Parameter-Efficient Score Calibration for Text-Image Retrieval: Interpretable Thresholding with Vision-Language Models

Taku Fujitomi, Naoya Sogi, Takashi Shibata, and Makoto Terao
Visual Intelligence Research Laboratories
NEC Corporation
Kanagawa, Japan
{taku-fujitomi, naoya-sogi}@nec.com, t.shibata@ieee.org, m-terao@nec.com

***Abstract*—Embedding-based retrieval with Vision-Language Models (VLMs) enables high-performance text-image retrieval by projecting queries and candidates into a shared vector space and ranking them via cosine similarity. However, setting relevance thresholds manually requires effort and lacks adaptability to differences across users, queries, and models. To overcome this, we introduce a fast, lightweight score calibration method that normalizes cosine similarity scores using a formula based on the mean of estimated non-relevant candidates. This approach lessens sensitivity to candidate set size and relevance ratio. By interpreting the calibrated scores as relevance probabilities, we can estimate precision and recall at any threshold, making it possible for users to intuitively specify preferred metrics and directly control results. Experiments on diverse datasets show that our calibration method improves text-image relevance classification at global thresholds and delivers more stable retrieval quality according to user-specified criteria, allowing intuitive and consistent control over the relation between threshold and retrieval performance.**

***Index Terms*—score calibration, vision-language models, text-image retrieval, embedding-based retrieval.**

I. Introduction

Recently, embedding-based retrieval using Vision-Language Models (VLMs) [1]–[4] has received significant attention as an approach for ranking query-candidate pairs by embedding both into a shared vector space and computing the relevance score either via cosine similarity or classification models. This framework enables high performance across modalities, and is particularly prevalent in text-image retrieval tasks, where relevant images are fetched based on arbitrary text queries [1], [5]–[7].

For practical text-image retrieval systems, it is common to display images to users in descending order of relevance scores. To prevent showing completely irrelevant images, threshold-based filtering—using empirically determined thresholds—is generally employed [8]. However, this manual threshold adjustment is challenging for inexperienced users and is sensitive to changes in query, retrieval model, or other factors, often requiring extensive tuning efforts.

To address this issue, various methods have been proposed for automatic threshold estimation based on the relevance score distribution [9]–[16]. These approaches often optimize a fixed evaluation metric (such as the F1 score) and thereby alleviate the burden of threshold selection. Nonetheless, users may have diverse requirements—such as prioritizing precision or recall depending on their specific use case—rendering rigid, metric-dependent thresholding insufficiently adaptable.

Furthermore, VLMs tend to produce score distributions that substantially fluctuate from query to query. Utilizing a query-independent fixed threshold can result in significant variance in the quality of returned lists. In textual retrieval, score calibration via neural network models based on query and cosine similarity has been proposed to mitigate such instability [8], but these require expensive per-candidate online inference. Moreover, available text-image pair datasets are limited in query diversity, increasing the risk of overfitting for large-parameter neural models.

In this work, we propose a fast, parameter-efficient score calibration method that performs per-query calibrations by taking into account the candidate score distribution of the VLM and supports effective candidate filtering with a single, global threshold even in online scenarios. Inspired by the batchwise softmax treatment of cosine similarities in InfoNCE used as a loss function for VLMs, we design a novel calibration method. Our method adjusts the softmax denominator at inference time based on the score distribution, making the calibrated scores independent of the candidate set size and the relevant data ratio. Our method tunes only three parameters, preselected via Bayesian optimization to minimize cross-entropy error. This compact formulation enables fast score calibration while mitigating overfitting under limited labeled queries.

As shown in Fig. 1, we further propose a score *"re"calibration* step on top of the calibrated scores. Given a user-specified target metric (e.g., precision or recall), recalibration maps each calibrated score to the *estimated metric value (recalibrated score)*. Consequently, a threshold on the recalibrated scores directly corresponds to the desired operating point, enabling intuitive control of the retrieved set.

Comprehensive evaluations on multiple captioned image datasets and object detection datasets show that our method improves relevant image classification at query-independent global fixed thresholds and that our score recalibration yields

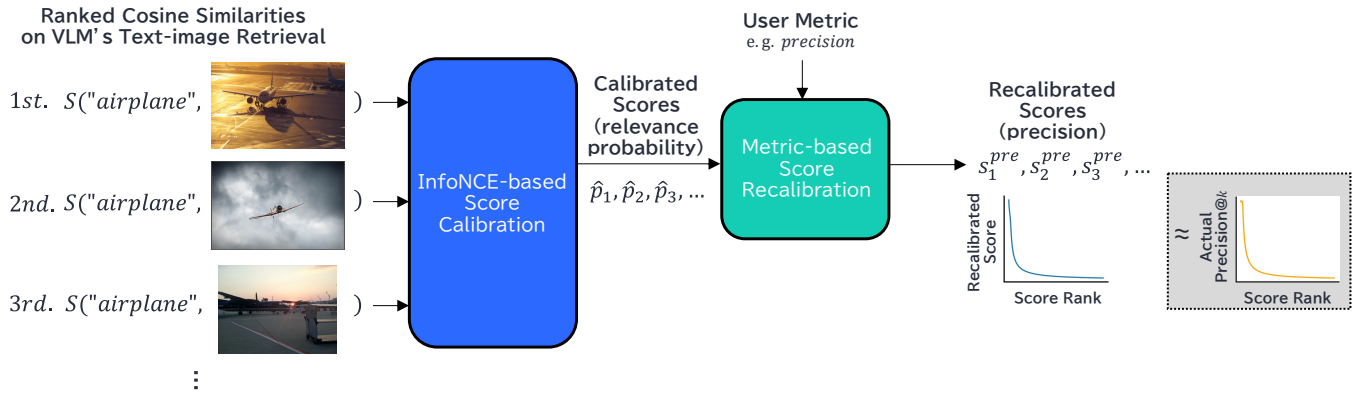

Fig. 1. Overview of our score calibration and recalibration method.

estimated metrics that are close to user-specified criteria for precision or recall. This provides clear and consistent control over the mapping from thresholds to retrieval quality as measured by user metrics.

Our code is available at https://github.com/nec-fujitomi/pesc-retrieval.

II. Related Work

A. Text-Image Retrieval and Vision-Language Models

Recent text-image retrieval systems are primarily built on strong Vision-Language Models (VLMs) such as CLIP [3], BLIP [4], [17], and SigLIP [18], [19]. These models project images and text into a shared space, allowing for fast retrieval based on similarity scores. However, in real-world applications, simply ranking items is often not enough. Systems need to decide where to stop or truncate the results to provide only the relevant items to the user. Finding the optimal cut-off point k is essential to balance precision and recall, yet this remains a challenge in the multi-modal field.

B. Ranked List Truncation

The problem of where to cut a ranked list has a long history in information retrieval [10], [11]. Traditional methods often use statistical models to solve this. For example, some approaches assume that scores for relevant and non-relevant items follow specific distributions, such as a mixture of Normal and Exponential distributions [12], [13], [15]. By modeling these distributions, they estimate a threshold that separates relevant from non-relevant. However, modern VLMs create complex embedding spaces where these simple statistical assumptions often do not hold.

To solve this, researchers have recently turned to machine learning. Methods like BiCut [14] and AttnCut [16] use neural networks to learn truncation points without relying on fixed statistical rules. While these models are flexible and perform well, they require a large amount of labeled data to train. In text-image retrieval, getting enough high-quality labels for every query is often too expensive or impractical.

C. Score Calibration

Another direction is score calibration, where scores are calibrated (e.g., via neural models) instead of thresholded directly [8]. While this preserves per-query ranking and improves filtering by a global threshold, such methods demand per-query neural inference and large training resources, and due to insufficient data their applicability in text-image retrieval is limited.

In contrast, our method provides lightweight score calibration without query-dependent neural models, thereby combining the adaptability of calibration with the efficiency required for online multimodal retrieval.

III. Method

This section introduces a novel, fast, and parameter-efficient score calibration that normalizes cosine similarities using the empirical distribution of candidate scores. We also describe the recalibration method that maps calibrated scores (relevance probabilities) to scores aligned with the precision and recall metrics. An overview of the method is shown in Fig. 1.

A. Softmax calibration of cosine similarity

The standard InfoNCE loss applies a temperature-scaled softmax to cosine similarities in the batch, normalizing them for cross-entropy loss computation. Let T denote a query text and I_n denote a candidate image. A calibration parameterized by the temperature τ is applied to the cosine similarity $S(T, I_n)$ as follows:

$$\hat{S}(T, I_n; \tau) = \exp\left(\frac{S(T, I_n)}{\tau}\right). \tag{1}$$

Under InfoNCE for text-to-image retrieval, the probability p_n^{t2i} for candidate n in a batch of size N_b is given by:

$$p_n^{t2i} = \frac{\hat{S}(T, I_n; \tau)}{\sum_{m=1}^{N_b} \hat{S}(T, I_m; \tau)}. \tag{2}$$

This batch-wise softmax calibration evaluates each candidate's score relative to others in the batch, and optimizing

it with cross-entropy error results in a globally calibrated similarity measure.

B. Issues with softmax calibration in inference

This candidates' probability p_n^{t2i} provides a reasonable approximation of relevance probability, but applying batchwise softmax normalization to an inference-time candidate pool by replacing N_b with the number of candidates N_c poses two key problems:

- **Dependence on candidate set size:** As the number of candidates N_c increases, the growing denominator suppresses the Softmax probability p_n^{t2i}, and direct comparison of scores across different candidate pools becomes invalid for fixed thresholds.
- **Dependence on relevant data ratio:** If multiple relevant data points exist within the candidate pool, p_n^{t2i} is suppressed after calibration even when the data is a true positive.

Both of these issues can be mitigated by rewiring the normalization to depend on the mean of the non-relevant candidate scores.

C. Improved score calibration based on non-relevant mean

To eliminate the dependence of the softmax denominator at inference time on the candidate set size and the relevant data ratio, we redefine the calibration formula using relevance labels.

Assume relevance labels t_m are known for all candidates, with $t_m = 1$ for relevant and $t_m = 0$ for non-relevant samples. The mean of the non-relevant data is defined as follows:

$$M(T, I_{1:N_c}, t_{1:N_c}; \tau) = \frac{\sum_{m=1}^{N_c}(1-t_m)\hat{S}(T, I_m; \tau)}{\sum_{m=1}^{N_c}(1-t_m)}. \quad (3)$$

Then, for each candidate I_n, the calibrated score is:

$$p_n = \frac{\hat{S}(T, I_n; \tau)}{\hat{S}(T, I_n; \tau) + (\beta - 1)\, M(T, I_{1:N_c}, t_{1:N_c}; \tau)}, \quad (4)$$

where β is a weighting parameter. In InfoNCE, only one relevant example is present in the batch; in such cases, setting $\beta = N_b$ recovers the original softmax as in (2) for relevant data.

D. Practical score calibration via estimated non-relevant probabilities

In inference scenarios, relevance labels t_m are unknown. Hence, we estimate the non-relevant mean M by modeling non-relevance probabilities.

1) Estimating non-relevant probabilities: Given the inherent negative correlation between cosine similarity and non-relevance probability, we invert and scale the similarity $S(T, I_n)$ with a shift α and temperature τ, mapping it through a sigmoid function:

$$f(T, I_n, \alpha, \tau) = \mathrm{Sigmoid}\left(-\frac{S(T, I_n) - \alpha}{\tau}\right). \quad (5)$$

This $f(\cdot)$ serves as an estimate of the probability that candidate I_m is non-relevant.

2) Estimating non-relevant mean: Next, we compute the non-relevant mean $\hat{M}$ as a weighted average of $\hat{S}$, weighted by the estimated non-relevance probabilities:

$$\hat{M}(T, I_{1:N_c}; \alpha, \tau) = \frac{\sum_{n=1}^{N_c} f(T, I_n; \alpha, \tau)\, \hat{S}(T, I_n; \tau)}{\sum_{n=1}^{N_c} f(T, I_n; \alpha, \tau)}. \quad (6)$$

3) Calibration score computation: Finally, using $\hat{M}$ instead of M in (4), we compute the calibration score for candidate I_n as:

$$\hat{p}_n = \frac{\hat{S}(T, I_n; \tau)}{\hat{S}(T, I_n; \tau) + (\beta - 1)\, \hat{M}(T, I_{1:N_c}; \alpha, \tau)}. \quad (7)$$

The parameters α, β, and τ are determined by optimizing to minimize cross-entropy error on a training set. For VLM scores trained with InfoNCE, β is set to N_b to efficiently optimize the remaining parameters.

E. Score recalibration for comparison with evaluation metrics

Since minimizing the cross-entropy error is equivalent to minimizing the Kullback-Leibler (KL) divergence [20], the calibrated scores can be interpreted probabilistically. Although applying a threshold to these scores enables reliable sampling, global metrics such as precision and recall on the retrieved set cannot be directly controlled by a single threshold on relevance probability.

To resolve this, we exploit estimation techniques that use relevance probabilities to infer the count of relevant samples [12], and recalibrate every candidate's score as the estimated metric value achieved by choosing it as a threshold. In effect, for any candidate, its score after recalibration can be interpreted as the precision or recall of the corresponding filtered set.

Specifically, let θ denote a threshold and define the four elements of the confusion matrix as:

$$\mathrm{TP}_\theta = \sum_{\substack{m=1 \\ \hat{p}_m \geq \theta}}^{N_c} \hat{p}_m, \quad \mathrm{FN}_\theta = \sum_{\substack{m=1 \\ \hat{p}_m < \theta}}^{N_c} \hat{p}_m,$$
$$\mathrm{FP}_\theta = \sum_{\substack{m=1 \\ \hat{p}_m \geq \theta}}^{N_c} (1 - \hat{p}_m), \quad \mathrm{TN}_\theta = \sum_{\substack{m=1 \\ \hat{p}_m < \theta}}^{N_c} (1 - \hat{p}_m). \quad (8)$$

Then, for each candidate n, the estimated precision and recall at threshold $\hat{p}_n$ are computed as:

$$s_n^{pre} = \frac{TP_{\hat{p}_n}}{TP_{\hat{p}_n} + FP_{\hat{p}_n}}, \quad s_n^{rec} = \frac{TP_{\hat{p}_n}}{TP_{\hat{p}_n} + FN_{\hat{p}_n}}. \quad (9)$$

By adopting s_n^{pre} or s_n^{rec} as the recalibrated score, we enable direct comparison and selection of thresholds by the target metric, supporting consistent control over retrieval result properties aligned with user needs.

IV. Experiments

This section explains our experimental setup and results. We introduce the datasets, evaluation metrics, implementation details, and report key findings.

A. Datasets

We use three types of datasets. For standard text-image retrieval, where captions act as queries, we use MS COCO [21] and Flickr30k [22]. Each caption corresponds to one relevant image in the candidate set.

To test cases with multiple relevant candidates per query, we use object class labels from COCO and incident class labels from Incidents1M [23]. Here, class names serve as queries, and all images containing the object or incident are treated as relevant.

For parameter optimization, we use object synset labels from Visual Genome [24]. We select labels that appear in at least 1% of candidate images and use their names as queries.

B. Evaluation Metrics

We evaluate binary classification with global fixed thresholds. Metrics are Precision-Recall AUC (PR AUC) and Precision at 95% Recall (P@R95%) as in the previous work [8]. Since our method preserves query-level ranking, these scores do not change within a query. Therefore, we focus on performance under global thresholds across all queries, resulting in the overall metrics varying by score calibration methods.

We also test score recalibration (Sec. III-E). We compute the Mean Absolute Error (MAE) between a target threshold and the actual precision/recall at the threshold, excluding cases that is impossible to achieve due to the VLM's ranking capability. Thresholds range from 0.1 to 0.9 in steps of 0.1. We then average errors over all queries.

C. Implementation Details

For parameter tuning, we apply Bayesian optimization on Visual Genome queries [24], minimizing cross-entropy loss. Cosine similarities are computed using pretrained models: CLIP [3] (arch: clip_feature_extractor, type: ViT-B-16), BLIP-2 [4] (arch: blip2_image_text_matching, type: coco) on LAVIS [25] and SigLIP 2 [19] (google/siglip2-base-patch16-naflex) on Hugging Face. For CLIP and BLIP-2, the β parameter is fixed to the pretraining batch size. Parameters used in experiments are listed in Table I.

Since no prior non-neural calibration methods exist for text-image retrieval, we adapt one from document retrieval: the normal-exponential mixture model [11]. It assumes relevant scores follow a normal distribution and non-relevant scores an exponential distribution. In this method, we conducted 100 trials of Expectation-Maximization (EM) algorithm [26] with a maximum of 100 steps each, and adopted the distribution parameters with the highest likelihood among the optimization results.

TABLE I
Values of the parameters used in our experiments. Except for β in CLIP and BLIP-2, parameters are computed by optimizing the cross-entropy loss via Bayesian optimization using Visual Genome object synset labels [24].

	Parameters		
	α	β	τ
For Ours with CLIP	0.6723	32	0.0258
For Ours with BLIP-2	0.8775	14	0.0320
For Ours with SigLIP 2	0.4033	23.162	0.0161

D. Results

Tables II and III present the results of binary relevance classification using a global threshold across different vision-language models and datasets.

Caption-based query results. For caption-based queries, our method consistently outperforms both raw cosine similarity and the mixture model baseline across all tested models (Table II). In the MS COCO captions dataset, our method achieves PR-AUC improvements of 85.7%, 35.4%, and 59.2% over raw cosine similarity for CLIP, BLIP-2, and SigLIP 2, respectively. Similarly, on Flickr30k captions, we observe gains of 22.6%, 16.9%, and 22.4% for the three models.

The mixture model performs particularly poorly in these caption retrieval scenarios, often showing near-zero performance metrics. This failure can be attributed to the inherent characteristics of caption-based retrieval: in both MS COCO and Flickr30k datasets, each query caption corresponds to only one relevant image among thousands of candidates. This extreme class imbalance and sparsity of positive examples severely weakens the mixture model's ability to fit meaningful Gaussian distributions to the score distribution. Furthermore, the mixture model fails to preserve the relative ranking order of retrieved candidates. This leads to unstable and unreliable retrieval results.

Object-class query results. For object-class queries, our method performs comparably to or better than raw cosine similarity across most experimental conditions (Table III). On COCO classes, our method achieves the best PR-AUC for CLIP (3.6% improvement) and BLIP-2 (13.1% improvement), while maintaining competitive performance with SigLIP 2. For P@R95%, our method shows substantial improvements, particularly for BLIP-2 (34.4% gain) and SigLIP 2 (33.7% gain). On Incidents1M classes, our method consistently improves P@R95% across all models, with gains of 48.1%, 82.9%, and 10.4% for CLIP, BLIP-2, and SigLIP 2, respectively.

The challenge in these object-class datasets lies in the scarcity of relevant images. Only 3.6% of images are relevant in COCO classes and 2.0% in Incidents1M classes. This overwhelming dominance of negative examples makes the mixture model unstable, as it struggles to accurately model the small positive class distribution. The relative improvements over raw cosine similarity are more modest than those observed

in caption-based retrieval because increasing the relevant data ratio makes the estimation of the non-relevant mean in (6) more difficult. Nevertheless, our method demonstrates consistent gains, particularly in precision at high recall rates, which is crucial for practical retrieval applications.

Score recalibration accuracy. Table IV presents the mean absolute error (MAE) between user-specified precision/recall thresholds and the actual precision/recall achieved in retrieval results. Our method maintains both precision and recall MAE below 0.2590 even in the worst-case scenario (CLIP on COCO classes for precision), demonstrating robust calibration performance across different models and datasets. More impressively, for SigLIP 2, our method achieves MAE as low as 0.1042 for recall on Incidents1M classes. These low error rates indicate that our recalibrated scores provide reliable probability estimates that closely match the true likelihood of relevance, and that user-specified thresholds correspond well to the actual retrieval metrics achieved.

By contrast, the mixture model baseline yields MAE values exceeding 0.4 in several cases (e.g., 0.4524 for SigLIP 2 recall on COCO classes, 0.4256 for CLIP recall on Incidents1M classes). This poor calibration performance stems from two factors: a fundamental mismatch between the mixture model's assumptions about score distributions and the actual distributions in embedding-based retrieval, and the scarcity of relevant candidates. These results confirm that our method offers more stable, accurate, and transparent control over retrieval precision and recall, enabling users to reliably specify their desired operating points.

Computational efficiency. Fig. 2 shows the measured running time for score calibration across different numbers of candidate images. The results demonstrate that our proposed method is more than 800× faster than the conventional mixture model approach, completing calibration for 100,000 candidates in only 0.55 milliseconds. The dramatic speed advantage stems from fundamental algorithmic differences: the conventional mixture model must execute the EM algorithm at each search to fit Gaussian distributions to the observed score distribution, which is computationally expensive. In our implementation, the mixture model runs EM 100 times with different initializations and selects the result with the highest likelihood to reduce sensitivity to initialization.

In contrast, our proposed method applies a single, closed-form calibration analogous to the softmax function, requiring only element-wise operations on the score vector. Even if we limit the mixture model to a single EM run (eliminating the 100 trials, resulting in approximately 100× speedup), our proposed method remains significantly faster. This computational efficiency makes our approach practical for real-time, large-scale retrieval applications where latency is critical.

TABLE II
RESULTS OF BINARY RELEVANCE CLASSIFICATION ON **CAPTION DATASETS**. THE ROW WITH "-" DENOTES THE RESULTS USING RAW COSINE SIMILARITY SCORES. PR-AUC AND P@R95% ARE CALCULATED ACROSS ALL QUERIES.

Model	Method	PR-AUC (↑)	P@R95% (↑)
COCO captions			
CLIP	-	0.1427	0.0123
	Mixture	0.0001	0.0002
	Ours	**0.2650**	**0.0136**
BLIP-2	-	0.4970	0.1016
	Mixture	0.4975	0.0397
	Ours	**0.6729**	**0.1102**
SigLIP 2	-	0.3180	0.0202
	Mixture	0.0001	0.0002
	Ours	**0.5063**	**0.0280**
Flickr30k captions			
CLIP	-	0.4902	0.0474
	Mixture	0.0005	0.0010
	Ours	**0.6012**	**0.0559**
BLIP-2	-	0.7819	0.3210
	Mixture	0.4854	0.0508
	Ours	**0.9142**	**0.5121**
SigLIP 2	-	0.6449	0.0831
	Mixture	0.0005	0.0010
	Ours	**0.7891**	**0.1450**

TABLE III
RESULTS OF BINARY RELEVANCE CLASSIFICATION ON **OBJECT CLASS DATASETS**. THE ROW WITH "-" DENOTES THE RESULTS USING RAW COSINE SIMILARITY SCORES.

Model	Method	PR-AUC (↑)	P@R95% (↑)
COCO classes			
CLIP	-	0.3896	**0.0504**
	Mixture	0.1243	0.0363
	Ours	**0.4036**	0.0499
BLIP-2	-	0.4800	0.0363
	Mixture	0.3830	0.0396
	Ours	**0.5429**	**0.0488**
SigLIP 2	-	**0.5998**	0.0710
	Mixture	0.1376	0.0363
	Ours	0.5969	**0.0949**
Incidents1M classes			
CLIP	-	0.3590	0.0335
	Mixture	0.0121	0.0197
	Ours	**0.3894**	**0.0496**
BLIP-2	-	**0.4183**	0.0264
	Mixture	0.1146	0.0234
	Ours	0.4161	**0.0483**
SigLIP 2	-	0.4034	0.0443
	Mixture	0.0119	0.0197
	Ours	**0.4289**	**0.0489**

V. CONCLUSION

We proposed a simple and efficient score calibration method for embedding-based text-image retrieval with Vision-Language Models, addressing the challenge of intuitive and stable thresholding across varying queries and retrieval settings. Our method calibrates relevance scores by normalizing cosine similarity using the mean of estimated non-relevant candidates, which is expected to alleviate sensitivity to candidate set size and relevance ratio. Experimental results demonstrated that our approach provides more stable and user-controllable

thresholding compared to conventional baselines. Moreover, the proposed calibration runs faster than the conventional method. Additionally, score recalibration enables direct interpretation of thresholds in terms of precision and recall, allowing users to intuitively specify retrieval criteria. These results indicate the effectiveness of our method for improving the practicality of real-world text-image retrieval systems.

TABLE IV
RESULTS OF ERROR EVALUATION FOR THE THRESHOLD AND ACTUAL PRECISION OR RECALL ON **OBJECT CLASS DATASETS**.

Model	Method	MAE (↓) Prec.	Rec.
COCO classes			
CLIP	Mixture	0.3949	0.3965
	Ours	**0.2590**	**0.2314**
BLIP-2	Mixture	0.1988	**0.1850**
	Ours	**0.1561**	0.1984
SigLIP 2	Mixture	0.3932	0.4524
	Ours	**0.1727**	**0.1236**
Incidents1M classes			
CLIP	Mixture	0.3896	0.4256
	Ours	**0.2101**	**0.2292**
BLIP-2	Mixture	0.3797	0.2916
	Ours	**0.1503**	**0.1379**
SigLIP 2	Mixture	0.4283	0.4259
	Ours	**0.1149**	**0.1042**

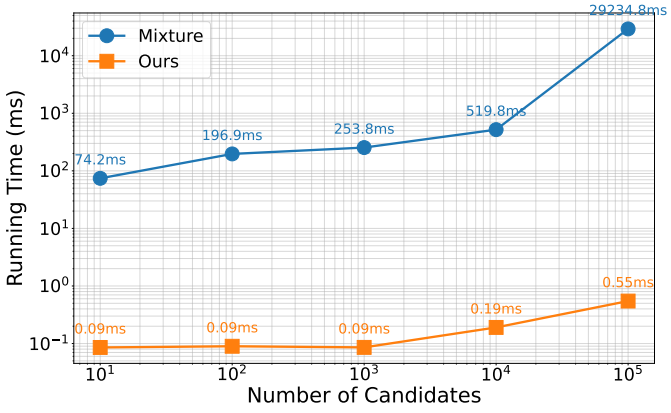

Fig. 2. Comparison of running time for score calibration. Measurements for both methods were performed using an Intel Xeon Gold 6226R processor. Each reported value is the average over 30 trials with random input score values.

REFERENCES

[1] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov, "Devise: A deep visual-semantic embedding model," in *NeurIPS*, vol. 2, 2013, pp. 2121 – 2129.

[2] J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, and Y. Wu, "Coca: Contrastive captioners are image-text foundation models," *TMLR*, 2022.

[3] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *ICML*, vol. 139, 2021, pp. 8748–8763.

[4] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *ICML*, vol. 202, 2023, pp. 19 730–19 742.

[5] Z. Chen, J. Wu, W. Wang, W. Su, G. Chen, S. Xing, M. Zhong, Q. Zhang, X. Zhu, L. Lu *et al.*, "Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks," in *CVPR*, 2024, pp. 24 185–24 198.

[6] P. Kaur, H. S. Pannu, and A. K. Malhi, "Comparative analysis on cross-modal information retrieval: A review," *Computer Science Review*, vol. 39, no. 2, p. 100336, 2021.

[7] M. Cao, S. Li, J. Li, L. Nie, and M. Zhang, "Image-text retrieval: A survey on recent research and development," in *IJCAI*, 2022, pp. 5410–5417.

[8] N. Rossi, J. Lin, F. Liu, Z. Yang, T. Lee, A. Magnani, and C. Liao, "Relevance filtering for embedding-based retrieval," in *CIKM*, 2024, pp. 4828–4835.

[9] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[10] D. Bahri, Y. Tay, C. Zheng, D. Metzler, and A. Tomkins, "Choppy: Cut transformer for ranked list truncation," in *SIGIR*, 2020, pp. 1513–1516.

[11] A. Arampatzis, J. Kamps, and S. Robertson, "Where to stop reading a ranked list? threshold optimization using truncated score distributions," in *SIGIR*, 2009, pp. 524–531.

[12] A. Arampatzis and A. van Hameran, "The score-distributional threshold optimization for adaptive binary classification tasks," in *SIGIR*, 2001, pp. 285–293.

[13] R. Manmatha, T. Rath, and F. Feng, "Modeling score distributions for combining the outputs of search engines," in *SIGIR*, 2001, pp. 267–275.

[14] Y.-C. Lien, D. Cohen, and W. B. Croft, "An assumption-free approach to the dynamic truncation of ranked lists," in *ICTIR*, 2019, pp. 79–82.

[15] A. Arampatzis and S. Robertson, "Modeling score distributions in information retrieval," *Information Retrieval*, vol. 14, pp. 26–46, 2011.

[16] C. Wu, R. Zhang, J. Guo, Y. Fan, Y. Lan, and X. Cheng, "Learning to truncate ranked lists for information retrieval," in *AAAI*, vol. 35, no. 5, 2021, pp. 4453–4461.

[17] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *ICML*, vol. 162, 2022, pp. 12 888–12 900.

[18] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid loss for language image pre-training," in *ICCV*, 2023, pp. 11 975–11 986.

[19] M. Tschannen, A. Gritsenko, X. Wang, M. F. Naeem, I. Alabdulmohsin, N. Parthasarathy, T. Evans, L. Beyer, Y. Xia, B. Mustafa, O. Hénaff, J. Harmsen, A. Steiner, and X. Zhai, "Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features," *arXiv preprint arXiv:2502.14786*, 2025.

[20] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[21] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014, pp. 740–755.

[22] B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik, "Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models," in *ICCV*, 2015, pp. 2641–2649.

[23] E. Weber, N. Marzo, D. P. Papadopoulos, A. Biswas, A. Lapedriza, F. Ofli, M. Imran, and A. Torralba, "Detecting natural disasters, damage, and incidents in the wild," in *ECCV*, 2020, pp. 331–350.

[24] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *IJCV*, vol. 123, no. 1, pp. 32–73, 2017.

[25] D. Li, J. Li, H. Le, G. Wang, S. Savarese, and S. C. H. Hoi, "Lavis: A library for language-vision intelligence," *arXiv preprint arXiv:1412.3555*, 2022.

[26] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–22, 1977.