Latent Space Cartography Applied to Wikidata: Relational Displacement Analysis Reveals a Silent Tokenizer Defect in mxbai-embed-large

Emma Leonhart
github.com/EmmaLeonhart/latent-space-cartography

Abstract

We apply latent space cartography — the systematic mapping of structure in pre-trained embedding spaces (Liu et al., 2019) — to three general-purpose text embedding models using Wikidata knowledge graph triples as probes. The method is a standard application of TransE-style relational displacement analysis (Bordes et al., 2013) to frozen (non-KGE) embeddings: given any embedding model and any knowledge base, it discovers which relations manifest as consistent vector displacements and which do not. Applied to mxbai-embed-large (1024-dim), nomic-embed-text (768-dim), and all-minilm (384-dim), the procedure identifies 30 relations that are consistent across all three models, confirming that these are properties of the semantic relationships rather than artifacts of any single model. A correlation between geometric consistency and prediction accuracy (r = 0.861, 95% CI [0.773, 0.926]) reproduces across models, meaning the consistency metric predicts which discovered operations will be useful without held-out evaluation.

The primary empirical finding is a previously unreported defect in how the Ollama runtime serves mxbai-embed-large: 147,687 cross-entity embedding pairs at cosine similarity ≥ 0.95, with diacritic-bearing input collapsing into a single `[UNK]`-dominated attractor region. Crucially, this is **not** an inherent property of the model and is **not** long-standing: a version bisection over 21 Ollama releases (Section 5.4) localizes it to a runtime regression introduced in **Ollama v0.14.0 (released 2026-01-10)**. The identical mxbai-embed-large registry blob is completely healthy on Ollama $\leq$ v0.13.4 (diacritical collision rate ≈ 0, indistinguishable from an ASCII control) and defective on every release from v0.14.0 through the current v0.24.0 ($\approx$ 10–11%). On affected versions, "Hokkaidō" has cosine similarity 1.0 with "Éire", "Djazaïr", and "Filasṭīn" — completely unrelated words in different languages — while having cosine similarity of only 0.45 with its own ASCII equivalent "Hokkaido"; these collisions occupy the densest regions of the embedding space (71% in the densest quartile). The defect is silent: it affects any RAG system, semantic search engine, or knowledge graph application serving mxbai-embed-large via an affected Ollama version with non-ASCII input, and standard benchmarks (MTEB, etc.) do not test for it. The method, all code, and all data are publicly available.

1 Introduction

That embedding spaces encode relational structure as vector arithmetic is well established. The word2vec analogy `king - man + woman` $\approx$ `queen` (Mikolov et al., 2013) demonstrated this for distributional word embeddings. TransE (Bordes et al., 2013) formalized the insight for knowledge graphs, training embeddings such that `h + r` $\approx$ `t` for each triple (head, relation, tail). Subsequent work introduced rotations (RotatE; Sun et al., 2019), complex-valued embeddings (ComplEx; Trouillon et al., 2016), geometric constraints for hierarchical relations (box embeddings; Vilnis et al., 2018), and extensive theoretical analysis of which relation types admit which geometric representations (e.g., Wang et al., 2014; Kazemi & Poole, 2018).

Preprint.

The KGE research program is *constructive*: it builds embedding spaces optimized for relational reasoning. A complementary *cartographic* approach — mapping the structure that pre-trained spaces already encode — has been explored through visual analysis tools (Liu et al., 2019) and probing classifiers (Conneau et al., 2018; Hewitt & Manning, 2019), but these techniques are typically applied to answer specific hypotheses about specific models. Systematic relational mapping across all predicates in a knowledge base, applied to frozen general-purpose embeddings, remains underexplored.

We apply standard TransE-style relational displacement analysis to frozen text embeddings, systematically sweeping over all predicates in a Wikidata knowledge graph. The procedure is not methodologically novel — it packages known techniques (displacement consistency, leave-one-out evaluation) into a replicable pipeline. What is novel is what the pipeline found when applied to a domain that standard benchmarks do not cover.

The paper has three contributions:

1. **Cross-model relational mapping.** Applied to three models (mxbai-embed-large, nomic-embed-text, all-minilm), the procedure identifies 30 relations that manifest as consistent displacements across all three — confirming that the mapped structure is a property of the semantic relationships, not any particular model. A correlation between consistency and prediction accuracy (r = 0.861) means the consistency metric is self-calibrating.
2. **Discovery of a silent serving regression.** The same procedure, applied to a domain-specific seed (Engishiki, a Japanese historical text), surfaced a large-scale defect in mxbai-embed-large as served by the Ollama runtime: 147,687 cross-entity embedding pairs at cosine ≥ 0.95. Diacritic-bearing input collapses into a single `[UNK]`-dominated attractor region regardless of text content. This is a serving-stack regression, not a property of the published model weights: the same registry blob is healthy under older Ollama and the failure is silent and benchmark-invisible.
3. **Exact provenance via version bisection.** We bisect the regression over 21 Ollama releases and localize it to **Ollama v0.14.0 (2026-01-10)**: clean on $\leq$ v0.13.4 (diacritical collision rate ≈ 0), defective on every release v0.14.0 $\rightarrow$ v0.24.0 ($\approx$ 10–11%), with an unchanged model blob throughout. Controlled pairs characterize the symptom on affected versions: the diacritical form of a word (e.g., "Hokkaidō") is more similar to an unrelated diacritical word ("Éire", cosine 1.0) than to its own ASCII equivalent ("Hokkaido", cosine 0.45) — ruling out diacritic *stripping* and pointing to `[UNK]`-token *dominance* in Ollama's tokenization path, not a flaw in the model itself.

1.1 Key Findings

1. **Relational displacement generalizes across models.** Of 159 predicates tested ($\geq$10 triples each), 86 produce consistent displacement vectors in mxbai-embed-large, with 30 universal across all three models. Functional (many-to-one) relations encode as consistent displacements; symmetric relations do not — matching the predictions of the KGE literature (Wang et al., 2014).
2. **Consistency predicts accuracy.** The correlation between geometric consistency and prediction accuracy (r = 0.861, 95% CI [0.773, 0.926]) means the consistency metric functions as a self-calibrating quality indicator. This correlation is not tautological: consistency is computed over all triples, while MRR uses leave-one-out evaluation where each prediction excludes the test triple.
3. **A silent serving regression, bisected to Ollama v0.14.0.** The procedure revealed 147,687 cross-entity embedding pairs at cosine ≥ 0.95 — short diacritical strings collapsing, regardless of language/script/meaning, into a single `[UNK]`-dominated region. A version bisection localizes the cause to Ollama v0.14.0 (2026-01-10): the same model blob is clean on Ollama $\leq$ v0.13.4 and defective on $\geq$ v0.14.0. Controlled pairs characterize the symptom: "Hokkaidō" $\leftrightarrow$ "Éire" = 1.0 cosine, "Hokkaidō" $\leftrightarrow$ "Hokkaido" = 0.45 cosine.
4. **The regression is silent and systemic.** Standard benchmarks (MTEB, etc.) do not test diacritic-rich input at scale, and the failure raises no error. Any RAG system or semantic search serving mxbai-embed-large via Ollama $\geq$ v0.14.0 silently fails on queries containing

diacritical marks — returning results from the [UNK] attractor region — and has done so since that release shipped on 2026-01-10.

5. **Domain-specific seeds expose domain-specific failures.** The Engishiki seed (a Japanese historical text) naturally reaches romanized non-Latin terminology that standard benchmarks never touch. This is not a limitation but an experimental design choice: different seeds probe different regions of the embedding space.

2 Related Work

2.1 Knowledge Graph Embedding

TransE (Bordes et al., 2013) established that relations can be modeled as translations (h + r $\approx$ t) in learned embedding spaces. Subsequent work analyzed which relation types each model can represent: TransE handles antisymmetric and compositional relations but cannot model symmetric ones; RotatE (Sun et al., 2019) handles symmetry via rotation; ComplEx (Trouillon et al., 2016) handles symmetry and antisymmetry via complex-valued embeddings. Wang et al. (2014) and Kazemi & Poole (2018) provided systematic analyses of the relation type expressiveness of different KGE architectures. Our work does not introduce a new embedding method but applies the known displacement test systematically to frozen general-purpose (non-KGE) embedding spaces.

2.2 Word Embedding Analogies

Mikolov et al. (2013) showed that king - man + woman $\approx$ queen holds in word2vec. Subsequent work (Linzen, 2016; Rogers et al., 2017; Schluter, 2018) showed these analogies are less robust than initially claimed, often reflecting frequency biases and dataset artifacts. Ethayarajh et al. (2019) formalized the conditions under which analogy recovery succeeds, showing it requires the relation to be approximately linear and low-rank in the embedding space. Our work is consistent with these findings: the relations we recover are exactly those that satisfy the linearity condition (functional, bijective), and those that fail are those the theory predicts will fail (symmetric, many-to-many).

2.3 Latent Space Cartography

Liu et al. (2019) introduced *latent space cartography* as a visual analysis framework for interpreting vector space embeddings, enabling discovery of relationships, definition of attribute vectors, and verification of findings across latent spaces. Their work demonstrated the cartographic approach on image generation models, cancer transcriptomes, and word embedding benchmarks. Our work extends this cartographic paradigm to systematic relational displacement analysis: rather than visual exploration, we sweep over all predicates in a knowledge graph and characterize which relations encode as consistent vector arithmetic. The individual techniques (displacement consistency, leave-one-out evaluation) are standard; we apply them systematically as a mapping procedure.

2.4 Neurosymbolic Integration

Logic Tensor Networks (Serafini & Garcez, 2016), Neural Theorem Provers (Rocktäschel & Riedel, 2017), and DeepProbLog (Manhaeve et al., 2018) integrate logical reasoning into neural architectures. These constructive approaches build systems that reason logically. Our work maps what relational structure existing spaces already encode, rather than building new systems to produce it.

2.5 Probing and Representation Analysis

Probing classifiers (Conneau et al., 2018; Hewitt & Manning, 2019) test what linguistic properties are encoded in learned representations. Our displacement consistency metric is analogous to a probe, but operates at the relational level and uses vector arithmetic rather than learned classifiers. Rather than testing specific hypotheses, we sweep over all available predicates in a knowledge base.

2.6 Embedding Defects and Failure Modes

The glitch token phenomenon (Li et al., 2024) documents poorly trained embeddings for low-frequency tokens in LLMs. Our collision finding extends this to sentence-embedding models, showing that entire *classes* of input (romanized non-Latin scripts, diacritical text) collapse into near-identical regions. Systematic relational probing detects these defects as a byproduct, providing a practical auditing tool for embedding quality.

2.7 Tokenizer-Induced Information Loss

WordPiece (Schuster & Nakajima, 2012) and BPE (Sennrich et al., 2016) tokenizers are known to struggle with out-of-vocabulary and non-Latin text. Rust et al. (2021) showed that tokenizer quality strongly predicts downstream multilingual model performance. Systematic relational probing provides a way to detect these failures geometrically: by probing a specific domain via BFS traversal, tokenizer-induced information loss becomes visible as large-scale embedding collisions.

3 Method

3.1 Problem Formulation

Given: - An embedding function $f : \text{Text} \to \mathbb{R}^d$ (any text embedding model) - A knowledge base $\mathcal{K} = \{(s, p, o)\}$ of subject-predicate-object triples

Find: The subset of predicates $P^* \subseteq P$ whose triples manifest as consistent displacement vectors in the embedding space.

Definition (Relational Displacement). For a triple $(s, p, o) \in \mathcal{K}$, the *relational displacement* is the vector $\mathbf{g}_{s,p,o} = f(o) - f(s)$, connecting the subject's embedding to the object's embedding. This is the standard TransE formulation applied without training.

Definition (Displacement Consistency). For a predicate p with triples $\{(s_1, p, o_1), \ldots, (s_n, p, o_n)\}$, the *mean displacement* is $\mathbf{d}_p = \frac{1}{n}\sum_{i=1}^{n} \mathbf{g}_{s_i,p,o_i}$. The *consistency* of p is the mean cosine alignment of individual displacements with the mean:

$$\text{consistency}(p) = \frac{1}{n}\sum_{i=1}^{n} \cos(\mathbf{g}_{s_i,p,o_i}, \mathbf{d}_p)$$

A predicate with consistency > 0.5 encodes as a **consistent relational displacement**: its triples are approximated by a single vector operation. This threshold is not novel — it corresponds to the standard criterion for meaningful directional agreement in high-dimensional spaces.

3.2 Data Pipeline: Knowledge Graph Traversal as Probing Strategy

The key methodological choice is using **breadth-first search through an existing knowledge graph** to generate embedding probes. This inverts the typical KGE pipeline. Standard KGE methods start with an embedding space and train it to encode known relations. Our method starts with a knowledge graph and uses its structure to *probe* an existing embedding space — the graph tells us which pairs of entities *should* be related, and the embedding tells us whether that relationship manifests geometrically.

BFS from a seed entity is not merely a data collection convenience. It is a **directed probing strategy**: by choosing a seed in a specific domain (e.g., Engishiki, a Japanese historical text), the traversal naturally reaches the entities and terminology that are most relevant to that domain. This means the method systematically tests the embedding space in regions where it may be weakest — regions populated by obscure, non-Latin, or domain-specific terminology that standard benchmarks never touch. A seed in Japanese history pulls in romanized shrine names, historical figures with diacritical marks, and linked entities from Arabic, Irish, and indigenous-language Wikipedia articles. A seed in geography or biography would probe different regions. The choice of seed controls *where* the map is drawn.

1. **Entity Import.** Two seed strategies: (a) Breadth-first search from Engishiki (Q1342448), seeding 500 entities then importing all their triples and linked entities. The BFS expansion produces **34,335 unique entities** (not 500), of which 1,781 contain diacritical marks. With aliases, the total embedding count reaches 41,725. (b) Broad P31 (instance of) sampling across country-level entities to provide a domain-general baseline. Both seeds contribute to the relational displacement analysis (Section 4.1); the collision analysis (Section 5.4) focuses on the Engishiki seed because its 1,781 diacritic-bearing labels trigger tokenizer collisions at scale.
2. **Embedding.** Each entity's English label is embedded using mxbai-embed-large (1024-dim) via Ollama. Aliases receive separate embeddings. Total: 41,725 embeddings from the Engishiki seed. Labels are short text strings (typically 1-5 words), consistent with how these models are used in practice for entity linking and retrieval.
3. **Relational Displacement Computation.** For each entity-entity triple, compute the displacement vector between subject and object label embeddings. Total: 16,893 entity-entity triples across 1,472 unique predicates. This is the standard `h + r` $\approx$ `t` test from TransE, applied without training.

3.3 Discovery Procedure

For each predicate p with ≥ 10 entity-entity triples:

1. Compute all relational displacements $\{\mathbf{g}_i\}$
2. Compute mean displacement $\mathbf{d}_p$
3. Compute consistency: mean alignment of each $\mathbf{g}_i$ with $\mathbf{d}_p$
4. Compute pairwise consistency: mean cosine similarity between all pairs of displacements
5. Compute magnitude coefficient of variation: stability of displacement magnitudes

Note on unit-norm embeddings. mxbai-embed-large returns L2-normalized embeddings (‖v‖ = 1.0000). Consequently, displacement magnitudes are a deterministic function of cosine similarity: ‖f(o) - f(s)‖ = sqrt(2(1 - cos(f(o), f(s)))). The MagCV metric therefore carries no information independent of cosine distance for this model. We retain it for cross-model comparability, as other models (e.g., BioBERT) do not necessarily normalize.

3.4 Prediction Evaluation

For each discovered operation (consistency > 0.5), we evaluate prediction accuracy using **leave-one-out**:

For each triple (s, p, o): 1. Compute $\mathbf{d}_p^{(-i)}$ = mean displacement excluding this triple 2. Predict: $\hat{\mathbf{o}} = f(s) + \mathbf{d}_p^{(-i)}$ 3. Rank all entities by cosine similarity to $\hat{\mathbf{o}}$ 4. Record the rank of the true object o

We report Mean Reciprocal Rank (MRR) and Hits@k for k $\in \{1, 5, 10, 50\}$.

3.5 Composition Test

To test whether operations can be chained, we find all two-hop paths $s \xrightarrow{p_1} m \xrightarrow{p_2} o$ where both p_1 and p_2 are discovered operations. We predict:

$$\hat{\mathbf{o}} = f(s) + \mathbf{d}_{p_1} + \mathbf{d}_{p_2}$$

and evaluate whether the true o appears in the top-k nearest neighbors. We test 5,000 compositions.

4 Results

4.1 Operation Discovery

Of 159 predicates with $\geq$10 triples, 86 (54.1%) produce consistent displacement vectors:

Category	Count	Alignment Range
Strong operations	32	> 0.7
Moderate operations	54	0.5 – 0.7
Weak/no operation	73	< 0.5

Table 1. Distribution of discovered operations by consistency.

The top 15 discovered operations:

Predicate	Label	N	Alignment	Pairwise	MagCV	Cos Dist
P8324	funder	25	0.930	0.859	0.079	0.447
P2633	geography of topic	18	0.910	0.819	0.097	0.200
P9241	demographics of topic	31	0.899	0.799	0.080	0.215
P2596	culture	16	0.896	0.790	0.063	0.202
P5125	Wikimedia outline	20	0.887	0.777	0.089	0.196
P7867	category for maps	29	0.878	0.763	0.099	0.205
P8744	economy of topic	30	0.870	0.749	0.094	0.182
P1740	cat. for films shot here	18	0.862	0.728	0.121	0.266
P1791	cat. for people buried here	13	0.857	0.714	0.121	0.302
P1465	cat. for people who died here	29	0.857	0.725	0.124	0.249
P163	flag	31	0.855	0.723	0.123	0.208
P2746	production statistics	11	0.850	0.696	0.048	0.411
P1923	participating team	32	0.831	0.681	0.042	0.387
P1464	cat. for people born here	32	0.814	0.653	0.145	0.265
P237	coat of arms	21	0.798	0.620	0.138	0.268

Table 2. Top 15 relations by displacement consistency (alignment with mean displacement). N = number of triples. Pairwise = mean cosine similarity between all pairs of displacements. MagCV = coefficient of variation of displacement magnitudes. Cos Dist = mean cosine distance between subject and object.

4.2 Prediction Accuracy

Leave-one-out evaluation of all 86 discovered operations:

Predicate	Label	N	Align	MRR	H@1	H@10	H@50
P9241	demographics of topic	21	0.899	1.000	1.000	1.000	1.000
P2596	culture	16	0.896	1.000	1.000	1.000	1.000
P7867	category for maps	29	0.878	1.000	1.000	1.000	1.000
P8744	economy of topic	30	0.870	1.000	1.000	1.000	1.000
P5125	Wikimedia outline	20	0.887	0.975	0.950	1.000	1.000
P2633	geography of topic	18	0.910	0.972	0.944	1.000	1.000
P1465	cat. for people who died here	29	0.857	0.966	0.966	0.966	0.966
P163	flag	31	0.855	0.937	0.903	0.968	1.000
P8324	funder	25	0.930	0.929	0.920	0.960	0.960
P1464	cat. for people born here	32	0.814	0.922	0.906	0.938	0.938
P237	coat of arms	21	0.798	0.858	0.762	0.952	1.000
P21	sex or gender	91	0.674	0.422	0.121	0.945	0.989
P27	country of citizenship	37	0.690	0.401	0.162	0.892	0.973

Table 3. Prediction results for selected operations (full table in supplementary). MRR = Mean Reciprocal Rank. H@k = Hits at rank k. The four predicates achieving MRR = 1.000 are functional predicates with highly consistent Wikidata naming conventions (e.g., every country has exactly one "Demographics of [Country]" article). Perfect MRR is expected when: (a) the predicate is strictly functional (one object per subject), (b) the displacement is consistent (alignment > 0.87), and (c) the object label is semantically close to a predictable transformation of the subject. Crucially, the string overlap null model (Section 4.4) confirms this is not a string manipulation artifact: these same predicates achieve string MRR of only 0.008–0.046 vs. vector MRR of 1.000. The embedding captures the semantic operation; the label convention merely makes the target unambiguous among 41,725 candidates.

Aggregate statistics across all 86 operations:

Metric	Value	95% Bootstrap CI
Mean MRR	0.350	—
Mean Hits@1	0.252	—
Mean Hits@10	0.550	—
Mean Hits@50	0.699	—
Correlation (alignment $\leftrightarrow$ MRR)	r = 0.861	[0.773, 0.926]
Correlation (alignment $\leftrightarrow$ H@1)	r = 0.848	[0.721, 0.932]
Correlation (alignment $\leftrightarrow$ H@10)	r = 0.625	[0.469, 0.760]
Effect size: strong vs moderate MRR (Cohen's d)	3.092	(large)

Table 4. Aggregate prediction statistics with bootstrap confidence intervals (10,000 resamples). All correlations survive Bonferroni correction across 3 tests (adjusted alpha = 0.017).

The correlation between displacement consistency and prediction accuracy (r = 0.861, 95% CI [0.773, 0.926]) is practically useful as a quality filter. We note that this correlation has a natural

mathematical component: when displacement variance is low (high consistency), the mean displacement is by construction a better predictor. However, the correlation is not fully tautological: consistency is computed over all triples, while MRR uses **leave-one-out** evaluation where each prediction excludes the test triple, and a high-consistency predicate could still have poor MRR if the predicted region is crowded with non-target entities. The effect size between strong (>0.7) and moderate (0.5-0.7) operations is Cohen's d = 3.092, indicating the 0.7 threshold cleanly separates high-performing from marginal operations.

4.3 Two-Hop Composition

Over 5,000 tested two-hop compositions ($S + d_1 + d_2$):

Metric	Value
Hits@1	0.058 (288/5000)
Hits@10	0.283 (1414/5000)
Hits@50	0.479 (2396/5000)
Mean Rank	1029.8

Table 5. Two-hop composition results.

Selected successful compositions (Rank ≤ 5):

Chain	Rank
Tadahira →[citizenship]→ Japan →[history of topic]→ history of Japan	1
Tadahira →[citizenship]→ Japan →[flag]→ flag of Japan	1
Tadahira →[citizenship]→ Japan →[cat. people buried here]→ Category:Burials in Japan	2
Tadahira →[citizenship]→ Japan →[cat. people who died here]→ Category:Deaths in Japan	2
Tadahira →[citizenship]→ Japan →[cat. associated people]→ Category:Japanese people	3
Tadahira →[citizenship]→ Japan →[head of state]→ Emperor of Japan	4
Tadahira →[sex or gender]→ male →[main category]→ Category:Male	5

Table 6. Successful two-hop compositions. Note: all examples involve Fujiwara no Tadahira because our dataset is seeded from Engishiki (Q1342448), a Japanese historical text. Tadahira is one of the most densely connected entities in this neighborhood, appearing in many two-hop paths. The composition mechanism itself is general — the examples reflect dataset composition, not a limitation of the method.

4.4 String Overlap Null Model

A potential concern is that the discovered displacements merely capture string-level patterns — e.g., the displacement for "history of topic" (P2184) might simply encode the string prefix "History of" rather than relational knowledge. We test this with a string overlap null model: for each triple (s, p, o), we rank all entities by longest common substring ratio with the subject label. If string overlap achieves comparable MRR to vector arithmetic, the displacement is trivially explained by surface patterns.

Result: Vector arithmetic outperforms string overlap in 39/39 tested predicates (100%). No predicate is trivially string-based.

Metric	Vector Arithmetic	String Overlap (LCS)	Token Overlap
Mean MRR	0.633	0.013	0.056
Predicates with MRR > 0.5	24	0	0

The gap is not marginal: mean vector MRR is 49× higher than string MRR. Even the strongest string overlap scores (max 0.093 for P163 "flag") are far below the corresponding vector MRR (0.937). The 24 predicates with vector MRR > 0.5 all have string MRR < 0.1, confirming that the embedding captures relational structure that cannot be recovered from label text alone.

Limitations of this baseline. The string overlap null model is deliberately simple — it tests whether vector arithmetic reduces to substring matching, not whether it outperforms all possible string-based methods. A more sophisticated baseline (e.g., regex pattern matching for predicates like "Demographics of [X]", or edit-distance heuristics) would likely close some of the gap for the most formulaic predicates. The 49× ratio should be interpreted as evidence that the displacement is not a trivial string artifact, not as a claim about the difficulty of the prediction task itself. For the most formulaic predicates (demographics-of, geography-of), the prediction is easy by any method — the interesting finding is that vector arithmetic also works for predicates without formulaic naming (flag, coat of arms, head of state).

4.5 Failure Analysis

Predicates that resist vector encoding:

Predicate	Label	N	Alignment	Pattern
P3373	sibling	661	0.026	Symmetric
P155	follows	89	0.050	Sequence (variable direction)
P156	followed by	86	0.053	Sequence (variable direction)
P1889	different from	222	0.109	Symmetric/diverse
P279	subclass of	168	0.118	Hierarchical (variable depth)
P26	spouse	138	0.135	Symmetric
P40	child	254	0.142	Variable direction
P47	shares border with	197	0.162	Symmetric
P530	diplomatic relation	930	0.165	Symmetric
P31	instance of	835	0.244	Too semantically diverse

Table 7. Predicates with lowest consistency. Pattern = our characterization of why the displacement is inconsistent.

Three failure modes emerge:

1. **Symmetric predicates** (sibling, spouse, shares-border-with, diplomatic-relation): No consistent displacement direction because `f(A) - f(B)` and `f(B) - f(A)` are equally valid. Alignment ≈ 0.
2. **Sequence predicates** (follows, followed-by): The displacement from "Monday" to "Tuesday" has nothing in common with the displacement from "Chapter 1" to "Chapter 2." The *relationship type* is consistent but the *direction in embedding space* is domain-dependent.
3. **Semantically overloaded predicates** (instance-of, subclass-of, part-of): "Tokyo is an instance of city" and "7 is an instance of prime number" produce wildly different displacement vectors because the predicate covers too many semantic domains.

Instance-of (P31) at 0.244 is particularly notable. It is the most important predicate in Wikidata (835 triples in our dataset) and a cornerstone of first-order logic, yet it does not function as a vector operation. This suggests that embedding spaces systematically under-represent relational structure: the space encodes *entities* well but *predicates* poorly.

4.6 Cross-Model Generalization

To test whether discovered operations are model-agnostic or artifacts of a single model's training, we ran the full pipeline on two additional embedding models: nomic-embed-text (768-dim) and all-minilm (384-dim). All three models were given identical input: the same Wikidata entities seeded from Engishiki (Q1342448) with --limit 500.

Model	Dimensions	Embeddings	Discovered	Strong (>0.7)
mxbai-embed-large	1024	41,725	86	32
nomic-embed-text	768	69,111	101	54
all-minilm	384	54,375	109	41

Table 8. Operations discovered per model. All three models discover operations despite different architectures and dimensionalities.

30 operations are universal — discovered by all three models. These include demographics-of-topic (avg alignment 0.925), culture (0.923), economy-of-topic (0.896), flag (0.883), coat of arms (0.777), and central bank (0.793). The universal operations are exclusively functional predicates, confirming the functional-vs-relational split across architectures.

Overlap Category	Count
Found by all 3 models	30
Found by 2 models	15
Found by 1 model only	30

Table 9. Cross-model operation overlap. 30 universal operations constitute the model-agnostic core.

Cross-model consistency correlations (alignment scores on shared predicates): mxbai vs all-minilm r = 0.779, mxbai vs nomic r = 0.554, nomic vs all-minilm r = 0.358. The positive correlations confirm that consistency is not random — predicates that work well in one model tend to work well in others, though the strength varies by model pair.

The same relational structure emerges across three unrelated embedding models with different architectures, different dimensionalities, and different training data. The discovered operations are properties of the semantic relationships themselves, not artifacts of any particular model.

5 Discussion

5.1 Relation Types and Displacement

The pattern across Tables 2 and 7 confirms what the KGE literature predicts: **consistent displacements emerge for functional (many-to-one) and bijective (one-to-one) relations, and fail for symmetric, transitive, or many-to-many relations.** Each country has one flag, one coat of arms, one head of state — these produce consistent displacements. Symmetric relations (sibling, spouse, shares-border-with) produce no consistent direction because `f(A) - f(B)` and `f(B) - f(A)` are equally valid.

That this pattern holds in general-purpose text embedding models — models with no relational training signal — confirms that the relational structure is a property of the semantic relationships themselves. Any embedding model that captures semantic similarity will encode functional relations as consistent displacements and fail on symmetric ones.

5.2 The Consistency-Accuracy Correlation

The r = 0.861 correlation between consistency and prediction accuracy is useful as a practical quality indicator but should not be overstated. There is a natural mathematical tendency for low-variance displacement vectors (high consistency) to produce better mean-based predictions — if all displacements point roughly the same direction, the mean will be a good predictor almost by construction.

The correlation is therefore partly a geometric property of high-dimensional spaces, not purely an empirical discovery about these specific embedding models. What *is* empirically informative is the magnitude of the effect size between strong and moderate operations (Cohen's d = 3.092), which suggests the consistency threshold at 0.7 cleanly separates operations that work well from those that do not. The correlation is practically useful as a quality filter, even if its theoretical status is less remarkable than "self-diagnostic" framing might suggest.

5.3 Collision Geography

We independently measure two properties of each embedding: (a) its local density (mean k-NN distance) and (b) whether it collides with a semantically distinct entity at cosine ≥ 0.95. Dense regions could in principle have few collisions if the model separates semantically distinct entities effectively even in crowded neighborhoods. The following results describe what we observe when diacritic-rich input is embedded.

5.4 The Embedding Collapse: a Diacritic-Tokenization Regression in the Ollama Runtime

A previously unreported regression in a widely-used serving stack. mxbai-embed-large is one of the most popular open-source embedding models, very commonly served via Ollama in RAG systems, semantic search, and knowledge graph applications. The defect we report — affecting over 16,000 entities and producing 147,687 colliding embedding pairs — appears to have gone undetected because standard embedding benchmarks (MTEB, etc.) do not systematically probe non-Latin or diacritic-rich inputs at scale; a BFS traversal from a domain-specific seed does, because the knowledge graph naturally reaches the obscure terminology that benchmarks miss. As Section 5.4.1 establishes by version bisection, the defect is **not** intrinsic to the model: it is a regression in the Ollama runtime introduced in v0.14.0 (2026-01-10).

The Jinmyōchō collapse. Our collision analysis finds 147,687 cross-entity embedding pairs with cosine similarity ≥ 0.95 that represent genuine semantic collisions: different text mapped to near-identical vectors. This count reflects *pairwise* collisions: if k entities cluster together, they contribute $\binom{k}{2}$ pairs. The 147,687 total arises from approximately 16,067 entities (of 41,725) participating in at least one collision, organized into clusters of varying size. "Jinmyōchō" collides with 504 unique texts spanning romanized Japanese (kugyō, Shōtai), Arabic (Djazaïr, Filasṭīn), Irish (Éire), Brazilian indigenous languages (Aikanã, Amanayé), and IPA characters — words that share no orthographic or semantic relationship whatsoever.

The symptom is `[UNK]` token dominance, not diacritic stripping. If the tokenizer simply stripped diacritics, "Hokkaidō" would become "Hokkaido" and "Djazaïr" would become "Djazair" — different strings that should produce different embeddings. The observed failure mode on affected Ollama versions is more severe:

1. Diacritic-bearing characters (ō, ū, ī, ï, ş, ṭ, é, â, etc.) are routed to the `[UNK]` (unknown) token in the tokenization Ollama applies to this model.
2. For short input strings where diacritical characters constitute a significant fraction of the content, the tokenized sequence becomes dominated by `[UNK]` tokens.
3. The model pools over this `[UNK]`-dominated sequence, producing an embedding that reflects the `[UNK]` token's representation rather than the actual text content.
4. **All short diacritical strings converge to the same `[UNK]`-dominated attractor region**, regardless of language, script, or meaning.

This is a property of the *runtime*, not the model weights. The same mxbai-embed-large registry blob does **not** exhibit this behavior under Ollama $\leq$ v0.13.4 — there, the model's own tokenizer handles diacritical text correctly and diacritical input is statistically indistinguishable from an ASCII control. The `[UNK]`-collapse symptom only appears once the input is tokenized by Ollama v0.14.0+ (Section 5.4.1). So the root cause is a change in how Ollama v0.14.0 builds or applies this model's tokenizer, not an incomplete vocabulary in the published model.

Controlled evidence. We embed test pairs to confirm the mechanism (full data in `collisions.csv`):

Pair	Cosine Similarity	Interpretation
"Hokkaidō" ↔ "Éire"	1.000	Different languages, different meanings — identical embedding
"Jinmyōchō" ↔ "Filasṭīn"	1.000	Japanese ↔ Arabic — identical embedding
"Djazaïr" ↔ "România"	1.000	Arabic ↔ Romanian — identical embedding
"naïve" ↔ "Zürich"	1.000	French ↔ German — identical embedding
"Hokkaidō" ↔ "Hokkaido"	0.450	Same word, diacritic vs. ASCII — **dissimilar**
"Tōkyō" ↔ "Tokyo"	0.500	Same word, diacritic vs. ASCII — **dissimilar**
"Tokyo" ↔ "Berlin"	0.751	Control: two capitals — normal similarity

Table 10. Controlled collision pairs. The diacritical version of a word is more similar to an unrelated diacritical word in a different language (cosine 1.0) than to its own ASCII equivalent (cosine ~0.45). This rules out diacritic stripping as the mechanism: if the model stripped diacritics and embedded the ASCII form, "Hokkaidō" would be close to "Hokkaido", not to "Éire". Instead, the `[UNK]` tokens overwhelm the embedding, and all `[UNK]`-dominated inputs converge to the same point.

5.4.1 5.4.1 Provenance: a runtime regression bisected to Ollama v0.14.0

A natural objection is that this is a long-standing flaw in mxbai-embed-large's tokenizer. It is not. We pinned the Ollama runtime to each of 21 stable releases spanning 2025-04 to 2026-05, pulled the *same* `mxbai-embed-large` registry tag in each, and re-ran the full Wikidata collision scan. The model blob is content-addressed and identical across every run; the only independent variable is the Ollama runtime version.

Ollama release	Date	Diacritical collision rate	Mean cosine	Verdict
v0.6.5, v0.12.9, v0.13.4	2025-04 → 2025-12-13	≈ 0.0%	~0.39	**clean** (= ASCII control)
v0.14.0	**2026-01-10**	**10.5%**	**0.59**	**defect — regression introduced here**
v0.14.1 … v0.15.4	2026-01 → 2026-02	10.5–11.6%	~0.59	defect
v0.17.0, v0.19.0, v0.20.2, v0.21.0, v0.22.0, v0.23.4, v0.24.0	2026-02 → 2026-05	10.3–11.1%	~0.59	defect

Table 11. Ollama version bisection. A clean, single-release boundary: every release through v0.13.4 (2025-12-13) is healthy; the regression appears at v0.14.0 (2026-01-10) and persists through the current v0.24.0. Because the model is byte-identical across the boundary, the defect is unambiguously a regression in the Ollama serving runtime, introduced in the v0.13.5 → v0.14.0 release. It is therefore recent (not "years old") and reproduces deterministically on a pinned v0.14.0+ runtime — which is how our CI now asserts it (a two-sided test: must be clean on v0.13.4, must reproduce on v0.14.0). Identifying the precise upstream commit within that release is left to Ollama maintainers; the v0.14.0 changelog notably includes an embedding-path change ("an error will now return when embeddings return `NaN` or `-Inf`").

The collapse zone is dense, not sparse. Geometric analysis of 16,067 colliding embeddings (vs. 74,760 non-colliding) reveals:

1. **Colliding embeddings are 2.4× denser than non-colliding ones.** Mean k-NN distance for colliding embeddings is 0.106, vs 0.258 for non-colliding (ratio 0.41×).
2. **71% of colliding embeddings fall in the densest quartile,** vs the expected 25% if uniformly distributed. Only 3.2% fall in the sparsest quartile.
3. **The collapse zone is not geometrically isolated.** The distance from a colliding embedding to its nearest non-colliding neighbor (mean 0.119) is nearly identical to the non-colliding-to-non-colliding distance (mean 0.121, ratio 0.98×).

This means the `[UNK]` attractor region sits *among* the well-structured embeddings, not apart from them. The colliding embeddings crowd into already-dense neighborhoods where the model cannot differentiate them from legitimate nearby entities.

The defect is silent and likely exploitable. The `[UNK]`-dominated embedding region has several concerning properties: (1) it is invisible to standard benchmarks, (2) the runtime returns a confident-looking embedding vector rather than an error, (3) any downstream system treating this vector as meaningful will silently produce wrong results. Because the regression shipped in a widely-used runtime on 2026-01-10 and persists through the current release, any RAG pipeline, semantic search engine, or knowledge graph application that has processed non-ASCII input through mxbai-embed-large served by Ollama $\geq$ v0.14.0 has, since that date, been mapping those inputs to a single undifferentiated region. The scale of affected systems is difficult to estimate, but given Ollama's popularity as a serving runtime and the prevalence of diacritical marks in non-English text, the impact is likely substantial.

The phenomenon is reminiscent of glitch tokens (Li et al., 2024) but at a different scale: entire *classes of input* (any text containing diacritical marks) rather than individual tokens, and in sentence-embedding models rather than LLMs.

Why the Engishiki seed matters. Engishiki (Q1342448) is a 10th-century Japanese text whose entities include romanized shrine names (Jinmyōchō, Shikinaisha), historical Japanese personal names, and linked entities from Arabic, Irish, and indigenous-language Wikipedia articles. This floods the embedding space with exactly the inputs that trigger `[UNK]` token dominance, making the phenomenon measurable at scale. The defect exists regardless of seed choice — any diacritical input triggers it — but the Engishiki seed makes it *statistically visible* by providing thousands of affected entities in a single BFS traversal.

5.5 Practical Implications

The diacritic-collapse regression has immediate practical consequences. Any system serving mxbai-embed-large via Ollama $\geq$ v0.14.0 for semantic search, RAG, or knowledge graph completion over non-ASCII text has been silently affected since 2026-01-10. A user querying "Hokkaidō" retrieves results from the `[UNK]` attractor region — potentially returning "Éire", "Djazaïr", or any other diacritical string — rather than results related to the Japanese island. The failure is silent: the runtime returns a valid-looking 1024-dimensional vector, and no error is raised.

The broader lesson is about the *serving stack*, not the model: a point-release of a popular inference runtime silently corrupted multilingual embeddings for a model that was, and remains, correct at the weights level. We deliberately do not generalize the mechanism to other models — we observed no such collapse on nomic-embed-text or all-minilm, and the defect vanishes on older Ollama for mxbai-embed-large itself. The practical recommendations are therefore: (1) test embedding *deployments* (model + runtime + version) with diacritic-rich input before and after every runtime upgrade, and (2) pin and record the serving-runtime version as part of any embedding-system provenance — a regression of this kind is invisible at the model level and to standard benchmarks.

5.6 Limitations

1. **Three embedding models.** We validate across mxbai-embed-large (1024-dim), nomic-embed-text (768-dim), and all-minilm (384-dim), finding 30 universal relations. All three are English-language text embedding models trained on similar corpora. Testing on multilingual models or domain-specific models (e.g., biomedical) would further characterize the generality of the three-regime structure.

2. **Collision geometry analysis covers one seed.** The distance metrics characterizing the embedding collision zone (Section 5.4) are computed from the Engishiki-seeded dataset. Multi-seed analysis would test whether the same crowding pattern holds across domains.

3. **Label embeddings only.** We embed entity *labels* (short text strings), not descriptions or full articles. This deliberately mirrors how these models are used in practice for entity linking and knowledge graph completion (short query strings, not full documents). Richer textual representations might shift some entities out of the sparse zone, but the label-only setting represents a common real-world deployment pattern for these models.

4. **Potential training data overlap.** The embedding models tested were trained on large web crawls that likely include Wikipedia content, and Wikidata entities often have corresponding Wikipedia articles. This raises the possibility that some discovered displacements reflect memorized associations from training data rather than emergent geometric structure. The cross-model consistency (30 universal operations across three independently trained models) provides partial mitigation: memorization patterns would be model-specific, while consistent operations across architectures suggest structural encoding. However, a definitive test would require embedding models trained on corpora that exclude Wikipedia, which we leave for future work.

5. **Mechanism localized empirically, not from source.** We establish by version bisection that the regression entered at Ollama v0.14.0 with the model byte-unchanged, which rules out an inherent model-tokenizer flaw and rules in an Ollama-side tokenization/serving change. We do not pinpoint the exact upstream commit or its internal cause from Ollama source; that requires a diff of the v0.13.5 $\rightarrow$ v0.14.0 release and is left to upstream maintainers. Whether other runtimes (llama.cpp, vLLM, sentence-transformers direct) exhibit the same collapse for this model is untested and we make no claim about them.

6. **Relational displacement, not full FOL.** We test which binary relations encode as consistent vector arithmetic. Full first-order logic includes quantifiers, variable binding, negation, and complex formula composition, none of which we test. Extending the displacement analysis to richer logical operations is future work.

6 Conclusion

We apply latent space cartography — systematic relational displacement analysis using knowledge graph triples — to three general-purpose text embedding models. The procedure, which packages standard TransE-style evaluation into a replicable pipeline, identifies 30 relations that manifest as consistent vector displacements across all three models. The functional-vs-symmetric split predicted by the KGE literature reproduces across models and domains.

The primary finding is a silent diacritic-collapse defect in mxbai-embed-large *as served by the Ollama runtime*, in which diacritic-bearing input collapses into a single `[UNK]`-dominated attractor region. A version bisection over 21 Ollama releases localizes it precisely: the model is byte-identical and healthy on Ollama $\leq$ v0.13.4, and the regression enters at v0.14.0 (2026-01-10), persisting through the current v0.24.0. Controlled pairs characterize the symptom on affected versions: the diacritical version of a word is more similar to an unrelated diacritical word in a different language (cosine 1.0) than to its own ASCII equivalent (cosine ~0.45). The defect affects 16,067 entities in our dataset (147,687 colliding pairs), is concentrated in the densest regions of the embedding space, and is invisible to standard benchmarks. It is a recent serving-runtime regression — not a years-old model flaw — that has silently degraded any non-ASCII embedding workload running on Ollama $\geq$ v0.14.0 since 2026-01-10.

The defect was discovered because the cartographic procedure, seeded from a Japanese historical text (Engishiki), naturally reached the diacritic-rich terminology that standard benchmarks never test. This suggests a broader lesson: systematic probing of embedding spaces with domain-specific knowledge graphs can surface defects that generic benchmarks miss. The practical recommendation is to test embedding models with representative non-ASCII input before deployment.

All code and data are publicly available.

References

Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., & Yakhnenko, O. (2013). Translating Embeddings for Modeling Multi-relational Data. *NeurIPS*, 26.

Conneau, A., Kruszewski, G., Lample, G., Barrault, L., & Baroni, M. (2018). What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. *ACL*.

Ethayarajh, K., Duvenaud, D., & Hirst, G. (2019). Towards understanding linear word analogies. *ACL*.

Hewitt, J., & Manning, C. D. (2019). A structural probe for finding syntax in word representations. *NAACL*.

Kazemi, S. M., & Poole, D. (2018). SimplE embedding for link prediction in knowledge graphs with baseline model comparison. *NeurIPS*.

Li, Y., Liu, Y., Deng, G., Zhang, Y., & Song, W. (2024). Glitch Tokens in Large Language Models: Categorization Taxonomy and Effective Detection. *Proceedings of the ACM on Software Engineering*, 1(FSE). https://doi.org/10.1145/3660799

Linzen, T. (2016). Issues in evaluating semantic spaces using word analogies. *RepEval Workshop*.

Liu, Y., Jun, E., Li, Q., & Heer, J. (2019). Latent Space Cartography: Visual Analysis of Vector Space Embeddings. *Computer Graphics Forum*, 38(3), 67–78. (Proc. EuroVis 2019).

Manhaeve, R., Dumančić, S., Kimmig, A., Demeester, T., & De Raedt, L. (2018). DeepProbLog: Neural probabilistic logic programming. *NeurIPS*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *NeurIPS*.

Rocktäschel, T., & Riedel, S. (2017). End-to-end differentiable proving. *NeurIPS*.

Rogers, A., Drozd, A., & Li, B. (2017). The (too many) problems of analogical reasoning with word vectors. *StarSem*.

Rust, P., Pfeiffer, J., Vulić, I., Ruder, S., & Gurevych, I. (2021). How good is your tokenizer? On the monolingual performance of multilingual language models. *ACL*.

Schluter, N. (2018). The word analogy testing caveat. *NAACL*.

Schuster, M., & Nakajima, K. (2012). Japanese and Korean voice search. *ICASSP*.

Sennrich, R., Haddow, B., & Birch, A. (2016). Neural machine translation of rare words with subword units. *ACL*.

Serafini, L., & Garcez, A. d'A. (2016). Logic Tensor Networks: Deep learning and logical reasoning from data and knowledge. *NeSy Workshop*.

Sun, Z., Deng, Z.-H., Nie, J.-Y., & Tang, J. (2019). RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. *ICLR*.

Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., & Bouchard, G. (2016). Complex embeddings for simple link prediction. *ICML*.

Vilnis, L., Li, X., Xiang, S., & McCallum, A. (2018). Probabilistic embedding of knowledge graphs with box lattice measures. *ACL*.

Wang, Z., Zhang, J., Feng, J., & Chen, Z. (2014). Knowledge graph embedding by translating on hyperplanes. *AAAI*.