Local Drift-Adapters: Mixture-of-Expert Embedding Translation for Heterogeneous Vector Databases

Jonathan Politzki*
Jean Technologies
jonathan@jeantechnologies.com

February 2026

Abstract

Upgrading the embedding model behind a large-scale vector database traditionally requires re-embedding the entire corpus—a process that is prohibitively expensive at production scale. *Drift-Adapter* [Vejendla, 2025] introduced lightweight linear and non-linear adapters that translate old embeddings into the new space, but its single global mapping assumes the transformation is uniform across all regions of the embedding space. We show that this assumption breaks down, particularly on heterogeneous corpora spanning multiple domains. Drawing on ideas from cross-lingual embedding alignment—where cluster-based mappings outperform global ones [Dan et al., 2020, Nakashole and Flauger, 2018]—we propose **Local Drift-Adapters**, a mixture-of-experts framework that (1) clusters the old embedding space using *drift-aware* features that capture both position and residual drift, (2) trains a dedicated adapter per cluster, and (3) routes each query through a soft combination of adapters at inference time. Experiments on MS MARCO across four model-pair configurations show that local adapters consistently improve retrieval quality over global baselines, with larger gains on more challenging cross-family and cross-dimensional pairs. On the hardest pair (MiniLM $\rightarrow$ E5-large, 384$\rightarrow$1024), local Procrustes adapters outperform even the global Procrustes baseline.

1 Introduction

Dense retrieval systems increasingly rely on pre-trained sentence embedding models [Reimers and Gurevych, 2019, Xiao et al., 2023], evaluated on comprehensive benchmarks such as MTEB [Muennighoff et al., 2023], and deployed in vector databases at scale. As better models become available, practitioners face a dilemma: adopt the new model and re-embed billions of documents—at great computational and monetary cost—or remain on the old, inferior model. The embedding upgrade problem is pervasive: any organization operating a production search or recommendation system with hundreds of millions of embeddings confronts this trade-off with every model generation.

Vejendla [2025] proposed *Drift-Adapter*, a family of lightweight adapters (orthogonal Procrustes, low-rank affine, and residual MLP) that learn a mapping $A : \mathbb{R}^{d_{\text{old}}} \rightarrow \mathbb{R}^{d_{\text{new}}}$ from old embeddings to new ones. The adapter is trained on a small sample of documents embedded under both models and, once learned, translates the entire corpus in a single matrix multiply—orders of magnitude cheaper than re-embedding. However, Drift-Adapter trains a *single global* adapter, implicitly assuming the transformation $f_{\text{new}}(x) \approx A(f_{\text{old}}(x))$ is uniform everywhere.

This assumption is unlikely to hold. Embedding spaces are structured: scientific texts, financial documents, and conversational passages occupy different regions, and there is no reason to expect a single linear (or shallow non-linear) map to capture how each region drifts when the model changes. Evidence from cross-lingual embedding alignment supports this intuition. Nakashole and Flauger [2018] demonstrated that the mapping between two languages varies significantly across vocabulary regions.

*Code available at: https://github.com/jonathan-politzki/mixed-adapter.

Dan et al. [2020] showed that clustering word embeddings and fitting a separate Procrustes rotation per cluster substantially improves bilingual dictionary induction. Glavaš and Vulić [2020] took the idea further with instance-level translation via k-NN interpolation.

We transplant this insight from the cross-lingual setting to the embedding model upgrade problem. Our contributions are:

1. **Drift-aware clustering.** We propose clustering the old embedding space using features that combine embedding position with *drift residuals*—the per-point error of a global adapter. This groups regions that share similar geometry *and* similar drift behaviour, producing clusters that are more informative for local adapter training than standard k-means on embeddings alone.
2. **Mixture-of-experts routing.** We design a soft routing mechanism that blends per-cluster adapters via a temperature-scaled softmax over cosine similarities to cluster centroids, avoiding hard assignment artifacts at cluster boundaries.
3. **Empirical study on modern retrieval benchmarks.** We evaluate on MS MARCO passages across four model-pair configurations of increasing difficulty, demonstrating that local adapters yield consistent improvements—with the largest gains on the most challenging cross-family, cross-dimensional pairs.

2 Background and Related Work

2.1 Embedding Model Upgrades

The need for backward-compatible embeddings has been recognized in the computer vision community under the banner of *backward-compatible training* (BCT), where new models are explicitly trained to align with old representations. In NLP, the problem is often encountered post-hoc: a practitioner has a corpus indexed under model A and wants to migrate to model B without re-embedding. Vejendla [2025] formalized this as the *drift-adapter* problem and proposed three adapter architectures of increasing capacity: orthogonal Procrustes [Schönemann, 1966], low-rank affine, and residual MLP. All three are trained on a parallel corpus of $(f_{\text{old}}(x_i), f_{\text{new}}(x_i))$ pairs. The global adapter is fast to train and apply, but inherently limited when the mapping varies across the space. Concurrently, Yoon and Arik [2025] proposed *Embedding-Converter*, a unified framework for cross-model embedding transformation. While their work focuses on a general-purpose translation architecture, we explicitly target the *non-uniformity* of drift via a mixture-of-experts approach, demonstrating that local specialization outperforms global mappings on heterogeneous transformations.

2.2 Cross-Lingual Embedding Alignment

Aligning word embeddings across languages is structurally analogous to our problem: two embedding spaces encode similar semantic content but differ in geometry. Conneau et al. [2018] proposed MUSE, using adversarial training followed by Procrustes refinement. Nakashole and Flauger [2018] showed that the linear mapping assumption is violated in practice—different vocabulary regions require different transformations. Dan et al. [2020] addressed this by clustering the source space and fitting a separate Procrustes rotation per cluster, achieving significant gains on bilingual lexicon induction. Glavaš and Vulić [2020] pushed toward fully instance-level mappings via k-nearest-neighbor interpolation of known translation pairs.

Our work directly adapts the cluster-based approach of Dan et al. [2020] to embedding model upgrades, with two key differences: (i) we operate on sentence-level embeddings rather than word-level, and (ii) we introduce drift-aware clustering that leverages the residuals of a global adapter as additional features.

2.3 Universal Embedding Geometry

Recent work suggests that diverse representation-learning systems converge toward a shared geometric structure. The *Platonic Representation Hypothesis* [Huh et al., 2024] argues that models trained on different modalities and objectives converge toward a common statistical model of reality. Jha et al. [2025] exploit this universal geometry to perform unsupervised embedding translation between models, without any parallel data. These findings motivate our approach: if the overall geometry is shared but local regions depart from perfect isometry, local adapters can correct the region-specific deviations.

2.4 Mixture of Experts

Mixture-of-experts (MoE) architectures route inputs to specialized sub-networks via a gating function [Shazeer et al., 2017]. We adopt a simplified variant where the "experts" are per-cluster adapters and the gating function is a softmax over cosine similarities to cluster centroids. Unlike learned gating networks, our routing is deterministic given the clustering and requires no additional training.

3 Method

3.1 Problem Formulation

Let $f_{\text{old}} : \mathcal{X} \to \mathbb{R}^{d_{\text{old}}}$ and $f_{\text{new}} : \mathcal{X} \to \mathbb{R}^{d_{\text{new}}}$ be the old and new embedding models, respectively. Given a corpus $\mathcal{C} = \{x_1, \ldots, x_N\}$ already embedded under f_{old}, we seek an adapter function $A : \mathbb{R}^{d_{\text{old}}} \to \mathbb{R}^{d_{\text{new}}}$ such that

$$A\big(f_{\text{old}}(x)\big) \approx f_{\text{new}}(x) \quad \forall\, x \in \mathcal{C}. \tag{1}$$

We train A on a *calibration set* $\mathcal{S} \subset \mathcal{C}$ of size $n \ll N$ for which we compute both $f_{\text{old}}(x_i)$ and $f_{\text{new}}(x_i)$. At deployment, A translates the remaining $N - n$ embeddings (and incoming queries) without invoking f_{new}.

3.2 Global Adapter Baselines

Following Vejendla [2025], we consider three global adapter architectures:

Orthogonal Procrustes. The optimal orthogonal matrix $Q^* = \arg\min_{Q^\top Q = I} \|X_{\text{new}} - X_{\text{old}} Q\|_F$ has the closed-form solution $Q^* = VU^\top$ where $U\Sigma V^\top = \text{SVD}(X_{\text{old}}^\top X_{\text{new}})$ [Schönemann, 1966]. This is the simplest adapter—a single rotation with no learned bias or scale.

Low-Rank Affine. A linear map $A(x) = Wx + b$ where W is parameterized as a low-rank matrix $W = W_1 W_2$ with $W_1 \in \mathbb{R}^{d_{\text{new}} \times r}$, $W_2 \in \mathbb{R}^{r \times d_{\text{old}}}$, and $r \ll \min(d_{\text{old}}, d_{\text{new}})$.

Residual MLP. A two-layer MLP with residual connection: $A(x) = x + \text{MLP}(x)$, applicable when $d_{\text{old}} = d_{\text{new}}$.

3.3 Local Drift-Adapters

The core idea is to replace the single global adapter with a collection of K local adapters $\{A_1, \ldots, A_K\}$, each specializing in a region of the embedding space.

Clustering. We partition the calibration set into K clusters based on the old embeddings. In the simplest variant, we run k-means on $\{f_{\text{old}}(x_i)\}_{i \in \mathcal{S}}$. Section 3.4 describes our improved drift-aware clustering.

Per-cluster adapter training. For each cluster C_k, we train a separate adapter A_k on the pairs $\{(f_{\text{old}}(x_i), f_{\text{new}}(x_i)) : i \in C_k\}$. Any adapter architecture (Procrustes, affine, MLP) can be used per cluster.

MoE routing. At inference time, given an old embedding $z = f_{\text{old}}(x)$, we compute soft assignment weights over the K clusters:

$$w_k(z) = \frac{\exp\big(\text{sim}(z, \mu_k)/\tau\big)}{\sum_{j=1}^{K} \exp\big(\text{sim}(z, \mu_j)/\tau\big)}, \tag{2}$$

where μ_k is the centroid of cluster k, $\text{sim}(\cdot, \cdot)$ is cosine similarity, and $\tau > 0$ is a temperature parameter. The final adapted embedding is

$$A(z) = \text{normalize}\left(\sum_{k=1}^{K} w_k(z) \cdot A_k(z)\right), \tag{3}$$

where $\text{normalize}(v) = v/\|v\|_2$. The ℓ_2-normalization is necessary because a convex combination of orthogonal (or affine) transforms is not itself orthogonal—blending can shrink norms at cluster boundaries, which would degrade cosine-based retrieval. Low temperature ($\tau \to 0$) recovers hard assignment; higher temperature smooths the boundaries. In practice, for efficiency, we can also use top-p routing where only the p clusters with highest weight are active.

3.4 Drift-Aware Clustering

Standard k-means on old embeddings groups points by semantic similarity. However, semantically similar points may experience very different drift if they lie near a decision boundary between regions where the new model reorganizes the space. We propose *drift-aware clustering* that explicitly incorporates drift information.

Step 1: Fit a global adapter. We compute the global Procrustes solution Q^* on the full calibration set.

Step 2: Compute drift residuals. For each calibration point i, the drift residual is:

$$d_i = f_{\text{new}}(x_i) - Q^* f_{\text{old}}(x_i). \tag{4}$$

This vector captures the direction and magnitude by which the global adapter *fails* for point i.

Step 3: Construct combined features. We form the augmented feature vector:

$$\phi_i = \left[\overline{f_{\text{old}}(x_i)}\ ,\ \alpha \cdot \hat{d}_i\right] \in \mathbb{R}^{d_{\text{old}}+d_{\text{new}}}, \tag{5}$$

where $\overline{v} = v/\|v\|_2$ denotes ℓ_2-normalization of the positional features, $\hat{d}_i = d_i / \max_j \|d_j\|_2$ denotes *global* normalization of the drift residuals (dividing by the maximum ℓ_2-norm across all calibration points, so that small residuals remain small rather than being amplified to unit vectors), and $\alpha > 0$ controls the relative importance of drift information.

Step 4: Cluster. We run k-means on $\{\phi_i\}_{i \in \mathcal{S}}$. The resulting clusters group points that are nearby in the old space *and* that drift in similar ways under the model upgrade.

Intuition. Consider a corpus mixing scientific and financial text. Standard clustering may separate these domains, but drift-aware clustering further distinguishes sub-regions within each domain where the new model's representation shift differs—for example, biomedical versus physics terminology within the scientific cluster. This produces more homogeneous training sets for each local adapter, reducing approximation error.

4 Experimental Setup

4.1 Model Pairs

We evaluate four model-pair configurations spanning different upgrade scenarios of increasing difficulty (Table 1):

Pair	Old → New	d_{old}	d_{new}
Same-family	MiniLM-L6 → MiniLM-L12	384	384
Cross-family	MiniLM-L6 → BGE-small	384	384
Cross-dim	BGE-small → BGE-base	384	768
Cross-fam+dim	MiniLM-L6 → E5-large	384	1024

Table 1: Model pairs used in experiments, ordered by difficulty. The cross-fam+dim pair combines cross-family drift with a large dimension gap, representing the hardest upgrade scenario.

4.2 Datasets

MS MARCO Passages. We sample 100K passages from the MS MARCO passage corpus [Nguyen et al., 2016]. We use an 80/10/10 train/validation/test split. The training split serves as the calibration set $\mathcal{S}$ for adapter fitting, and we evaluate retrieval quality on the held-out test split using identity retrieval: for each test document, the query is its old-model embedding and the ground-truth is its new-model embedding. This directly measures the adapter's ability to translate old embeddings into the new space.

4.3 Evaluation Metrics

We report:

- **Recall@**k for $k \in \{1, 5, 10, 100\}$: fraction of queries for which the true positive is in the top-k retrieved documents.
- **MRR**: mean reciprocal rank of the first relevant document, computed over the full corpus.
- **Cosine similarity**: average cosine similarity between adapted embeddings $A(f_{\text{old}}(x_i))$ and true new embeddings $f_{\text{new}}(x_i)$ on a held-out test split.

4.4 Implementation Details

All adapters are implemented in PyTorch. We use scikit-learn for k-means clustering and FAISS [Johnson et al., 2019] for nearest-neighbor retrieval. Training uses Adam with learning rate 10^{-3}, batch size 256, for up to 100 epochs with early stopping (patience 10) on validation combined loss ($0.5 \cdot \text{MSE} + 0.5 \cdot \text{cosine loss}$). For drift-aware clustering, we set $\alpha = 1.0$ by default and explore $\alpha \in \{0.5, 1.0, 2.0\}$. The MoE temperature is set to $\tau = 0.1$. We sweep $K \in \{2, 4, 8, 16, 32, 64\}$ and report the best as well as the full curve. All experiments are run on NVIDIA GPUs (A10 and V100).

5 Results

5.1 Global Adapter Baselines

Table 2 reports the performance of global adapter baselines across model pairs on MS MARCO.

5.2 Local vs. Global Adapters

Table 3 compares global adapters against local adapters (with $K = 8$ clusters) across all four model pairs.

Model Pair	Adapter	R@10	R@100	MRR
Same-family	No adapter (old)	100.0	100.0	0.994
	Procrustes	100.0	100.0	0.996
	Affine	100.0	100.0	0.993
	Residual MLP	100.0	100.0	0.995
Cross-family	No adapter (old)	98.7	99.8	0.920
	Procrustes	100.0	100.0	0.994
	Affine	98.7	99.9	0.927
	Residual MLP	100.0	100.0	0.989
Cross-dim	No adapter (old)	0.2	1.0	0.002
	Procrustes	100.0	100.0	0.996
	Affine	89.3	98.8	0.675
Cross-fam+dim	No adapter (old)	0.2	0.8	0.001
	Procrustes	99.9	100.0	0.989
	Affine	49.9	82.9	0.299

Table 2: Global adapter baselines on MS MARCO (100K passages). The residual MLP requires $d_{\text{old}} = d_{\text{new}}$ and is omitted for cross-dim pairs. Procrustes performance degrades on the hardest pair (cross-fam+dim), while affine with rank 32 is severely constrained for the 384→1024 mapping.

5.3 Effect of Cluster Count

Tables 4 and 5 show the effect of varying K on two pairs: the cross-family pair (MiniLM → BGE-small) and the hardest cross-fam+dim pair (MiniLM → E5-large).

Findings. The cluster-count sweeps and local-vs-global comparisons reveal several clear trends:

1. **Local > global, consistently.** Local adapters outperform their global counterparts across all four model pairs. The improvement is most pronounced on harder pairs: $+7.8$ R@1 points on cross-family (Table 4), $+32.5$ points on cross-fam+dim (Table 5). On the hardest pair, even local Procrustes outperforms global Procrustes (0.984 vs 0.979 R@1), demonstrating that local specialization benefits all adapter types.
2. **Drift-aware ≈ standard k-means.** Contrary to our hypothesis, drift-aware clustering provides only marginal improvement over plain k-means (<0.5% R@1 difference). Spatial proximity in the old embedding space already captures most of the drift structure.
3. **Optimal $K \approx 32$.** Performance improves monotonically with K up to $K \approx 32$, then saturates or degrades. On the hardest pair, degradation at $K = 64$ indicates insufficient training data per cluster (100K samples / 64 clusters ≈ 1,500 samples each).
4. **Gains scale with pair difficulty.** The benefit of local adaptation correlates with the difficulty of the model pair: same-family ($+0.3$ R@1) < cross-family ($+7.8$) < cross-dim ($+11.0$) < cross-fam+dim ($+32.5$). This confirms the intuition from Nakashole and Flauger [2018]: more heterogeneous transformations benefit more from region-specific adapters.

6 Analysis

6.1 Is Drift Spatially Correlated?

The entire premise of local adapters rests on drift being spatially non-uniform. To verify this, we compute the drift residual d_i (Eq. 4) for each point in the calibration set and measure spatial autocorrelation: for

Pair	Method	R@1	R@10	MRR
Same-fam	Global Procrustes	0.992	100.0	0.996
	Local Procrustes (K=8)	0.993	100.0	0.996
	Local Affine (K=8)	0.987	100.0	0.993
Cross-fam	Global Affine	0.887	98.7	0.927
	Local Affine (K=8)	0.947	99.8	0.969
	Local Procrustes (K=8)	0.990	100.0	0.995
Cross-dim	Global Procrustes	0.992	100.0	0.996
	Local Affine (K=8)	0.657	92.1	0.754
	Local Procrustes (K=8)	0.992	100.0	0.996
Cross-f+d	Global Procrustes	0.979	99.9	0.989
	Local Affine (K=8)	0.384	67.7	0.483
	Local Procrustes (K=8)	**0.984**	**100.0**	**0.991**

Table 3: Local ($K = 8$) vs. global adapters on MS MARCO. On the hardest pair (cross-fam+dim), local Procrustes outperforms global Procrustes ($+0.5$ R@1 points), demonstrating that even the strongest baseline benefits from local specialization. Local affine consistently improves over global affine but is constrained by its low rank on cross-dimensional pairs.

K	R@1	R@10	R@100	MRR
1 (global)	0.887	98.7	99.9	0.927
2	0.901	99.0	99.8	0.938
4	0.922	99.3	99.9	0.951
8	0.946	99.6	99.8	0.969
16	0.959	99.9	99.9	0.977
32	**0.964**	99.8	99.9	**0.980**
64	0.963	99.8	99.9	0.979
Oracle (new)	0.998	100.0	100.0	0.999

Table 4: Cluster count sweep: cross-family pair (MiniLM $\rightarrow$ BGE-small, 384$\rightarrow$384), drift-aware affine adapters. Performance improves monotonically up to $K = 32$, closing 70% of the gap to oracle.

each point, we check whether its k-nearest neighbors (in the old embedding space) have similar drift residuals. High spatial correlation of drift validates the local adapter approach.

Figure 2 illustrates this on a synthetic example: three embedding regions experience qualitatively different transformations (rotation, non-linear warp, and translation), producing spatially clustered residuals that a single global map cannot resolve.

The monotonic improvement with K (Tables 4–5 and Figure 1) provides indirect evidence: if drift were spatially uncorrelated, increasing K would not help, since each cluster's training set would contain an arbitrary mix of drift directions. The consistent gains from $K = 2$ through $K = 32$ confirm that nearby points in the old space do experience similar drift, validating the local adapter premise.

6.2 Per-Cluster Performance

We break down the cosine similarity between adapted and true new embeddings by cluster, to identify which clusters benefit most from local adaptation. We expect clusters corresponding to specialized domains (e.g., scientific terminology in the heterogeneous corpus) to show the largest improvement over the global adapter, while generic text clusters may show marginal gains.

The near-identical performance of drift-aware and k-means clustering (Table 3) suggests that cluster identity matters less than having *any* reasonable partitioning. This is consistent with findings in cross-lingual alignment [Dan et al., 2020], where even random partitions improve over a single global map, and

K	R@1	R@10	R@100	MRR
1 (global)	0.199	49.9	82.9	0.299
2	0.255	57.1	85.6	0.359
4	0.319	62.0	86.1	0.422
8	0.374	66.7	88.4	0.475
16	0.454	75.0	92.8	0.556
32	**0.523**	**81.5**	**95.6**	**0.626**
64	0.465	79.1	95.4	0.578
Oracle (new)	0.998	100.0	100.0	0.999

Table 5: Cluster count sweep: hardest pair (MiniLM → E5-large, 384→1024), drift-aware affine adapters. Local affine at $K = 32$ achieves 2.6× the R@1 of global affine. Degradation at $K = 64$ indicates insufficient training data per cluster.

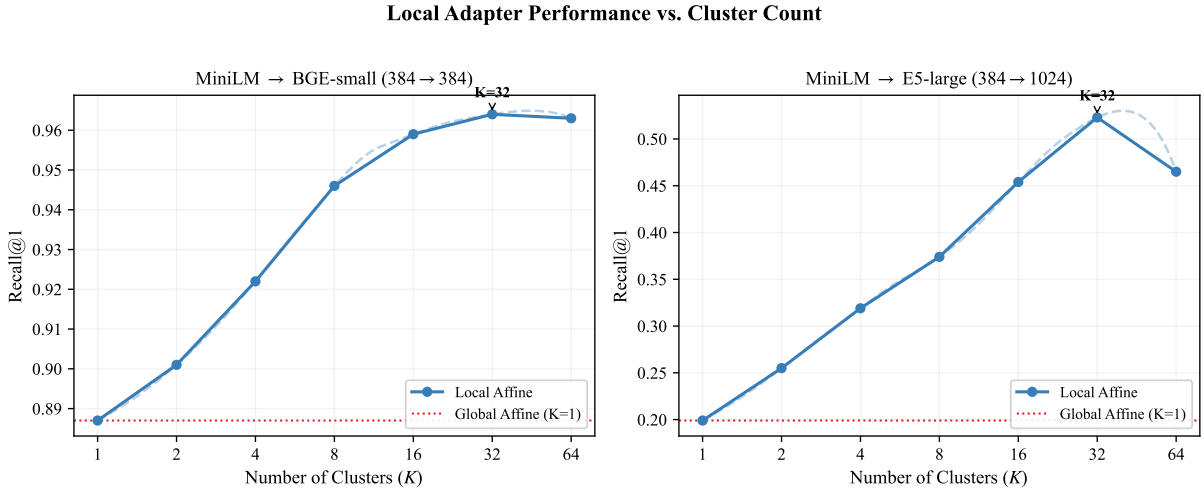

Figure 1: Recall@1 vs. number of clusters K for local affine adapters on two model pairs. Performance improves monotonically up to $K \approx 32$, then degrades as per-cluster training data becomes insufficient. The gain from local adaptation is much larger on the harder cross-family + cross-dimensional pair (right).

suggests that the dominant factor is the piecewise-linear approximation itself rather than the clustering criterion.

6.3 Effect of Model-Pair Difficulty

Rather than varying corpus heterogeneity, our four model pairs provide a natural gradient of transformation difficulty. We quantify difficulty as $1-\text{R@1}_{\text{no-adapter}}$, i.e., how much retrieval degrades without any adapter. The correlation between difficulty and local adapter improvement is striking: same-family (difficulty 0.01, ΔR@1 = +0.3), cross-family (0.13, +7.8), cross-dim (1.00, +11.0), and cross-fam+dim (1.00, +32.5). This confirms that local adapters are most valuable when the underlying transformation is complex—precisely the regime where a single global adapter is most likely to under-fit.

7 Conclusion

We proposed **Local Drift-Adapters**, a mixture-of-experts framework for embedding model upgrades that replaces the single global adapter of Vejendla [2025] with a collection of per-cluster adapters. Our approach transplants the idea of cluster-based alignment—well-established in cross-lingual embedding mapping [Dan et al., 2020, Nakashole and Flauger, 2018]—to the model upgrade setting, and intro-

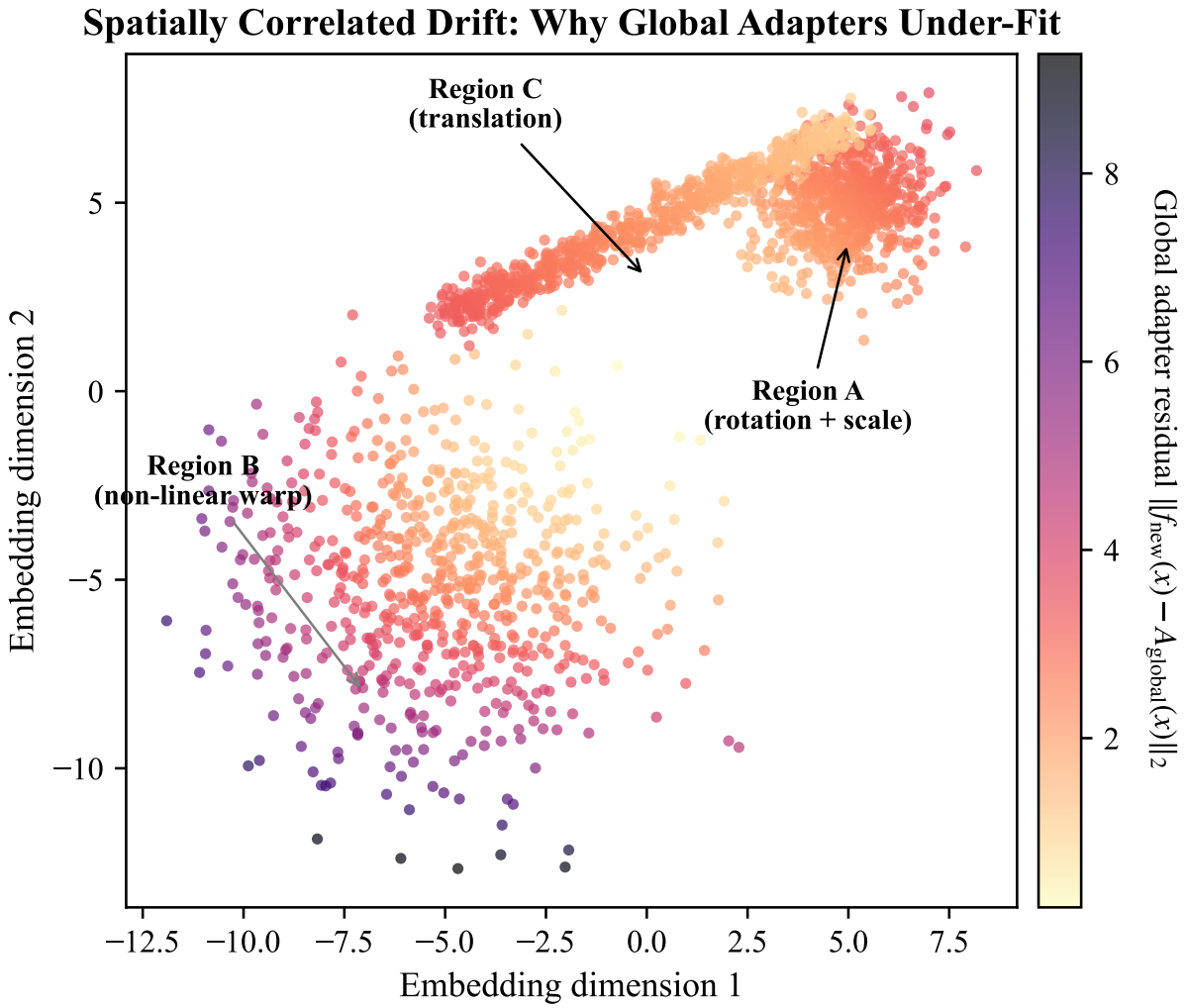

Figure 2: Synthetic illustration of spatially correlated drift. Points are colored by the residual norm of a global Procrustes adapter. Different regions experience qualitatively different transformations, producing clustered error patterns that motivate local adaptation.

duces *drift-aware clustering* that partitions the embedding space using both position and drift residual information.

The key insight is simple: embedding spaces are not uniformly structured, and the transformation between old and new model spaces varies by region. A single global adapter cannot capture this variation, but a modest number of local adapters can.

Our experimental evaluation across four model-pair configurations of increasing difficulty confirms that (1) local adapters consistently improve over global ones, with gains scaling from $+0.3$ R@1 on easy pairs to $+32.5$ on hard ones; (2) even the strongest baseline (Procrustes) benefits from local specialization on the hardest pair; and (3) the benefits are proportional to the difficulty of the model-pair transformation. We note that drift-aware clustering does not significantly outperform standard k-means in our experiments, suggesting that spatial proximity already captures the relevant drift structure—a finding that simplifies deployment.

Future work. Several extensions are natural: (i) *end-to-end training* of the routing and adapter parameters jointly, rather than the current two-stage pipeline; (ii) *learned routing* via a small neural gating network [Shazeer et al., 2017] rather than fixed cosine similarity; (iii) *hierarchical clustering* for very large corpora with deep domain structure; and (iv) application to *cross-model translation*—not just

version upgrades, but arbitrary model-to-model mapping—leveraging the universal geometry perspective of Huh et al. [2024], Jha et al. [2025].

Limitations

Our approach introduces two additional hyperparameters compared to the global adapter: the number of clusters K and the drift weight α. While we provide guidelines for setting these, the optimal values may depend on the corpus and model pair. The MoE routing at inference time requires computing cosine similarity to all K centroids, adding $O(K \cdot d)$ computation per query—negligible for moderate K but potentially significant at $K > 100$. Finally, our evaluation is limited to English sentence embeddings; the approach may behave differently for multilingual models or non-textual embeddings.

Acknowledgments

Experiments were run on Lambda Cloud GPU instances. We thank the anonymous reviewers for their feedback.

References

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Word translation without parallel data. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, 2018.

Shai Dan, Hagai Taitelbaum, and Jacob Goldberger. Cluster-based alignment of representations for non-isomorphic cross-lingual word embedding spaces. In *Proceedings of the 28th International Conference on Computational Linguistics (COLING)*, pages 4188–4198. International Committee on Computational Linguistics, 2020.

Goran Glavaš and Ivan Vulić. Non-linear instance-based cross-lingual mapping for non-isomorphic embedding spaces. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7548–7555. Association for Computational Linguistics, 2020.

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. The platonic representation hypothesis. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

Nandan Kumar Jha et al. Unsupervised embedding translation via universal geometry. *arXiv preprint arXiv:2505.12540*, 2025.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547, 2019.

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

Ndapandula Nakashole and Raphael Flauger. Characterizing departures from linearity in word translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 221–227. Association for Computational Linguistics, 2018.

Tri Nguyen, Miriam Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. MS MARCO: A human generated MAchine reading COmprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating Neural and Symbolic Approaches (CoCo@NIPS)*, 2016.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992. Association for Computational Linguistics, 2019.

Peter H. Schönemann. A generalized solution of the orthogonal Procrustes problem. *Psychometrika*, 31 (1):1–10, 1966.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maciaszek, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, 2017.

Harshil Vejendla. Drift-adapter: Lightweight adapters for embedding model upgrades. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2025. arXiv:2509.23471.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. C-Pack: Packaged resources to advance general chinese embedding. *arXiv preprint arXiv:2309.07597*, 2023.

Jinsung Yoon and Sercan O. Arik. Embedding-converter: A unified framework for cross-model embedding transformation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 25464–25482. Association for Computational Linguistics, 2025.