Factorized Semantic IDs for Controllable Coarse-to-Fine Outfit Retrieval

Anonymous

September 17, 2026

Abstract

We study controllable outfit completion: retrieving a compatible item for a partial outfit while letting the caller specify what the returned item must be, its role (*e.g.* "shoes") and, optionally, its coarse color, without scoring every catalog item at query time. Our approach gives every catalog item a short ID made of three independent fields: role and color, assigned by fixed rules, and a compatibility code that groups items a frozen, role-conditioned Transformer teacher treats as interchangeable for the same kinds of outfits. This compatibility code comes from clustering the teacher's own item embeddings, not from a learned decoder. At query time, the caller's role and color constraints are enforced first, by intersecting these fields the way a database query intersects indexes on several columns, before any embedding similarity is computed; only the resulting small candidate set is then reranked with the teacher's continuous compatibility score. Under *ground-truth* color control, this factorized routing at $L{=}8$ retains **98.8%** of a role+color oracle's Recall@10 while scoring only 1,032 candidates (**11.3**$\times$ compression). Under *counterfactual* color control, that is, deliberately requesting a color different from the true target's so the system must actually honor an unfamiliar constraint rather than default to what it would retrieve anyway, factorized routing at $L{=}8$ recovers **88.2%** of the top-10 items the exhaustive constrained teacher would itself select, against **10.8%** for a matched-budget random control; both results hold stably across 3 independently trained teacher seeds (Section 7). Unlike prior compatibility-ranking methods, which offer no native mechanism to enforce such constraints, and unlike autoregressive Semantic-ID methods, which require a decoder and a collaborative-filtering training signal that outfit-compatibility data does not provide, our factorized index adds this controllability to an existing compatibility scorer using only fixed rules and clustering, at a small fraction of the cost of scoring the full catalog.

1 Introduction

Fashion outfit completion (predicting a missing garment given a partial look) is typically framed as pure ranking: encode the partial outfit, encode the catalog, and retrieve the nearest candidate. OutfitTransformer [Sarkar et al., 2023] is a strong instance of this recipe: a Transformer set encoder produces one query embedding, which is compared against candidate embeddings by nearest-neighbor search. This is efficient (one forward pass per query, independent of catalog size), but it offers no native mechanism for the caller to *constrain* what comes back (*e.g.* "show me a compatible *shoe*", or "…in *this* color family") without a separate, ad hoc post-filtering step over the full ranked list.

Generative retrieval methods such as TIGER [Rajput et al., 2023] suggest an alternative: assign every item a single short discrete Semantic ID and predict that ID directly, token by token, so that decoding is itself constraint-aware. TIGER's codes are trained with collaborative-filtering signal (co-purchase sequences) that has no counterpart in outfit-compatibility datasets, and its inference is autoregressive: the model must generate a valid code one token at a time before any item can be retrieved. We adopt the idea that a structured discrete ID is useful for constrained retrieval, but not the training signal or the generation mechanism. Critically, our Semantic ID is not a single opaque code: it is *factorized* into three separately addressable fields, role, color, and compatibility code, each assigned by a fixed rule or a clustering step rather than predicted by a decoder. Because the fields are separate, a caller can constrain any subset of them (role alone, or role and color together) by directly intersecting per-field lookup tables, the same way a database query filters rows by intersecting indexes on several columns. This intersection happens before any embedding similarity is computed, so it is what shrinks the catalog down to a small candidate set; nearest-neighbor search over the compatibility field then narrows that set further, and only the resulting handful of items is scored with the teacher's continuous embedding. This distinguishes our method from a plain nearest-neighbor lookup over the teacher's embedding space, which has no discrete fields to intersect and therefore no native way to guarantee a hard

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

constraint such as "must be a shoe" without scoring, or at least filtering, the entire catalog first. We refer to this as **Transformer-derived factorized Semantic-ID routing with continuous reranking**, to distinguish it explicitly from autoregressive Semantic-ID generation.

Our approach. We first train a role-conditioned Transformer teacher with InfoNCE on Polyvore Outfits-ND [Vasileva et al., 2018] context/target pairs. This teacher is a compatibility scorer only; it is not this paper's contribution. Our contribution is what we build on top of it: we assign every catalog item three separate, independently queryable fields, role, color, and a compatibility code, together forming the factorized Semantic ID. Role and color are constraints the caller can specify directly at query time (*e.g.* "role = shoes, color = color_3"); the compatibility code is not specified by the caller. It is a fixed cluster label obtained once, before any retrieval happens, by running k-means on the teacher's own item embeddings. Item and query embeddings come from the same encoder but represent different things: an item embedding encodes what a single item offers, while a query embedding encodes what a partial outfit still needs, conditioned on the requested role. InfoNCE training pulls a query embedding toward the embedding of whichever item actually completed that outfit, so items that repeatedly fill the same kind of gap (*e.g.* shoes that suit casual, denim-based outfits) end up close together in embedding space, not because they look alike, but because training pulled them toward the same query region. The compatibility code therefore does not claim that two items are compatible with each other; it groups items the teacher treats as interchangeable for the same queries, which is also why routing to the clusters nearest a query loses little: the clusters left out are ones the teacher would have scored low regardless. Retrieval proceeds in three discrete steps: (1) intersect the role and, if given, color fields to restrict the catalog to matching items; (2) among those items, keep only the ones whose compatibility-code cluster is among the top-L clusters closest to the query, shrinking the candidate set further without computing a single per-item similarity score; (3) compute the teacher's continuous similarity score only for this small remaining candidate set and return the top 10. We evaluate this pipeline under both ground-truth and counterfactual color constraints, and we separately establish, through two independent learning attempts, that the compatibility code is best obtained by simple geometric k-means rather than by learning the partition jointly with routing objectives.

Our contributions:

1. A **factorized Semantic ID** $\mathrm{SID}(i) = [r_i, c_i, k_i]$ built on top of a role-conditioned continuous compatibility teacher, supporting native discrete-constraint retrieval (role, color) without candidate re-encoding or post-hoc filtering; color-code interventions reuse the same query embedding, though a role change requires re-encoding the query (Section 4).
2. A **coarse-to-fine retrieval procedure** that composes role/color partitioning with compatibility-code routing and continuous reranking, achieving $\sim$11–23$\times$ candidate-scoring compression beyond typed role/color filtering at the operating points we report (L=4/8), while retaining most of the exhaustive constrained teacher's ranking quality.
3. A **ground-truth color-control evaluation** showing factorized routing retains **98.8%** of a role+color oracle's Recall@10 at **11.3**$\times$ compression (L=8).
4. A **counterfactual color-control evaluation** (probing target colors different from the ground-truth color) showing the system recovers **88.2%** of the exhaustive constrained teacher's own top-10 choices under a non-target color constraint, an 8$\times$ margin over a matched-budget random control, which shows that counterfactual slot intervention preserves compatibility-aware ranking rather than merely satisfying the constraint by default.
5. A **controlled negative result**: two independently implemented learned-partitioning methods (an EMA joint codebook and a Sinkhorn-balanced joint codebook), each evaluated under a collapse-resistant composite selection metric, both underperform frozen role-wise k-means at matched code budgets.

Figure 1 summarizes the pipeline. Section 2 positions this against prior compatibility, sequential-recommendation, and generative-retrieval work. Section 3 formalizes the task and the factorized ID. Section 4 describes the frozen teacher. Section 5 describes how each ID slot is obtained, including the negative result on learned partitioning. Section 6 describes the coarse-to-fine retrieval procedure. Section 7 reports both control evaluations. Section 8 states the evaluation's current limitations plainly.

2 Related Work

Outfit compatibility learning. Vasileva et al. [2018] introduced the Polyvore Outfits dataset and type-aware subspace embeddings (57.8% FITB). Han et al. [2017] modeled outfit sequences with bidirectional LSTMs. SCE-Net [Tan

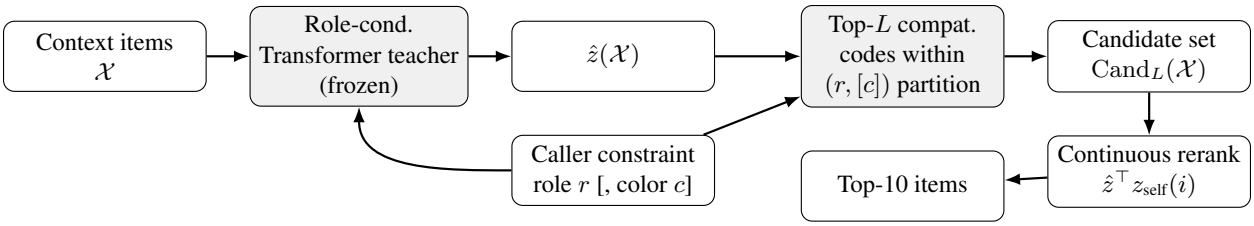


Figure 1: Coarse-to-fine factorized Semantic-ID retrieval. The teacher is a role-conditioned Transformer, frozen after training on InfoNCE; the caller-specified role r is fed into the teacher as a role token (Section 4) and, together with an optional color constraint c, also restricts routing to a matching partition of the catalog. The teacher produces one query vector $\hat{z}(\mathcal{X})$ per request; within the matching partition, the top-L nearest compatibility-code centroids to $\hat{z}(\mathcal{X})$ define a small candidate set $\mathrm{Cand}_L(\mathcal{X})$, which is reranked by the continuous teacher score. No component is autoregressive; the Semantic ID is used only to restrict the scoring budget.

et al., 2019] decomposes compatibility into unsupervised complementary conditions (59.1%), and CSA-Net [Lin et al., 2020] introduces category-specific subspace attention (63.7%). OutfitTransformer [Sarkar et al., 2023] encodes the partial outfit with a Transformer and a learned target token, producing one outfit-level query embedding retrieved by nearest-neighbor search (67.1% FITB). Our teacher (Section 4) follows this candidate-independent retrieval structure; we do not claim to improve on its reported accuracy; our contribution is the controllable, efficient retrieval structure we build on top of a comparable teacher.

Transformer-based recommendation. SASRec [Kang and McAuley, 2018] and BERT4Rec [Sun et al., 2019] apply self-attention over user interaction histories to predict the next item. Outfit completion instead operates on visual compatibility without user logs; our Transformer attends over image embeddings of context items rather than item-ID tokens, and is trained with a visual contrastive objective.

Generative retrieval and Semantic IDs. TIGER [Rajput et al., 2023] assigns items hierarchical RQ-VAE codes from collaborative-filtering embeddings and predicts the next item's code autoregressively. Differentiable Search Index [Tay et al., 2022] similarly predicts discrete document identifiers token-by-token for text retrieval. Both require a decoder capable of generating a valid code sequence and a training signal rich enough to shape it. Our Semantic ID is constructed differently: it is a *partition* of an already-trained continuous embedding space (role from metadata, color from a pretrained attribute encoder, compatibility code from k-means over teacher embeddings), used only to select a coarse-grained candidate set that a continuous score then reranks. This is closer in spirit to IVF/product-quantization-style approximate nearest-neighbor indexing [Jégou et al., 2011] (a coarse partition followed by exact scoring within the partition) than to autoregressive code generation, and we use this framing throughout rather than describing our system as generative retrieval.

Balanced clustering and codebook learning. MiniBatchKMeans [Sculley, 2010] provides the scalable geometric partitioning we use for the compatibility code. We also attempt to *learn* the partition jointly with a routing objective, using an EMA-updated codebook and a Sinkhorn-Knopp balanced assignment in the style of SwAV [Caron et al., 2020], which guarantees non-degenerate code occupancy during training. Section 5 reports that neither learned variant improves on frozen k-means for this task.

Multimodal fashion attribute learning. Forouzandehmehr et al. [2026] pretrain a multimodal slot-factorization encoder that produces typed attribute clusters (color, pattern, category, demographic) for fashion items. We reuse only their color slot (K=7) as the source of our color code c_i. We treat c_i as an *opaque* coarse color code throughout this paper and make no claim about what any individual code value visually represents.

Contrastive learning for fashion. FashionCLIP [Chia et al., 2022] provides the frozen 512-d image embeddings we use as input throughout; we do not fine-tune the backbone.

Positioning relative to prior work. Each thread above supplies one piece of our pipeline but not the combination. Compatibility learning methods [Vasileva et al., 2018, Han et al., 2017, Tan et al., 2019, Lin et al., 2020, Sarkar et al., 2023] train a strong continuous scorer, but none of them let a caller attach an explicit role or color constraint to a request; satisfying such a constraint would require filtering their ranked output after the fact, not before scoring. Generative retrieval [Rajput et al., 2023, Tay et al., 2022] shows that a discrete Semantic ID can make decoding itself constraint-aware, but only by learning to generate that ID with an autoregressive decoder and a training signal, co-purchase sequences, that outfit-compatibility datasets do not provide. Balanced-clustering and codebook methods [Sculley, 2010, Caron et al., 2020] supply mechanisms for obtaining a discrete partition, but say nothing on their own about outfit retrieval or query-time controllability. Our contribution combines these pieces rather than proposing a new one in isolation: we keep a continuous compatibility scorer as in the first thread, but attach to it a discrete, factorized index whose role and color fields come from fixed rules or a pretrained encoder and whose compatibility field comes from geometric clustering rather than learned generation. This lets role and color constraints be enforced exactly, by index intersection, before any embedding similarity is computed, using no training signal beyond what the compatibility scorer already requires.

Recent vision-language models and multimodal recommenders offer a different way to enforce a constraint: the caller states it as free text (or a reference image) and the model reasons over it inside a single forward pass. This handles open-vocabulary constraints (*e.g.* "navy" rather than one of our 7 fixed color codes) that our fixed-rule fields do not, but it moves constraint satisfaction back inside the network, with no guarantee that a stated hard constraint is actually honored, and at the inference cost of a full multimodal forward pass per query rather than a lookup. Standard approximate-nearest-neighbor libraries (*e.g.* Faiss IVF) sit closer to our approach, since they also partition the catalog before scoring, but an unconstrained ANN index partitions purely by embedding proximity; enforcing our role and color fields on top of it would still need either a post-hoc filter over the returned neighbors or a metadata-aware payload mask evaluated at query time. Our factorized index instead makes role and color first-class partition keys from the start, so the constraint is satisfied by construction of the index itself, not by a downstream check on an unconstrained result.

3 Problem Formulation

Given a catalog $\mathcal{C}$ of N fashion items and an outfit $\mathcal{O} = \{i_1, \ldots, i_n\}$, the *outfit completion* task asks: given context items $\mathcal{X} = \{i_1, \ldots, i_{n-1}\}$, identify an item $i^* \in \mathcal{C}$ compatible with $\mathcal{X}$. We evaluate under two protocols: the Vasileva nondisjoint FITB protocol [Vasileva et al., 2018] (4-way choice, 10,000 test questions) and full-catalog retrieval, in which the candidate pool is the entire N-item catalog.

Controllable completion. We additionally allow the caller to attach a discrete constraint to a completion request: a target **role** $r \in \mathcal{R}$ (*e.g.* "shoes"; always required) and, optionally, a target **color code** $c \in \mathcal{K}_{\text{color}}$ (never inferred by the model; always supplied by the caller, whether it matches the ground-truth item's color or not). A query is therefore the triple $(\mathcal{X}, r, c)$, with c optional.

Factorized Semantic ID. Every catalog item i is assigned

$$\text{SID}(i) = [\, r_i,\ c_i,\ k_i \,] \tag{1}$$

where $r_i \in \mathcal{R}$ ($|\mathcal{R}|$=8, including an `unknown` role) is deterministic from category metadata, $c_i \in \{0, \ldots, 6\}$ is an opaque color code, and $k_i \in \{0, \ldots, K_{r_i} - 1\}$ is a compatibility code fit by k-means independently within role r_i's own codebook of size K_{r_i}, with $\sum_r K_r = K = 512$ total codes (Section 5).

Coarse-to-fine candidate generation. Given a query $(\mathcal{X}, r, c)$, the teacher produces one query vector $\hat{z}(\mathcal{X})$ (Section 4). Let $\mathcal{C}_{r,c} = \{i \in \mathcal{C} : r_i{=}r,\ (c_i{=}c \text{ if } c \text{ given})\}$ be the exhaustive constraint-satisfying partition. Let $\text{Top}_L\big(\hat{z}(\mathcal{X}); r\big)$ denote the L nearest of role r's K_r compatibility-code centroids to $\hat{z}(\mathcal{X})$ by cosine similarity, ranked over role r's *full* codebook and computed independently of any color constraint c. The candidate set at budget L intersects this code membership with the exhaustive role[,color] partition:

$$\text{Cand}_L(\mathcal{X}, r, c) = \big\{\, i \in \mathcal{C}_{r,c} \ :\ k_i \in \text{Top}_L\big(\hat{z}(\mathcal{X}); r\big) \,\big\}. \tag{2}$$

Table 1: Role-conditioned teacher, 5 seeds, Polyvore Outfits-ND. Recall@K is full-catalog retrieval recall (235,878-item catalog), averaged over the same 5 seeds. The learned-vs.-geometric routing comparison (Table 2) uses the **seed-42** checkpoint only; the two control evaluations (Tables 3 and 4) additionally repeat the full pipeline against seeds 43 and 44. This table characterizes the teacher's own seed variance and is not propagated through the routing pipeline.

Seed	42	43	44	45	46
FITB (%)	68.91	66.69	69.37	69.05	65.78

Metric	**Mean $\pm$ std (5 seeds)**
FITB (%)	67.96 ± 1.45
Full-catalog Recall@1 (%)	0.384 ± 0.057
Full-catalog Recall@10 (%)	2.486 ± 0.387
Full-catalog Recall@50 (%)	7.380 ± 0.921

Because Top_L is chosen from role r's full codebook before the color constraint is applied, it is possible, when c is given, for none of the top-L codes to have any member in $\mathcal{C}_{r,c}$; this, not a color-conditioned centroid search, is the source of the empty candidate sets handled in Section 6. The final ranking reranks Cand_L continuously and returns the top-10:

$$\mathrm{Top10}(\mathcal{X}, r, c) = \underset{i \in \mathrm{Cand}_L(\mathcal{X},r,c)}{\arg\,\text{top-10}} \; \hat{z}(\mathcal{X})^\top z_{\text{self}}(i). \tag{3}$$

This is **not** autoregressive Semantic-ID generation: $\mathrm{SID}(i)$ is never decoded token-by-token; it is a fixed lookup key used only to construct Cand_L in Eq. 2 before continuous reranking in Eq. 3. Eqs. 2–3 describe the fixed-L case; the adaptive-L escalation used when Cand_L is empty (Section 6) is a runtime fallback and is not reflected in these equations. We refer to the overall procedure as Transformer-derived factorized Semantic-ID routing with continuous reranking.

4 Role-Conditioned Continuous Teacher

The teacher is a single symmetric Transformer encoder used for both context (query) and candidate (key) encoding.

Query encoding. Given context embeddings $\{\mathbf{v}_{x_j}\}_{j=1}^{N_{\text{ctx}}}$ and target role r:

1. Input projection: $\mathbf{h}_j = \mathrm{LayerNorm}(\mathbf{W}_{\text{in}}\mathbf{v}_{x_j}) \in \mathbb{R}^d$, d=256.
2. Role token: $\mathbf{r} = \mathrm{Embed}(r) \in \mathbb{R}^d$ (Embedding(8×256)).
3. Sequence: $[\mathbf{h}_1; \ldots; \mathbf{h}_{N_{\text{ctx}}}; \mathbf{r}; \mathbf{m}]$, where $\mathbf{m}$ is a learnable mask token.
4. Output: $\hat{z} = \mathrm{Normalize}(\mathbf{W}_{\text{out}}\, h_{\text{last}})$, where h_{last} is the encoder's output at the sequence's final position, *i.e.* the position of the mask token $\mathbf{m}$.

The encoder uses no positional encoding, so context order does not affect $\hat{z}$, consistent with permutation-invariant set encoders for unordered inputs [Lee et al., 2019]. Candidate items are encoded with the same weights but as a length-one sequence containing only the item itself, with no role token and no mask token appended: $z_{\text{self}}(i) = \mathrm{Normalize}(\mathbf{W}_{\text{out}}\,\mathrm{Enc}(\mathbf{h}_i)_1)$, where $\mathbf{h}_i = \mathrm{LayerNorm}(\mathbf{W}_{\text{in}}\mathbf{v}_i)$ and the subscript 1 denotes pooling at that single position. Training is end-to-end InfoNCE, temperature τ=0.07, in-batch negatives, batch size 256; 4 layers, 8 heads, d=256; no encoder freezing. At inference, $\hat{z}$ is generated once per query and all candidates are scored by a single batched dot product.

Role conditioning is load-bearing. The role token is not a cosmetic input: it is the mechanism by which the caller's role constraint r reaches the query encoder, and Sections 5–6 route only within the matching role partition. We verified with a gradient check that the role embedding receives non-zero, differentiated gradient for every active role, and that the resulting per-role query vectors are measurably different from each other (mean pairwise cosine similarity 0.51–0.53 across role pairs, well below 1.0).

Table 1 reports the 5-seed result. $67.96\% \pm 1.45\%$ FITB is in the same range as OutfitTransformer's reported 67.1%; we do not attempt a head-to-head reproduction (different codebase, backbone, and evaluation harness) and

do not claim to surpass it. This model is not this paper's contribution: it establishes a non-trivial, role-conditioned continuous compatibility scorer that the rest of the paper turns into a controllable, efficient retrieval system.

5 Factorized Semantic-ID Construction

Role slot. r_i is deterministic: item category metadata is mapped to one of 8 roles (`top`, `bottom`, `dress`, `outerwear`, `shoes`, `bag`, `accessory`, `unknown`) via a fixed lookup table. No learning is involved. After this mapping, the indexed 235,878-item catalog contains zero `unknown`-role items (Appendix A); the compatibility codebook below is therefore allocated only across the 7 typed roles, and `unknown` exists in $\mathcal{R}$ only as a mapping target for items excluded from the catalog.

Color slot. $c_i \in \{0, \ldots, 6\}$ is read directly from a pretrained attribute encoder [Forouzandehmehr et al., 2026] and treated as an opaque code (Section 2); we neither learn nor relabel it.

Compatibility code. k_i is fit by role-wise MiniBatchKMeans [Sculley, 2010] over $z_{\text{self}}(i)$ for the seed-42 teacher, with a total budget of $K{=}512$ codes allocated proportionally to role frequency: accessory 141, shoes 91, bag 83, top 67, bottom 57, dress 38, outerwear 35 (sum 512). Each role's codes are fit independently, so routing within a role never crosses into another role's codebook.

Attempts to learn the partition instead of clustering it. We tested whether k_i could be learned jointly with a routing objective rather than fit by unsupervised clustering, under two independent designs, both trained and evaluated against the same **role-conditioned** seed-42 teacher used throughout this paper. Each learned variant is frozen and evaluated on held-out data with a composite selection metric ($S = \text{TeacherTop10Coverage} - \alpha \cdot \text{Budget} - \beta \cdot \text{RoutingRegret}$, with $\alpha{=}2.0$ and $\beta{=}1.0$; Budget is the fraction of the exhaustive role partition scored, and RoutingRegret is the teacher-score gap between the best item in the full role partition and the best item in the routed bucket, matching BestScoreRegret in Section 7; optionally with an occupancy penalty, weight $\gamma{=}5.0$, equal to the mean shortfall below 95% active-code occupancy across roles) that penalizes both low coverage and inflated collapsed-codebook coverage:

1. **Unconstrained joint EMA partition**: both query and item codes are learned jointly via an EMA-updated codebook, role-conditioned with equal per-role code blocks, 64 codes for each of the 8 roles in $\mathcal{R}$ ($8 \times 64{=}512$). Naive nearest-centroid EMA updates produced severe codebook collapse (512 codes $\rightarrow$ 84–85 active) with deceptively inflated raw RouteRecall; our composite metric correctly rejected the collapsed checkpoints and selected an early, honest checkpoint instead.

2. **Sinkhorn-balanced joint partition**: the same equal-block architecture and joint learning objective as above, but item-to-code assignment is computed by Sinkhorn-Knopp balancing [Caron et al., 2020] at every step, guaranteeing $\geq$95% code occupancy per role throughout training (no collapse observed across 15 epochs).

Both learned variants use this equal 64-code-per-role allocation, which statically reserves one block for the `unknown` role even though the indexed catalog contains zero `unknown`-role items (Section 5); this leaves 448 codes actually usable to either learned method, versus the full 512 the k-means baseline distributes across the 7 typed roles. We flag this as a structural disadvantage for both learned baselines relative to k-means.

Table 2 reports both comparisons. Every learned variant underperforms its matched k-means baseline, including the Sinkhorn variant, which is the only learned method that provably avoids collapse throughout training. We conclude that geometric k-means over the teacher's own embedding space is a strong, hard-to-beat partitioning for this compatibility task, and we **freeze the role-frequency-proportional k-means partition** as the compatibility tokenizer used in Sections 6–7. We do not claim this result generalizes beyond this architecture and data scale (Section 8).

Is the compatibility code redundant with color? We compute the normalized mutual information (NMI) between k_i and c_i within each role. NMI ranges 0.027 (accessory) to 0.126 (bottom), uniformly low, and the mean color entropy remaining within a compatibility-code bucket is 1.305–1.610 nats (max 1.946, close to the $\ln 7$ ceiling), indicating the compatibility code is not simply rediscovering color. The full range across roles is reported in Appendix B.

Table 2: Learned partitioning vs. frozen role-wise k-means, $K{=}512$, $L{=}4$, role-conditioned seed-42 teacher, test split. Values are RouteRecall (fraction of queries whose true target's code is retrieved within the top-L codes). Each learned method is compared against the k-means baseline it was evaluated against in its own experiment; the two baselines differ slightly in code-allocation scheme (equal per-role blocks vs. role-frequency-proportional), so absolute values are not directly comparable across rows; only the within-row learned-vs.-baseline delta is. Both rows use the same role-conditioned teacher checkpoint.

Comparison	**k-means baseline**	**Learned method**	Δ
Joint EMA partition (equal-block k-means)	40.51%	38.54%	-1.97pp
Sinkhorn joint partition (proportional k-means)	36.70%	34.15%	-2.55pp

6 Constrained Coarse-to-Fine Retrieval

Given a query $(\mathcal{X}, r, c)$, retrieval proceeds in three stages, formalized in Eqs. 2–3: (1) rank role r's full compatibility codebook by distance to the query and take the top-L codes, independently of any color constraint; (2) intersect the items carrying one of those codes with the exhaustive role[,color] partition $\mathcal{C}_{r,c}$; (3) rerank the resulting candidate set continuously and return the top-10. L is an explicit efficiency/coverage knob: larger L scores more candidates and more closely approaches the role[,color]-filtered oracle; smaller L scores fewer candidates at some coverage cost. We report $L \in \{1, 2, 4, 8\}$ throughout.

Empty-candidate handling. At small L, Cand_L can be empty if no item in $\mathcal{C}_{r,c}$ carries one of the top-L codes selected in step (1) (*e.g.* a rare role[,color] combination whose items happen to fall outside those codes). Our policy escalates L through the fixed schedule $1 \to 2 \to 4 \to 8$, in that order, until Cand_L is non-empty, capped at the full partition $\mathcal{C}_{r,c}$; because the top-L code set only grows as L increases, this escalation is monotonic, each larger L strictly adds to, and never removes from, the smaller L's candidate set. Across our ground-truth control evaluation, this resolves 100% of queries to a non-empty candidate set at every reported L (final empty rate 0.0%); we report the resolution rate rather than describing every request as "feasible" by construction (see Section 7).

Efficiency. We report a compression ratio, defined per setting as the size of the relevant exhaustive filtered baseline divided by $|\mathrm{Cand}_L|$: for settings with a color constraint, the baseline is the exhaustive role+color partition $\mathcal{C}_{r,c}$; for settings without a color constraint, it is the exhaustive role-only partition $\mathcal{C}_r$. This measures how much scoring work the compatibility code saves *beyond* the constraint the caller already gave, rather than comparing against the full 235,878-item catalog, which would inflate the number by conflating the constraint's own selectivity with the routing's.

7 Experiments

Setup. Polyvore Outfits-ND nondisjoint [Vasileva et al., 2018]: 53,306 training outfits, 251,008 unique items, 235,878-item retrieval catalog. Figure 2 summarizes the full retrieval pipeline together with both control evaluations reported below. Sections 7.1 and 7.2 headline the frozen seed-42 teacher checkpoint (Section 4) and the frozen proportional role-wise k-means compatibility code (Section 5), and additionally repeat the identical pipeline (same k-means allocation scheme, refit per seed's own embedding space) against the independently trained seed-43 and seed-44 role-conditioned teachers, with no change to any hyperparameter, threshold, or reported metric between seeds. Table 2's learned-vs.-geometric comparison uses only the seed-42 teacher (Section 8). The teacher checkpoints, role conditioning, and k-means allocation scheme were fixed before Sections 7.1–7.2 were run and were not changed afterward.

7.1 Ground-truth color control

We evaluate four retrieval conditions against 10,000 full-catalog retrieval queries with a known ground-truth target color: **A** (role-only filtering, exhaustive), **B** (role+color filtering, exhaustive; the ceiling any color-aware method can reach), **C** (compatibility-code routing only, no color constraint used), and **D** (factorized: role+color filtering *and* compatibility-code routing).

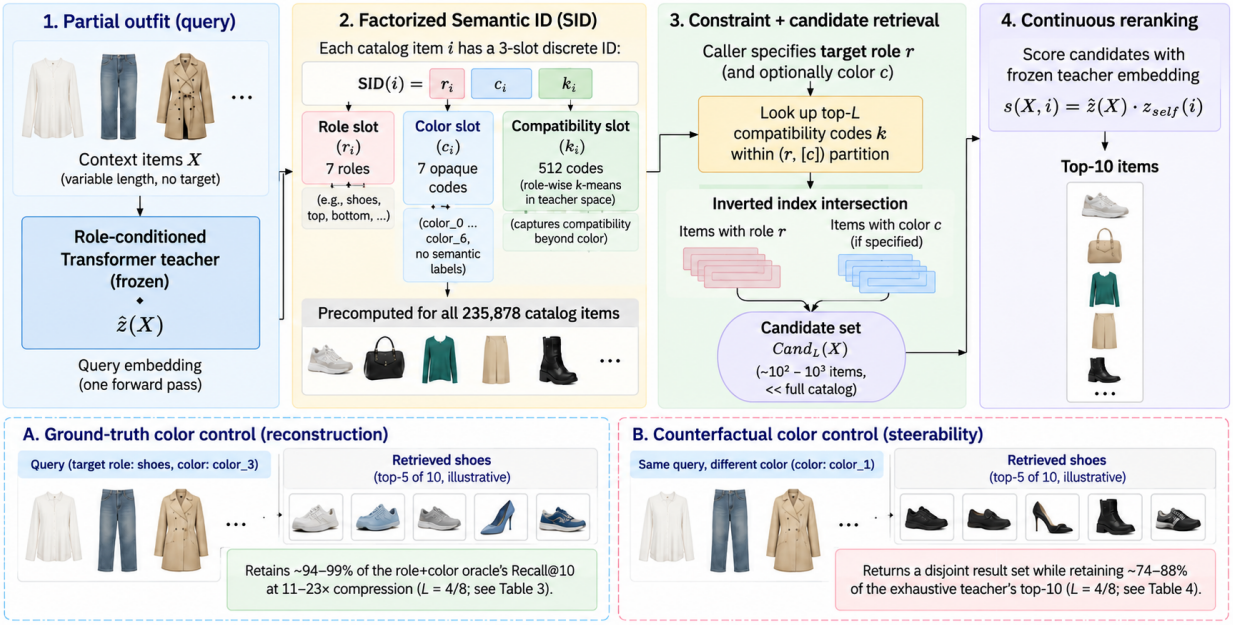


Figure 2: Factorized Semantic-ID routing with role and color-slot control. A partial outfit and requested target role are encoded once by a frozen, role-conditioned Transformer to obtain the query embedding $\hat{z}(\mathcal{X})$. Each catalog item is indexed by $\text{SID}(i) = [r_i, c_i, k_i]$, where r_i is its semantic role, c_i is an opaque seven-way color code, and k_i is a role-specific compatibility code obtained by k-means in the teacher embedding space. The requested role and optional color code are intersected with the query's top-L compatibility-code buckets to form a small candidate set, which is reranked using continuous teacher similarity. Bottom: under ground-truth color control, the factorized ID retains 93.6–98.8% of the exhaustive role-and-color oracle's Recall@10 at 11.3–22.7× compression ($L = 4/8$; see Table 3); under counterfactual color-code intervention, it returns a disjoint result set by construction of the color partition while retaining 74–88% of the exhaustive constrained teacher's top-10 at the same operating points (see Table 4). Product images are illustrative rather than actual dataset retrievals.

Table 3 shows the factorized condition D closes almost all of the gap between compatibility-only routing (C, 32.7% retention) and the role+color oracle (B): L=8 retains 98.8% of B's Recall@10 while scoring 11.3× fewer items than exhaustive role+color filtering, and L=4 retains 93.6% at 22.7× compression. All reported bootstrap CIs are tight relative to the point estimates (e.g. Compression at both L values has a CI width under 1×), indicating the headline numbers are not an artifact of a favorable single draw of the 10,000-query test set. The L=8 retention CI upper bound (101.0%) slightly exceeds 100% because it is a ratio of two independently bootstrapped means, both of which can shift above their point estimate in the same resample; the two-sided CI is reported as computed rather than clipped. Color and compatibility code are complementary: C alone (no color) is far weaker than D, confirming the color constraint is doing real work and is not redundant with the compatibility code, consistent with the NMI diagnostic in Section 5.

Multi-seed check. Repeating the full D pipeline against independently trained seed-43 and seed-44 teachers (Table 3, bottom two rows) gives 93.2% ± 0.8% retention and 22.5× ± 0.2× compression at L=4, and 97.5% ± 1.9% retention and 11.2 × ± 0.1× compression at L=8, across seeds 42–44, consistent with the seed-42 point estimates in every case and with compression varying by under 1% across seeds. Retention is the more seed-sensitive of the two quantities (seed 43's L=8 retention, 95.3%, is the low end of the observed range), but no seed drops below 92% retention at either L.

Per-role breakdown. We repeat the same computation within each of the 7 typed roles (Appendix C) to check whether the aggregate numbers hide role-specific failures. At seed 42, Retention@L4 ranges from 86.8% (accessory, the role with the *most* compatibility codes, 141) to 97.6% (shoes) and 99.3% (bag); dress and outerwear (the roles with

Table 3: Ground-truth color control, 10,000 queries. Rows A–D are the seed-42 teacher. Recall@10 is exact target-item retrieval. Budget is the number of items scored per query. Retention@10 is Recall@10 relative to the role+color oracle (B). Compression is the exhaustive baseline each row has access to (role-only for A/C, role+color for B/D) divided by that row's budget. Bracketed ranges are 95% query-clustered bootstrap CIs ($B = 2000$ resamples of the 10,000 queries). The bottom two rows repeat the full D pipeline (teacher, k-means refit, evaluation) independently against the seed-43 and seed-44 role-conditioned teachers and report the mean $\pm$ std across all 3 seeds; Budget is omitted for these rows because per-seed candidate counts were not logged individually (only the aggregate Compression ratio was). Recall@10 varies more across seeds in absolute terms ($7.27\% \pm 0.61\%$ at $L = 4$) than Retention@10 does ($93.2\% \pm 0.8\%$), because B's own Recall@10 (the denominator) moves with the same teacher-seed variance as D's numerator, so the ratio is the more stable quantity.

Condition	Recall@10 (%)	Budget	Retention@10	Compression
A: role-only (exhaustive)	2.83 [2.50, 3.16]	41,129	–	$1.0\times$
B: role+color (exhaustive, oracle)	8.10 [7.54, 8.65]	11,684	100% (ref)	$1.0\times$
C: compat-code only ($L = 4$, no color)	2.65 [2.33, 2.97]	1,861	32.7%	$22.1\times$ (vs. A)
D: factorized ($L = 4$)	7.58 [7.06, 8.09]	515	**93.6** [90.1, 97.1]%	**22.7** [22.4, 23.0]$\times$ (vs. B)
D: factorized ($L = 8$)	8.00 [7.47, 8.56]	1,032	**98.8** [96.6, 101.0]%	**11.3** [11.2, 11.5]$\times$ (vs. B)
D ($L = 4$), 3-seed mean $\pm$ std (42, 43, 44)	7.27 ± 0.61	–	$93.2 \pm 0.8\%$	$22.5 \pm 0.2\times$ (vs. B)
D ($L = 8$), 3-seed mean $\pm$ std (42, 43, 44)	7.62 ± 0.82	–	$97.5 \pm 1.9\%$	$11.2 \pm 0.1\times$ (vs. B)

the *fewest* codes, 38 and 35) retain 90.0% and 97.2% respectively, not the disproportionate failure a code-starvation hypothesis would predict. This ordering is seed-dependent, however: repeating the per-role breakdown at seeds 43 and 44 (Appendix C) shows `top` and `bottom`, not accessory, as the weakest roles for ground-truth retention on those two seeds. We therefore do not claim a single consistently-weakest role for ground-truth retention. Counterfactual coverage is different: accessory is the weakest role at every one of the 3 teacher seeds there (Appendix C), which we attribute to it being the most visually and stylistically heterogeneous role rather than to its code budget.

7.2 Counterfactual color control

Ground-truth color control does not establish *steerability*: a system could satisfy the color constraint only because it usually agrees with what the teacher would have retrieved anyway. We therefore probe colors *different* from the true target's color. From 3,000 subsampled queries, we issue every one of the 6 non-true color codes as an explicit constraint (up to 18,000 requests total); all 49 role$\times$color combinations in this pool have at least one catalog item (*i.e.* none of the 7×7 role/color combinations is empty), so no request is dropped for lacking any matching items. Because the requested color is deliberately wrong, there is no ground-truth item to score exact recall against; instead we score against the *exhaustive constrained teacher's own ranking*, *i.e.* the teacher's own scores restricted to the (role, counterfactual-color) partition: TeacherTop10Coverage (overlap with the top-10 items the teacher would rank highest within that partition), a matched-budget **Random** control (same candidate-set size, uniformly sampled from the partition), Teacher-NDCG@10 (relevance = teacher score, not ground truth), and BestScoreRegret (score gap to the best-scoring item in the exhaustive partition). We treat 85–90% TeacherTop10Coverage as a practically useful steerability level for a deployed system; this target was set before running this evaluation and is not tuned post hoc against the results below.

Table 4 shows a consistent large margin over the matched-budget random control at every L (8–30$\times$), confirming that the routing genuinely honors the counterfactual color constraint rather than defaulting to what it would have retrieved unconstrained. At L=4 and L=8 the query-clustered bootstrap CIs on SID coverage and the matched-budget random control do not overlap at all (e.g. L=4: [73.5, 74.7]% vs. [4.9, 5.8]%), so the margin is not an artifact of query-sampling noise. We recommend L=8 as the primary operating point: it recovers 88.2% of the teacher's own top-10 exactly, with Teacher-NDCG@10 of 0.998, at 10.8$\times$ compression. L=4 is a more aggressive alternative, trading exact top-10 overlap (74.1%, below our stated 85–90% target) for higher compression (21.8$\times$); its Teacher-NDCG@10 (0.995) clears that bar because near-miss substitutions still preserve rank quality, so we report both operating points rather than leading on the score-based metric alone.

Multi-seed check. Repeating the full counterfactual pipeline against independently trained seed-43 and seed-44 teachers (Table 4, bottom two rows) gives SID coverage of $74.2\% \pm 0.2\%$ (vs. random $5.5\% \pm 0.1\%$) at L=4 and

Table 4: Counterfactual color control (micro-averaged over queries and counterfactual colors), n=18,000 requests, seed-42 teacher. Exhaustive partition size (role+counterfactual-color) averages 4,902 items over this pool, smaller than Table 3's 11,684, because this evaluation subsamples to 3,000 queries and pools over non-true colors specifically; the two tables' candidate counts are not directly comparable. Compression is exhaustive/(candidate-set size); the L=1 row includes 0.38% of requests whose candidate set was empty and resolved by the fallback policy of Section 6. Bracketed ranges at $L \in \{4, 8\}$ are 95% bootstrap CIs clustered by the 3,000 base queries (B=2000 resamples; the ∼6 counterfactual-color requests per query share the same query embedding and are resampled together, not treated as independent). The bottom two rows repeat the full pipeline independently against the seed-43 and seed-44 role-conditioned teachers (each with its own 3,000-query subsample and refit k-means) and report the mean ± std across all 3 seeds.

L	**TeacherTop10Cov. (%)**	**Random (%)**	**Teacher-NDCG@10**	**BestScoreRegret**	**Compression**
1	37.41	1.23	0.9152	0.0289	89.07×
2	55.82	2.54	0.9795	0.0107	44.06×
4	74.12 [73.54, 74.68]	5.34 [4.92, 5.77]	0.9950 [0.9948, 0.9952]	0.0041	21.78×
8	**88.20** [87.83, 88.57]	10.81 [10.21, 11.39]	**0.9984** [0.9983, 0.9984]	**0.0014**	10.78×
4, 3-seed mean ± std (42,43,44)	74.2 ± 0.2	5.5 ± 0.1	0.9954 ± 0.0005	–	–
8, 3-seed mean ± std (42,43,44)	88.4 ± 0.5	10.9 ± 0.1	0.9985 ± 0.0002	–	–

$88.4\% \pm 0.5\%$ (vs. random $10.9\% \pm 0.1\%$) at L=8, across seeds 42–44. Both the SID coverage level and the SID-vs-random margin are markedly more stable across teacher seeds here than the ground-truth retention numbers above (std under 0.5pp at both L values, versus up to 1.9pp for ground-truth L=8 retention), so the counterfactual steerability claim (the paper's central controllability result) does not depend on a favorable draw of the teacher training seed.

Per-role breakdown. The same per-role check (Appendix C) shows SID coverage at L=4 ranging from 66.7% (accessory) to 81.4% (outerwear), and at L=8 from 82.7% (accessory) to 93.3% (outerwear), again with accessory as the weakest role and the two smallest-codebook roles (dress, outerwear) among the *strongest*, not the weakest. The margin over the matched-budget random control remains large in every role at every L (smallest observed margin: accessory at L=4, 66.7% vs. 3.1%, a 21.5× ratio). A macro-average (computed per counterfactual color, then averaged) is within 0.7pp of the micro-average at every L, confirming no single color code dominates the result (Appendix D).

Feasibility, stated precisely. Every one of the 49 role×color combinations touched by this evaluation has at least one catalog item; we do not claim every individual query is trivially satisfiable, only that no combination is structurally empty. Any query whose *true* target role/color could not be resolved during preprocessing was excluded upstream of this evaluation (Section 6).

8 Limitations

Ground-truth per-role ordering is seed-dependent. We report a per-role breakdown of both control evaluations at all 3 teacher seeds (Appendix C) and find no disproportionate failure in the smallest-codebook roles (`dress`, `outerwear`) at any seed. For counterfactual coverage, `accessory` (the largest codebook) is the weakest role at all 3 seeds. For ground-truth retention, however, the weakest role changes across seeds (accessory at seed 42; `top`/`bottom` at seeds 43/44), so we do not identify a single role as the ground-truth bottleneck, only a stable one for counterfactual coverage.

9 Conclusion

We presented a factorized Semantic ID that turns a frozen, role-conditioned compatibility teacher into a controllable and efficient retrieval system. Splitting an item's identity into three independently queryable fields, role, color, and a k-means compatibility code, lets a caller constrain retrieval by index intersection before any embedding similarity is computed, rather than by post-hoc filtering over a full ranked list or by autoregressive code generation. Across two control evaluations and three independently trained teacher seeds, factorized routing retains 93.6–98.8% of

an exhaustive role+color oracle's Recall@10 at 11.3–22.7× compression, and recovers 74–88% of the exhaustive constrained teacher's own top-10 choices under counterfactual color constraints, in both cases with a large, seed-stable margin over a matched-budget random control. A controlled negative result further shows that, at this scale, simple geometric k-means partitioning of the teacher's own embedding space is a stronger foundation for the compatibility code than two independent attempts to learn the partition jointly with a routing objective.

The per-role seed variations discussed in Section 8, along with practical next steps like wall-clock latency benchmarks and open-vocabulary attribute extensions, point to natural directions for future work rather than fundamental bottlenecks. We view the results above as evidence that a factorized discrete index, built without any new training signal beyond the compatibility teacher itself, can make an existing compatibility scorer controllable at a small fraction of its exhaustive retrieval cost.

AI Use Statement

This work used AI assistance (Claude) throughout the research process: implementation of experiments and evaluation pipelines, debugging, literature search, and drafting and restructuring of manuscript text. The authors verified all reported numbers against underlying experiment logs, made all final experimental and methodological decisions, and take full responsibility for the theoretical claims and final text of this paper.

Reproducibility Statement

Code for all experiments will be released upon paper acceptance. Hyperparameters and dataset statistics are reported in Sections 4–7 and Appendix A. The Polyvore Outfits-ND dataset is publicly available from Vasileva et al. [2018]. FashionCLIP weights are at `https://huggingface.co/patrickjohncyh/fashion-clip`. The seed-42 teacher checkpoint used for every routing and control result is identified by SHA256 hash in our internal experiment log and will be released alongside the code.

References

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Patrick John Chia, Giuseppe Attanasio, Franco Maria Nardini, Luca Maria Aiello, and Gianni Casanova. Contrastive language and vision learning of general fashion concepts. *Scientific Reports*, 12:18958, 2022.

Najmeh Forouzandehmehr, Topojoy Biswas, Evren Korpeoglu, and Kannan Achan. Attribute-conditioned multimodal slot factorization for controllable fashion retrieval. *arXiv preprint arXiv:2608.12570*, 2026. Accepted at OARS RecSys 2026 Workshop.

Xintong Han, Zuxuan Wu, Yu-Gang Jiang, and Larry S Davis. Learning fashion compatibility with bidirectional lstms. In *Proceedings of the ACM International Conference on Multimedia*, pages 1078–1086, 2017.

Hervé Jégou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2011.

Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *IEEE International Conference on Data Mining (ICDM)*, pages 197–206, 2018.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 3744–3753, 2019.

Yen-Liang Lin, Son Tran, and Larry S. Davis. Fashion outfit complementary item retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3311–3319, 2020.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, Maciej Kula, Ed Chi, and Maheswaran Sathiamoorthy. Recommender systems with generative retrieval. In *Advances in Neural Information Processing Systems*, volume 36, pages 10299–10315, 2023.

Rohan Sarkar, Navaneeth Bodla, Mariya I Vasileva, Yen-Liang Lin, Anurag Beniwal, Alan Lu, and Gerard Medioni. OutfitTransformer: Learning outfit representations for fashion recommendation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2262–2271, 2023.

D. Sculley. Web-scale k-means clustering. In *Proceedings of the 19th International Conference on World Wide Web (WWW)*, pages 1177–1178, 2010.

Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1441–1450, 2019.

Reuben Tan, Mariya I. Vasileva, Kate Saenko, and Bryan A. Plummer. Learning similarity conditions without explicit supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10373–10382, 2019.

Yi Tay, Vinh Q. Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, Tal Schuster, William W. Cohen, and Donald Metzler. Transformer memory as a differentiable search index. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Maria I Vasileva, Andreas Veit, Prashant Fedorov, Rajiv Shah, and Serge Belongie. Learning type-aware embeddings for fashion compatibility. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 390–405, 2018.

A Implementation Details

Data preprocessing. Polyvore Outfits-ND nondisjoint split: 53,306 training outfits, 251,008 items, 235,878-item retrieval catalog (items with resolvable image embeddings). Images encoded with FashionCLIP ViT (frozen, image-only).

Teacher architecture. Input projection Linear(512,256)+LayerNorm; role embedding Embedding(8×256); learnable mask token $\mathbf{m} \in \mathbb{R}^{256}$; Transformer: 4 pre-LN layers, 8 heads, FFN width $4d$, dropout 0.1, no positional encoding (order-invariant context). Output projection Linear(256,256)+L2-normalize. Trained end-to-end with InfoNCE, τ=0.07, batch size 256, AdamW.

Compatibility k-means. MiniBatchKMeans [Sculley, 2010] fit independently per role over frozen seed-42 teacher item embeddings $z_{\text{self}}(i)$, with total code budget K=512 allocated proportionally to role frequency (accessory 141, shoes 91, bag 83, top 67, bottom 57, dress 38, outerwear 35).

Learned-partitioning negative-result implementations. *Joint EMA partition*: role-conditioned codebook with equal 64-code blocks per role (8×64=512), items assigned to nearest current centroid and centroids updated by EMA; selection via the composite metric S = TeacherTop10Coverage $- \alpha \cdot$ Budget $- \beta \cdot$ RoutingRegret rather than raw RouteRecall, specifically to avoid selecting collapsed checkpoints with inflated raw scores. *Sinkhorn-balanced joint partition*: same architecture, but assignment at every training step is computed by Sinkhorn-Knopp normalization [Caron et al., 2020] over query×code similarity, with hard gates requiring $\geq$95% active codes per role and a bucket-size ratio (p95/mean) $\leq 3\times$; both gates held throughout 15 training epochs. Both learned methods use the equal 64-code-per-role allocation described in Section 5, which leaves 448 of the 512 codes usable since the `unknown` role has no catalog items; the k-means baselines they are compared against in Table 2 do not share this restriction.

B Per-Role NMI and Color-Entropy Diagnostic

Normalized mutual information between compatibility code k_i and color code c_i, computed within each of the 7 typed roles, and mean color entropy remaining within a compatibility-code bucket (max possible $\ln 7 \approx 1.946$ nats, for a uniform distribution over 7 color codes). We report the two extremal roles bounding the observed range; the remaining five typed roles fall between these values.

Role	**NMI**$(k_i; c_i)$	**Mean color entropy in bucket (nats)**
accessory (min NMI)	0.027	1.610
bottom (max NMI)	0.126	1.305

Table 5: Extremal per-role NMI and in-bucket color entropy. Across all 7 typed roles, NMI ranges 0.027–0.126 (uniformly low) and mean in-bucket color entropy ranges 1.305–1.610 nats, close to the 1.946-nat ceiling, indicating substantial residual color diversity within any single compatibility code.

C Per-Role Breakdown of Both Control Evaluations

Both control evaluations (Sections 7.1–7.2) are aggregated across the 7 typed roles in the main text. Table 6 and Table 7 report the same quantities broken down by role, seed-42 teacher, to check whether the aggregate numbers conceal a role-specific failure (in particular, whether the two smallest-codebook roles, `dress` (38 codes) and `outerwear` (35 codes), underperform).

Table 6: Ground-truth color control (Section 7.1) by role, 10,000 queries, seed-42 teacher. Retention is D's E2E@10 relative to that role's own B (role+color oracle) Recall@10.

Role	n	**B Recall@10**	**D@L4**	**Ret. L4**	**D@L8**	**Ret. L8**
dress	739	14.88%	13.40%	90.0%	14.21%	95.5%
top	1234	8.27%	7.70%	93.1%	7.94%	96.1%
bottom	1124	6.85%	6.05%	88.3%	6.67%	97.4%
outerwear	597	12.06%	11.73%	97.2%	11.56%	95.8%
shoes	1963	8.46%	8.25%	97.6%	8.86%	104.8%
bag	1644	8.94%	8.88%	99.3%	9.18%	102.7%
accessory	2699	5.04%	4.37%	86.8%	4.74%	94.1%

Table 7: Counterfactual color control (Section 7.2) by role, 3,000 base queries, seed-42 teacher. SID and Random are TeacherTop10Coverage; NDCG is Teacher-NDCG@10 for SID.

Role	n	**SID L4**	**Rand L4**	**NDCG L4**	**SID L8**	**Rand L8**	**NDCG L8**
dress	218	79.08%	9.25%	0.9955	92.55%	18.89%	0.9988
top	401	69.58%	5.65%	0.9941	85.54%	11.25%	0.9979
bottom	303	75.08%	6.89%	0.9946	89.27%	14.62%	0.9987
outerwear	183	81.40%	13.59%	0.9966	93.32%	26.38%	0.9990
shoes	601	77.33%	4.15%	0.9958	90.14%	8.68%	0.9987
bag	501	80.21%	4.47%	0.9969	92.22%	9.20%	0.9991
accessory	793	66.73%	3.07%	0.9934	82.74%	5.92%	0.9975

In both evaluations, the two smallest-codebook roles (`dress`, `outerwear`) are among the *strongest* performers, not the weakest, contradicting a naive code-starvation hypothesis.

Some ground-truth retention entries in Table 6 exceed 100% (`shoes` at L=8, 104.8%; `bag` at L=8, 102.7%). This is not an error: D's candidate set at a given L is a *subset* of B's exhaustive role+color set, so D's top-10 differs from B's top-10 only by excluding some items B ranked above the true target; when one of those excluded items

previously displaced the target out of B's top-10, removing it can promote the target into D's top-10, making D's per-role Recall@10 exceed B's on that role's (comparatively small, $n<2000$) query subset. This is a small-sample ratio artifact, consistent with the main text's aggregate $L=8$ retention CI (Table 3) having an upper bound of 101.0% for the same reason.

`accessory` (the largest codebook, 141 codes) is consistently the weakest role in the *counterfactual* evaluation across all 3 teacher seeds (66.7%/67.2%/67.0% SID coverage at $L=4$ for seeds 42/43/44 respectively), still far above its matched random control in every case (smallest margin: 66.7% vs. 3.1% at $L=4$, a 21.5$\times$ ratio). The same is *not* true for ground-truth retention: `accessory` is weakest for seed 42 (86.8% at $L=4$, Table 6), but for seeds 43 and 44 `top` and `bottom` are weakest instead (89.1–89.2% and 84.6–85.6% respectively, both below accessory's 88.9–93.5% on those seeds). We therefore do not claim a consistently weakest role for ground-truth retention, only for counterfactual coverage. We attribute `accessory`'s counterfactual weakness to it aggregating a visually and stylistically heterogeneous set of item types (jewelry, sunglasses, hats, scarves; Section 5) rather than to its code allocation, though this does not explain why the same role is not uniquely weakest under ground-truth retention.

D Macro-Averaged Counterfactual Control

Averaging per counterfactual color code first, then across codes (rather than pooling all requests), gives $L = 4$ macro-TeacherTop10Coverage $= 74.83\%$ and macro-Teacher-NDCG@10 $= 0.9953$, both within 0.7 pp / 0.0003 of the micro-averaged values in Table 4, indicating no single color code dominates the reported margin.