Small Model, Big Retrieval: Unified Heterogeneous Retrieval via Multi-Label-Aware Contrastive Fine-Tuning

Shaobai Jiang* **Dantong Liu*** **Federico Gonzalez** **Pedro Herrero-Vidal**
You-Lin Chen **Xiaoyu Chu** **Prithviraj Sen**
Amazon
{shaobaij, lidanton, fedeigon, phvidal,
cyoulin, xiaoychu, prithsen}@amazon.com

Abstract

Contrastive fine-tuning of a single embedding model across heterogeneous retrieval tasks is challenging when the tasks differ in label structure. In product-compliance retrieval, some tasks have multi-label supervision — every item carries a set of compliance categories, and many in-batch items share at least one category with the query, producing frequent in-batch false negatives. Other tasks (e.g., cross-catalog product matching) have no category labels: each query has one canonical match. Existing multi-label contrastive methods require every batch item to carry comparable labels, so they cannot serve a single embedding model trained on both label-bearing and label-free tasks. We propose **label-aware false-negative exclusion**: a zero-cost binary mask on the InfoNCE denominator that drops same-label candidates from the negative set rather than including them to positives. The mask has no effect on label-free items, so the same loss handles both task types. A 0.6 B-parameter embedding model fine-tuned this way handles four retrieval tasks simultaneously, lifting average Recall by 49% (0.489 $\rightarrow$ 0.729) and matching the 13$\times$ larger off-the-shelf Qwen3-Embedding-8B at 32$\times$ lower embedding dimensionality. Fine-tuning further enables **learned query fusion**: concatenating regulatory text with an example product, which hurts retrieval before fine-tuning, becomes the strongest query type after, beating either alone by 6–11%.

1 Introduction

Modern e-commerce platforms list very large numbers of products across many marketplaces, where three problems recur. (1) **Regulatory compliance** — items legal in some jurisdictions are restricted in others (e.g., kangaroo-meat products cannot be sold in California [1]), so the catalog must be screened against the rules of each marketplace. (2) **Cross-catalog product matching** — supplier and partner-feed listings must be linked to their counterparts in the internal catalog, a long-standing entity-resolution problem in e-commerce (Peeters et al., 2023). (3) **Variation-family grouping** — products that are functionally identical but differ in non-functional attributes (e.g., color, size) should be linked into a single family. All three reduce to retrieval over the internal catalog: given a query (a SOP [2], an external listing, or a reference product), return the matching items.

At catalog scale, the standard solution is a funnel: a cheap, high-recall candidate generator (typically combining keyword/rule-based filters with embedding-based retrieval) narrows the catalog before a downstream high-precision model makes the final decision. A single shared embedding model across the three retrieval settings is desirable, but training one model jointly via in-batch contrastive learning exposes a structural mismatch in *how the tasks are labeled*. Regulatory retrieval is **multi-label** — each product carries a set of compliance categories, so within any batch many items share a category with the query, turning most in-batch negatives into false negatives. Cross-catalog matching has no category labels: each external listing maps to one canonical internal product. Existing multi-label contrastive losses require every batch item to carry comparable labels, so they cannot serve a single embedding model trained on both label-bearing and label-free tasks.

We fine-tune a single Qwen3-Embedding-0.6B (Zhang et al., 2025) model on four retrieval tasks via task-aware instruction prompts (Fig. 1). The core method is **label-aware false-negative exclusion**: in-batch items that share a compliance label with the query are removed from the negative

* Equal contribution.

[1] California Penal Code §653o; see https://california.public.law/codes/penal_code_section_653o.

[2] SOPs are structured natural-language specifications that operationalize each regulation for retrieval, defining the regulatory category and its scope (mock-up in Appendix C)

EMNLP 2026 Industry Track, pages 2355–2368
October 24-29, 2026 ©2026 Association for Computational Linguistics

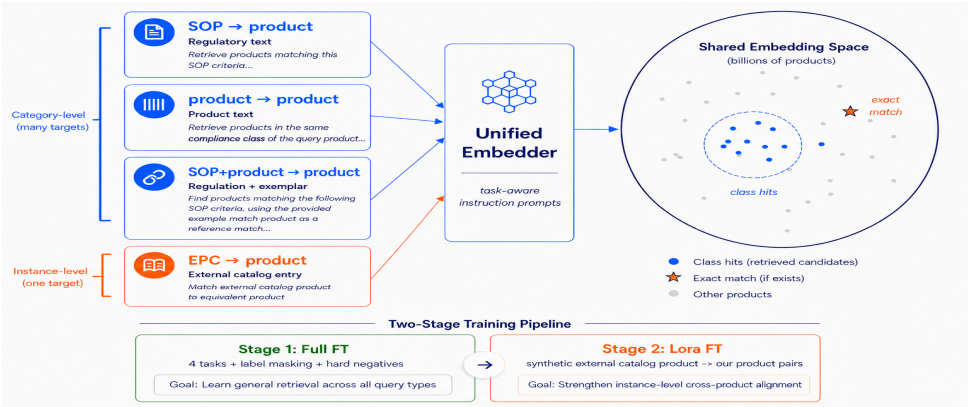

Figure 1: Overview of the heterogeneous retrieval system. A single embedding model unifies four retrieval tasks — three category-level retrieval tasks (returning all products in a compliance category) and one cross-catalog matching task (linking one external listing to its internal counterpart) — through task-aware instruction prompts over a shared embedding space. Training has two stages: full fine-tuning with label masking and hard negatives (Stage 1), then LoRA adaptation on synthetic external–product pairs (Stage 2).

set (a zero-cost binary mask on the InfoNCE denominator), rather than included to additional positives as in multi-label contrastive losses. The mask has no effect on label-free items, so the loss reduces to standard InfoNCE wherever no labels apply, making the method directly compatible with mixed-label batches. Combined with hard negative mining (de Souza P. Moreira et al., 2025), controlled lexical diffusion, and two-stage training, our fine-tuned Qwen3-Embedding-0.6B (Zhang et al., 2025) improves average recall by 49% (0.489 → 0.729).

The closest line of work is multi-label supervised contrastive learning (MulSupCon (Zhang and Wu, 2024) and follow-ups (Wei et al., 2024; Audibert et al., 2025; Lan et al., 2026)), which treats label-overlapping items as additional positives — pulling all same-category items together. We use the same multi-label signal more conservatively: *shared labels are reliable enough to mask false negatives, but not to designate new positives*. Two products sharing a compliance category may differ in brand, regulatory boundary, etc., so forcing them together blurs distinctions retrieval ranking depends on. Exclusion also fits our heterogeneous batches: on label-free tasks the mask has no effect, while on label-bearing tasks it removes false negatives without category clustering. Empirically, exclusion beats multi-positive inclusion on every task (§5), with the largest gap on instance-level retrieval. Softening inclusion does not close that gap: graded variants that give label-overlapping items a similarity-weighted partial-positive weight, or re-weight pairs adaptively, still trail exclusion by 3.3% and 4.1% average recall (Appendix G.3). Handling false negatives itself is well studied in contrastive learning, identifying them from an assumed class prior or from embedding similarity (Appendix A). We instead identify them from ground-truth label overlap, which costs nothing to compute and catches false negatives that the current model scores as dissimilar — precisely those similarity-based detection misses early in training.

Our Contributions are (1) **Label-aware false-negative exclusion** Excluding label-overlapping items from the InfoNCE denominator yields up to 5.7% retrieval gain at zero compute cost, naturally handles mixed-label batches by reducing to standard InfoNCE when labels are absent, and outperforms multi-positive inclusion on every task (§5). (2) **Unified heterogeneous retrieval.** A single fine-tuned 0.6 B-parameter embedding model serves three category-level retrieval tasks and one cross-catalog matching task, matching or exceeding the 13× larger off-the-shelf Qwen3-Embedding-8B at 32× lower embedding dimensionality (§4). (3) **Learned query fusion.** Combining regulatory text with an example product as a single query *hurts* retrieval in the base model but becomes the strongest query type after fine-tuning, outperforming either query type alone by 6–11% (§4.3). (4) **Zero-shot transfer.** Variation-family retrieval improves by +16% Recall@100 despite being unseen during training (§4.2).

2 Training Framework

Base model and multi-task batching. We adopt Qwen3-Embedding-0.6B (Zhang et al., 2025), which produces 1024-dim L2-normalized embeddings and supports 100+ languages. The size is a pragmatic trade-off: evaluation over very

large item indexes makes larger embedding models prohibitive at inference. We train on all four tasks simultaneously using heterogeneous batching with task-proportional sampling; the loss is computed over the combined batch rather than per-task sub-batches, since well-blended batches avoid the "zigzag" gradient updates of alternating-task training (Lee et al., 2025).

2.1 Label-Aware False Negative Exclusion

As discussed in §1, multi-label structure causes same-category items to co-occur frequently within a training batch, and standard contrastive losses treat them as negatives. Prior multi-label contrastive methods (Zhang and Wu, 2024; Wei et al., 2024; Audibert et al., 2025) address this by including label-overlapping items as additional weighted positives. We take the opposite approach: we *exclude* them from the negative set and keep a single designated positive per query. We show in §5 that this exclusion strategy consistently outperforms multi-positive inclusion and other variants.

Contrastive loss. For each training instance i with query q_i, positive document d_i^+, and K mined hard negatives $\{d_{i,k}^-\}_{k=1}^K$, we adopt the modified InfoNCE loss (van den Oord et al., 2019) used in Qwen3-Embedding training (Zhang et al., 2025):

$$\mathcal{L}_i = -\log\big(e_\tau(q_i, d_i^+) \,/\, Z_i\big) \tag{1}$$

where $e_\tau(a,b) = \exp\big(s(a,b)/\tau\big)$, $s(\cdot,\cdot)$ denotes cosine similarity, τ is the temperature, and Z_i sums over the positive, K hard negatives, and all in-batch items used as additional negatives:

$$\begin{aligned} Z_i = e(q_i, d_i^+) &+ \sum_k e(q_i, d_{i,k}^-) \\ &+ \sum_{j \neq i} \big[m(q_i, d_j) e(q_i, d_j) \\ &\quad + m(q_i, q_j) e(q_i, q_j) \\ &\quad + m(d_i^+, d_j) e(d_i^+, d_j) \big] \end{aligned} \tag{2}$$

Concretely, q_i is the task-aware query (a SOP, a known compliant product, or an external-catalog listing, depending on the task), d_i^+ is the canonical matching product, and the in-batch items in Z_i are queries and documents drawn from any of the four training tasks (§3).

The false-negative problem. By default ($m(a,b) = 1$ for all pairs), every non-paired in-batch item acts as a negative. In our multi-task setting, items from the same compliance category frequently co-occur within a batch, turning true positives into false negatives that the loss actively pushes apart. Multi-label structure compounds this: a single item may belong to multiple classes simultaneously, so the probability that two random in-batch items share at least one label grows with label cardinality.

Label-aware mask (illustrated in Fig. 5). We introduce a binary mask $m(a,b)$ into the denominator of Eq. (2). If items a and b share at least one compliance label, b is excluded from the negative set for a:

$$m(a,b) = \begin{cases} 0 & \text{if } \mathcal{Y}(a) \cap \mathcal{Y}(b) \neq \emptyset \\ 1 & \text{otherwise} \end{cases} \tag{3}$$

where $\mathcal{Y}(a)$ is the set of compliance labels for item a. For instance-level tasks (External→Product), the external-feed query has $\mathcal{Y}(q_i) = \emptyset$, so $m(q_i, \cdot) = 1$ and the loss reduces to standard InfoNCE—the desired behavior since false negatives are negligible there. The label sets $\mathcal{Y}(\cdot)$ come from internal compliance classifier outputs over the evaluation catalog, so each evaluated product carries its multi-label annotation with no additional manual labeling effort; computing the mask is a single set-intersection per pair. The mask is applied only to off-diagonal in-batch pairs ($j \neq i$ in Eq. (2)); a query is never compared with itself.

Matryoshka multi-scale loss. The final training loss wraps Eq. (1) with Matryoshka Representation Learning (Kusupati et al., 2024), computing the contrastive loss at dimensions $\{32, 64, 128, 256, 512, 1024\}$ and summing across scales, enabling reduced-dimensional evaluation without retraining.

3 Data Pipeline

Task formulation. Our training data is derived from binary compliance categories collected through an internal labeling pipeline over an approximately two-year period, spanning multiple languages and regulatory programs. The dataset comprises thousands of distinct SOPs, each with a substantial average number of human-labeled positive products. A single product may belong to multiple compliance categories. We reformulate classification labels as retrieval pairs: for each SOP, positive products serve as relevant documents. We construct three

category-level tasks: **SOP→Product** (regulatory-text query), **Product→Product** (known-compliant-product query), and **SOP+Product→Product** (regulatory text concatenated with an example product). We additionally incorporate one **instance-level** task, **External→Product**, linking external-catalog feed entries to internal-catalog counterparts. The training set totals 2M unique query–document pairs; each task uses a task-specific instruction template, and documents share a common template (Appendix D).

Hard negative mining. Hard negatives provide stronger training signal than random in-batch negatives (Gao et al., 2021), but mining must avoid false negatives. We adopt positive-aware mining (de Souza P. Moreira et al., 2025) with Qwen3-Embedding-8B as teacher (a stronger teacher surfaces harder negatives), then apply two filters: (i) a **score-based filter** that discards any candidate scoring above $s^+ \times \rho$ (ρ=0.95), where s^+ is the teacher's query–positive similarity, and (ii) a **label-overlap filter** that discards any candidate sharing at least one compliance label with the query. We retain the top-2 surviving candidates per query (ablation in §G.1).

Controlled Lexical Diffusion (CLD). Authentic external–product training pairs are scarce; we augment them with CLD, a six-stage curriculum where token-removal probability p rises linearly from 0.0 to 0.15. For positives, CLD removes tokens shared between external-feed and product listings (set intersection), forcing matching on sparser lexical overlap; for negatives, it removes tokens unique to either listing (set difference), making them superficially more similar. Only free-text fields are diffused; structured attributes (brand, model number, dimensions) are preserved as stable anchors.

Two-stage training. Stage 1 fully fine-tunes Qwen3-Embedding-0.6B on all four tasks with heterogeneous batching and label-aware exclusion (§2.1). Stage 2 applies LoRA (Hu et al., 2021) from the Stage 1 checkpoint on CLD-generated external–product pairs. A separate stage is necessary because CLD's token perturbations can invalidate compliance labels, which would corrupt label masking for category-level tasks if mixed into Stage 1. LoRA preserves the multi-task representations while adapting to the augmented external-feed distribution. Configuration in Appendix F.

4 Experiments

4.1 Evaluation Data, Setup, and Metrics

For category-level retrieval, we hold out 100 SOPs spanning multiple languages and regulatory programs, totaling 201K human-labeled products; we construct SOP, Product, and SOP+Product queries from these. For instance-level retrieval, we use 49K human-labeled external–product pairs, plus 48K human-labeled product variant pairs (functionally identical products differing only in non-functional attributes such as color or size); the variant task is excluded from training to test zero-shot transfer. We evaluate at two scales: a **298K development pool** (201K category-level + 97K instance-level products, controlled ablations at full and 128 dimensionality), and a **large-scale index** (200M products sampled from a larger internal catalog, with all instance-level targets guaranteed included). The large-scale index uses 128-dimensional embeddings for computationally constrained retrieval. The primary metric is Recall@K. For the 298K pool, K equals the per-class positive count. For the large-scale index, $K \in \{100, 1000\}$ is fixed. For instance-level tasks, K=1 reduces Recall@K to Hit@K. For category-level tasks at large scale, ground-truth labels come from internal compliance classifier outputs; training and evaluation SOPs are disjoint, and core findings are validated on the human-labeled 298K pool.

4.2 Large-scale retrieval

We evaluate our fine-tuned model on a large-scale item index spanning multiple marketplaces and languages, using 128-dimensional embeddings.[3] Table 1 reports per-stage results across all five retrieval tasks.[4]

Stage 1 vs. Stage 2. Stage 1 drives the dominant category-level gains, improving Recall@1000 by +61–224% across the three SOP tasks. Stage 2 adds substantial External→Product improvement (R@100: +10.3%) while category-level recall degradation is less than 3.5% relative from Stage 1. The final model improves category-level R@1000 by +57–233% over the base and reaches near-ceiling external-feed retrieval (R@1000 = 0.972), all with a 0.6B model at no increase in inference cost. While absolute category-level recall at fixed

[3] At this index size, ANN retrieval cost motivates 128-dimensional embeddings.

[4] Extended results including comparison with a prior internal baseline are provided in Appendix H.

Table 1: Retrieval performance on large-scale item index (128d). Stage 1: full fine-tuning; Stage 2: + external-feed LoRA. †Variation Family task is excluded from training. Bold marks the best result across conditions; Δrel compares the final model (+S2) to the base.

	Base		+S1		+S2		Δrel	
Task	R@100	R@1000	R@100	R@1000	R@100	R@1000	R@100	R@1000
Category-level retrieval								
SOP→Product	.024	.079	**.088**	.256	.085	**.264**	+248%	+233%
Product→Product	.100	.212	**.141**	**.342**	.138	.333	+37%	+57%
SOP+Product→Product	.064	.158	.134	**.362**	**.143**	.361	+125%	+128%
Instance-level retrieval								
External→Product	.467	.629	.854	.924	**.942**	**.972**	+102%	+54%
Product→Var.†	.705	.832	.688	.866	**.818**	**.940**	+16%	+13%

K=1000 appears modest, the embedding-based retrieval serves as a high-recall candidate generator where K can be set much larger before a downstream classifier makes the final decision. The +233% relative improvement translates directly into broader coverage. The final model also outperforms the prior internal baseline on all five tasks by +69-281% Recall@1000 on SOP-query tasks, and +10% on variant retrieval despite never training on it (Appendix H).

Zero-shot transfer to Product Variation family retrieval. The Variation family task retrieves child products sharing identical grouping attributes (brand, item form, etc.) but differing in non-functional themes (size, color), and is excluded from training. Stage 1 alone yields mixed results (-2% at R@100), as category-level retrieval does not require distinguishing individual products within a class. Stage 2 dramatically improves variant retrieval (R@100: +16%, R@1000: +13% over base). external-feed LoRA trains instance-level retrieval, aligning one specific external product to one specific product despite surface-form differences, precisely the skill that variant identification requires.

4.3 Development pool analysis

We use a 298K development pool for controlled analysis of the pipeline, with results shown in Table 2. The off-the-shelf Qwen3-Embedding-8B is reported as a scaling reference, not a scientific baseline: it is 13× more expensive at inference, which makes it unsuitable for retrieval over catalogs at this scale, and it is not fine-tuned. The effect of the mask is measured separately in §5, where every model compared is the same fine-tuned 0.6B and only the loss changes.

Learned query fusion. In the base model, concatenating SOP text with an example product *hurts* retrieval: Product-only (0.537) > SOP+Product (0.513) > SOP-only (0.367). After fine-tuning, the ranking reverses: SOP+Product (0.742) > SOP-only (0.697) > Product-only (0.668), outperforming either query type alone by 6–11%. The model learns to fuse complementary signals: the SOP defines *what to look for* (regulatory scope) and the example product defines *what it looks like* (product attributes), all without any architectural modification. This provides a practical few-shot mechanism: a single example product dramatically improves retrieval for new SOPs.

Table 2: Retrieval on 298K pool (Recall@K, K = per-class positive count). Superscripts show relative performance loss when truncating from full to 128 dimensions. †Untrained direction. ‡Off-the-shelf reference. Gemma in Appendix I.

	Full dim		128d	
Task	Base	FT	Base	FT
Qwen3-Emb. 0.6B (1024d)				
SOP→Product	.367	**.697**	$.298^{-19\%}$	$\mathbf{.677}^{-3\%}$
Product→Product	.537	**.668**	$.496^{-8\%}$	$\mathbf{.667}^{-0\%}$
SOP+Product→Product	.513	**.742**	$.436^{-15\%}$	$\mathbf{.733}^{-1\%}$
External→Product	.538	**.809**	$.410^{-24\%}$	$\mathbf{.789}^{-2\%}$
Product→SOP†	.749	**.838**	$.630^{-16\%}$	$\mathbf{.819}^{-2\%}$
Qwen3-Emb. 8B‡ (4096d)				
SOP→Product	.649	—	$.547^{-16\%}$	—
Product→Product	.594	—	$.565^{-5\%}$	—
SOP+Product→Product	.692	—	$.627^{-9\%}$	—
External→Product	.802	—	$.724^{-10\%}$	—
Product→SOP†	.814	—	$.735^{-10\%}$	—

Dimensional robustness. Matryoshka-trained models support truncation to lower dimensions without retraining. The superscripts in Table 2 show that base models lose 8–24% per task when truncated from full to 128 dimensions, while fine-tuned models lose at most 3%. Fine-tuning concentrates task-relevant information into the first few embedding dimensions, so truncation discards little useful signal. This enables 128-dimensional

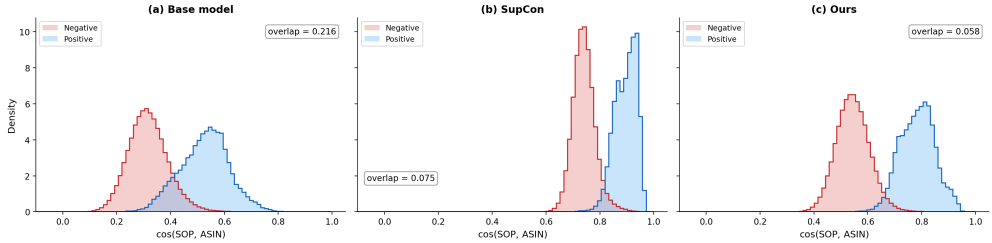

Figure 2: SOP–Product cosine similarity distributions. Overlap area (lower is better) reported per panel.

retrieval at 8× reduced ANN cost with minimal retrieval degradation.

Reverse retrieval without reverse training. The model is trained on SOP→Product but never the reverse. Fine-tuning improves Product→SOP from 0.749 to 0.838 (+12%), surpassing the off-the-shelf 8B (0.814), despite swapped prompt roles never seen in training. This indicates genuine semantic alignment rather than prompt-slot memorization.

Fine-tuning vs. model scaling. Our fine-tuned 0.6B matches or exceeds the off-the-shelf Qwen3-Embedding-8B on all five tasks at full dimensionality. At 128d, it still outperforms the 8B at its full 4096d on four of five tasks (External→Product: −1.6% relative). The 8B is larger and is used without fine-tuning, so we treat it as a scaling reference rather than as evidence for our method's effectiveness . It answers a different and practical question: can a fine-tuned small model replace a large model used off the shelf? This is the choice an industrial system faces. The effect of the mask is measured separately in §5, where every model compared is the same fine-tuned 0.6B and only the loss changes. For compute-constrained settings where the larger model is impractical, task-aware fine-tuning of the 0.6B model closes, and in most cases eliminates, the gap with the off-the-shelf 8B, delivering comparable or superior retrieval at 13× lower inference cost and 32× lower index size.

Unified vs. per-task models. An alternative design is one specialist embedding model per task. In our setting all tasks retrieve against the same target - the internal catalog - so a unified model lets the catalog be embedded and indexed *once*; Per-task specialists would instead multiply both embedding generation and ANN index footprint by the number of tasks, and re-incur both on every catalog update, which is prohibitive at the scale of our catalog.

Cross-architecture generalization. To verify that the gains are not specific to Qwen3, we apply the same training pipeline to EmbeddingGemma-0.3B (Vera et al., 2025), an encoder-only model. Gemma shows comparable improvement (+47% average category-level Recall@K vs. +49% for Qwen3), confirming that the gains stem from the contrastive framework rather than the model architecture. Full results about Gemma are in Appendix I.

5 Ablation Studies

Table 3 compares three strategies for handling in-batch false negatives: (1) InfoNCE with label-aware masking (ours), (2) Vanilla InfoNCE without masking: the identical multi-task fine-tune with only the mask removed, serving as the multi-task baseline, and (3) SupCon, which treats label-overlapping items as additional positives (implementation in Appendix G.2). SupCon is *hard* inclusion: it assigns every label-overlapping item full positive weight. To test whether softer inclusion recovers the gap, we additionally evaluate two graded variants: a soft-label loss that gives the mined hard negative a partial-positive weight set by a softmax over query–candidate similarities, and circle loss (Sun et al., 2020), which adaptively reweights pairs by their distance from an optimal similarity. Both still underperform exclusion and per-task results and discussion are in Appendix G.3. Additional ablations are in Appendix G.1 and G.4.

Exclusion outperforms inclusion. SupCon degrades every task; the damage is worst on External→Product (−16.8%) where external-feed products carry no category labels so SupCon cannot add positives, yet still distorts the shared embedding space by pulling same-category products together, destroying the per-product distinctions instance-level retrieval requires.

Why exclusion wins: embedding geometry. Retrieval quality is governed by overlap between positive and negative cosine distributions (Fig. 2). SupCon reduces overlap from 0.216 (base) to 0.075, but pulling all same-category items together shifts negatives rightward nearly as much as positives.

Table 3: Ablation: InfoNCE with label-aware masking (Ours) vs. removing the mask (w/o) and replacing InfoNCE with SupCon. Recall@K, dim=1024. †Not directly trained. Ours, w/o, and $\Delta\%$ report range over two independent seeds; $\Delta\% = (\text{Ours} - \text{w/o})/\text{Ours}$. External→Product and Product→SOP were not part of the two-seed study (placeholders, to be filled). SupCon is a single-seed main-pool comparison; its Rec. and $\Delta\%$ are relative to the main-pool Ours in Table 2.

	Mask vs. no-mask (2 seeds)			SupCon	
Task	Ours	w/o	Avg $\Delta\%$	Rec.	Avg $\Delta\%$
SOP→Product	.689–.695	.657–.660	-4.9%	.662	−5.0%
Product→Product	.690–.696	.678–.688	-1.5%	.618	−7.5%
SOP+Product→Product	.756–.760	.739–.746	-2.1%	.686	−7.5%
External→Product	.806–.809	.79–.794	-2.2%	.673	−16.8%
Product→SOP†	.839	.825	-1.7%	.809	−3.6%

Our method does not pull same-category items together; it only prevents false negatives from being pushed apart, concentrating the learning signal on the designated positive and producing the lowest overlap (0.058). Retrieval benefits more from clean positive–negative separation than from category-cluster compactness. A gradient view makes the same prediction. With one designated positive, the entire numerator gradient pulls the query toward that single pair; inclusion spreads this pull across every same-label positive, weakening the signal on the true pair, while the resulting category-cluster contraction drags nearby negatives along so positive–negative separation barely improves. This hurts instance-level tasks most, since they must separate individual products within a category rather than compact them — consistent with the largest degradation on External→Product (−16.8%).

Masking provides complementary gains. Removing the mask from InfoNCE degrades every category-level task, most on SOP→Product where same-category products co-occur most frequently in batches and are incorrectly penalized as negatives, and least on Product→Product; the untrained Product→SOP also benefits. To check that these gains are not an artifact of a single run, the Ours columns of Table 3 report range over two independent training seeds: the mask's gain stays clearly positive on the SOP-query tasks. Masking adds zero compute cost but reliably improves retrieval wherever false-negative collisions are frequent.

6 Conclusion

When category-level and instance-level retrieval tasks coexist in the same training batch, existing multi-label contrastive methods that include label-overlapping items as additional positives cannot be applied uniformly. We show that excluding false negatives from the contrastive loss is simpler, cheaper, and strictly better than including them as positives. A single 0.6B-parameter model trained with this approach unifies four heterogeneous retrieval tasks, matches or exceeds the 13× larger Qwen3-Embedding-8B at 32× lower dimensionality, and generalizes to unseen retrieval tasks excluded from training. Contrastive fine-tuning also teaches the model to fuse complementary query signals, combining regulatory text with an example product, providing a practical few-shot retrieval mechanism.

Limitations

We evaluate on e-commerce product compliance; generalization to other heterogeneous retrieval domains (legal, biomedical) is plausible but unverified. The label-aware mask assumes reliable multi-label annotations; noisy or partial labels could cause it to wrongly exclude informative negatives. The mask is also less suited to very coarse label sets: if categories are few and broad, most of a batch shares a label with the query and is excluded, leaving too few in-batch negatives to learn from. Our compliance labels are fine-grained (thousands of SOPs), so this does not arise here. Large-scale category-level ground truth comes from internal classifier outputs, not human annotation; we mitigate via the human-labeled 298K pool, but classifier biases may still influence absolute numbers. Stage 2 LoRA adapts to a synthetic external-feed distribution that may differ from real-world feeds in ways our perturbation does not capture. Finally, the problem we study is not well represented by existing public benchmarks: retrieval over a shared catalog where some items carry multi-label annotations and some tasks carry no shared labels at all. General retrieval suites (MTEB (Muennighoff et al., 2023), BEIR (Thakur et al., 2021)) provide no per-item label sets, and the closest public multi-label resource (MultiEURLEX (Chalkidis et al., 2021)) is a classification dataset without a retrieval

pool. We therefore cannot evaluate the mask on any dataset other than our own, which is why the cross-domain generalization (e.g., legal, biomedical) noted above remains unverified. We also cannot release the internal catalog and its compliance labels, so our results cannot be independently replicated. Building a public heterogeneous retrieval benchmark that reflects this setting is future work.

Ethics Statement

Scope of the model. The embedding model studied in this paper is evaluated only as a high-recall candidate-generation component in a controlled research setup. It is not a standalone compliance decision-maker, and the recall numbers reported here are intended for *cross-model benchmarking under fixed experimental conditions*. They should not be interpreted as operational system recall. In practical high-recall retrieval settings, a candidate generator can be configured with larger top-K values and combined with complementary non-neural filters before subsequent review or classification.

Data and privacy. Training data is derived from internal compliance classifications used for this study. No personal data, customer queries, or seller-private information are used. All examples in the paper are anonymized.

References

Alexandre Audibert, Aurélien Gauffre, and Massih-Reza Amini. 2025. Multi-label contrastive learning : A comprehensive study. *Preprint*, arXiv:2412.00101.

Ilias Chalkidis, Manos Fergadiotis, and Ion Androutsopoulos. 2021. MultiEURLEX – a multi-lingual and multi-label legal document classification dataset for zero-shot cross-lingual transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6974–6996, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ching-Yao Chuang, Joshua Robinson, Lin Yen-Chen, Antonio Torralba, and Stefanie Jegelka. 2020. Debiased contrastive learning. *Preprint*, arXiv:2007.00224.

Gabriel de Souza P. Moreira, Radek Osmulski, Mengyao Xu, Ronay Ak, Benedikt Schifferer, and Even Oldridge. 2025. Nv-retriever: Improving text embedding models with effective hard-negative mining. *Preprint*, arXiv:2407.15831.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6894–6910.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *Preprint*, arXiv:2004.10964.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Tri Huynh, Simon Kornblith, Matthew R. Walter, Michael Maire, and Maryam Khademi. 2022. Boosting contrastive self-supervised learning with false negative cancellation. *Preprint*, arXiv:2011.11765.

Nora Kassner and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7811–7818.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 18661–18673.

Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, and Ali Farhadi. 2024. Matryoshka representation learning. *Preprint*, arXiv:2205.13147.

Mengfei Lan, Lecheng Zheng, and Halil Kilicoglu. 2026. Biohicl: Hierarchical multi-label contrastive learning for biomedical retrieval with mesh labels. *Preprint*, arXiv:2604.15591.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2025. Nv-embed: Improved techniques for training llms as generalist embedding models. *Preprint*, arXiv:2405.17428.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *Preprint*, arXiv:2308.03281.

Niklas Muennighoff, Hongjin Su, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, and Douwe Kiela. 2024. Generative representational instruction tuning. *arXiv preprint arXiv:2402.09906*.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. MTEB: Massive text embedding benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.

Ralph Peeters, Reng Chiz Der, and Christian Bizer. 2023. Wdc products: A multi-dimensional entity matching benchmark. *Preprint*, arXiv:2301.09521.

Abhilasha Ravichander, Matt Gardner, and Ana Marasovic. 2022. Condaqa: A contrastive reading comprehension dataset for reasoning about negation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8729–8744.

Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2021. Contrastive learning with hard negative samples. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2023. One embedder, any task: Instruction-finetuned text embeddings. *Preprint*, arXiv:2212.09741.

Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. 2020. Circle loss: A unified perspective of pair similarity optimization. *Preprint*, arXiv:2002.10857.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogenous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation learning with contrastive predictive coding. *Preprint*, arXiv:1807.03748.

Henrique Schechter Vera and 1 others. 2025. Embeddinggemma: Powerful and lightweight text representations. *Preprint*, arXiv:2509.20354.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Zeqiang Wei, Kai Jin, Zheng Zhang, and Xiuzhuang Zhou. 2024. Multi-label contrastive hashing. *Pattern Recognition*, 149:110239.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Pingyue Zhang and Mengyue Wu. 2024. Multi-label supervised contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 16786–16793.

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

A Related Work

General-purpose text embedding models. Existing work has produced a family of general-purpose embedding models trained on large-scale multi-task mixtures. E5 (Wang et al., 2022) uses weakly supervised contrastive pretraining over web-scale text pairs; GTE (Li et al., 2023) extends this with multi-stage contrastive learning across diverse NLP tasks; INSTRUCTOR (Su et al., 2023) introduces task-specific instruction prompts so that a single model handles retrieval, classification, and clustering without per-task fine-tuning; and NV-Embed (Lee et al., 2025) adds a two-stage pipeline that separates retrieval training from non-retrieval instruction tuning. Our work shares the instruction-prompted multi-task paradigm but targets a different regime: *domain-specific* fine-tuning on a narrow set of compliance retrieval tasks. We show that a fine-tuned 0.6B model outperforms a 13× larger general-purpose model, demonstrating that for specialized retrieval, task-aware fine-tuning is more cost-effective than model scaling.

Domain-specific embedding adaptation. General-purpose embedding models often underperform on specialized domains due to vocabulary mismatch and distributional shift (Thakur et al., 2021). Continued pretraining on in-domain corpora (Gururangan et al., 2020) and contrastive fine-tuning on domain-specific pairs are common remedies. Our results extend this line of work: domain-aware contrastive fine-tuning of a 0.6B model not only closes the gap with but surpasses the off-the-shelf 8B model on all domain-specific tasks, at 13× lower inference cost and 32× smaller index size.

Hard negative mining. The quality of negative examples is critical for contrastive learning (Xiong et al., 2021; Robinson et al., 2021). de Souza P. Moreira et al. (2025) introduce positive-aware filtering, using the positive relevance score as an upper bound to discard likely false negatives during mining. We adopt this strategy with a stronger teacher model (Qwen3-Embedding-8B) for candidate generation and show that combining mined

hard negatives with in-batch label-aware masking yields complementary gains.

False negatives and multi-label contrastive learning. MulSupCon (Zhang and Wu, 2024) and follow-ups (Wei et al., 2024; Audibert et al., 2025; Lan et al., 2026) extend SupCon to multi-label settings, weighting positives by label overlap. Our contrast with these methods — why exclusion rather than weighted multi-positive inclusion suits heterogeneous batches that mix label-bearing and label-free tasks — is discussed in §1 (paragraph 4) and empirically compared in §5. Beyond multi-label methods, false negatives in contrastive learning have also been addressed without class labels: Chuang et al. (2020) correct the contrastive loss in distribution, re-weighting the negative term under an assumed class prior; Huynh et al. (2022) detect likely false negatives via embedding similarity and discard or promote them; and de Souza P. Moreira et al. (2025) filter mined negatives whose teacher score exceeds a positive-anchored threshold. These similarity-driven estimates are complementary to our mask, which uses ground-truth label overlap and therefore also removes false negatives that the current model scores as dissimilar — precisely those that similarity-based detection misses early in training. Our contribution is thus not false-negative handling per se, but a label-overlap exclusion mechanism that operates in-loss, costs nothing to compute, and degrades gracefully to standard InfoNCE under heterogeneous label structures.

Few-shot and exemplar-augmented retrieval. Augmenting queries with exemplar documents has been explored in retrieval-augmented generation and few-shot settings (Muennighoff et al., 2024). We demonstrate that combining regulatory text with an example product (exemplar-augmented query) reverses performance from the weakest to the strongest query type after fine-tuning, providing evidence for this approach in industrial compliance retrieval.

Matryoshka representations. Kusupati et al. (2024) introduce Matryoshka Representation Learning, training embeddings to be effective at multiple dimensionalities. Our base model incorporates this approach, and we show that domain-specific fine-tuning preserves and even enhances this dimensional robustness, enabling $8\times$ embedding compression with minimal retrieval degradation.

Negation understanding in language models. Negation remains a known challenge for language models and embedding systems (Kassner and Schütze, 2020; Ravichander et al., 2022). We provide empirical evidence that our fine-tuned model effectively leverages negation information in regulatory text (e.g., "keywords not of interest" in SOPs), contrary to common assumptions.

B Problem Setting

Products listed on large e-commerce platforms may not always satisfy regulatory requirements; identifying such products is important for customer safety and platform integrity. This setting requires matching very large product collections against many regulatory rules, a scale at which a single end-to-end classifier is impractical. We study a retrieval formulation in which a high-recall embedding model narrows the item collection to a candidate pool for subsequent classification. Our experiments evaluate whether a compact 0.6B-parameter model can improve this candidate-generation step relative to a prior internal baseline, while preserving low inference and index cost.

C SOP

A structural mock-up of an SOP is shown in Fig. 3, illustrating the class description, positive keywords, and negative keywords that define a compliance category.

I'm interested in classifying products into one of the following categories: Category A

`<class_description>`
Describe the categories of interest: A natural-language description of the regulatory category, including its scope, the kinds of products it covers, and any distinguishing functional or material attributes.
`</class_description>`

`<keywords_of_interest>`
Products that should be classified as Category A often contain keywords like *[positive keyword 1]*, *[positive keyword 2]*, ...
`</keywords_of_interest>`

`<keywords_not_of_interest>`
Products that should be excluded from this classification task contain keywords like *[exclusion keyword 1]*, *[exclusion keyword 2]*, ...
`</keywords_not_of_interest>`

Figure 3: Mock-up SOP query template. The class description defines the regulatory category, positive keywords guide retrieval, and negative keywords explicitly exclude unrelated products.

D Prompt Design

Fig. 4 shows prompt templates for the five retrieval tasks.

E Label-Aware Masking

Fig. 5) illustrated the Label-aware mask in an example batch.

F Training Hyperparameters

Table 4 reports the complete training configuration. Stage 1 performs full fine-tuning on the multi-task compliance dataset; Stage 2 adapts the Stage 1 checkpoint with LoRA on synthetic external–product pairs.

Table 4: Training hyperparameters. Values reported as Stage 1 / Stage 2; a single value indicates the setting is shared.

Hyperparameter	**Stage 1 / Stage 2**
Model	
Base model	Qwen3-0.6B / S1 ckpt-1000
Training type	Full / LoRA
Infrastructure	
Nodes $\times$ GPUs	8×8 / 1×8
Per-GPU batch	16
Grad. accum. steps	2
Effective batch	2048 / 256
DeepSpeed	ZeRO-3 / ZeRO-2
Precision	bf16
Optimization	
Learning rate	10^{-5} / 2×10^{-5}
LR scheduler	Cosine
Warmup ratio	0.01 / 0.05
Loss function	
Loss type	Matryoshka InfoNCE
Temperature τ	0.02
Matryoshka dims	{32, 64, . . . , 1024}
Hard negatives K	2 / —
LoRA (Stage 2 only)	
Rank / α / drop.	— / 16 / 32 / 0.05
Data	
Tasks	4 (excl. var.) / ext.–item
Task weights (Stage 1)	.52 / .19 / .19 / .10
Max seq. length	3072

Notes. The Matryoshka InfoNCE loss computes a separate InfoNCE loss at each truncated dimension $d \in \{32, 64, 128, 256, 512, 1024\}$ and returns the unweighted sum. All six dimensions are evaluated at every training step. The denominator (Eq. 2) includes query-to-document (QD), query-to-query (QQ), and document-to-document (DD) similarity blocks, following Qwen3-Embedding's training formulation (Zhang et al., 2025). An identity mask sets any in-batch pair with cosine similarity ≥ 0.999 to $-\infty$, preventing near-duplicate items from acting as false negatives; each query's own document block (positive + hard negatives) is exempt.

Stage 2 disables label masking because the synthetic external-feed data lacks namespace labels, and omits hard negatives because the synthetic generation process (Section 3) already controls negative difficulty through token-level perturbation. LoRA with rank 16 and a higher learning rate (2×10^{-5}) preserves the multi-task representations from Stage 1 while efficiently adapting to the augmented external-feed distribution.

G Additional Ablations

G.1 Hard Negatives

Table 5 compares 1, 2 and 5 hard negatives per positive. At checkpoint-1000 (dim=1024), using 2 hard negatives outperforms 5 on all tasks by a consistent margin (-0.004 average Recall@K). Examining the training progression reveals that 5 hard negatives provides a slight advantage at earlier checkpoints (400–800) but degrades faster, while 2 hard negatives peaks later and reaches a higher final performance. This suggests that additional hard negatives accelerate early learning but introduce more noise from borderline examples, leading to faster overfitting. We select 2 hard negatives as the best trade-off between training signal strength and stability.

Table 5: Ablation: hard negatives per positive (Recall@K, dim=1024, checkpoint-1000). Using 2 hard negatives provides the best trade-off.

Task	HN=1	HN=2	HN=5
SOP→Product	0.680	**0.695**	0.691
Product→Product	0.668	**0.696**	0.668
SOP+Product→Product	0.738	**0.756**	0.741

G.2 SupCon variant for ablation

The supervised contrastive (SupCon) variant used in Section 5 replaces the standard single-positive InfoNCE objective with a *per-label* SupCon loss adapted from Khosla et al. (2020). All items in the batch (queries, positive documents, and hard negative documents) are pooled into a single set of N embeddings. Each item inherits its namespace membership labels from the training data: queries receive `query_labels`, positive documents receive `pos_labels`, and hard negatives receive `neg_labels`.

For each namespace label l present in the batch, the loss identifies all items sharing that label and treats them as mutual positives. For anchor i with

(a) Product Index and Query Prompt

```
Product:
item_name: [product name]
manufacturer: [manufacturer]
product_type: [type code]
product_group: [group code]
item_type_keyword: [type keyword]
[additional structured attributes]
```

(b) SOP Query Prompt

```
Retrieve products that match the following Standard
Operating Procedure (SOP) criteria:
SOP: I'm interested in classifying products into one of the
following categories: Category B
⟨class_description⟩
[abstract description of the category, including its scope
and any distinguishing functional or material
attributes. . . ]
⟨/class_description⟩
⟨keywords_of_interest⟩
[positive keywords in target marketplace language]. . .
⟨/keywords_of_interest⟩
⟨keywords_not_of_interest⟩
[exclusion keywords in target marketplace language]. . .
⟨/keywords_not_of_interest⟩
```

(c) SOP+Product Query Prompt (Exemplar-Augmented)

```
Find products matching the following SOP criteria, using the
provided example match product as a reference match.
SOP: I'm interested in classifying products into one of the
following categories: Category B
⟨class_description⟩... ⟨/class_description⟩
⟨keywords_of_interest⟩... ⟨/keywords_of_interest⟩
⟨keywords_not_of_interest⟩... ⟨/keywords_not_of_interest⟩
Example Match Product:
item_name: [product name]
manufacturer: [manufacturer]
product_type: [type code]
product_group: [group code]
[additional structured attributes]
```

(d) External Query Prompt

```
Match external catalog products to equivalent products:
Retrieve products that share the same attributes,
specifications, and features as the query product from an
external catalog, prioritizing semantic similarity over
exact keyword matches. Missing information in any attribute
does not disqualify a match as long as there is no explicit
contradiction.
Query Product:
[external catalog product attributes]
```

(e) Variation Query Prompt

```
Retrieve variant products for the query product:
Two products are variants if they are functionally identical.
Use semantic matching rather than exact string matching when
comparing functional attributes. Color, size, and quantity
are non-functional and may vary within a variation family.
Query Product:
item_name: [product name]
brand: [brand]
bullet_point: [bullet point]. . .
color: [color value]
. . .
```

Figure 4: Prompt templates for the five retrieval tasks. Blue text denotes the system instruction prefix that defines the task. **(a)** Product documents are indexed using structured product attributes (shared across all tasks as the retrieval target). **(b)** SOP queries contain a category description, positive keywords, and negative keywords in the target marketplace language. **(c)** Exemplar-augmented queries concatenate the SOP with an example matching product, providing both textual criteria and a concrete reference. **(d)** External-feed queries match products across external catalogs via semantic attribute similarity. **(e)** Variation queries retrieve functionally identical products, ignoring non-functional attributes such as color and size.

label l, the per-label loss is:

$$\mathcal{L}_{i,l} = -\frac{1}{|P_l(i)|} \sum_{p \in P_l(i)} \log \frac{\exp(z_i \cdot z_p/\tau)}{\sum_{a \neq i} \exp(z_i \cdot z_a/\tau)}, \tag{4}$$

where $P_l(i)$ is the set of items sharing label l with i (excluding i itself), and the denominator sums over all $N-1$ other items regardless of label. The per-label decomposition prevents transitive contamination: a multi-label item belonging to namespaces A and B only pulls toward A-members when computing the A sub-loss, not toward B-members. For instance-level tasks (External, Variation) that lack namespace labels, the loss degenerates to standard single-positive InfoNCE.

This SupCon formulation replaces rather than augments the InfoNCE objective during training. All other hyperparameters (temperature τ=0.02, 2 hard negatives, 8-node training, Matryoshka dimensions) are held constant to isolate the effect of including same-category items as additional positives versus excluding them as false negatives.

G.3 Inclusion-family variants: soft-label and circle loss

SupCon (§5) is the *hard* form of inclusion: every label-overlapping item is treated as a full positive. To test whether *graded* inclusion — assigning partial rather than full positive weight to items other than the designated positive — closes the gap to exclusion, we evaluate two additional objectives under the identical training recipe as our model.

Soft-label weighting. The soft-label contrastive loss gives the mined hard negative a partial *positive* weight in the numerator, instead of treating it purely as a negative. For a query q with positive p and hard negative h, the per-sample loss is the weighted sum $-w_1 \log \frac{e_\tau(q,p)}{Z} - w_2 \log \frac{e_\tau(q,h)}{Z}$, where the weights $[w_1, w_2] = \mathrm{softmax}([\cos(q,p), \cos(q,h)]/\tau_w)$ are computed from the query–candidate cosine similarities and detached from the gradient (τ_w=0.1).

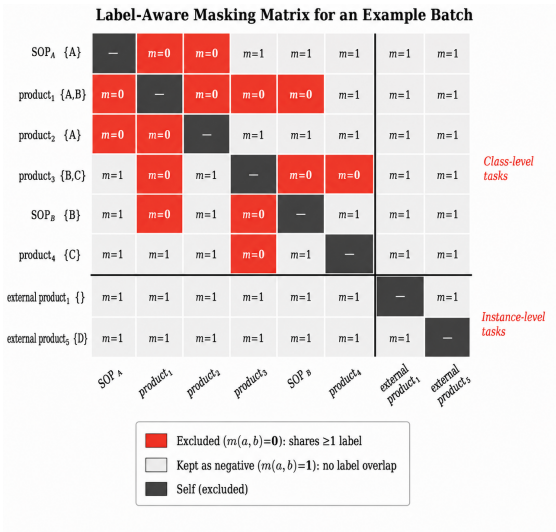

Figure 5: Red: excluded ($m(a,b)=0$, shares ≥ 1 label). Gray: kept as negative. External row: no category labels, no masking.

When the hard negative is far from the query, $w_2 \to 0$ and the loss reduces to standard InfoNCE; when it is close, w_2 grows and the hard negative receives partial-positive gradient. Inclusion here is thus soft and self-calibrating, driven by embedding similarity rather than by label overlap.

Circle loss. Circle loss (Sun et al., 2020) keeps the same positive and negative pairs but re-weights each adaptively, applying a larger gradient to pairs farther from their optimal similarity — an adaptive rather than fixed weighting.

Results. Table 6 reports Recall@K on the three category-level query types, each condition measured against the matched InfoNCE+mask baseline. Both variants underperform exclusion on every category-level task. On average, soft-label loses -3.3% and circle loss -4.1% relative Recall@K; adding SupCon's -5.0 to -7.5% on the same tasks (Table 3), the four objectives order by how aggressively each assigns positive weight away from the single designated pair: exclusion (none) $>$ soft-label (a soft weight on the hard negative) $>$ circle loss (adaptive re-weighting) $>$ SupCon (full weight on all same-category items).

G.4 Negation keyword ablation

SOPs contain a `<keywords_not_of_interest>` field listing terms the compliance category explicitly excludes. We train and evaluate a variant with this field stripped from all SOP queries. Removing negation keywords degrades average Recall@K by -3.6% relative (Table 7), with the largest impact on SOP→Product (-5.3%), where negation keywords directly constrain retrieval scope. The effect is consistent across checkpoints, confirming that the model has learned to leverage negation information, challenging the common assumption that embedding models cannot handle negation (Kassner and Schütze, 2020).

Table 6: Variants vs. exclusion (Recall@K, dim=1024, checkpoint-1000, three category-level query types). S→P: SOP→Product; P→P: Product→Product; S+P→P: SOP+Product→Product. $\Delta\%$ is relative to the matched InfoNCE+mask baseline (Ours).

		Soft-label		Circle	
Task	Ours	Rec.	$\Delta\%$	Rec.	$\Delta\%$
S→P	**.695**	.656	-5.6%	.631	-9.2%
P→P	**.696**	.676	-3.0%	.683	-1.9%
S+P→P	**.756**	.741	-2.0%	.740	-2.0%
Average	**.715**	.691	-3.3%	.685	-4.1%

Table 7: Ablation: removing negation keywords from SOP queries (Recall@K, dim=1024).

Task	Ours	– Neg. KW	$\Delta\%$
SOP→Product	**0.695**	0.674	-3.1%
Product→Product	**0.696**	0.661	-5.3%
SOP+Product→Product	**0.756**	0.738	-2.4%

H Extended large-scale results

Table 8 provides the full large-scale evaluation including comparison with a prior internal baseline. Our fine-tuned 0.6B model (Stage 2) outperforms the prior internal baseline on all five tasks. The advantage is largest on SOP-query tasks (+69–281% at Recall@1000), where the prior internal baseline, which was not trained with SOP-specific contrastive objectives, performs below our proposed model. On external-feed product matching, where the prior internal baseline was specifically optimized, our model still holds a narrow advantage (+1.5% at R@1000). On variant retrieval, our model outperforms the prior internal baseline by +10% at R@1000 despite never training on this task, while the prior internal baseline was explicitly trained on variation pairs.

I Cross-Architecture Generalization

Table 9 reports results for EmbeddingGemma-0.3B (Vera et al., 2025), an encoder-only model with 768-dimensional embeddings, trained using the same Stage 1 pipeline as Qwen3-0.6B (Stage 2 external-feed LoRA not conducted). Gemma improves average category-level Recall@K by +47% relative (0.437→0.642), closely matching Qwen3's +48% (0.473→0.702). The per-task pattern is par-

Table 8: Full retrieval results on large-scale item index (128d). Base = Qwen3-Embedding-0.6B; S1 = Stage 1 (full fine-tuning); S2 = Stage 2 (+ external-feed LoRA); Internal = prior internal baseline. †Variant task excluded from our training. Best per row in **bold**.

Task	Metric	Base	+S1	+S2	Internal
SOP→Product	R@100	.024	.088	**.085**	.019
	R@500	.059	.182	**.196**	.046
	R@1000	.079	.256	**.264**	.069
Product→Product	R@100	.100	**.141**	.138	.133
	R@500	.173	**.263**	.260	.224
	R@1000	.212	**.342**	.333	.280
SOP+Product→Product	R@100	.064	.134	**.143**	.074
	R@500	.122	.274	**.279**	.156
	R@1000	.158	**.362**	.361	.214
External→Product	R@100	.467	.854	**.942**	.871
	R@500	.582	.906	**.966**	.938
	R@1000	.629	.924	**.972**	.957
Product→Var.†	R@100	.705	.688	**.818**	.581
	R@500	.801	.823	**.916**	.783
	R@1000	.832	.866	**.940**	.855

Table 9: Cross-architecture generalization: EmbeddingGemma-0.3B (768d), Stage 1 only. Same training framework as Qwen3-0.6B. Superscripts show relative performance loss when truncating from full to 128 dimensions. †Untrained direction.

	Full dim (768)		128d	
Task	Base	FT	Base	FT
SOP→Product	0.364	0.607	$0.293^{-19\%}$	$0.583^{-4\%}$
Product→Product	0.515	0.647	$0.484^{-6\%}$	$0.634^{-2\%}$
SOP+Product→Product	0.433	0.671	$0.364^{-16\%}$	$0.654^{-3\%}$
External→Product	0.457	0.694	$0.392^{-14\%}$	$0.658^{-5\%}$
Product→SOP†	0.610	0.826	$0.570^{-7\%}$	$0.814^{-1\%}$

allel: SOP→Product shows the largest gain (+67% vs. +90% for Qwen3), and dimensional robustness is similarly preserved (at most −5% under 6× truncation to 128d). This consistency across an encoder-only and a decoder-only architecture confirms that the gains stem from the task-aware contrastive framework, not model-specific properties.