AutoRelAnnotator: Calibrated Model Cascades for Cost-Efficient Relevance Evaluation in Sponsored Search

Md Omar Faruk Rokon[1,*], Shasvat Desai[1,*], Hong Yao[1] and Kuang-chih Lee[1]

[1]*Walmart Global Tech, Sunnyvale, CA, USA*

Abstract

How can we generate high-quality relevance annotations at scale without the cost and delays of human labeling? Relevance annotations are the backbone of search ranking systems which is needed for training data preparation, NDCG evaluation, and root cause analysis. However, human annotation is slow and off-the-shelf LLMs suffer from accuracy on domain-specific tasks. We propose a **calibrated model cascade**, a systematic approach for cost-efficient offline relevance annotation by routing queries through progressively larger fine-tuned classifiers. Our central insight is that **accuracy and cost are orthogonal optimizations**: domain-specific fine-tuning drives accuracy, cascading drives cost, and per-class isotonic calibration adds a small but reliable gain on top. Our contribution is threefold: (a) we decompose the gains and show that fine-tuning contributes $\sim$20 accuracy points while cascading is approximately accuracy-neutral but halves compute cost, (b) we introduce per-class isotonic calibration as one component of the cascade, contributing a small but statistically significant gain (+0.6 points over the strongest calibration baseline), and (c) we validate the system in production across six offline use cases, processing 150M+ annotations and enabling faster experimentation cycles. Our work is a building block for scalable, high-quality offline annotation pipelines in search and advertising systems.

Keywords

Relevance Annotation, Model Cascades, Confidence Calibration, Sponsored Search, LLM Classification, E-Commerce Search

1. Introduction

How can a search team evaluate ranking quality, diagnose relevance failures, or prepare training data—when human annotation takes days and costs hundreds of thousands of dollars? This is not a hypothetical problem. In our production environment, a team of 3 trained annotators produces approximately 3,000 labels per day at $0.50 per annotation, with 3–5 business day turnaround for batches exceeding 10,000 labels. These constraints bottleneck virtually every stage of the search ML lifecycle: (a) **training data preparation**: annotating query-product pairs for ranking model training, (b) **NDCG evaluation**: assessing search quality using graded relevance judgments, (c) **root cause analysis**: rapidly annotating suspect queries when production metrics degrade, (d) **guardrail calibration**: determining thresholds for production safety systems, and (e) **feature evaluation**: measuring the impact of A/B tests on relevance. None of these use cases require real-time inference. However, these require high accuracy at manageable cost. This offline setting is crucial. Unlike online serving where latency is paramount, batch annotation allows us to run multiple models, use ensemble fallbacks, and invest in calibration. These are impractical at serving time but highly effective for annotation quality.

Can off-the-shelf LLMs fill this gap? Unfortunately, no. GPT-4 achieves only 68.2% and Claude-3 only 70.1% accuracy on our 5-class relevance task which are sub par compared to the 89% inter-annotator agreement we observe. LLM cascade methods [1, 2, 3] reduce costs by routing among such models, but inherit their accuracy ceiling which is a fundamental limitation, since no routing policy can substantially exceed the accuracy of its constituent models [4].

We take a fundamentally different approach. We propose a **calibrated model cascade**, a systematic approach for offline relevance annotation that decouples two orthogonal problems: (a) *accuracy improve-*

ECOM'26: SIGIR Workshop on eCommerce, Jul 24, 2026, Melbourne, Australia

*Corresponding author.

mdomarfaruk.rokon@walmart.com (M. O. F. Rokon); shasvat.desai@walmart.com (S. Desai); hong.yao0@walmart.com (H. Yao); kuang-chih.lee@walmart.com (K. Lee)

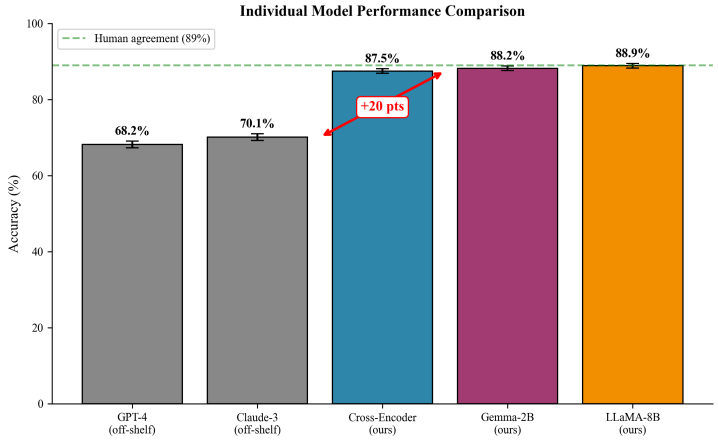

Figure 1: Accuracy-cost tradeoff: fine-tuning improves accuracy, while cascading reduces compute cost.

ment through domain-specific fine-tuning, and (b) *cost reduction* through calibrated cascading. Rather than routing among low performing models, we first fine-tune classifiers achieving higher accuracy individually, then apply per-class isotonic calibration to route queries through a three-model cascade (Cross-Encoder → Gemma-2B → LLaMA-8B). Figure 1 illustrates these orthogonal gains.

Contribution. The contribution of this work is threefold:

a. We show that *accuracy and cost are orthogonal optimizations* for offline relevance annotation: fine-tuning contributes a 20+ point accuracy gain while cascading is approximately accuracy-neutral but halves compute cost.

b. We introduce per-class isotonic calibration as one component of the cascade, contributing a small but statistically significant gain (+0.6 points over the strongest calibration baseline, $p < 0.05$).

c. We validate the system across six offline use cases, processing over 150 million annotations and reducing turnaround from 5 days to 1–3 hours.

Our work in perspective. Our approach can be seen as a first step towards fully automated offline annotation pipelines. The cascade is modular where each model can be independently updated without affecting routing. It can be a practical building block for any organization needing large-scale relevance annotations.

2. Related Work

LLM Cascades and Routing. Cascade architectures route inputs through progressively more expensive models. FrugalGPT [1] learns a DistilBERT scoring function to select among LLM APIs, achieving up to 98% cost reduction. ABC [2] uses ensemble agreement as a training-free deferral signal for 2–25× cost savings; its extension to open-ended generation via semantic agreement appeared at EMNLP 2025 [5]. RouteLLM [3] trains routers on preference data for binary model selection. Hybrid LLM [6] learns a quality predictor for routing between two models. Zellinger and Thomson [7] provide a probabilistic framework for threshold optimization using Markov-copula models. Dekoninck et al. [8] derive optimal strategies for unified routing and cascading. C3PO [9] applies conformal prediction for probabilistic cost bounds.

While these cascade methods achieve impressive cost reductions, our work differs in two ways: (a) we fine-tune classifiers before cascading, decoupling accuracy gains (from fine-tuning) from cost gains (from cascading), rather than routing among off-the-shelf models that inherit their accuracy limitations on domain-specific tasks, and (b) we apply per-class calibration for routing decisions. Three recent papers move closer to our approach. CascadeBERT [10] cascades multiple BERT variants with calibrated confidence which is our closest pre-LLM precursor, which we extend to heterogeneous LLM architectures. Cascade-Aware Training [11] fine-tunes the smallest model with downstream awareness,

but targets generative tasks only. GATEKEEPER [12] modifies the training loss for calibration-aware deferral. Our approach differs in using standard fine-tuning plus *post-hoc* per-class isotonic calibration, maintaining modularity.

Confidence Calibration. Neural networks produce miscalibrated confidence scores [13]. Post-hoc methods include temperature scaling, Platt scaling [14], histogram binning [15], and isotonic regression [16]. Nixon et al. [17] demonstrate that class-conditional miscalibration is far worse than global ECE suggests. Kull et al. [18] formalize classwise calibration and propose Dirichlet calibration as an alternative. We choose per-class isotonic regression for its nonparametric flexibility. Jitkrittum et al. [4] provide theoretical conditions under which confidence-based cascade deferral succeeds, noting that well-calibrated scores are essential.

Relevance Classification. Cross-encoders [19, 20] jointly encode query-document pairs for relevance scoring. DeBERTa-v3 cross-encoders arguably match models 11× larger on reranking tasks [21]. LoRA [22] enables efficient fine-tuning of LLaMA [23] and Gemma [24]. LLMs as annotators have gained attention [25, 26], but studies consistently find that generic LLMs underperform on domain-specific tasks [27].

3. Our Approach

Why do existing cascades plateau at 68–72% accuracy on domain-specific tasks? The answer is straightforward: they route among off-the-shelf models whose individual accuracy is limited [1, 2]. No matter how sophisticated the routing, the cascade cannot substantially exceed the accuracy of its constituent models.

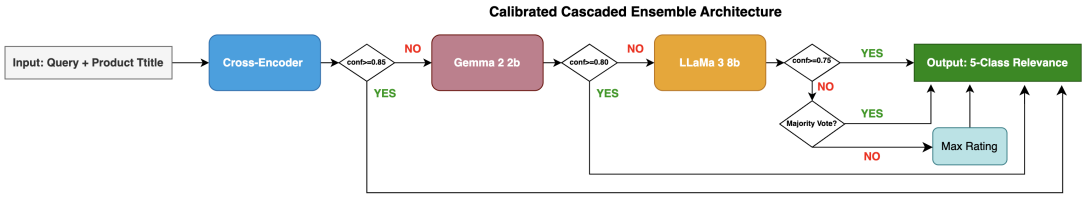

Figure 2: Offline cascade with calibrated deferral at each stage and an ensemble fallback for ambiguous cases.

Our key insight is to **decouple accuracy improvement from cost reduction**: (a) first, we invest in domain-specific fine-tuning to create high-accuracy classifiers (87–89%), and (b) then, we apply calibrated cascading to reduce computational cost while preserving this accuracy.

3.1. System Architecture

Our cascade consists of three domain-specific fine-tuned classifiers ordered by cost. Figure 2 illustrates the full pipeline.
Cross-Encoder (DeBERTa-v3-base, 184M params): Processes query-title pairs in ∼5ms. We fine-tuned with a 5-class classification head. **Gemma-2B**: We fine-tuned with LoRA (r=16, α=16). Inference ∼50ms. **LLaMA-8B** (Meta LLaMA-3-8B): Our largest cascade model with similar LoRA fine-tuning. Inference ∼200ms. All latency measured on NVIDIA A100 (40GB), batch size 64, FP16.

3.2. Domain-Specific Fine-Tuning

How do we obtain reliable confidence scores? Unlike generative approaches requiring regex parsing of free-form text, our models use classification heads producing well-defined confidence:

$$p(y|q,t) = \text{softmax}(\mathbf{W} \cdot \mathbf{h}_{\text{pool}} + \mathbf{b}) \tag{1}$$

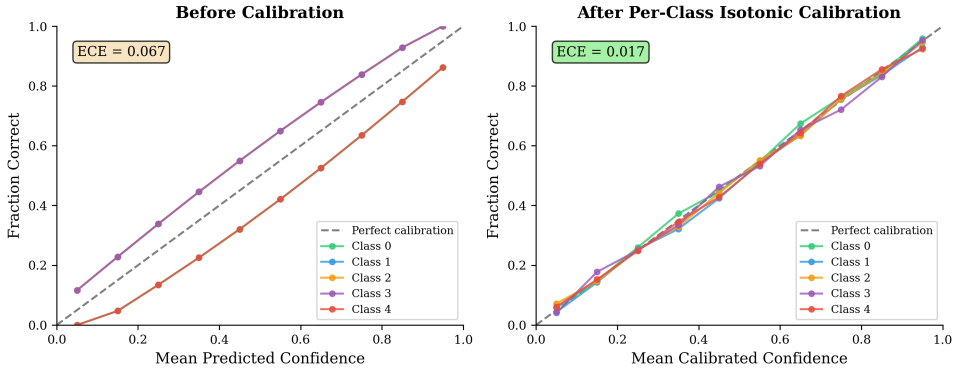

Figure 3: Calibration curves before and after per-class isotonic calibration.

Algorithm 1 Calibrated Cascade Inference

Require: Query q, title t, thresholds $\{\tau_m\}$
for model $m \in$ [CrossEncoder, Gemma, LLaMA] **do**
  $(\hat{y}_m, \hat{p}_m) \leftarrow m(q, t)$
  $p_m^* \leftarrow f_{m,\hat{y}_m}(\hat{p}_m)$ *{Per-class calibration}*
  if $p_m^* \geq \tau_m$ **then**
    return $\hat{y}_m$
  end if
end for
$\hat{y} \leftarrow \text{MajorityVote}(\hat{y}_1, \hat{y}_2, \hat{y}_3)$
if tie **then** $\hat{y} \leftarrow \max(\hat{y}_1, \hat{y}_2, \hat{y}_3)$
return $\hat{y}$

where $\mathbf{h}_{\text{pool}}$ is the pooled sequence representation, $\mathbf{W} \in \mathbb{R}^{5\times d}$, and $y \in \{0, 1, 2, 3, 4\}$. For the Cross-Encoder, $\mathbf{h}_{\text{pool}}$ is the final [CLS] hidden state. For Gemma and LLaMA, $\mathbf{h}_{\text{pool}}$ is the mean-pooled final hidden state across tokens; LoRA targets attention projections and the classifier head trains end-to-end. Training uses cross-entropy loss with early stopping on validation loss.

3.3. Per-Class Isotonic Calibration

Why not use raw confidence for routing? Raw confidences are miscalibrated [13], and prior cascades either ignore calibration [2] or apply global calibration that treats all classes identically [1]. We observe that confidence distributions vary across classes: models are overconfident on extreme classes (0, 4) and underconfident on middle classes (1, 2, 3) which is consistent with Nixon et al. [17], who show standard ECE underestimates class-conditional miscalibration by 3–5×.

We learn separate isotonic calibration functions per predicted class, following Zadrozny and Elkan [16]:

$$p_c^* = f_c(\hat{p}) \quad \text{where } f_c : [0, 1] \to [0, 1] \tag{2}$$

is monotonically non-decreasing. For each model m and class c: (a) we collect predictions where $\hat{y}_i = c$, (b) compute correctness $z_i = \mathbf{1}[\hat{y}_i = y_i]$, and (c) fit $f_c = \text{IsotonicRegression}(\hat{p}, z)$. We choose isotonic regression over Dirichlet calibration [18] for (a) its nonparametric flexibility, and (b) monotonicity preservation for threshold-based routing. Figure 3 shows calibration curves before and after per-class calibration.

3.4. Cascade Decision Logic

Given calibrated confidence thresholds τ_1, τ_2, τ_3 for each model, the cascade proceeds as mentioned in the Algorithm 1.

Hyperparameter	Cross-Enc.	Gemma-2B	LLaMA-8B
Learning rate	1e-5	5e-4	5e-4
Optimizer	AdamW	AdamW	AdamW
Max epochs	20	5	5
Batch size	128	4	4
Warmup ratio	0.1	0.1	0.1
LR schedule	linear	linear	linear
LoRA r, α	—	16, 16	16, 16
Train (H100-hrs)	$\sim$8	$\sim$16	$\sim$68

Table 1
Training hyperparameters and single-H100 wall-clock training cost for the three cascade models.

The ensemble fallback uses majority voting with a max-prediction tiebreaker that biases toward higher relevance—appropriate for offline use cases where over-penalizing relevant ads is costlier than mild over-scoring.

3.5. Implementation Details

How are the cascade models trained? Table 1 reports training hyperparameters for the three classifiers. We trained on NVIDIA H100 GPUs and benchmarked inference latency on A100 (40GB) for production parity. The Cross-Encoder fine-tunes all 184M parameters; Gemma-2B and LLaMA-8B use LoRA [22] on attention projections together with a 5-class classification head. All models use AdamW with linear warmup (10% of total steps) followed by linear decay, and early stopping on validation cross-entropy.

4. Experiments and Evaluation

4.1. Research Questions

We structure our evaluation around four questions: **Q1:** How do fine-tuned classifiers compare to off-the-shelf LLMs? **Q2:** Does per-class calibration improve cascade routing over global methods? **Q3:** What is the cost-accuracy tradeoff of the cascade versus single-model baselines? **Q4:** How do fine-tuning, cascading, and calibration each contribute to the final accuracy and cost?

4.2. Dataset and Setup

We fine-tune on 1.2M query-product pairs from a major e-commerce platform's sponsored search, collected over 18 months. We evaluate on a held-out set of 100K pairs annotated by trained raters on a 5-point scale (0=Embarrassing through 4=Excellent and class distribution approximately balanced at 18–22% per class). The evaluation set is split: 50% for calibration fitting, and 50% for final evaluation. All metrics use bootstrap CIs (1,000 resamples) and differences $>$0.5% are significant at $p < 0.01$.

4.3. Individual Model Performance (Q1)

We showed our model performance comparison with 95% bootstrap confidence intervals in Table 2 .
Observation 1: Fine-tuning closes a 20-point accuracy gap. All fine-tuned models achieve 87–89% vs. 68–70% for off-the-shelf LLMs. Even the smallest Cross-Encoder (184M params) outperforms GPT-4 by 20 points while being 100$\times$ faster. We argue that classification heads produce well-defined probability distributions, whereas generative prompting is inherently fragile and poorly calibrated.
Observation 1b: Smaller models saturate earlier. Figure 4 reports accuracy as a function of training-set size for each model. The Cross-Encoder gains only 6.2 points from 50K to 1.2M examples (81.3$\rightarrow$87.5), while LLaMA-8B gains 9.6 points (79.3$\rightarrow$88.9) and Gemma-2B gains 9.3 points (78.9$\rightarrow$88.2). The pattern is consistent with capacity-driven scaling: smaller models reach their plateau with modest data, while

Model	Accuracy (95% CI)	Latency
GPT-4 (off-the-shelf)	68.2 ± 0.9%	500ms
Claude-3 (off-the-shelf)	70.1 ± 0.9%	450ms
Cross-Encoder (ours)	87.5 ± 0.6%	5ms
Gemma-2B (ours)	88.2 ± 0.6%	50ms
LLaMA-8B (ours)	88.9 ± 0.6%	200ms

Table 2
Accuracy and latency of individual models.

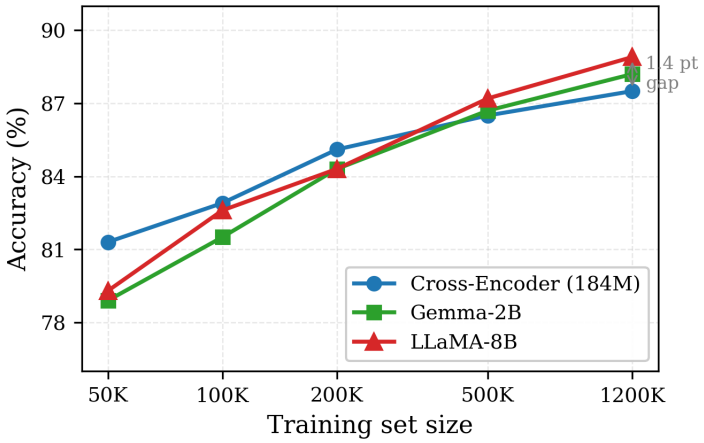

Figure 4: Learning curves by training-set size; smaller models saturate earlier while larger models keep improving.

Method	ECE↓	Casc. Acc.	Bal. Acc.
No calibration	.067±.008	88.2±0.6	88.0±0.6
Temperature	.040±.005	88.5±0.6	88.3±0.6
Platt scaling	.027±.004	88.5±0.6	88.3±0.6
Histogram bin.	.010±.003	88.5±0.6	88.3±0.6
Isotonic (global)	.012±.003	88.5±0.6	88.3±0.6
Iso. (per-class)	.017±.004	**89.1±0.6**	**89.0±0.6**

Table 3
Calibration comparison for cascade routing; per-class isotonic gives the best cascade accuracy.

larger models continue to benefit. This sample-efficiency gradient justifies the cascade ordering—the Cross-Encoder is a reliable cheap first stage, while the larger models earn their inference cost on the harder long tail.

4.4. Calibration Method Comparison (Q2)

We presented our calibration methods comparison in Table 3.

Observation 2: Per-class calibration yields the best cascade accuracy despite higher global ECE. The most stringent comparison is against the strongest calibration baseline—global isotonic regression—rather than against the uncalibrated cascade. Per-class isotonic improves cascade accuracy from 88.5% to 89.1% over global isotonic (+0.6 points, $p < 0.05$, paired bootstrap), and from 88.2% to 89.1% over no calibration (+0.9 points). Global ECE (0.017 for per-class vs. 0.010 for histogram binning) masks the class-conditional improvements that matter for routing (see Figure 3).

XE	Gem	Lla	Acc. (95% CI)	Savings
0.92	0.98	0.90	88.9 ± 0.6%	46.8%
0.90	0.90	0.90	89.0 ± 0.6%	44.2%
0.85	0.85	0.85	89.1 ± 0.6%	39.8%
0.85	**0.80**	**0.75**	**89.1 ± 0.6%**	**50.1%**

Table 4
Representative thresholded cascade configurations and their cost savings.

Decision Source	Traffic	Accuracy (95% CI)
Cross-Encoder (confident)	74.5%	91.2 ± 0.7%
Gemma (confident)	12.5%	86.8 ± 1.9%
LLaMA (confident)	5.9%	84.5 ± 2.9%
Ensemble fallback	7.1%	79.2 ± 3.0%
Overall	100%	**89.1 ± 0.6%**

Table 5
Traffic and accuracy by decision source in the production cascade.

4.5. Cascade Performance (Q3)

We illustrated cascaded performance across threshold configurations in Table 4.

Observation 3: The cascade achieves ensemble-level accuracy at half the cost. The production configuration achieves 89.1% accuracy with 50.1% cost savings. For reference, the full ensemble (all three models) achieves 89.3% at 1.28× cost, while LLaMA-8B alone achieves 88.9% at 1.0×. The cascade matches ensemble accuracy at lower cost and lower latency.

Observation 4: The Cross-Encoder resolves 74.5% of queries at higher-than-standalone accuracy. Table 5 shows prediction origins. The Cross-Encoder handles 74.5% of queries at 91.2% which is well above its standalone 87.5% ($p < 0.001$), because calibration routes only high-confidence predictions to the cheap first stage. The ensemble fallback handles 7.1% at 79.2%, acceptable for inherently ambiguous samples.

Configuration	Acc.	Rel. Cost
Off-the-shelf LLMs (API)		
GPT-4 (single)	68.2%	—
GPT-4 + Claude-3 (cascade)	71.4%	—
Fine-tuned (ours, local A100)		
LLaMA-8B (single)	88.9%	1.00×
Cascade, uncalibrated	88.2%	0.50×
Cascade, global isotonic	88.5%	0.50×
Cascade, per-class iso.	**89.1%**	**0.50×**

Table 6
Ablation of fine-tuning, cascading, and calibration. Relative cost is normalized to fine-tuned LLaMA-8B.

4.6. Decomposing the Gains (Q4)

How much of our 89.1% accuracy comes from fine-tuning versus cascading versus calibration? Table 6 isolates each component's contribution.

Observation 5: Fine-tuning contributes the bulk of the accuracy gain. A single fine-tuned LLaMA-8B classifier (88.9%) closes 20.7 points of the gap over GPT-4 (68.2%). By contrast, cascading off-the-shelf LLMs (GPT-4 → Claude-3) gains only 3.2 points over GPT-4 alone (68.2→71.4%), consistent

with the cascade-bound argument of Jitkrittum et al. [4]: routing cannot exceed the accuracy of its constituents by much.

Observation 6: Cascading delivers cost reduction without sacrificing accuracy. Moving from a single fine-tuned LLaMA-8B (88.9%, 1.00×) to an uncalibrated cascade gives 88.2% at 0.50× cost—a 0.7-point drop for 50% cost savings. Calibration recovers and then exceeds the single-model accuracy.

Observation 7: Calibration contributes a small but consistent gain. Per-class isotonic calibration adds 0.9 points over the uncalibrated cascade and 0.6 points over the strongest calibration baseline (global isotonic), both statistically significant ($p < 0.05$).

The decomposition supports our central claim: *accuracy and cost are orthogonal optimizations*—fine-tuning drives accuracy, cascading drives cost, and calibration provides a small additive gain that is reliable across deployment windows.

5. Production Deployment and Discussion

How does the system perform in real-world offline workflows? Since deployment in Q3 2024, the cascade has processed over 150 million annotations across six offline use cases: (a) training data preparation (100M+ pairs), (b) NDCG evaluation (10K+ annotations per experiment), (c) root cause analysis (1-2M annotations within hours), (d) guardrail calibration, (e) feature evaluation via A/B tests, and (f) ANN dictionary preparation. All use cases are *fully offline*. The cascade is never in the serving path. Median turnaround dropped from 5 days to 1-3 hours. Three teams actively use the system across Advertising, Marketing, and Recommendations.

We discuss several practical insights:

a. Does input format affect accuracy? Yes, dramatically. Using the exact template from training is critical and even minor variations degrade accuracy by 10–15%.

b. How fresh must calibrators be? We retrain quarterly using a rolling 50K-sample window. Stale calibrators lead to suboptimal routing that silently degrades accuracy.

c. Are high thresholds always better? High thresholds (e.g., 0.98) render intermediate models useless. Lower thresholds (0.80–0.85) better utilize the full cascade.

d. How does our system compare to related work? A comparable study fine-tuned LLaMA-2 7B for 3-class sponsored search relevance, achieving 89.43% [28]. Our system achieves nearly identical accuracy on a harder 5-class task while halving compute which suggests cascade architecture provides efficiency gains beyond single-model approaches.

Limitations. Our evaluation covers a single domain (sponsored search), so the 20-point accuracy gain reflects fine-tuning in this setting rather than cascade architecture alone. The system also requires substantial labeled data (1.2M examples for fine-tuning and 50K for calibration). Latency comparisons use local A100 inference vs. cloud APIs, although our primary efficiency claim (50% savings vs. running LLaMA-8B on every query) is deployment-independent. Finally, proprietary data prevents external reproduction, though we provide detailed methodology for replication on other datasets.

6. Conclusion

We presented a calibrated model cascade for cost-efficient offline relevance annotation. The core insight is that **accuracy and cost are orthogonal optimizations**: fine-tuning contributes ∼20 accuracy points, cascading halves compute cost without sacrificing accuracy, and per-class isotonic calibration adds a small but reliable gain (+0.6 points over the strongest baseline) on top. The system has been deployed in production, processing 150M+ annotations across six offline use cases. We argue that our work is a building block for scalable offline annotation pipelines, with natural extensions including intra-model early exits [29], human-in-the-loop deferral, and adaptive model orderings.

Declaration on Generative AI

During the preparation of this work, the author(s) used Generative AI only for formatting assistance in plotting code used to produce figures, and not for drafting, analysis, interpretation, or any other part of the manuscript. The author(s) reviewed, corrected, and validated the generated code and figures and take(s) full responsibility for the publication's content.

References

[1] L. Chen, M. Zaharia, J. Zou, Frugalgpt: How to use large language models while reducing cost and improving performance, arXiv preprint arXiv:2305.05176 (2023).

[2] S. Kolawole, D. Dennis, A. Talwalkar, V. Smith, Agreement-based cascading for efficient inference, Transactions on Machine Learning Research (2024). ArXiv:2407.02348.

[3] I. Ong, A. Almahairi, V. Wu, W.-L. Chiang, T. Wu, J. E. Gonzalez, M. W. Kadous, I. Stoica, Routellm: Learning to route llms with preference data, arXiv preprint arXiv:2406.18665 (2024).

[4] W. Jitkrittum, N. Gupta, A. K. Menon, H. Narasimhan, A. Rawat, S. Kumar, When does confidence-based cascade deferral suffice?, Advances in Neural Information Processing Systems 36 (2023) 9891–9906.

[5] D. Soiffer, S. Kolawole, V. Smith, Semantic agreement enables efficient open-ended llm cascades, in: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track, 2025, pp. 2499–2537.

[6] D. Ding, A. Mallick, C. Wang, R. Sim, S. Mukherjee, V. Ruhle, L. V. Lakshmanan, A. H. Awadallah, Hybrid LLM: Cost-efficient and quality-aware query routing, in: Proceedings of the 12th International Conference on Learning Representations, 2024.

[7] M. Zellinger, M. Thomson, Rational tuning of LLM cascades via probabilistic modeling, arXiv preprint arXiv:2501.09345 (2025).

[8] J. Dekoninck, M. Baader, M. Vechev, A unified approach to routing and cascading for llms, arXiv preprint arXiv:2410.10347 (2024).

[9] A. Valkanas, S. Pal, P. Rumiantsev, Y. Zhang, M. Coates, C3po: Optimized large language model cascades with probabilistic cost constraints for reasoning, arXiv preprint arXiv:2511.07396 (2025).

[10] L. Li, Y. Lin, D. Chen, S. Ren, P. Li, J. Zhou, X. Sun, Cascadebert: Accelerating inference of pre-trained language models via calibrated complete models cascade, in: Findings of the Association for Computational Linguistics: EMNLP 2021, 2021, pp. 475–486.

[11] C. Wang, S. Augenstein, K. Rush, W. Jitkrittum, H. Narasimhan, A. S. Rawat, A. K. Menon, A. Go, Cascade-aware training of language models, arXiv preprint arXiv:2406.00060 (2024).

[12] S. Rabanser, N. Rauschmayr, A. Kulshrestha, P. Poklukar, W. Jitkrittum, S. Augenstein, C. Wang, F. Tombari, Gatekeeper: Improving model cascades through confidence tuning, in: The Thirty-ninth Annual Conference on Neural Information Processing Systems, 2025.

[13] C. Guo, G. Pleiss, Y. Sun, K. Q. Weinberger, On calibration of modern neural networks, in: International conference on machine learning, PMLR, 2017, pp. 1321–1330.

[14] J. Platt, et al., Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods, Advances in large margin classifiers 10 (1999) 61–74.

[15] B. Zadrozny, C. Elkan, Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers, in: Icml, volume 1, 2001.

[16] B. Zadrozny, C. Elkan, Transforming classifier scores into accurate multiclass probability estimates, in: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, 2002, pp. 694–699.

[17] J. Nixon, M. W. Dusenberry, L. Zhang, G. Jerfel, D. Tran, Measuring calibration in deep learning., in: CVPR workshops, volume 2, 2019.

[18] M. Kull, M. Perello Nieto, M. Kängsepp, T. Silva Filho, H. Song, P. Flach, Beyond temperature

scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration, Advances in neural information processing systems 32 (2019).
[19] R. Nogueira, K. Cho, Passage re-ranking with BERT, arXiv preprint arXiv:1901.04085 (2019).
[20] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using Siamese BERT-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, 2019, pp. 3982–3992.
[21] H. Déjean, S. Clinchant, T. Formal, A thorough comparison of cross-encoders and llms for reranking splade, arXiv preprint arXiv:2403.10407 (2024).
[22] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, et al., Lora: Low-rank adaptation of large language models., ICLR 1 (2022) 3.
[23] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., Llama: Open and efficient foundation language models, arXiv preprint arXiv:2302.13971 (2023).
[24] G. Team, T. Mesnard, C. Hardin, R. Dadashi, S. Bhupatiraju, S. Pathak, L. Sifre, M. Rivière, M. S. Kale, J. Love, et al., Gemma: Open models based on gemini research and technology, arXiv preprint arXiv:2403.08295 (2024).
[25] F. Gilardi, M. Alizadeh, M. Kubli, Chatgpt outperforms crowd workers for text-annotation tasks, Proceedings of the National Academy of Sciences 120 (2023) e2305016120.
[26] X. He, Z. Lin, Y. Gong, A.-L. Jin, H. Zhang, C. Lin, J. Jiao, S. M. Yiu, N. Duan, W. Chen, Annollm: Making large language models to be better crowdsourced annotators, in: Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track), 2024, pp. 165–190.
[27] Y. Wang, Z. Yu, Z. Zeng, L. Yang, C. Wang, H. Chen, et al., PandaLM: An automatic evaluation benchmark for LLM instruction tuning optimization, in: Proceedings of the 12th International Conference on Learning Representations, 2024.
[28] M. O. F. Rokon, A. Simion, W. Du, M. Wen, H. Yao, K.-c. Lee, Enhancement of e-commerce sponsored search relevancy with llm, in: Proceedings of the ACM SIGIR Workshop on eCommerce (eCom'24). https://sigir-ecom. github. io/eCom24Papers/paper_19. pdf, 2024.
[29] T. Schuster, A. Fisch, J. Gupta, M. Dehghani, D. Bahri, V. Tran, Y. Tay, D. Metzler, Confident adaptive language modeling, Advances in Neural Information Processing Systems 35 (2022) 17456–17472.