1–6, Halifax NS Canada. ACM.

705
706
707
708
709
710

Yanda Chen, Ruiqi Zhong, Narutatsu Ri, Chen Zhao, He He, Jacob Steinhardt, Zhou Yu, and Kathleen McKeown. 2023. [Do Models Explain Themselves? Counterfactual Simulatability of Natural Language Explanations](#). *arXiv preprint*. ArXiv:2307.08678 [cs].

711
712
713
714
715
716
717
718

Daniel Cohen, Bhaskar Mitra, Oleg Lesota, Navid Reksabsaz, and Carsten Eickhoff. 2021. [Not All Relevance Scores are Equal: Efficient Uncertainty and Calibration Modeling for Deep Retrieval Models](#). In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 654–664. ArXiv:2105.04651 [cs].

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. [Overview of the TREC 2019 deep learning track](#). *arXiv preprint*. ArXiv:2003.07820 [cs].

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, Ellen M. Voorhees, and Ian Soboroff. 2021. [TREC Deep Learning Track: Reusable Test Collections in the Large Data Regime](#). In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2369–2375, Virtual Event Canada. ACM.

Domenico Dato, Claudio Lucchese, Franco Maria Nardini, Salvatore Orlando, Raffaele Perego, Nicola Tonello, and Rossano Venturini. 2017. [Fast Ranking with Additive Ensembles of Oblivious and Non-Oblivious Regression Trees](#). *ACM Transactions on Information Systems*, 35(2):1–31.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Guglielmo Faggioli, Laura Dietz, Charles L. A. Clarke, Gianluca Demartini, Matthias Hagen, Claudia Hauff, Noriko Kando, Evangelos Kanoulas, Martin Potthast, Benno Stein, and Henning Wachsmuth. 2023a. [Perspectives on Large Language Models for Relevance Judgment](#). In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 39–50, Taipei Taiwan. ACM.

Guglielmo Faggioli, Thibault Formal, Stefano Marchesin, Stéphane Clinchant, Nicola Ferro, and Benjamin Piwowarski. 2023b. [Query Performance Prediction for Neural IR: Are We There Yet?](#) *arXiv preprint*. ArXiv:2302.09947 [cs].

Fernando Ferraretto, Thiago Laitz, Roberto Lotufo, and Rodrigo Nogueira. 2023. [ExaRanker: Explanation-Augmented Neural Ranker](#). *arXiv preprint*. ArXiv:2301.10521 [cs].

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. [On Calibration of Modern Neural Networks](#). *arXiv preprint*. ArXiv:1706.04599 [cs].

Maria Heuss, Daniel Cohen, Masoud Mansoury, Maarten de Rijke, and Carsten Eickhoff. 2023. [Predictive Uncertainty-based Bias Mitigation in Ranking](#). In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM '23*, pages 762–772, New York, NY, USA. Association for Computing Machinery.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk,

776	Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. 2022. Language Models (Mostly) Know What They Know . <i>arXiv preprint</i> . ArXiv:2207.05221 [cs].	<i>ACM SIGIR Conference on Research and Development in Information Retrieval</i> , pages 2583–2593, Taipei Taiwan. ACM.	830 831 832
786	Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT . In <i>Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval</i> , pages 39–48, Virtual Event China. ACM.	Sabrina J. Mielke, Arthur Szlam, Emily Dinan, and Y.-Lan Boureau. 2022. Reducing conversational agents’ overconfidence through linguistic calibration . <i>arXiv preprint</i> . ArXiv:2012.14983 [cs].	833 834 835 836
792	Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2022. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation .	Allan H. Murphy and Robert L. Winkler. 1977. Reliability of Subjective Probability Forecasts of Precipitation and Temperature . <i>Journal of the Royal Statistical Society Series C: Applied Statistics</i> , 26(1):41–47.	837 838 839 840
795	Thiago Laitz, Konstantinos Papakostas, Roberto Lotufo, and Rodrigo Nogueira. 2024. InRanker: Distilled Rankers for Zero-shot Information Retrieval . <i>arXiv preprint</i> . ArXiv:2401.06910 [cs].	Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT . In <i>arXiv:1901.04085 [cs]</i> . ArXiv: 1901.04085.	841 842 843
799	Lei Li, Yongfeng Zhang, and Li Chen. 2020. Generate Neural Template Explanations for Recommendation . In <i>Proceedings of the 29th ACM International Conference on Information & Knowledge Management, CIKM ’20</i> , pages 755–764, New York, NY, USA. Association for Computing Machinery.	Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document Ranking with a Pre-trained Sequence-to-Sequence Model . In <i>Findings of the Association for Computational Linguistics: EMNLP 2020</i> , pages 708–718, Online. Association for Computational Linguistics.	844 845 846 847 848 849
805	Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries . In <i>Text Summarization Branches Out</i> , pages 74–81, Barcelona, Spain. Association for Computational Linguistics.	Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019a. Multi-Stage Document Ranking with BERT . <i>arXiv preprint</i> . ArXiv:1910.14424 [cs].	850 851 852
809	Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching Models to Express Their Uncertainty in Words . <i>Transactions on Machine Learning Research</i> .	Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019b. Document Expansion by Query Prediction . Technical Report arXiv:1904.08375, arXiv. ArXiv:1904.08375 [cs] type: article.	853 854 855 856 857
812	Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization . <i>arXiv preprint</i> . ArXiv:1711.05101 [cs, math].	Gustavo Penha and Claudia Hauff. 2021. On the Calibration and Uncertainty of Neural Learning to Rank Models for Conversational Search . In <i>Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume</i> , pages 160–170, Online. Association for Computational Linguistics.	858 859 860 861 862 863 864
815	Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2023a. Fine-Tuning LLaMA for Multi-Stage Text Retrieval . <i>arXiv preprint</i> . ArXiv:2310.08319 [cs] version: 1.	John Platt. 2000. Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. <i>Adv. Large Margin Classif.</i> , 10.	865 866 867
819	Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. 2023b. Zero-Shot Listwise Document Reranking with a Large Language Model . <i>arXiv preprint</i> . ArXiv:2305.02156 [cs].	Tao Qin and Tie-Yan Liu. 2013. Introducing LETOR 4.0 Datasets . <i>arXiv preprint</i> . ArXiv:1306.2597 [cs].	868 869
823	Jiaxin Mao, Tetsuya Sakai, Cheng Luo, Peng Xiao, Yiqun Liu, and Zhicheng Dou. 2019. Overview of the NTCIR-14 We Want Web Task.	Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. 2023. Large Language Models are Effective Text Rankers with Pairwise Ranking Prompting . <i>arXiv preprint</i> . ArXiv:2306.17563 [cs].	870 871 872 873 874 875
826	Chuan Meng, Negar Arabzadeh, Mohammad Aliannejadi, and Maarten De Rijke. 2023. Query Performance Prediction: From Ad-hoc to Conversational Search . In <i>Proceedings of the 46th International</i>	Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S. Jaakkola, and Regina Barzilay. 2023. Conformal Language Modeling . <i>arXiv preprint</i> . ArXiv:2306.10193 [cs].	876 877 878 879
829		Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2021. ColBERTv2: Effective and Efficient Retrieval via Lightweight Late Interaction . Technical Report	880 881 882 883

884	arXiv:2112.01488, arXiv. ArXiv:2112.01488 [cs]	Manya Wadhwa, Jifan Chen, Junyi Jessy Li, and Greg	942
885	type: article.	Durrett. 2023. Using Natural Language Explanations to Rescale Human Judgments . <i>arXiv preprint</i> . ArXiv:2305.14770 [cs].	943
886	Anna Shtok, Oren Kurland, David Carmel, Fiana Raiber,		944
887	and Gad Markovits. 2012. Predicting Query Performance by Query-Drift Estimation . <i>ACM Transactions on Information Systems</i> , 30(2):11:1–11:35.		945
888		Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc	946
889		Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models . <i>arXiv preprint</i> . ArXiv:2203.11171 [cs].	947
890	Mark D. Smucker and Charles L.A. Clarke. 2012. Time-based calibration of effectiveness measures . In <i>Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval</i> , pages 95–104, Portland Oregon USA. ACM.		948
891			949
892			950
893		Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel,	951
894		Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022a. Emergent Abilities of Large Language Models . <i>Transactions on Machine Learning Research</i> .	952
895			953
896	Yukihiro Tagami, Shingo Ono, Koji Yamamoto, Koji		954
897	Tsukamoto, and Akira Tajima. 2013. CTR prediction for contextual advertising: learning-to-rank approach . In <i>Proceedings of the Seventh International Workshop on Data Mining for Online Advertising</i> , pages 1–8, Chicago Illinois. ACM.		955
898			956
899			957
900		Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten	958
901		Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022b. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models . <i>Advances in Neural Information Processing Systems</i> , 35:24824–24837.	959
902	Paul Thomas, Seth Spielman, Nick Craswell, and		960
903	Bhaskar Mitra. 2023. Large language models can accurately predict searcher preferences . <i>arXiv preprint</i> . ArXiv:2309.10621 [cs].		961
904			962
905			963
906	Katherine Tian, Eric Mitchell, Allan Zhou, Archit	Eric W. Weisstein. Bonferroni Correction . Publisher: Wolfram Research, Inc.	964
907	Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea		965
908	Finn, and Christopher D. Manning. 2023. Just Ask for Calibration: Strategies for Eliciting Calibrated Confidence Scores from Language Models Fine-Tuned with Human Feedback . <i>arXiv preprint</i> . ArXiv:2305.14975 [cs].		966
909		Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie	967
910		Fu, Junxian He, and Bryan Hooi. 2023. Can LLMs Express Their Uncertainty? An Empirical Evaluation of Confidence Elicitation in LLMs . <i>arXiv preprint</i> . ArXiv:2306.13063 [cs].	968
911			969
912			970
913	Hugo Touvron, Louis Martin, Kevin Stone, Peter Al-	Le Yan, Zhen Qin, Xuanhui Wang, Michael Bender-	971
914	bert, Amjad Almahairi, Yasmine Babaei, Nikolay	sky, and Marc Najork. 2022. Scale Calibration of Deep Ranking Models . In <i>Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining</i> , pages 4300–4309, Washington DC USA. ACM.	972
915	Bashlykov, Soumya Batra, Prajwal Bhargava, Shruti		973
916	Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton		974
917	Ferrer, Moya Chen, Guillem Cucurull, David Esiobu,		975
918	Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller,		976
919	Cynthia Gao, Vedanuj Goswami, Naman Goyal, An-		977
920	thony Hartshorn, Saghar Hosseini, Rui Hou, Hakan		978
921	Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa,		979
922	Isabel Kloumann, Artem Korenev, Punit Singh Koura,		
923	Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Di-		
924	ana Liskovich, Yinghai Lu, Yuning Mao, Xavier Mar-		
925	tinet, Todor Mihaylov, Pushkar Mishra, Igor Moly-		
926	bog, Yixin Nie, Andrew Poulton, Jeremy Reizen-		
927	stein, Rashi Rungta, Kalyan Saladi, Alan Schel-		
928	ten, Ruan Silva, Eric Michael Smith, Ranjan Sub-		
929	ramanian, Xiaoqing Ellen Tan, Binh Tang, Ross		
930	Taylor, Adina Williams, Jian Xiang Kuan, Puxin		
931	Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, An-		
932	gela Fan, Melanie Kambadur, Sharan Narang, Aure-		
933	lien Rodriguez, Robert Stojnic, Sergey Edunov, and		
934	Thomas Scialom. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models . <i>arXiv preprint</i> . ArXiv:2307.09288 [cs].		
935			
936			
937	Miles Turpin, Julian Michael, Ethan Perez, and		
938	Samuel R. Bowman. 2023. Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting . <i>arXiv preprint</i> . ArXiv:2305.04388 [cs].		
939			
940			
941			
		Meike Zehlike, Ke Yang, and Julia Stoyanovich. 2023a. Fairness in Ranking, Part I: Score-Based Ranking . <i>ACM Computing Surveys</i> , 55(6):1–36.	993
			994
			995

996	Meike Zehlike, Ke Yang, and Julia Stoyanovich. 2023b.	to their use of a fixed set of 7 examples as demon-	1046
997	Fairness in Ranking, Part II: Learning-to-Rank and	strations for few-shot prompting.	1047
998	Recommender Systems . <i>ACM Computing Surveys</i> ,	The prompt we use for the conditional explana-	1048
999	55(6):1–41.	tion approach is as follows:	1049
1000	George Zerveas, Navid Rekabsaz, Daniel Cohen, and	“For the following query and document, explain	1050
1001	Carsten Eickhoff. 2022. Mitigating Bias in Search	why they are {relevant/nonrelevant}. Query:	1051
1002	Results Through Contextual Document Reranking	{query} Document: {document} Output:”	1052
1003	and Neutrality Regularization . In <i>Proceedings of</i>		
1004	<i>the 45th International ACM SIGIR Conference on</i>		
1005	<i>Research and Development in Information Retrieval</i> ,		
1006	pages 2532–2538, Madrid Spain. ACM.		
1007	Yun Zhou and W. Bruce Croft. 2007. Query perfor-	B Details of Competing Methods	1053
1008	mance prediction in web search environments . In	Here we provide more details and explanations for	1054
1009	<i>Proceedings of the 30th annual international ACM</i>	competing methods evaluated in the experiments.	1055
1010	<i>SIGIR conference on Research and development in</i>	No calibration (NC) . Our approach here employs	1056
1011	<i>information retrieval</i> , pages 543–550, Amsterdam	a BERT-based ranker that has been previously fine-	1057
1012	The Netherlands. ACM.	tuned using the MS MARCO dataset (Bajaj et al.,	1058
1013	Honglei Zhuang, Zhen Qin, Kai Hui, Junru Wu, Le Yan,	2018). Despite its strong performance in ranking	1059
1014	Xuanhui Wang, and Michael Berdersky. 2023a. Be-	tasks on TREC (in-domain) and NTCIR (out-of-	1060
1015	yond Yes and No: Improving Zero-Shot LLM	domain), this ranker lacks scale calibration. This	1061
1016	Rankers via Scoring Fine-Grained Relevance Labels .	is attributed to the binary relevance labels in MS	1062
1017	<i>arXiv preprint</i> . ArXiv:2310.14122 [cs].	MARCO, which differ from the multi-level rele-	1063
1018	Honglei Zhuang, Zhen Qin, Rolf Jagerman, Kai Hui,	levance judgments used in TREC and NTCIR.	1064
1019	Ji Ma, Jing Lu, Jianmo Ni, Xuanhui Wang, and	Post-hoc calibration (PC) . In this method, we	1065
1020	Michael Bendersky. 2023b. RankT5: Fine-Tuning	adjust the output scores of the finetuned rankers	1066
1021	T5 for Text Ranking with Ranking Losses . In <i>Pro-</i>	using a learnable function. Following the approach	1067
1022	<i>ceedings of the 46th International ACM SIGIR Con-</i>	of Yan et al. (2022) , we apply Platt scaling (Platt,	1068
1023	<i>ference on Research and Development in Information</i>	2000), adapted for regression calibration. Given the	1069
1024	<i>Retrieval, SIGIR ’23</i> , pages 2308–2313, New York,	output ranking scores of the model s , the calibrated	1070
1025	NY, USA. Association for Computing Machinery.	scores are computed as $s' = \exp(ws + b)/2$,	1071
1026	Shengyao Zhuang, Honglei Zhuang, Bevan Koopman,	where w and b are learnable parameters. It is impor-	1072
1027	and Guido Zuccon. 2024. A Setwise Approach for	tant to note that under this method, the parameters	1073
1028	Effective and Highly Efficient Zero-shot Ranking	of the finetuned rankers remain fixed; only the pa-	1074
1029	with Large Language Models . ArXiv:2310.09497	rameters w and b are optimized using the scale	1075
1030	[cs].	calibration data. This approach maintains the orig-	1076
1031	A Prompts Used in Experiments	inal ranking performance of the ranker, provided	1077
1032	The format of the prompt ⁴ we employ for literal	the final value of w is positive.	1078
1033	explanation is as follows:	Full calibration (FC) . This strategy involves	1079
1034	“For the following query and document, judge	directly fine-tuning a BERT-based ranker with	1080
1035	whether they are relevant or non-relevant, and pro-	“query-[SEP]-document” style inputs on scale cali-	1081
1036	vide an explanation. Output ‘Relevant’ or ‘Nonrel-	bration data. Utilizing the monoBERT checkpoint	1082
1037	evant’. Do not repeat the content of the query or	allows for a direct comparison with PC methods -	1083
1038	the document. Query: {query} Document: {docu-	they use the same initial checkpoint and training	1084
1039	ment} Output:”	data, but optimize different parameters. Starting	1085
1040	This method parallels the prompt used by Fer-	with a general-purpose BERT checkpoint sets up a	1086
1041	raretto et al. (2023) to generate explanations for	direct comparison with our NLE-based calibration	1087
1042	query-document relevance for training generative	approaches - they share the same initial weights	1088
1043	rankers. However, our approach differs due to	and training data, but the format of their input data	1089
1044	its simplicity and broader generalizability across	differs significantly.	1090
1045	datasets, utilizing zero-shot prompting in contrast	LLM prompting with rubrics (PR) . The underly-	1091
		ing concept is that the LLM should be able to com-	1092
		prehend the given rubrics and automatically pro-	1093
		duced calibrated scores that align with the rubrics,	1094

⁴We experimented with different prompts, but stick to this one for clear performance gains and ease of formatting.

thereby removing the need for any post-hoc calibration steps. It is important to note that in this method, the LLM itself essentially functions as the ranker, without the integration of an additional neural ranking model. For the TREC dataset, we adopt the prompt method used by Thomas et al. (2023). Conversely, for NTCIR, we adapt the approach to incorporate the specific rubric outlined in the task description of NTCIR-14 WWW-2 (Mao et al., 2019). It is noteworthy that the NTCIR rubric is defined based on scores provided by two annotators (e.g., “relevance=3: One annotator rated as highly relevant, one as relevant”), focusing on the quantitative synthesis of annotations. In contrast, the TREC rubric is more qualitatively oriented, emphasizing the explanation of query-document relationships.

Post-hoc calibrating LLM predictions (PL). In this method, we use the LLM as a zero-shot classifier to determine whether a given query and document pair is relevant (denoted as 1) or not (denoted as 0). To mitigate the bias inherent in the most probable generation and to minimize instances of tied scores, which complicate the derivation of rankings, we sample the LLM’s responses 20 times for each input and calculate the average of these scores. Subsequently, we employ Platt scaling to these averaged scores and refine the parameters using the training set. The prompt we use is similar to that of Zhuang et al. (2023a), with one significant modification: we instruct the LLM to output either “relevant” or “nonrelevant” in lieu of “yes” or “no”. This alteration stems from our observation that the LLM exhibits a strong prior towards generating affirmative responses such as “Yes, I can help you with this request...”, which could potentially skew the predictions. By specifying the terms “relevant” and “nonrelevant”, we aim to reduce this bias and achieve more accurate relevance predictions.

Full calibration with NLEs (NLE). Building upon our methods for generating and aggregating natural language explanations (NLEs) for query-document pairs, as discussed in Section 3, we proceed to fine-tune a BERT model (not finetuned for retrieval) to process meta NLEs and yield scale-calibrated ranking scores. In the scenario of the conditional explanation approach (Section 3.3.2), where each input is represented with two meta NLEs, one for relevance and one for non-relevance, our method involves an additional processing step. Specifically, we concatenate the [CLS] hidden states obtained from encoding both NLEs. This concate-

nated representation then feeds into an additional linear layer, which is responsible for transforming these combined hidden states into a final ranking score. This approach allows the model to integrate insights from both relevance perspectives.

C Class-balanced ECE and the limitations of ECE

We discuss in Section 4.2 the limitations inherent in using the expected calibration error (ECE) (Guo et al., 2017) for scale calibration data, especially when faced with imbalanced label distributions. To counter this challenge, we propose a straightforward modification: the adoption of class-balanced ECE (CB-ECE). This section uses model outputs on TREC as illustrative examples to provide detailed insights and empirical support for our approach.

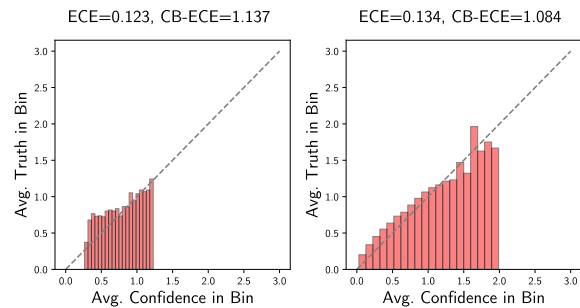


Figure 3: Reliability diagrams for two models on TREC: The left diagram shows a model with ranking scores densely concentrated on the lower part of the scale, which exhibits better ECE performance due to ECE’s failure to account for prediction coverage across the target scale. On the right, the CB-ECE penalizes this undesirable behavior, indicating that the model providing better coverage across the scale is more effectively calibrated.

We employ reliability diagrams (Murphy and Winkler, 1977), a widely-used technique in calibration, to visualize calibration performance. In ECE calculations, samples are sorted by their predicted values and divided into several buckets. The calibration error for each bucket is determined by the absolute difference between the mean labels and mean predictions of samples within that bucket, further *weighted* by the number of samples falling in the bucket. Reliability diagrams concretely visualize this concept, with each bucket represented by a bar. The mean predictions are indicated on the x-axis, while the mean labels (i.e., scale) are

represented by the height of the bars. In an ideal scenario, a perfectly calibrated model would yield a reliability diagram aligning with the diagonal line, indicating that mean predictions correspond accurately to the labels for every bucket. However, ECE overlooks a critical aspect of scale calibration: **coverage**. This refers to the model’s ability to adequately span the target scale range. Figure 3 illustrates this point with two diagrams. The left diagram shows a model whose outputs are narrowly concentrated around lower values on TREC’s 0-3 scale. The right diagram, conversely, exhibits outputs covering a broader range on the target scale, also closely aligned with the diagonal line. Relying solely on ECE values would misleadingly suggest that the first model is better calibrated (ECE: $0.123 < 0.134$; lower is better). However, we argue that a model with predictions confined to a limited portion of the desired scale, particularly skewed towards certain values popular in the evaluation set, cannot be considered well-calibrated. CB-ECE offers a solution by re-adjusting the significance of each target scale. In the example shown in Fig. 3, CB-ECE more accurately indicates that the second model exhibits better scale calibration (CB-ECE: $1.137 > 1.084$; lower is better), which is more in line with our intuitive understanding of model calibration.

D Implementation Details

For all experiments that involve LLM, we employ the LLaMA2-13B-Chat model (Touvron et al., 2023), hosted locally through vLLM⁵ and using quantized weights⁶, operated on an A100 GPU. In this study, our neural rankers are based on BERT (Devlin et al., 2019). We initialize these rankers with weights fine-tuned on MS MARCO, referred to as monoBERT⁷ in Table 2, or with weights without retrieval-oriented fine-tuning, simply denoted as BERT⁸. For fine-tuning on scale calibration data, we use the AdamW optimizer (Loshchilov and Hutter, 2019) with a learning rate of $3e^{-6}$ and effective batch size of 64 query-document pairs or meta NLEs.⁹ The training is

⁵<https://github.com/vllm-project/vllm>

⁶<https://huggingface.co/TheBloke/Llama-2-13B-chat-AWQ>

⁷<https://huggingface.co/veneres/monobert-msmarco>

⁸<https://huggingface.co/bert-base-uncased>

⁹Note that due to larger model size and more complex input representations, the batch size for fine-tuning neural ranking models is much smaller than training LTR models.

conducted over a maximum of 10 epochs, selecting the best model based on validation set loss. To mitigate the impact of randomness due to the limited dataset size in terms of query numbers, each experiment is run with 5 different random seeds. The metrics reported are averaged across these five runs and the statistical significance is determined using t-tests with Bonferroni correction (Weisstein) at the 95% confidence level.

In relation to the components and hyperparameters used in Algorithm 1, following Quach et al. (2023), we employ ROUGE-L (Lin, 2004) as the text similarity function \mathcal{S} , set the similarity threshold λ to 0.35, the sampling budget k_l (maximum number of responses) to 20 and k_s (maximum number of sentences in the meta NLE) to 30.

We use PyGaggle¹⁰ for acquiring the ranking scores from pre-trained neural ranking models, which are then used for the post-hoc calibration baseline. We use WIG and NQC’s implementations in QPP4CS (Meng et al., 2023)¹¹ for the downstream evaluation of QPP.

E Case Study

We randomly show a pair of query and document from TREC, as well as LLM-generated explanations for them in order to predict calibrated ranking scores. The example is shown in Table 5. The superior effectiveness of our methods in assessing the relevance of the query “example of what a family advocate does” to a specific document can be attributed to several aspects. Firstly, the use of NLEs generated by an LLM provides a deeper contextual understanding. For instance, the first approach’s explanation elucidates the direct relevance by highlighting how the document exemplifies the role of a family advocate, particularly in advocating for a child’s best interests. This detailed breakdown goes beyond mere textual similarity, offering a richer interpretation of the document’s relevance to the query. Furthermore, our second approach’s dual perspective in explaining both relevance and non-relevance brings a comprehensive understanding. It captures the essence of the document’s role description and contrasts it with the lack of specific examples, thereby offering a deeper analysis of the document-query relationship. By feeding these elaborate explanations into BERT, our methods effectively leverage more informative data, allowing

¹⁰<https://github.com/castorini/pygaggle>

¹¹<https://github.com/ChuanMeng/QPP4CS>

Query: “Example of what a family advocate does”
Document: “This person advocates for the best interest of the child, and is tasked with investigating the family situation and advising the court what custody situation would be in the best interests of the child.”
Label: 2 (Highly relevant: The passage has some answer for the query, but the answer may be a bit unclear, or hidden amongst extraneous information).

Approach	Explanation	Score
Uncalibrated monoBERT	N/A	0.872
Calibrated monoBERT	N/A	0.397
Literal Explanation	here’s my answer: relevant the query asks for an example of what a family advocate does, and the document provides an example of a family advocate’s role, specifically advocating for the best interests of the child and investigating the family situation to advise the court on the appropriate custody arrangement. this is directly related to the query and the document provides relevant information. the document provides a clear example of what a family advocate does, making it relevant to the query. the document directly addresses the query’s question, making it a relevant match. sure, i can help you with that!	1.980
Conditional Explanation	<p>Relevance: the query “example of what a family advocate does” is relevant to the provided document because the document describes the role of a family advocate, which is to advocate for the best interests of the child in family law cases. the document highlights the key aspects of a family advocate this is an example of what a family advocate might do in practice. this connection between the query and document makes them relevant to each other. 3. focus on the child’s best interests: the document emphasizes the importance of prioritizing the child’s best interests when making decisions about child custody.</p> <p>Non-Relevance: the query “example of what a family advocate does” is not relevant to the document provided because the document does not provide an example of what a family advocate does, but rather describes the responsibilities of a family advocate. the document states that a family advocate advocates for the best interests of the child and investigates the family situation to advise the court on the custody situation that would be in the best interests of the child. this is a description of the role and responsibilities of a family advocate, but it does not provide an example of what a family advocate does in a specific situation.</p>	2.053

Table 5: An example showing natural language explanations generated by the LLM lead to better calibrated scores.

1267 for a more refined and accurate relevance assess-
1268 ment. Consequently, our methods demonstrate a
1269 notable improvement in capturing the subtleties
1270 and depths of relevance, as evidenced by the scores
1271 closer to the label compared to the baseline ap-
1272 proaches.