Embeddings for Preferences, Not Semantics*

Carter Blair
Harvard University
carterblair@g.harvard.edu

Ariel D. Procaccia
Harvard University
arielpro@g.harvard.edu

Milind Tambe
Harvard University
tambe@g.harvard.edu

Abstract

Modern AI is opening the door to collective decision-making in which participants express their views as free-form text rather than voting on a fixed set of candidates. A natural idea is to embed these opinions in a vector space so that the substantial literature on facility location problems and fair clustering can be brought to bear. But standard text embeddings measure semantic similarity, whereas distances in facility location problems and fair clustering require what we call *preferential similarity*: a participant's agreement with a piece of text should be inversely related to their distance from it. Off-the-shelf embeddings inherit a coarse preference signal through a correlation between semantic and preferential similarity, but fail to capture preferences when the correlation breaks. We formalize this as an invariance problem: text embedding models encode both a preference-relevant signal (stance and values) and semantic nuisance (style and wording), and the two are observationally correlated, so a geometry that relies on nuisance can appear preference-correct even when it is not. We show that synthetic training data designed to break this correlation provably shifts the optimal scorer away from nuisance-dominated cosine and significantly improves preference prediction across 11 online deliberation datasets.

1 Introduction

Many emerging systems for collective decision-making allow participants to express their preferences in free-form text instead of voting on predetermined candidate sets. For example, two influential online deliberation platforms, Polis and Remesh, allow participants to write statements and vote on statements written by others [Small et al., 2021]. Similarly, in generative social choice, participants express their preferences using free-form text, which are then aggregated into a slate of representative statements [Fish et al., 2026]. A commonality among these systems is that they require methods to group participants and to estimate a participant's utility for statements they did not vote on. Since inputs are free-form text, a natural idea is to embed each participant's text using a text embedding model. This would permit the grouping of participants via clustering in the embedding space and distances could be used to estimate participants' utility for statements they did not vote on. Further, it would allow for novel applications of ideas from the facility location and fair clustering literature [Feldman et al., 2016, Chen et al., 2019, Micha and Shah, 2020, Kellerhals and Peters, 2024].

However, off-the-shelf text embedding models are mainly trained and evaluated on semantic tasks such as retrieval, textual similarity, and natural-language inference [Muennighoff et al., 2023]. These tasks tend to reward placing texts close together in embedding space when they discuss the same topic or answer a similar query. They do not necessarily require close points to be mutually endorsable or, in other words, preferentially similar.

The distinction between semantic and preferential similarity is important to understand if embeddings are to be used inside preference aggregation procedures. To make the importance clear, it is useful to

*Code: https://github.com/cartgr/Embeddings-for-Preferences
Model: https://huggingface.co/cartgr/embeddings-for-preferences-st5-xl

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Table 1: A hard triplet illustrating the mismatch between semantic and preferential similarity. The semantic distractor uses the anchor's wording but reverses its stance, while the preference match preserves the stance but changes the wording. A base embedding model ranks the distractor closer to the anchor. A preference-tuned and topic-specific projected embedding produce the correct ranking.

		Similarity		
Role	**Text**	**Base**	**Tuned (§5)**	**Projected (§7)**
Anchor	"It should be **legal**. Religion has **no** place in politics if we truly want to be free."			
Distractor	"It should be **illegal**. Religion has **a** place in politics if we truly want to be free."	**0.87**	0.79	−0.74
Match	"To ensure true freedom, it's crucial that legislation remains secular, without religious influence."	0.82	**0.81**	**0.99**

imagine two statements from either side of a controversial political debate. They may share the same style and topic, as well as many of the same words, all of which a generic text embedding model may pick up on. However, they would not be mutually endorsed. Table 1 gives an example where a small alteration of a statement produces near-identical surface similarity to the anchor yet significantly alters the preferential similarity. A standard embedding model (ST5-XL) scores the altered distractor statement as more similar than an opinion-aligned alternative expressed in different words.

We frame the mismatch between semantic and preferential geometry as an invariance problem [Achille and Soatto, 2018]: a preference geometry should be invariant to wording and style, and sensitive only to stance and values. Generic embedding models do not have this invariance, because they encode topical and stylistic features that are useful for retrieval and similarity tasks but unrelated to whether two statements would be endorsed by the same person. On natural deliberation data, this gap is partly hidden because semantic and preferential similarity are correlated: people who share a stance often share wording. In §4.1 we make the confounding explicit by decomposing the cosine margin into a preference component and a nuisance component. Cosine weights the two equally, so when they agree (as on natural data) it looks correct, but when they disagree (as on the hard triplet in Table 1) the nuisance dominates and cosine fails.

Our method follows from this diagnosis. We synthesize triplets consisting of an anchor, a preference match with different wording, and a semantic distractor with high surface overlap but opposite stance. Training on these triplets forces cosine to down-weight nuisance variation that ordinary data leaves confounded with preference. We prove that, under this hard-triplet distribution, the Bradley-Terry risk is strictly decreased by reducing the nuisance contribution below cosine's unit weighting. Empirically, our method significantly improves performance on hard triplets and on preference prediction across 11 online deliberation datasets.

When per-topic votes are available, which is common in online deliberation platforms, the same framework suggests an even simpler method: learning a low-rank projection of the frozen embedding. Despite its simplicity, this projected embedding performs very well in practice.

To summarize, our contributions are as follows:

1. We diagnose the mismatch between semantic and preferential similarity as an invariance problem, and formalize it via a decomposition of the cosine margin into preference signal and nuisance.
2. We introduce a hard-triplet synthesis procedure and prove that training on it strictly decreases the Bradley-Terry risk relative to standard cosine.
3. We demonstrate substantial gains on hard triplets and on preference prediction across 11 online deliberation datasets.
4. We show that when per-topic votes are available, a low-rank projection of frozen embeddings outperforms full preference tuning.

2 Related Work

A growing body of work on collective decision-making over free-form text relies on some geometry over participants or statements: Polis derives opinion maps from vote matrices [Small et al., 2021],

generative social choice groups statements in an LLM-defined feature space to produce representative slates [Fish et al., 2026], Blair et al. [2026] model consensus as a region of embedding space, and De et al. [2026] use cosine similarity of embeddings as participant utility when auditing justified representation in slates of questions. Our goal in this paper is to answer a prerequisite question. Namely, do distances in general-purpose sentence embedding spaces reflect preferential similarity? And, if not, can they be realigned so that they do?

Another line of work fine-tunes sentence encoders for opinion-related tasks, including stance-aware embeddings for opinion mining [Ghafouri et al., 2024] and sparsity-aware embeddings for contradiction retrieval [Xu et al., 2024]. However, in §6 we find that neither works well for our task. Our recipe is also related to SimCSE [Gao et al., 2021], which uses NLI entailments as positives and contradictions as hard negatives. Our task is different and we go a step further by engineering triplets where the nuisance signal intentionally points in the wrong direction (see Appendix E.4 for the relevant ablation). An extended related work discussion is in Appendix A.

3 Evaluation Setup

Before presenting our diagnosis and method, we describe the evaluation data and metrics used throughout. We evaluate on 11 datasets from three deliberation platforms. Generative social choice (GSC) [Fish et al., 2026] provides surveys in which participants write free-text opinions and then rate AI-generated statements (two abortion surveys and one on chatbot personalization). Remesh provides binary agree/disagree votes on others' open-ended responses across three topics (campus protests, foreign intervention, right to assemble). Polis [Small et al., 2021] provides comment-level agree/disagree votes across five conversations (Seattle minimum wage, Bowling Green, Brexit, Canadian electoral reform, universal basic income). Some participants author comments in addition to voting. Together these span text lengths from short Polis comments to multi-paragraph GSC opinions and cover 1,462 participants, 3,958 statements, and 1.46M pairwise preference triplets. Dataset details and URLs are in Appendix B.

For each participant, we construct preference triplets (a, p, n) where a is the participant's own written text (the anchor), p is a statement they rated more favorably, and n is one they rated less favorably. Candidate models score each triplet by computing a similarity margin $s(a, p) - s(a, n)$; the triplet is correct if this margin is positive. We call the fraction of correctly ordered triplets *triplet accuracy*, and *cosine accuracy* when the scorer s is cosine similarity. In §7 we also evaluate the ideal-point scorer, for which s is the learned distance-based utility. Training details for our methods are given in §5.2, §7, and Appendix D.

4 Diagnosing Embedding Models

We begin by characterizing the preference signal in existing embedding models. We introduce a formal framework (§4.1), show that cosine is an approximation to the ideal-point utility margin, present empirical evidence for a correlation between nuisance and preference signal in the natural-data regime (§4.2), and diagnose the hard-triplet failure (§4.3).

4.1 Formal framework

Let $\psi : \mathcal{X} \to \mathbb{R}^d$ denote a pretrained encoder producing unit-norm embeddings. Within the embedding, preference on a given topic is governed by a *preference subspace* $S \subseteq \mathbb{R}^d$ of dimension $k \ll d$ that carries the stance-relevant structure. Let P_S denote orthogonal projection onto S and $P_{S^\perp}$ onto its complement, and abbreviate $\psi_S := P_S\psi$, $\psi_\perp := P_{S^\perp}\psi$. Each participant v writes anchor text a_v and has *ideal point* $u_v := \psi_S(a_v) \in S$, which is the projection of their own embedding onto S. Different participants have different anchors and thus different ideal points in S. For example, pro-choice and pro-life users live in opposite regions along the stance axis while sharing S as the relevant direction of variation.

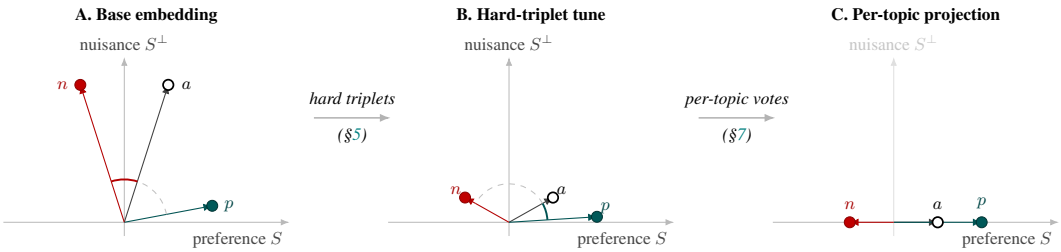

Figure 1: A hard triplet: anchor a, preference-match p (same stance, different wording), and semantic distractor n (opposite stance, same wording) with preference subspace S horizontal and nuisance $S^\perp$ vertical. (A) In the pretrained embedding, n shares a's nuisance component, so cosine ranks n above p. (B) Fine-tuning on counterfactual hard triplets downweights $\psi_\perp$ (Theorem 1). (C) With per-topic labels, a rank-r projection $L^\top$ maps onto S directly, discarding $S^\perp$.

We model the utility of candidate statement j as the negative squared Euclidean distance to the participant's ideal point within S.[2] Expanding the square,

$$U^*(v,j) \;=\; -\|\psi_S(a_v) - \psi_S(x_j)\|^2 \;=\; 2\,\langle \psi_S(a_v), \psi_S(x_j)\rangle - \|\psi_S(a_v)\|^2 - \|\psi_S(x_j)\|^2.$$

Pairwise preferences follow Bradley-Terry: $\Pr[p \succ n \mid v] = \big(1 + e^{-(U^*(v,p) - U^*(v,n))}\big)^{-1}$. For within-anchor rankings, the $\|\psi_S(a_v)\|^2$ term is constant across candidates and cancels out, so the ranking-relevant utility margin is

$$U^*(v,p) - U^*(v,n) \;=\; 2\,\underbrace{\langle \psi_S(a_v), \psi_S(x_p) - \psi_S(x_n)\rangle}_{\Delta_S} \;+\; \underbrace{\|\psi_S(x_n)\|^2 - \|\psi_S(x_p)\|^2}_{\Delta_{\text{norm}}}. \qquad (1)$$

The first term Δ_S is bilinear in the anchor and item projections and the second Δ_{norm} is an item-specific quadratic that depends only on candidate norms within S.

Cosine as an approximation. A cosine scorer on an off-the-shelf embedding model does not know S and uses all d dimensions equally. Because P_S and $P_{S^\perp}$ map into orthogonal subspaces, the cosine margin decomposes additively as

$$s(a_v,p) - s(a_v,n) \;=\; \underbrace{\langle \psi_S(a_v), \psi_S(x_p) - \psi_S(x_n)\rangle}_{\Delta_S} \;+\; \underbrace{\langle \psi_\perp(a_v), \psi_\perp(x_p) - \psi_\perp(x_n)\rangle}_{\Delta_T}. \qquad (2)$$

Comparing (2) with (1), cosine captures the bilinear part of the utility margin (up to a factor of two) but misses the item-norm difference Δ_{norm} entirely, and contributes an out-of-subspace nuisance Δ_T that has no utility counterpart. Cosine is therefore an approximation to the ideal-point utility margin, tight when Δ_{norm} is small and when Δ_T aligns with Δ_S. In §7 we find empirical evidence that this model describes real preferences well.

Figure 1 illustrates the three stages in this paper: the diagnosis that base cosine overweights the nuisance direction (§4), hard-triplet fine-tuning which suppresses Δ_T (§5), and a per-topic rank-r projection which projects onto S directly (§7).

4.2 Proximity bands: cosine carries a preference signal

With the framework in place, we first ask what cosine similarity on base embedding models captures on natural deliberation data. This diagnostic is pairwise (between an anchor statement and a candidate statement) rather than triplet-based. For each participant, we compute cosine similarity from their anchor embedding to every statement they voted on, bin these pairs into quintiles by similarity, and compute the approval rate per band. We use the binary-vote datasets (Remesh and Polis), where approval is directly observed.

Approval rises monotonically across all four encoders (Figure 2). The spread between the most distant and most similar quintile is roughly between 15 and 20 percentage points for each encoder.

[2] The derivations below extend to the general Mahalanobis case $-\|\psi_S(a_v) - \psi_S(x_j)\|_M^2$ for any PSD M on S: absorbing $\sqrt{M}$ into the encoder reduces it to the Euclidean case with no other change.

From this it is clear that cosine carries some preference signal, as statements closer to a participant's own (according to cosine similarity) are more likely to be approved by that participant.

However, where this signal comes from is less clear. A high cosine score can arise because the candidate is close to the anchor in the preference subspace, or because it shares topic, wording, style, or affect that happen to correlate with preference in naturally occurring text on deliberation platforms. More specifically, at the pair level similarity can be decomposed as

$$s(a_v, x_j) = \langle \psi(a_v), \psi(x_j) \rangle = \underbrace{\langle \psi_S(a_v), \psi_S(x_j) \rangle}_{s_S(a_v, x_j)} + \underbrace{\langle \psi_\perp(a_v), \psi_\perp(x_j) \rangle}_{s_T(a_v, x_j)}.$$

Figure 2 shows that the sum is predictive of approval but it does not identify whether the cause is the preference term s_S, the nuisance term s_T, or the two terms moving together. On natural deliberation data it could be the case that people who share a stance often also share wording and style. If this were the case, semantic similarity and preferential similarity would be observationally correlated.

The same ambiguity appears in the triplet margin used throughout the paper. For a triplet (a, p, n) with p preferred to n, differencing the pairwise scores gives

$$s(a, p) - s(a, n) = \Delta_S + \Delta_T.$$

Cosine gives Δ_S and Δ_T unit weight, but their effective strength on a dataset can be very different. The contribution of each term depends not only on the scorer's coefficient, but also on the norm, variance, dimensionality, and label-alignment of the component in the observed triplets. If Δ_S is large and reliably aligned with approval, then cosine succeeds because the embedding already contains a strong preference geometry. If Δ_T is large and merely correlated with approval, then cosine can also succeed on natural data, but for the wrong reason — namely, by using semantic features that happen to track preferences in the observed distribution.

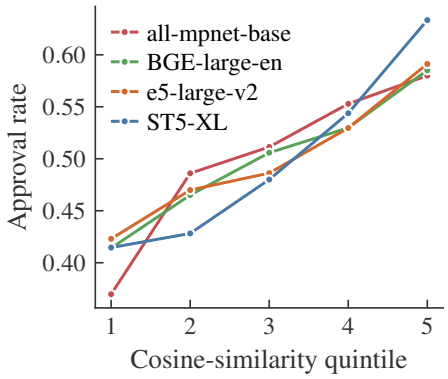

Figure 2: Approval rate by cosine-similarity quintile, pooled across the binary-vote evaluation datasets.

The proximity bands therefore leave open two explanations. Base embeddings may genuinely put nearby points near each other because they are preferentially similar. Alternatively, base embeddings may mostly capture semantic similarity, which may appear preference-aware only because semantic similarity is correlated with preferential similarity in data from online deliberation platforms. The next diagnostic is designed to ascertain which of the two explanations is correct by putting the two sources of signal in conflict.

4.3 Hard triplets: separating preference from surface similarity

To test whether the apparent preference signal in base embedding models results from a correlated nuisance component or from a true preference awareness, we construct hard triplets that deliberately decouple preference alignment from surface similarity. For each anchor, we generate two comparison statements: a *semantic distractor* that shares the anchor's vocabulary but takes the opposite stance, and a *preference match* that shares the anchor's opinion but uses different vocabulary. A model that relies on surface features will score the distractor higher, whereas a model that captures preferential similarity will score the match higher.

We generate 875 such triplets (100 per evaluation dataset where available) by rewriting real participant anchors with GPT-4o (prompt in Appendix G). On these triplets, cosine fails across every encoder family we test. Base sentence-T5-XL reaches only 48.3%, near chance. Three widely-used encoders perform far worse: e5-large-v2 (26.7%), BGE-large (6.3%), and all-mpnet-base (8.2%). These three rank the semantic distractor above the preference match 70–95% of the time. Meanwhile these same encoders score 58–60% on the natural-data triplets (Table 2). The drops, of up to roughly 50 percentage points, from natural to hard data suggest that cosine's apparent preference signal on natural data comes almost entirely from a nuisance component that is correlated with preferences on natural data.

Table 2: Triplet accuracy (%) on the natural datasets (test split) vs. the hard triplets described in §4.3.

Encoder	**Natural (mean, 11 ds.)**	**Hard** (n=875)	Δ
sentence-T5-XL	65.2	48.3	-16.9
e5-large-v2	60.4	26.7	-33.7
BGE-large-en-v1.5	59.2	6.3	-52.9
all-mpnet-base-v2	57.8	8.2	-49.6

In sum, hard triplets show that the natural-data signal from cosine on base embedding models is not robust when surface similarity and preference alignment conflict. This suggests using such conflicts not just as a diagnostic, but as supervision.

5 Decorrelated Preference Tuning

The hard-triplet diagnostic suggests a simple intervention: train on cases where surface similarity and preference alignment disagree. However, §4.2 and §4.3 suggest that these cases are rare in natural data. Therefore, we construct synthetic counterfactual triplets that break this correlation; we tune the encoder on them so that cosine learns to follow the preference-aligned item rather than the distractor that is similar in a superficial sense. We call this procedure *decorrelated preference tuning* (DPT).

5.1 Why hard triplets are corrective

We first formalize the direction in which hard-triplet training should move a scorer. Consider the class of bilinear scorers

$$s_A(a, x) = \psi(a)^\top A \psi(x),$$

with symmetric $A \in \mathbb{R}^{d\times d}$. To make the nuisance weight explicit, this can be written as

$$A(B, \lambda) = B + \lambda P_{S^\perp},$$

where $B = P_S B P_S$ only acts on the preference subspace and λ scales the nuisance. This family includes cosine on unit-norm embeddings when $B = P_S$ and $\lambda = 1$, and the S-projected scorer when $B = P_S$ and $\lambda = 0$.

For a triplet (a, p, n), the margin is

$$\psi(a)^\top A(B, \lambda)(\psi(p) - \psi(n)) = \Delta_B + \lambda \Delta_T,$$

where $\Delta_B := \psi_S(a)^\top B(\psi_S(p) - \psi_S(n))$ and where $\Delta_T = \langle \psi_\perp(a), \psi_\perp(p) - \psi_\perp(n)\rangle$ is the nuisance margin. Thus, λ controls how much the scorer relies on the nuisance part of cosine.

Let $\mathcal{G} = \sigma(\psi_S(a), \psi_S(p), \psi_S(n))$ denote the σ-algebra generated by the in-subspace parts of the triplet. Conditioning on $\mathcal{G}$ means holding the preference subspace projections fixed while averaging over the nuisance subspace variation.

The hard-triplet condition can be formalized as $\mathbb{E}[\Delta_T \mid \mathcal{G}] \leq 0$ almost surely, with strict inequality on a set of positive probability. In words, once the preference-relevant parts of the triplet are fixed, the nuisance margin does not help the preferred item and, on average, it favors the distractor. Since we assume the preferred item in the hard triplet is aligned with the anchor in the preference subspace,[3] this condition formalizes the idea that, in hard triplets, the preference and nuisance components point in opposite directions.

For the scorer $A(B, \lambda)$, define the Bradley-Terry population risk as

$$R(B, \lambda) = \mathbb{E}[\log(1 + e^{-\Delta_B - \lambda\Delta_T})].$$

The following result shows that, on any triplet distribution satisfying the hard-triplet condition, risk is reduced by decreasing the weight on the nuisance.

Theorem 1. *If $\mathbb{E}[\Delta_T \mid \mathcal{G}] \leq 0$ a.s., with strict inequality on a set of positive probability, then*

$$R(B, \lambda) < R(B, 1) \quad \textit{for every } \lambda \in [0, 1).$$

[3]The following proof of Theorem 1, however, does not require this.

The result does not require knowing S or directly optimizing λ. Instead, simply constructing a distribution of triplets where Δ_S points in the right direction and Δ_T points in the wrong direction for each triplet almost surely rewards scorers that put relatively more weight on the preference aligned part of the embedding. The proof is in Appendix C; its main steps are showing $R(B, \cdot)$ is convex, $R'(B, 0) > 0$, then using the convexity (and consequent non-decreasing derivative) to show that $R(B, \cdot)$ increases as λ moves from 0 toward 1.

5.2 Counterfactual hard-triplet generation and fine-tuning

Theorem 1 suggests that training on a distribution of hard triplets should implicitly cause the encoder to place less weight on the nuisance subspace. We now describe how we synthesize such a distribution and how we train on it.

To begin, we assemble a pool of 2,000 political and social issues from the Habermas Machine dataset [Tessler et al., 2024] and Kialo, [4] filtered by GPT-4o-mini for genuine policy debatability, and prompt Claude Sonnet 4 to generate 5 opinions per issue spanning the stance spectrum. We then sample anchors from this pool and prompt GPT-4o to rewrite each anchor into two versions. The *preference match* preserves the anchor's stance and values but changes vocabulary, framing, and sentence structure. The *semantic distractor* preserves much of the anchor's wording and structure but flips the stance. Together these form a triplet (a, p, n) in which the preferred item is stance-aligned but less surface-similar, while the dispreferred item is surface-similar but stance-opposed. Applying the same rewrite prompt to real participant anchors from the 11 evaluation datasets produces the held-out hard triplets used in §4.3 and §6. All prompts and model details are in Appendix G.

We remark that the training data is completely synthetic and was generated independently of the 11 benchmark datasets used for evaluation. So, any performance gains on the natural data benchmark are a strong indicator of generalization. In addition, the way the triplets are generated is specifically designed to approximate the hard-triplet condition, $\mathbb{E}[\Delta_T \mid \mathcal{G}] \leq 0$.

We adapt the pretrained encoder ψ with LoRA [Hu et al., 2022] ($r = 16$, $\alpha = 48$, targeting query and value projections). For a triplet (a, p, n) we train with Bradley-Terry loss over cosine differences:

$$\mathcal{L}_{\mathrm{BT}}(a, p, n) = \log\Big(1 + e^{-\left(\cos(\psi(a),\psi(p)) - \cos(\psi(a),\psi(n))\right)}\Big).$$

Training and inference both use cosine geometry, so what we optimize is what is used downstream. We do not train to convergence on the hard-triplet distribution since this would lead to negative weight on the nuisance subspace instead of the desired invariance.

6 Empirical Results

Fine-tuning substantially reduces the hard-triplet failure identified in §4.3. Our DPT-tuned sentence-T5-XL lifts hard-triplet (from the evaluation set) cosine accuracy from 48.3% to 80.0% and natural-triplet mean accuracy from 65.2% to 68.6%. Further, the recipe is not specific to sentence-T5: applying the same training to three widely-used BERT-derived encoders, e5-large-v2, BGE-large, and all-mpnet-base, produces similar patterns, with hard-triplet accuracy lifting by roughly 20 to 50 percentage points across models (Table 3). The natural-data lift is significant at the participant level under a paired Wilcoxon test ($p = 3.4 \times 10^{-9}$, pooled over the 442 test-split participants in the 11 datasets) and significant on 10 of 11 datasets under a triplet-level McNemar test (Appendix E.8).

Table 4 shows where the resulting DPT-tuned geometry sits relative to a broader set of off-the-shelf and stance-aware baselines, including methods explicitly designed for viewpoint or contradiction-sensitive similarity. It shows a representative cut of the full 25-model comparison (Appendix F). Our DPT-tuned sentence-T5-XL achieves the highest mean accuracy at 68.6%, compared to 65.2% for the next-best embedding (the untuned ST5-XL). Our method is best on 8 of 11 datasets, and no other model is best on more than one.

[4] https://huggingface.co/datasets/timchen0618/Kialo

Table 3: Base vs DPT-tuned cosine accuracy on the 11 natural-data datasets (test split) and on the 875 hard triplets across 5 training seeds. The recipe is hyperparameter-identical across models except for learning rate and hard-triplet count, which are val-selected per encoder.

	Natural (%)		**Hard (%)**		**Gain (pp)**	
Encoder	base	tuned	base	tuned	nat.	hard
sentence-T5-XL	65.2	68.6 ± 0.2	48.3	80.0 ± 0.4	+3.4	+31.7
e5-large-v2	60.4	63.0 ± 0.3	26.7	78.7 ± 0.5	+2.6	+51.9
BGE-large-en-v1.5	59.2	61.1 ± 0.2	6.3	32.8 ± 3.0	+1.9	+26.5
all-mpnet-base-v2	57.8	58.6 ± 0.2	8.2	27.4 ± 2.2	+0.8	+19.2

Table 4: Triplet accuracy (%) across the 11 evaluation datasets, abridged to seven representative models. A full comparison across 25 embedding models is in Appendix F.

	GSC			**Remesh**			**Polis**					
Model	AbG	AbV	Chat	Camp	Frgn	RTA	Seatl	BG	Brxt	Can	UBI	**Mean**
ST5-XL + DPT (ours)	**81.8**	**76.4**	65.9	**67.5**	**65.6**	69.5	**69.8**	**65.1**	**68.4**	**62.5**	61.9	**68.6**
ST5-XL	81.1	70.5	**69.7**	65.8	61.1	69.9	60.7	57.6	56.4	55.5	69.2	65.2
OAI text-emb-3-large	79.6	74.5	56.1	63.9	61.1	**71.4**	58.1	57.7	51.6	50.0	68.1	62.9
ST5-Large	75.5	65.5	57.2	64.6	59.9	69.2	58.8	58.1	49.9	56.8	**72.2**	62.5
BGE-large-en	64.7	59.5	56.5	62.5	57.6	68.0	54.2	55.5	53.2	52.9	66.3	59.2
StanceAware-SBERT	58.5	58.2	36.4	64.8	59.7	68.2	68.1	56.1	52.5	52.3	68.3	58.5
BGE-SparseCL (arguana)	62.4	61.7	42.7	59.2	57.2	66.1	50.1	56.1	57.5	52.6	63.0	57.2

7 Per-Topic Projected Embeddings

Decorrelated preference tuning trains a single encoder whose cosine geometry is more preference-aware without using any data from the target topic. This section asks what changes when participant votes over statements are available, as is often the case for online deliberation platforms. The section serves three purposes. First, it gives a stronger topic-specific deployment mode where the base embedding model is held fixed and a low-rank linear map is learned from participant votes. Second, it tests the ideal-point model from §4.1. And third, it sharpens the diagnosis of cosine similarity on base embedding models. If a low-rank linear map works on a frozen encoder, it suggests that the encoder already contains a preference signal. DPT makes this structure more visible to cosine by implicitly down-weighting the nuisance component, and the per-topic map recovers it from votes.

To begin, we fit the ideal-point utility directly on frozen sentence-T5-XL. Let a be a participant v's anchor and x_j a candidate statement. We learn a rank-r linear map $L^\top : \mathbb{R}^d \to \mathbb{R}^r$ and score candidates by squared Euclidean distance in the mapped space:

$$U(v, j) = -\left\|L^\top \psi(a_v) - L^\top \psi(x_j)\right\|^2 = 2\langle L^\top \psi(a_v), L^\top \psi(x_j)\rangle - \|L^\top \psi(x_j)\|^2 - \|L^\top \psi(a_v)\|^2. \quad (3)$$

We refer to $\tilde{\psi}(x) = L^\top \psi(x)$ as a *projected embedding*. This form is the ideal-point model with the preference subspace and metric learned from votes. The same map is applied to anchors and candidates, and utility is distance rather than inner product. Fitting L separately for each of the 11 deliberation datasets with Bradley-Terry loss on participant votes (3-way train/validation/test split, rank $r = 20$, 5 seeds) reaches 77.6% macro-mean accuracy over the datasets on the test split. Applied to the hard evaluation triplets, with no training on hard data, the same per-topic linear map reaches 81.1% macro-mean accuracy (Table 5, first row). This suggests that the frozen encoder already

Table 5: Each row relaxes one structural commitment of the ideal-point scorer; all are fit on frozen base sentence-T5-XL with Bradley-Terry loss across the 11 evaluation datasets (3-way participant split, val-selected hyperparameters, five seeds).

Scorer on frozen ψ	**Form**	**Nat. (%)**	**Hard (%)**
Ideal-point scorer *(ours)*	$-\|L^\top \psi(a) - L^\top \psi(x)\|^2$	77.6 ± 0.4	81.1 ± 0.5
Nonlinear *[drops linearity]*	$-\|\phi(\psi(a)) - \phi(\psi(x))\|^2$	73.5 ± 0.6	66.5 ± 3.1
Asymmetric *[drops tied projection]*	$-\|L_a^\top \psi(a) - L_x^\top \psi(x)\|^2$	77.0 ± 0.2	73.7 ± 0.4
Inner product *[drops distance]*	$\langle L^\top \psi(a), L^\top \psi(x)\rangle$	77.9 ± 0.3	73.3 ± 0.2

contains preference signal and that L simply learns to read it out. This may also explain why DPT is so sample efficient — the geometry only needs to be realigned rather than learned from scratch.

We now turn to the question of whether the model proposed in §4.1 is empirically justified. The ideal-point scorer commits to three main structural claims about $\tilde{\psi}$: the scorer is *linear* in ψ, *tied* between anchor and item (both projected by the same L), and a *distance* rather than an inner product. Table 5 relaxes each claim in turn. Replacing the linear $L^\top$ with a shared nonlinear MLP ϕ while keeping the squared-distance form costs 4.1% on the natural data and 14.6% on the hard evaluation triplets, indicating that ψ is already rich enough that added per-text nonlinearity overfits rather than helps. Projecting anchor and item through independent L_a, L_j matches natural-data accuracy but loses 7.4% on the hard evaluation triplets. Dropping the item-quadratic $-\|L^\top \psi(j)\|^2$ for the bilinear cross-term alone (same rank, same shared L) leaves natural accuracy unchanged but loses 7.8% on hard evaluation triplets. A possible explanation for this is that the item-norm encodes statement quality that directional similarity discards.

Our model also assumes that the learned space is low-dimensional. To test this we swept across ranks $r \in \{1, 2, 5, 10, 20, 50, 100\}$. Figure 3 shows that accuracy rises from $r = 1$ and plateaus at $r = 20$. Per-topic supervision is data-efficient as well. Fixing $r = 20$, the projected embedding crosses the universal DPT cosine at roughly 50 labeled triplets (Appendix E.10).

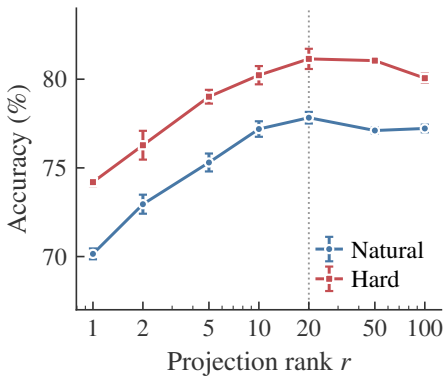

Figure 3: Per-topic scorer accuracy versus projection rank r. Mean±std over five seeds, macro-averaged over 11 datasets.

8 Discussion

This paper develops methods for learning a preference geometry over text, motivated by the design choices that deliberative systems must make. A key aspect of deliberation is exposing participants to a range of viewpoints [Mutz, 2006, Fishkin and Luskin, 2005, Sunstein, 2002]. Online platforms such as Polis, Remesh, Frankly, and the Stanford Online Deliberation Platform face this problem at scale, and must decide who should deliberate together, which comments to surface, and which viewpoints are missing [Small et al., 2021, Fishkin et al., 2019, Frankly, 2026]. A preference geometry gives one way to represent the viewpoints before making these decisions.

Two examples illustrate the point: forming deliberation groups, and aggregating the views expressed during deliberation. The composition of a deliberation group affects the quality of discussion, with quality highest at moderate levels of within-group disagreement [Esterling et al., 2015, Karpowitz and Mendelberg, 2007]. In practice, participants are assigned to small groups at random or by algorithms that balance demographics across groups alongside other objectives such as maximizing distinct pairwise meetings across rounds [Barrett et al., 2023, 2024], where demographic categories act as a coarse proxy for views [Mansbridge, 1999]. Yang and Bachmann [2025] move beyond demographics by using voting data as a proxy of views. A preference embedding space takes the final step, providing a direct measure of views in which objectives like balancing positions across rooms or controlling the level of within-room disagreement can be specified directly. Aggregation also has a natural formulation as voting in a metric space [Bulteau et al., 2021, Feldman et al., 2016]. Since platforms typically surface several comments rather than one, the relevant problem is selecting a slate, which corresponds geometrically to clustering. Here k-median and k-center capture utilitarian and egalitarian aggregation, while proportional fairness requires any large, aligned subgroup to have a representative [Chen et al., 2019, Micha and Shah, 2020, Kellerhals and Peters, 2024].

A preference geometry can also aid generative social choice [Fish et al., 2026, Boehmer et al., 2025], where generative models expand the set of candidate statements. Before generation, it can identify groups who deserve representation, during generation it can guide the generative model, and after generation, it can measure how well a generated statement represents its intended group. Appendix E.11 gives preliminary evidence that the tuned and projected geometries identify such groups better than the base embedding: on Remesh, users agree more with comments from their own cluster than from others, and the gap is significantly larger under the DPT-tuned and projected geometries.

Limitations and future work. First, our evaluation focuses on within-participant rankings, not whether absolute distances are calibrated, and since utility is latent, direct verification is difficult. Yet, Bradley-Terry training does push correctly-ordered pairs apart by a margin that scales with preference strength, and Appendix E.12 shows that similarity in the tuned and projected spaces tracks continuous Likert ratings better than the base model. Second, the gap between the universal tune and the per-topic projected embedding suggests that preferences have both a shared component across topics and a topic-specific component that only voting data on that topic recovers. A natural next step is to generate hard triplets conditioned on a target topic, producing a topic-specific embedding without per-topic votes. Finally, an embedding aimed at capturing preferences is an empirical and imperfect representation of preferences. It can support sense-making and help surface diverse views, but it should not be treated as a perfect representation of any individual's considered judgment, and it should not be used as a basis for binding decisions [Revel and Pénigaud, 2026].

Acknowledgments

This work was partially supported by the National Science Foundation under grant IIS-2229881; by the Office of Naval Research under grants N00014-24-1-2704 and N00014-25-1-2153; and by grants from the Cooperative AI Foundation and the Foresight Institute. This work was also partially supported by ONR MURI N00014-24-1-2742. Carter Blair is supported by an NSERC PGS D and a Cooperative AI PhD Fellowship.

AI Acknowledgment

The AI coding assistant Claude Code helped us write and debug code and run experiments, ChatGPT helped with the proof of Theorem 1, and AI assistants helped with writing and editing. We take full responsibility for all content in this paper. LLMs that are part of the method are described in Appendix G.

References

Alessandro Achille and Stefano Soatto. Emergence of invariance and disentanglement in deep representations. *Journal of Machine Learning Research*, 19(50):1–34, 2018.

Elliot Anshelevich, Aris Filos-Ratsikas, Nisarg Shah, and Alexandros A. Voudouris. Distortion in social choice problems: The first 15 years and beyond. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4294–4301, 2021. doi: 10.24963/ijcai.2021/589. Survey Track.

Jake Barrett, Kobi Gal, Paul Gölz, Rose M. Hong, and Ariel D. Procaccia. Now we're talking: Better deliberation groups through submodular optimization. In *Proceedings of the 37th AAAI Conference on Artificial Intelligence (AAAI)*, pages 5490–5498, 2023. doi: 10.1609/aaai.v37i5.25682.

Jake Barrett, Philipp C. Verpoort, and Kobi Gal. A new heuristic algorithm for balanced deliberation groups, 2024. URL https://arxiv.org/abs/2410.21451.

Carter Blair and Kate Larson. Generating fair consensus statements with social choice on token-level MDPs, 2025. URL https://arxiv.org/abs/2510.14106.

Carter Blair, Kate Larson, and Edith Law. Reflective verbal reward design for pluralistic alignment. In *Proceedings of the 34th International Joint Conference on Artificial Intelligence (IJCAI)*, 2025. doi: 10.24963/ijcai.2025/1141.

Carter Blair, Ben Armstrong, Shiri Alouf-Heffetz, Nimrod Talmon, and Davide Grossi. Probably approximately consensus: On the learning theory of finding common ground, 2026. URL https://arxiv.org/abs/2604.21811.

Niclas Boehmer, Sara Fish, and Ariel D. Procaccia. Generative social choice: The next generation. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pages 4627–4659, 2025. URL https://proceedings.mlr.press/v267/boehmer25a.html.

Laurent Bulteau, Gal Shahaf, Ehud Shapiro, and Nimrod Talmon. Aggregation over metric spaces: Proposing and voting in elections, budgeting, and legislation. *Journal of Artificial Intelligence Research*, 70:1413–1439, 2021. doi: 10.1613/jair.1.12388.

Xingyu Chen, Brandon Fain, Liang Lyu, and Kamesh Munagala. Proportionally fair clustering. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pages 1032–1041, 2019. URL https://proceedings.mlr.press/v97/chen19d.html.

Jay Chooi, Paul Gölz, Ariel D. Procaccia, Benjamin Schiffer, and Shirley Zhang. Finding common ground in a sea of alternatives, 2026. URL https://arxiv.org/abs/2603.16751.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, volume 30, pages 4299–4307, 2017.

Joshua Clinton, Simon Jackman, and Douglas Rivers. The statistical analysis of roll call data. *American Political Science Review*, 98(2):355–370, 2004. doi: 10.1017/S0003055404001194.

Soham De, Lodewijk Gelauff, Ashish Goel, Smitha Milli, Ariel D. Procaccia, and Alice Siu. Question the questions: Auditing representation in online deliberative processes. In *Proceedings of the ACM Web Conference 2026*, pages 1640–1650, 2026. doi: 10.1145/3774904.3792474.

Kevin M. Esterling, Archon Fung, and Taeku Lee. How much disagreement is good for democratic deliberation? *Political Communication*, 32(4):529–551, 2015. doi: 10.1080/10584609.2014.969466.

Michal Feldman, Amos Fiat, and Iddan Golomb. On voting and facility location. In *Proceedings of the 2016 ACM Conference on Economics and Computation (EC '16)*, pages 269–286, 2016. doi: 10.1145/2940716.2940725.

Sara Fish, Paul Gölz, David C. Parkes, Ariel D. Procaccia, Gili Rusak, Itai Shapira, and Manuel Wüthrich. Generative social choice. *Journal of the ACM*, 73(2), 2026. doi: 10.1145/3799709.

James Fishkin, Nikhil Garg, Lodewijk Gelauff, Ashish Goel, Kamesh Munagala, Sukolsak Sakshuwong, Alice Siu, and Sravya Yandamuri. Deliberative democracy with the online deliberation platform. In *The 7th AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*, pages 1–2, 2019. URL https://www.humancomputation.com/2019/assets/papers/144.pdf. Demo.

James S. Fishkin and Robert C. Luskin. Experimenting with a democratic ideal: Deliberative polling and public opinion. *Acta Politica*, 40(3):284–298, 2005. doi: 10.1057/palgrave.ap.5500121.

Frankly. Frankly: Enabling constructive dialogue. https://frankly.org/, 2026. Accessed: 2026-05-01.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6894–6910, 2021. doi: 10.18653/v1/2021.emnlp-main.552.

Sean M. Gerrish and David M. Blei. Predicting legislative roll calls from text. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pages 489–496, 2011.

Vahid Ghafouri, Jose Such, and Guillermo Suarez-Tangil. I love pineapple on pizza != I hate pineapple on pizza: Stance-aware sentence transformers for opinion mining. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 21046–21058, 2024. doi: 10.18653/v1/2024.emnlp-main.1171.

Umberto Grandi, Robert Bredereck, Théo Delemazure, Ulle Endriss, Jan Maly, Nick Mattei, Nicolas Maudet, Oliviero Nardi, and Stanisław Szufa. Social choice with text: Collective decision making in the LLM era, 2026. URL https://hal.science/hal-05548117. Preprint available at HAL Open Science Archive (hal-05548117), March 2026.

Kunal Handa, Yarin Gal, Ellie Pavlick, Noah Goodman, Jacob Andreas, Alex Tamkin, and Belinda Z. Li. Bayesian preference elicitation with language models, 2024. URL https://arxiv.org/abs/2403.05534.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Joshua Introne. Measuring belief dynamics on Twitter. *Proceedings of the International AAAI Conference on Web and Social Media*, 17(1):387–398, 2023. doi: 10.1609/icwsm.v17i1.22154.

Christopher F. Karpowitz and Tali Mendelberg. Groups and deliberation. *Swiss Political Science Review*, 13(4):645–662, 2007. doi: 10.1002/j.1662-6370.2007.tb00092.x.

Leon Kellerhals and Jannik Peters. Proportional fairness in clustering: A social choice perspective. In *Advances in Neural Information Processing Systems*, volume 37, pages 28471–28494, 2024.

Michael Laver, Kenneth Benoit, and John Garry. Extracting policy positions from political texts using words as data. *American Political Science Review*, 97(2):311–331, 2003. doi: 10.1017/S0003055403000698.

Sean Lee, Aamir Shakir, Darius Koenig, and Julius Lipp. Open source strikes bread - new fluffy embeddings model. https://www.mixedbread.com/blog/mxbai-embed-large-v1, 2024. Accessed: 2026-05-01.

Belinda Z. Li, Alex Tamkin, Noah Goodman, and Jacob Andreas. Eliciting human preferences with language models. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=LvDwwAgMEW.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning, 2023. URL https://arxiv.org/abs/2308.03281.

Jane Mansbridge. Should Blacks represent Blacks and women represent women? A contingent "yes". *The Journal of Politics*, 61(3):628–657, 1999. doi: 10.2307/2647821.

Luke Merrick, Danmei Xu, Gaurav Nuti, and Daniel Campos. Arctic-embed: Scalable, efficient, and accurate text embedding models, 2024. URL https://arxiv.org/abs/2405.05374.

Evi Micha and Nisarg Shah. Proportionally fair clustering revisited. In *47th International Colloquium on Automata, Languages, and Programming (ICALP 2020)*, volume 168 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 85:1–85:16, 2020. doi: 10.4230/LIPIcs.ICALP.2020.85.

Niklas Muennighoff, Nouamane Tazi, Loic 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, 2023. doi: 10.18653/v1/2023.eacl-main.148.

Diana C. Mutz. *Hearing the Other Side: Deliberative versus Participatory Democracy*. Cambridge University Press, 2006. ISBN 9780521612289.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith B. Hall, Daniel Cer, and Yinfei Yang. Sentence-T5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874, 2022. doi: 10.18653/v1/2022.findings-acl.146.

OpenAI. New embedding models and API updates. https://openai.com/index/new-embedding-models-and-api-updates/, 2024. Accessed: 2026-05-01.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744, 2022.

Keith T. Poole and Howard Rosenthal. A spatial model for legislative roll call analysis. *American Journal of Political Science*, 29(2):357–384, 1985. doi: 10.2307/2111172.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741, 2023.

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 (EMNLP-IJCNLP)*, pages 3982–3992, 2019. doi: 10.18653/v1/D19-1410.

Manon Revel and Théophile Pénigaud. AI-enhanced deliberative democracy and the future of the collective will. *Philosophy & Technology*, 39(81), 2026. doi: 10.1007/s13347-026-01044-1.

Ludovic Rheault and Christopher Cochrane. Word embeddings for the analysis of ideological placement in parliamentary corpora. *Political Analysis*, 28(1):112–133, 2020. doi: 10.1017/pan.2019.26.

Jonathan B. Slapin and Sven-Oliver Proksch. A scaling model for estimating time-series party positions from texts. *American Journal of Political Science*, 52(3):705–722, 2008. doi: 10.1111/j.1540-5907.2008.00338.x.

Christopher T. Small, Michael Bjorkegren, Timo Erkkilä, Lynette Shaw, and Colin Megill. Polis: Scaling deliberation by mapping high dimensional opinion spaces. *RECERCA. Revista de Pensament i Anàlisi*, 26(2):1–26, 2021. doi: 10.6035/recerca.5516.

Christopher T. Small, Ivan Vendrov, Esin Durmus, Hadjar Homaei, Elizabeth Barry, Julien Cornebise, Ted Suzman, Deep Ganguli, and Colin Megill. Opportunities and risks of LLMs for scalable deliberation with Polis, 2023. URL https://arxiv.org/abs/2306.11932.

Cass R. Sunstein. The law of group polarization. *Journal of Political Philosophy*, 10(2):175–195, 2002. doi: 10.1111/1467-9760.00148.

Michael Henry Tessler, Michiel A. Bakker, Daniel Jarrett, Hannah Sheahan, Martin J. Chadwick, Raphael Koster, Georgina Evans, Lucy Campbell-Gillingham, Tantum Collins, David C. Parkes, Matthew Botvinick, and Christopher Summerfield. AI can help humans find common ground in democratic deliberation. *Science*, 386(6719):eadq2852, 2024. doi: 10.1126/science.adq2852.

Keyon Vafa, Suresh Naidu, and David M. Blei. Text-based ideal points. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5345–5357, 2020. doi: 10.18653/v1/2020.acl-main.475.

Teemu Vahtola, Mathias Creutz, and Jörg Tiedemann. It is not easy to detect paraphrases: Analysing semantic similarity with antonyms and negation using the new SemAntoNeg benchmark. In *Proceedings of the Fifth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 249–262, 2022. doi: 10.18653/v1/2022.blackboxnlp-1.20.

Voyage AI. voyage-3 & voyage-3-lite: A new generation of small yet mighty general-purpose embedding models. https://blog.voyageai.com/2024/09/18/voyage-3/, 2024. Accessed: 2026-05-01.

Voyage AI. voyage-3-large: The new state-of-the-art general-purpose embedding model. https://blog.voyageai.com/2025/01/07/voyage-3-large/, 2025. Accessed: 2026-05-01.

Voyage AI. The Voyage 4 model family: Shared embedding space with MoE architecture. https://blog.voyageai.com/2026/01/15/voyage-4/, 2026. Accessed: 2026-05-01.

Henning Wachsmuth, Shahbaz Syed, and Benno Stein. Retrieval of the best counterargument without prior topic knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 241–251, 2018. doi: 10.18653/v1/P18-1023.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training, 2022. URL https://arxiv.org/abs/2212.03533.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. C-Pack: Packed resources for general Chinese embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 641–649, 2024. doi: 10.1145/3626772.3657878.

Haike Xu, Zongyu Lin, Yizhou Sun, Kai-Wei Chang, and Piotr Indyk. SparseCL: Sparse contrastive learning for contradiction retrieval, 2024. URL https://arxiv.org/abs/2406.10746.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report, 2024. URL https://arxiv.org/abs/2407.10671.

Joshua C. Yang and Fynn Bachmann. Bridging voting and deliberation with algorithms: Field insights from vTaiwan and Kultur Komitee. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pages 3207–3223, 2025.

Dun Zhang, Jiacheng Li, Ziyang Zeng, and Fulong Wang. Jasper and Stella: Distillation of SOTA embedding models, 2024. URL https://arxiv.org/abs/2412.19048.

Yifan Zhang, Ge Zhang, Yue Wu, Kangping Xu, and Quanquan Gu. Beyond Bradley-Terry models: A general preference model for language model alignment. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, 2025. URL https://arxiv.org/abs/2410.02197.

A Extended Related Work

Work on collective decision-making over free-form text increasingly relies on some representation of participants or statements. Polis derives a low-dimensional opinion map from the participant-by-comment vote matrix via PCA and clustering [Small et al., 2021, 2023], generative social choice and PROSE group statements in an LLM-defined feature space to produce representative slates [Fish et al., 2026, Boehmer et al., 2025], Blair et al. [2026] model approximate consensus as a region of sentence-embedding space, and De et al. [2026] audit justified representation of question slates using cosine similarity of question embeddings as participant utility. Complementary work aggregates text without an explicit embedding geometry, for example through reward models, token-level policies, or sampling-based social-choice procedures [Tessler et al., 2024, Blair and Larson, 2025, Chooi et al., 2026, Grandi et al., 2026].

A separate line of work studies the mismatch between semantic overlap and stance. Vahtola et al. [Vahtola et al., 2022] show that generic sentence embeddings struggle when negation or antonymy flips meaning; Introne [Introne, 2023] and Ghafouri et al. [Ghafouri et al., 2024] fine-tune encoders to separate opposing viewpoints for stance detection or opinion retrieval; and counter-argument retrieval methods add explicit dissimilarity terms or sparsity-aware scoring to retrieve contradictions [Wachsmuth et al., 2018, Xu et al., 2024]. We have a similar starting point, but make a unique diagnosis and propose a formal model, then use these to inspire a novel method.

A long tradition in political science estimates low-dimensional ideal points from political data. The canonical approach models legislators' roll-call votes as a function of their positions in a latent spatial model [Poole and Rosenthal, 1985, Clinton et al., 2004]. A parallel line recovers positions directly from text: Wordscores scales manifestos against reference documents [Laver et al., 2003], Wordfish estimates time-series party positions from speech [Slapin and Proksch, 2008], and later work joins text with votes [Gerrish and Blei, 2011], combines topic models with ideal points [Vafa et al., 2020], or augments word embeddings with speaker metadata [Rheault and Cochrane, 2020]. Our §7 finding — that preferences on a frozen sentence embedding are linearly accessible through a rank-$\sim$20 projection — is consistent with this tradition, recovered here as a property of a general-purpose pretrained encoder on deliberation data from non-legislators.

Our work also connects to preference learning more broadly. Bradley-Terry objectives are standard in reward modeling and preference-based fine-tuning [Christiano et al., 2017, Ouyang et al., 2022, Rafailov et al., 2023], where they reshape a generative policy's output distribution. Zhang et al. [2025] also embed responses but score preferences with a skew-symmetric operator to express within-user cyclic preferences, a concern that does not arise in our cross-user setting. We use the Bradley-Terry objective to reshape a reusable embedding geometry that can be consumed by metric social choice and other downstream geometric procedures [Anshelevich et al., 2021, Bulteau et al., 2021], rather than only a per-query reward model.

B Dataset Details

The GSC datasets come from two studies by Fish et al. [2026]. In each, participants first write free-text opinions describing their views on a topic, then rate a set of AI-generated policy statements on a Likert scale. We construct triplets from all pairs of statements with distinct ratings for each participant. The abortion generation survey uses participants who authored the original opinions; the abortion validation survey uses a separate cohort who rate the same statements and give verbal feedback to the statements. The chatbot personalization survey follows the same structure. Data for the abortion surveys is available at `https://github.com/generative-social-choice/gsc_abortion` and for chatbot personalization at `https://github.com/generative-social-choice/chatbot_personalization`.

The Remesh datasets come from the Polarized Issues corpus, which collects binary agree/disagree votes on open-ended responses to political questions using the Remesh platform. Participants first write their own response to a prompt, then vote on responses written by other participants. We use three topics: campus protests, foreign intervention, and right to assemble. The data is available at `https://github.com/akonya/polarized-issues-data`.

The Polis datasets come from publicly available Polis conversations from Small et al. [2021]. In Polis, participants write short comments and vote agree, disagree, or pass on comments written

Table 6: Evaluation datasets. Participants authored anchors (own text) and voted on statements (others' text); pairwise preference triplets derive from their vote orderings. GSC uses a small fixed statement pool; Remesh and Polis use open pools seeded by participants themselves. We use all 11 datasets for evaluation.

Source	Dataset	Abbreviation	Participants	Anchors	Statements	Triplets
GSC	Abortion (generation users)	AbG	100	500	5	846
	Abortion (validation users)	AbV	100	993	10	3,513
	Chatbot Personalization	Chat	100	788	6	1,104
Remesh	Campus Protests	Camp	298	1,176	1,011	21,989
	Foreign Intervention	Frgn	289	857	764	12,428
	Right to Assemble	RTA	295	1,445	1,274	28,340
Polis	Seattle $15/hour	Seatl	13	13	29	1,103
	Bowling Green AA	BG	222	531	607	1,363,191
	Brexit Consensus	Brxt	11	50	50	2,468
	Canadian Reform	Can	19	150	151	19,366
	NZ UBI	UBI	15	15	51	4,643
Total			1,462	6,458	3,958	1,458,991

by others. We restrict to participants who both authored at least one comment and voted on at least five others, using their authored comments as anchors. These datasets have the shortest texts and sparsest preference signal of any platform in our evaluation. The data is available at https://github.com/compdemocracy/openData.

Licenses. The GSC datasets are released under AGPL-3.0, the Polarized Issues (Remesh) data under CC-BY-4.0, and the Polis openData under CC-BY-4.0. The Habermas Machine dataset used for training-data generation is released under CC-BY-4.0 (data) and Apache-2.0 (code), and the Kialo dataset under MIT. The embedding models used are sentence-T5-XL (Apache-2.0), e5-large-v2 (MIT), BGE-large-en-v1.5 (MIT), and all-mpnet-base-v2 (Apache-2.0). All licenses permit research use.

C Proof of Theorem 1

Fix a symmetric in-subspace operator B with $B = P_S B P_S$, and write

$$\ell(u) = \log(1 + e^{-u}).$$

For a triplet (a, p, n), recall that

$$\Delta_B = \psi_S(a)^\top B\big(\psi_S(p) - \psi_S(n)\big), \qquad \Delta_T = \big\langle \psi_\perp(a), \psi_\perp(p) - \psi_\perp(n) \big\rangle.$$

Thus, the risk can be written as

$$R(B, \lambda) = \mathbb{E}\big[\ell(\Delta_B + \lambda \Delta_T)\big].$$

Our goal is to show that $R(B, \lambda) < R(B, 1)$ for every $\lambda \in [0, 1)$.

To do so, we will first note bounds which will be required below. Since the encoder outputs unit-norm embeddings and orthogonal projection cannot increase the norm (by the Pythagorean theorem),

$$\|\psi_\perp(a)\| \le 1, \qquad \|\psi_\perp(p) - \psi_\perp(n)\| \le \|\psi_\perp(p)\| + \|\psi_\perp(n)\| \le 2.$$

Therefore,

$$|\Delta_T| = \big|\langle \psi_\perp(a), \psi_\perp(p) - \psi_\perp(n) \rangle\big| \le 2$$

everywhere. Also, since B is fixed and the embeddings have norm at most one,

$$|\Delta_B| \le 2\|B\|_{\mathrm{op}}.$$

Since $|\Delta_B + \lambda \Delta_T| \le 2\|B\|_{\mathrm{op}} + 2|\lambda|$ for each fixed λ, and ℓ is continuous, $\ell(\Delta_B + \lambda \Delta_T)$ is bounded and therefore integrable.

We now justify differentiating $R(B, \lambda)$ with respect to λ. The derivative of the logistic loss is

$$\ell'(u) = -\frac{1}{1+e^u},$$

so $|\ell'(u)| \le 1$ for every $u \in \mathbb{R}$. Fix $\lambda \in \mathbb{R}$ and, for nonzero h, define the difference quotient random variable

$$Q_h = \frac{\ell(\Delta_B + (\lambda+h)\Delta_T) - \ell(\Delta_B + \lambda\Delta_T)}{h}.$$

For each realization of the triplet, Δ_B and Δ_T are fixed finite numbers, and the map

$$\eta \mapsto \ell(\Delta_B + \eta\Delta_T)$$

is differentiable. Hence, as $h \to 0$,

$$Q_h \to \ell'(\Delta_B + \lambda\Delta_T)\Delta_T$$

pointwise. Moreover, by the mean value theorem, for each $h \neq 0$ there exists some ξ_h between λ and $\lambda + h$ such that

$$Q_h = \ell'(\Delta_B + \xi_h\Delta_T)\Delta_T.$$

Therefore

$$|Q_h| \le |\ell'(\Delta_B + \xi_h\Delta_T)|\,|\Delta_T| \le 2,$$

using $|\ell'| \le 1$ and $|\Delta_T| \le 2$. The constant function 2 is integrable, so dominated convergence gives

$$R'(B,\lambda) = \lim_{h\to 0} \frac{R(B,\lambda+h) - R(B,\lambda)}{h} = \lim_{h\to 0} \mathbb{E}[Q_h] = \mathbb{E}\big[\ell'(\Delta_B + \lambda\Delta_T)\Delta_T\big].$$

Thus, for every $\lambda \in \mathbb{R}$,

$$R'(B,\lambda) = \mathbb{E}\big[\ell'(\Delta_B + \lambda\Delta_T)\Delta_T\big].$$

We now evaluate this derivative at $\lambda = 0$. Let

$$\mathcal{G} = \sigma\big(\psi_S(a), \psi_S(p), \psi_S(n)\big)$$

be the σ-algebra generated by the preference-subspace parts of the triplet. Since $B = P_S B P_S$, the quantity Δ_B is a function only of the projected variables in $\mathcal{G}$. Hence $\ell'(\Delta_B)$ is $\mathcal{G}$-measurable. By the tower property,

$$R'(B,0) = \mathbb{E}\big[\ell'(\Delta_B)\Delta_T\big] = \mathbb{E}\Big[\mathbb{E}\big[\ell'(\Delta_B)\Delta_T \mid \mathcal{G}\big]\Big].$$

Since $\ell'(\Delta_B)$ is $\mathcal{G}$-measurable, the pull-out property gives

$$\mathbb{E}\big[\ell'(\Delta_B)\Delta_T \mid \mathcal{G}\big] = \ell'(\Delta_B)\,\mathbb{E}[\Delta_T \mid \mathcal{G}].$$

Therefore

$$R'(B,0) = \mathbb{E}\Big[\ell'(\Delta_B)\,\mathbb{E}[\Delta_T \mid \mathcal{G}]\Big].$$

The hard-triplet condition states that

$$\mathbb{E}[\Delta_T \mid \mathcal{G}] \le 0 \quad \text{a.s.},$$

with strict inequality on a set of positive probability. Also,

$$\ell'(u) = -\frac{1}{1+e^u} < 0 \qquad \text{for every } u \in \mathbb{R},$$

and hence $\ell'(\Delta_B) < 0$ everywhere. Therefore the product

$$\ell'(\Delta_B)\,\mathbb{E}[\Delta_T \mid \mathcal{G}]$$

is nonnegative almost surely and strictly positive on a set of positive probability. Hence

$$R'(B,0) > 0.$$

It remains to show that this positivity at zero implies the desired comparison with $\lambda = 1$. The loss ℓ is convex because

$$\ell''(u) = \frac{e^u}{(1+e^u)^2} > 0 \qquad \text{for every } u \in \mathbb{R}.$$

For each realization of the triplet, the map

$$\lambda \mapsto \Delta_B + \lambda \Delta_T$$

is affine, so

$$\lambda \mapsto \ell(\Delta_B + \lambda \Delta_T)$$

is convex. Taking expectations preserves convexity, and hence $R(B, \cdot)$ is convex. Since $R(B, \cdot)$ is differentiable, its derivative is nondecreasing. Therefore, for every $\lambda \in [0, 1]$,

$$R'(B, \lambda) \geq R'(B, 0) > 0.$$

Now fix any $\bar{\lambda} \in [0, 1)$. By the mean value theorem, there exists some $c \in (\bar{\lambda}, 1)$ such that

$$R(B, 1) - R(B, \bar{\lambda}) = R'(B, c)(1 - \bar{\lambda}).$$

Since $c \in [0, 1]$, we have $R'(B, c) > 0$, and since $1 - \bar{\lambda} > 0$,

$$R(B, 1) - R(B, \bar{\lambda}) > 0.$$

Equivalently,

$$R(B, \bar{\lambda}) < R(B, 1).$$

Because $\bar{\lambda} \in [0, 1)$ was arbitrary,

$$R(B, \lambda) < R(B, 1) \qquad \text{for every } \lambda \in [0, 1).$$

□

D Additional Experimental Details

Hyperparameter selection. The sentence-T5-XL recipe reported in the main text was selected by a sweep over learning rate $\in$ {1.25e-5, 5e-5, 1.25e-4, 5e-4}, hard-triplet count $\in$ $\{250, 500, 750, 1000, 2000\}$, LoRA rank $\in$ $\{4, 8, 16, 32, 64\}$, and LoRA $\alpha = 3r$ on a held-out validation split. The selected configuration is lr = 1.25e-4, $n_{\text{hard}} = 750$, $r = 16$, $\alpha = 48$, which corresponds to 46 gradient steps at batch size 16 on a random 750 of the 10,000-triplet synthetic pool, sampled uniformly without replacement and resampled independently for each training seed. For cross-model transfer (e5, BGE, all-mpnet) we fix rank and alpha and re-select $(\text{lr}, n_{\text{hard}})$ per encoder from the same grid; selected configurations are $(5\text{e-}5, 1000)$ for e5-large, $(1.25\text{e-}4, 500)$ for BGE-large, and $(1.25\text{e-}4, 1000)$ for all-mpnet. We AdamW with a linear-decay schedule and a 10% warmup ratio for all models.

Compute. All experiments were run on a single NVIDIA A100 GPU. Training a single DPT seed takes less than 20 minutes.

E Additional Experiments

E.1 Error Decomposition

To understand the residual errors of the tuned model, we classify a stratified sample of 913 errors using GPT-4o. Each error is a triplet where the tuned model incorrectly ranks the dispreferred statement higher. We present the full triplet (anchor texts, preferred, dispreferred) and ask the classifier to select from six categories, presented in randomized order to avoid position bias.

Surface similarity remains the dominant failure mode at 61%, which indicates room for further improvement through refined hard-triplet training. Insufficient anchor signal accounts for 20.5% and is concentrated in Polis datasets where comments are short: 73% of Bowling Green errors fall in this category, compared to 6% for GSC. Subtle value distinctions at 16.9% represent cases where both options broadly match the anchor's views but differ on fine-grained nuance below the resolution of sentence-level similarity. The high number of "insufficient anchor signal" errors is notable since this is technically something no embedding model can solve. This must be addressed upstream by encouraging participants to write more or perhaps by giving them scaffolding to aid in their articulation of their preferences [Blair et al., 2025, Li et al., 2025, Handa et al., 2024].

Table 7: LLM-classified error categories for tuned model errors (913 triplets, stratified across datasets, capped at 5 per participant).

Category	**Count**	**%**
Surface similarity	557	61.0
Insufficient anchor signal	187	20.5
Subtle value distinction	154	16.9
Both options plausible	10	1.1
Style/register mismatch	3	0.3
None of the above	2	0.2

E.2 LoRA Rank Ablation.

We vary LoRA rank $r \in \{4, 8, 16, 32, 64\}$ with $\alpha = 3r$, holding all other hyperparameters at the selected ST5-XL configuration. Table 8 reports held-out test accuracy averaged across the 11 main evaluation datasets. Accuracy peaks at r=16: 67.6% at r=4, 68.7% at r=16, 67.1% at r=32, and 63.8% at r=64. With only 46 optimizer steps on 750 triplets, higher ranks introduce more free parameters than the data can constrain, and the extra capacity starts overwriting the pretrained geometry that the tuning is meant to realign, not replace.

Table 8: LoRA rank ablation on sentence-T5-XL (single seed, mean test accuracy across 11 datasets).

LoRA rank r	4	8	16	32	64
Mean test acc. (%)	67.6	68.3	**68.7**	67.1	63.8

E.3 Loss Ablation: Bradley-Terry vs InfoNCE.

We replace the pairwise Bradley-Terry loss with InfoNCE (temperature 0.05). For each anchor in a batch of 16, the positive is its own preference match and the negatives are its own semantic distractor plus the other 15 anchors' preference matches as in-batch negatives, giving 16 negatives per anchor. All other hyperparameters are held at the selected ST5-XL configuration. The two losses yield essentially identical mean test accuracy (68.7% for BT vs 68.6% for InfoNCE), so the gain comes from the counterfactual triplet construction itself, not the choice of pairwise objective. We report Bradley-Terry in the main results because its score is exactly the signed cosine difference used at inference, and because it operates on each triplet independently — no batch size or batch composition to tune.

E.4 Normal-Correlation Triplet Ablation.

The hard-triplet construction in §4.3 engineers a worst-case correlation between stance and wording so that the nuisance margin satisfies the hard-triplet condition $\mathbb{E}[\Delta_T \mid \mathcal{G}] \leq 0$. The purpose of this is to force the model to rely less on the nuisance subspace and more on the preference subspace. We test whether this engineering is necessary by generating 2,000 *normal-correlation* training triplets in which stance and wording move together. This mimics more traditional NLI pair training where the correlation between the desired signal and a nuisance signal is not explicitly broken [Gao et al., 2021]. The rewrite prompt mirrors the hard prompt but flips the role assignment, so the preference match shares vocabulary with the anchor while the semantic distractor uses different vocabulary. The full prompt is in Appendix G. By construction $\mathbb{E}[\Delta_T \mid \mathcal{G}] > 0$, the opposite of the hard-triplet condition used in Theorem 1. We sweep the same hyperparameter grid used for the hard-triplet selection in Appendix D and select the configuration with the highest validation macro-mean accuracy on the 11 natural datasets, which gives lr $=$ 1.25e-5, $n = 2000$, $r = 64$. Re-running this configuration over five seeds yields 67.0 $\pm$ 0.1% mean test accuracy and 59.3 $\pm$ 0.2% accuracy on the hard evaluation triplets. The matched hard-trained DPT configuration in Table 3 reaches 68.6 $\pm$ 0.2% test and 80.0 $\pm$ 0.4% on the hard evaluation triplets. Normal-correlation training therefore recovers a small lift over base cosine, but it is significantly worse than DPT.

E.5 Hard-Positive vs. Hard-Negative Ablation.

Each hard triplet has two hard sides: the preference match keeps the anchor's stance in new wording (a hard positive), and the semantic distractor keeps the anchor's wording but reverses its stance (a hard negative). To measure what each side contributes, we fix a random sample of 2,000 anchors from the training pool. For each anchor we make one additional GPT-4o call with a mirrored prompt (Appendix G), which produces an *easy positive* (same stance, nearly the same wording) and an *easy negative* (opposite stance, different wording). We then recombine the four rewrites so that one side changes at a time: *hard negative only* (anchor, easy positive, hard negative), where both candidates are lexically close to the anchor; *hard positive only* (anchor, hard positive, easy negative), where both candidates are lexically distant; *both sides hard*, our standard construction; and *easy–easy* (anchor, easy positive, easy negative). Mean word-level Jaccard overlap with the anchor confirms the manipulation: it is 0.12 for hard positives versus 0.91 for easy positives, and 0.70 for hard negatives versus 0.12 for easy negatives. Every condition uses the selected sentence-T5-XL configuration (750 triplets sampled per seed from the 2,000, 46 steps) and is evaluated exactly as in the main text. We use five seeds.

Table 9: Hard-positive vs. hard-negative ablation on sentence-T5-XL (same 2,000 anchors in every row, five seeds). Natural is the mean test accuracy over the 11 datasets; Hard is accuracy on the held-out hard evaluation triplets.

Training triplets	**Positive**	**Negative**	**Nat. (%)**	**Hard (%)**
Both sides hard *(ours)*	hard	hard	**68.5** $\pm$ 0.1	**79.9** $\pm$ 0.6
Hard negative only	easy	hard	66.8 $\pm$ 0.2	78.9 $\pm$ 0.4
Hard positive only	hard	easy	66.0 $\pm$ 0.2	74.1 $\pm$ 0.5
Easy–easy	easy	easy	64.3 $\pm$ 0.1	71.1 $\pm$ 0.8
Untuned sentence-T5-XL	–	–	65.2	48.3

Table 9 shows that both sides hard is best on both metrics. The hard negative carries most of the hard-triplet gain: hard-negative-only training reaches 78.9% hard accuracy, against 79.9% for both sides and 74.1% for hard-positive-only. On natural data, however, neither side alone matches the full construction, which is 1.7 points above hard-negative-only and 2.5 points above hard-positive-only. Each hard side also improves on easy–easy training on both metrics, so the hard positive contributes even though the hard negative contributes more. Easy–easy triplets resemble the normal-correlation triplets of Appendix E.4, but here they are trained at the DPT configuration rather than at a configuration selected for them, so the two sets of numbers differ.

E.6 Sensitivity to the Rewrite Prompt and Generator.

The hard-triplet rewrite asks the generator to balance two goals at once (preserve or reverse the stance, while controlling word overlap), so we test whether DPT depends on the exact wording of the prompt in Appendix G or on GPT-4o. Using the same 2,000 anchors as Appendix E.5, we regenerate both rewrites under four configurations: GPT-4o with two alternative prompts, and the original prompt with GPT-4o-mini and with Claude Sonnet 4.6. Prompt B restates the task in prose, presents the preference match first, uses a different worked example, and frames the two rewrites as statements the person would agree with or reject. Prompt C is a terse benchmark-builder instruction with no worked example and no language about values. Both prompts are given in Appendix G. The control is the original GPT-4o rewrites of the same 2,000 anchors, which is also the both-sides-hard row of Table 9. Training and evaluation are as in Appendix E.5.

Across all five configurations, hard-triplet accuracy stays between 79.2% and 80.6%, within about one seed standard deviation of the control (79.9 $\pm$ 0.6%). Natural accuracy is at most 0.9 points below the control (prompt C), and every configuration remains well above the untuned encoder. The two configurations whose distractors share the least wording with the anchor (prompt C and GPT-4o-mini) have the lowest natural accuracy, consistent with the construction working through the wording contrast. Both the much smaller GPT-4o-mini and a model from a different family reproduce the control's hard-triplet accuracy, so DPT depends neither on the exact prompt nor on a single generator's notion of an opposite stance.

Table 10: Sensitivity of DPT to the rewrite prompt and generator (sentence-T5-XL, same 2,000 anchors in every row, five seeds). Overlap is the mean word-level Jaccard overlap of the preference match and of the semantic distractor with the anchor.

Generator	**Prompt**	**Overlap (match / distr.)**	**Nat. (%)**	**Hard (%)**
GPT-4o	original (control)	0.12 / 0.70	68.5 ± 0.1	79.9 ± 0.6
GPT-4o	B	0.15 / 0.70	68.3 ± 0.1	80.6 ± 0.2
GPT-4o	C	0.19 / 0.54	67.6 ± 0.1	79.2 ± 0.4
GPT-4o-mini	original	0.16 / 0.55	67.9 ± 0.2	79.9 ± 0.2
Claude Sonnet 4.6	original	0.07 / 0.71	68.1 ± 0.2	80.3 ± 0.4
Untuned sentence-T5-XL	–	–	65.2	48.3

E.7 Robustness to Label Noise.

LLM-generated triplets will sometimes be wrong, for example when the generator fails to reverse the stance of the semantic distractor. We test robustness to such errors directly. Starting from the selected sentence-T5-XL configuration, we swap the preference match and the semantic distractor in a random $\rho \in \{10, 30, 50, 70, 90\}\%$ of the 750 training triplets, chosen independently for each seed, and leave everything else unchanged. We use five seeds per level.

Table 11: DPT under label noise (sentence-T5-XL, five seeds). A fraction ρ of the training triplets has its preference match and semantic distractor swapped. The $\rho = 0$ row is the DPT model of Table 3.

Label noise ρ	**Nat. (%)**	**Hard (%)**
0%	68.6 ± 0.2	80.0 ± 0.4
10%	68.5 ± 0.3	77.1 ± 0.8
30%	67.8 ± 0.4	64.1 ± 1.9
50%	65.6 ± 0.4	48.6 ± 1.1
70%	63.3 ± 0.4	34.6 ± 2.3
90%	61.2 ± 0.3	17.3 ± 1.4
Untuned sentence-T5-XL	65.2	48.3

Performance degrades smoothly and monotonically as noise increases (Table 11). At 10% noise, natural accuracy is unchanged (68.5% vs. 68.6%) and hard-triplet accuracy falls by 2.9 points. At exactly 50%, where the labels carry no information about which candidate is preferred, DPT returns to the untuned encoder (65.6% vs. 65.2% natural, 48.6% vs. 48.3% hard). It falls below the untuned encoder only once most labels are inverted. This is consistent with Theorem 1, which requires the hard-triplet condition only in conditional expectation rather than on every triplet, so a minority of flawed triplets weakens the effect without reversing it.

E.8 Participant-Level Significance.

Table 3 reports dataset-mean accuracies across five training seeds, which is the right unit for comparing tuning recipes but does not directly speak to whether the gain is reliable on the unit of deployment, the individual participant. We therefore run a paired test on per-participant accuracies. For each of the 11 evaluation datasets we replay the test-split scoring loop under both base sentence-T5-XL and the DPT-tuned encoder, keeping the per-triplet outcome (1 if the encoder ranks the preference match above the distractor, 0 otherwise) and the per-participant accuracy (e.g., the accuracy for participant i using the base model is a_i^{base}). Each participant then has a pair $(a_i^{\text{base}}, a_i^{\text{tuned}})$, and each triplet has a pair of binary outcomes. We report a paired Wilcoxon signed-rank test on the participant accuracies and a McNemar exact test on the triplet outcomes.

Pooled across all 442 participants in the 11 test cohorts, the mean accuracy lift is +2.0 pp (median +0.9 pp). Tuned beats base on 51.6% of participants, ties on 17.2%, and loses on 31.2%. The paired Wilcoxon test yields $p = 3.4 \times 10^{-9}$ and the paired t-test yields $p = 1.8 \times 10^{-8}$. McNemar over the 24,834 discordant triplets returns $b = 14{,}891$ wins for tuned and $c = 9{,}943$ wins for base, with $p = 9.5 \times 10^{-218}$. Per-dataset results are in Table 12. The participant-level Wilcoxon reaches $p < 0.05$ on 5 of 11 datasets, while the triplet-level McNemar reaches $p < 0.05$ on 10 of 11. The

lone non-mover under both is GSC chatbot, where DPT shows the smallest accuracy lift across the entire benchmark ($\Delta = +0.4$ pp). The Wilcoxon non-significance on the four small Polis cohorts (Seattle, Brexit, Canadian, UBI) reflects power rather than effect, since each shows tuned-mean above base-mean and the triplet-level McNemar reaches $p < 0.05$ on all four. The per-dataset and pooled accuracies in Table 12 are means of per-participant accuracies, which weight each participant equally rather than each triplet (as Table 18 and Table 4 do).

Table 12: Paired test of DPT-tuned vs base sentence-T5-XL at the participant and triplet level (test split, seed-42 DPT). Δ is the per-participant accuracy difference in percentage points. Wilcoxon is the paired signed-rank test on participant accuracies, McNemar is the exact binomial test on per-triplet wins and losses.

Dataset	n	base (%)	tuned (%)	Δ (pp)	Wilcoxon p	McNemar p
GSC Abortion (gen)	30	78.1	84.5	+6.4	4.3×10^{-2}	1.5×10^{-3}
GSC Abortion (val)	30	71.8	78.0	+6.2	9.9×10^{-3}	3.5×10^{-8}
GSC Chatbot (gen)	30	65.1	65.5	+0.4	7.5×10^{-1}	1.0
Remesh Campus Protests	90	63.7	65.9	+2.2	3.5×10^{-5}	1.3×10^{-12}
Remesh Foreign Interv.	87	62.2	63.3	+1.1	4.3×10^{-2}	3.4×10^{-3}
Remesh Right to Assemble	89	69.3	70.3	+1.0	7.5×10^{-2}	1.4×10^{-2}
Polis Seattle	4	70.4	72.9	+2.6	7.5×10^{-1}	4.1×10^{-2}
Polis Bowling Green	67	61.1	61.8	+0.7	3.0×10^{-2}	7.2×10^{-189}
Polis Brexit	4	69.9	72.0	+2.1	0.62	5.4×10^{-3}
Polis Canadian	6	50.7	57.1	+6.3	6.2×10^{-2}	6.9×10^{-10}
Polis UBI	5	55.0	58.0	+3.0	0.12	1.4×10^{-2}
Pooled	**442**	**65.6**	**67.6**	**+2.0**	$\mathbf{3.4\times10^{-9}}$	$\mathbf{9.5\times10^{-218}}$

E.9 Per-Topic Scorer: Rank Sweep

Figure 3 in §7 shows accuracy as a function of projection rank r. Table 13 reports the exact macro-mean values with standard deviations across five seeds.

Table 13: Per-topic scorer accuracy as a function of rank (macro-mean over 11 datasets, mean±std over five seeds).

Rank r	1	2	5	10	20	50	100
Natural (%)	70.1 ± 0.3	72.9 ± 0.6	75.4 ± 0.5	77.1 ± 0.4	**77.6** ± 0.4	77.1 ± 0.2	77.3 ± 0.2
Hard (%)	74.0 ± 0.4	76.4 ± 0.8	79.2 ± 0.4	80.4 ± 0.6	**81.1** ± 0.5	81.0 ± 0.5	80.3 ± 0.4

E.10 Per-Topic Scorer: Data Efficiency

Fixing r=20, we vary the number of labeled triplets per topic and evaluate held-out natural accuracy. Figure 4 plots the learning curve across individual datasets and the macro-mean against the base and universal DPT-tuned cosine baselines. Averaging over datasets, $K\approx50$ labeled triplets already cross the universal DPT baseline, and accuracy saturates by $K\approx1{,}000$.

E.11 User Clustering Coherence on Remesh

Here we test whether the DPT-tuned and per-topic projected embeddings provide a geometry in which clusters are coherent in approval behavior, beyond what the base ST5-XL geometry already provides. The Remesh deliberation transcripts are well suited to this question as each participant authors one or more comments, votes *Agree* or *Disagree* on a sample of comments written by others, and the comments tend to be more substantive than those from Polis. Further, in contrast to GSC, there are many statements that were voted on so the approval level for any cluster can be estimated. Taken together, this allows us to get another angle on how well the geometry aligns with participants' expressed approval preferences in a setting akin to how it may be deployed.

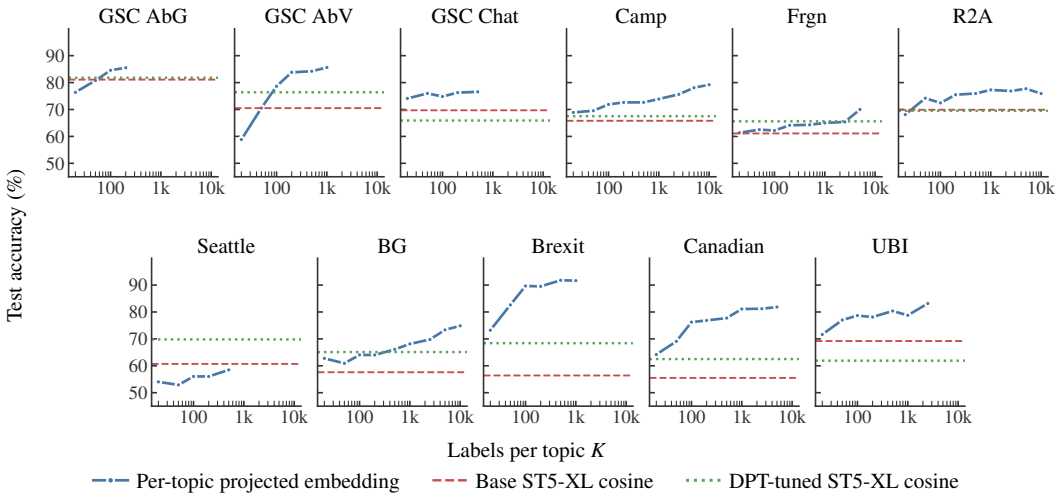

Figure 4: Data efficiency of the rank-20 per-topic projected embedding on base sentence-T5-XL.

For each participant u we form a user vector by mean-pooling the encoder embeddings of their authored comments. We do this under three geometries, the base sentence-T5-XL encoder, the DPT-tuned encoder of §5, and the rank-20 per-topic projection $L^\top \psi$ of §7, with L taken from the ideal-point scorer fit on the same dataset. We then run k-means on the user vectors for $k \in \{3, 5, 8, 10\}$. For every user u assigned to cluster c, we compute their *within-cluster approval rate*, defined as the fraction of *Agree* votes among the cluster-c-authored comments that u actually voted on. The denominator counts only votes that exist, and comments that were never shown to u are excluded. The analogous *across-cluster* rate uses the comments not in u's cluster.

We aggregate per-user rates within a cluster by vote-weighted mean and then macro-average over the clusters at a given k. The headline statistic is the lift $\Delta = \overline{\text{within}} - \overline{\text{across}}$, which is approximately zero under random assignment. Each cell in Table 14, indexed by dataset, encoder, and k, averages over five k-means seeds. We additionally compute a permutation null by shuffling cluster labels, with 50 permutations per seed. All shuffle lifts have absolute value at most 0.005, so we report raw lifts directly.

Table 14: Within−across approval-rate lift Δ (%) on Remesh user clusters, by encoder (base ST5-XL; DPT-tuned ST5-XL; rank-20 per-topic projection of base ST5-XL via $L^\top \psi$). Each cell is the mean of five k-means seeds; per-seed shuffle null has $|\Delta| \leq 0.5\%$. Best entry per row in **bold**.

Dataset	k	**Base**	**Tuned**	**Proj**
Campus Protests	3	9.7	**11.1**	9.5
	5	6.4	8.5	**11.5**
	8	8.7	11.0	**11.1**
	10	8.8	10.4	**10.7**
Foreign Intervention	3	2.4	5.8	**10.1**
	5	6.4	5.1	**11.4**
	8	7.7	**9.0**	0.7
	10	6.0	**7.1**	2.6
Right to Assemble	3	1.4	1.8	**2.0**
	5	2.8	**3.7**	2.9
	8	**4.5**	3.3	2.3
	10	2.1	3.4	**3.9**

Table 14 reports the per-configuration lifts. Both adapted geometries exceed the base on a strong majority of the 12 cells. Clustering separates users most cleanly on Campus Protests, where $\Delta \approx 0.10$ for both the tuned encoder and the projection at all values of k. It separates them less cleanly on

Foreign Intervention, where the projection collapses at $k \geq 8$, and only weakly on Right to Assemble, where the global within-rate is approximately 0.78 and there is little disagreement left to partition.

Aggregating across the 12 configurations, the mean lift is 5.6% for base, 6.7% for DPT-tuned, and 6.6% for the per-topic projected embedding. The mean rank, with lower being better, is 2.50, 1.75, and 1.75 respectively. Paired comparisons over configurations give $\Delta_{\text{DPT-tuned}} - \Delta_{\text{base}} = +1.11$ pp, with 95% bootstrap CI $[+0.4, +1.8]$ and Wilcoxon $p = 0.016$, and $\Delta_{\text{projected}} - \Delta_{\text{base}} = +1.00$ pp, with CI $[-1.3, +3.1]$ and $p = 0.38$. The wider interval for the projection is driven by the two values of k on Foreign Intervention where the rank-20 projection collapses. DPT-tuned and the projected embedding are statistically indistinguishable at this scale, with $\Delta = +0.1$ pp, CI $[-1.9, +2.3]$, and $p = 0.97$. These results suggest that the DPT-tuned embedding provides a geometry which would be useful for the types of tasks identified in §8.

E.12 Likert-Rating Correlation on GSC

The triplet metric used throughout the paper collapses each participant's continuous Likert rating of a candidate statement into a binary preference between two statements for the GSC datasets. However, the GSC surveys release the underlying ratings on a 0 to 6 scale, which lets us ask the more direct question of how strongly cosine similarity between a participant's free-text opinion and a candidate statement tracks the rating they actually gave it, which can give us some indication of whether distances in the DPT-tuned and per-topic projected embedding track latent utility as we would hope.

For each of the three GSC surveys, we recover from the raw CSVs every (u, s, r) triple in which participant u rated statement s with Likert score $r \in [0, 6]$. We assemble an anchor pool per participant from all of their free-text responses, which include the open-ended opinion text in the generation and chatbot surveys together with the per-rating justification text that all three surveys collect. The validation cohort writes only justifications. When correlating with the rating of statement s, we exclude the user's justification text written about s from the anchor pool. The remaining texts are mean-pooled to give a single anchor embedding per rating row. We then compute Spearman rank correlation between cosine similarity and Likert rating, pooled across all rating rows in a survey, under three geometries. These are the base sentence-T5-XL encoder, the DPT-tuned encoder of §5, and the rank-20 per-topic projected embedding $L^\top \psi$ of §7, with L fit from the ideal-point scorer on the same survey's votes.

Table 15 shows that both adapted geometries lift the pooled correlation substantially over the base encoder on every survey. On the two abortion surveys, which are the most explicitly preference-laden topic in GSC, the per-topic projected embedding more than doubles the chatbot-survey correlation gap and pushes pooled correlation past 0.7 on the validation cohort. The chatbot survey gives the smallest absolute correlations across the board, and it is the one survey on which the global tune beats the per-topic projection.

Table 15: Pooled Spearman rank correlation between cosine similarity (anchor → statement) and the participant's Likert rating, across the three GSC surveys. Higher is better. Best per row in **bold**.

Survey	n ratings	Base	Tuned	Proj
GSC Abortion (gen)	500	0.436	0.614	**0.626**
GSC Abortion (val)	1000	0.415	0.544	**0.721**
GSC Chatbot	1100	0.254	**0.385**	0.331

If we accept that Likert ratings give us a noisy view of the participants' latent utility, then this experiment provides some evidence that the distances in the DPT-tuned embedding and the per-topic projected embedding are more closely related to utility than distances in the base embedding model.

E.13 Stacking the Per-Topic Probe on the Tuned Encoder

The per-topic metric scorer of §7 is fit on the base sentence-T5-XL encoder. A natural follow-up question is whether the rank-20 projection would perform better if trained on the encoder that has been globally tuned by DPT. We re-run the same val-selected metric probe pipeline on the DPT-

tuned encoder, holding all other choices fixed, including the rank, the validation grid, the three-way participant split, and the five seeds.

The natural-test macro is essentially unchanged, moving from 77.6 (base ST5-XL) to 78.0 (DPT-tuned). The hard evaluation triplet performance lifts from 81.1 (base ST5-XL) to 87.4 (DPT-tuned), a 6.3 percentage-point gain, with non-negative gains on every dataset and seven datasets gaining over five points.

Table 16: Per-dataset macro-mean test and 875-triplet hard accuracy of the rank-20 metric probe, fit at val-selected hyperparameters on either the base encoder or the DPT-tuned encoder. Mean and standard deviation over five seeds. Best per row in **bold**.

Dataset	Test (base)	Test (tuned)	Hard (base)	Hard (tuned)
GSC Abortion (gen)	**88.3 ± 0.8**	87.5 ± 0.3	78.0 ± 0.0	**79.2 ± 0.4**
GSC Abortion (val)	85.5 ± 2.1	**86.3 ± 0.5**	82.8 ± 2.7	**86.8 ± 0.4**
GSC Chatbot	**76.7 ± 1.5**	76.0 ± 1.2	67.5 ± 1.3	**77.4 ± 0.6**
Polis Seattle	74.6 ± 4.0	**76.2 ± 0.8**	98.5 ± 3.4	**100.0 ± 0.0**
Polis Bowling Green	72.2 ± 0.3	**73.9 ± 0.8**	84.2 ± 1.3	**93.8 ± 2.2**
Polis Brexit	83.4 ± 1.7	**91.5 ± 1.0**	79.2 ± 1.5	**88.8 ± 1.9**
Polis Canadian	**75.7 ± 1.1**	75.5 ± 1.0	77.4 ± 1.5	**85.2 ± 1.8**
Polis UBI	**78.4 ± 0.7**	72.1 ± 0.3	86.7 ± 0.0	86.7 ± 0.0
Remesh Campus	**76.9 ± 0.2**	76.5 ± 0.3	85.4 ± 1.1	**92.2 ± 0.4**
Remesh Foreign	**67.0 ± 0.1**	66.5 ± 0.2	80.8 ± 1.8	**88.0 ± 0.7**
Remesh RTA	74.9 ± 0.5	**76.4 ± 0.1**	71.8 ± 0.8	**83.6 ± 0.5**
Macro mean	77.6	**78.0**	81.1	**87.4**

This asymmetry between natural and hard data can be explained using our formal framework from §4.1. The ideal-point scorer projects into a learned rank-20 subspace and never sees the orthogonal complement, so the cosine decomposition of (2) does not literally apply to the probe's score, but the same mechanism reappears inside the projection. Our hypothesis is that val-selection on natural data tends to pull a few directions into the column space of L that are not strictly preference-aligned, because on natural data those directions correlate positively with preference and so improve validation accuracy. They are nuisance directions that look like signal when semantic and preferential similarity are correlated. When the resulting probe is then evaluated on hard triplets, those nuisance directions contribute against preference, and the projected embedding pays for them with degraded hard-triplet accuracy.

By suppressing the out-of-subspace nuisance globally, the DPT-tuning leaves val-selection with cleaner candidate directions to choose among, and the resulting tuned L contains less of the regime-dependent contamination. We see this directly in the geometry of the learned subspaces. Letting Q_{base} and Q_{tuned} denote orthonormal bases of the two probes' columns, the singular values of $Q_{\text{base}}^{\top} Q_{\text{tuned}}$ are the cosines of the principal angles between the two rank-20 subspaces. Across the eleven datasets, the largest cosine averages 0.95 and the median averages 0.80, while the smallest averages 0.17. The two probes share a core preference subspace where most directions are well aligned, and each has a small number of directions that are nearly orthogonal to anything in the other. Those near-orthogonal residues are the encoder-specific component of each fit, and on hard evaluation triplets it is likely the case that the tuned encoder's residue is benign while the base encoder's residue carries the nuisance signal that costs accuracy.

A second fingerprint comes from decomposing the score margin as $2\langle L^{\top}\psi(a),\, L^{\top}(\psi(p) - \psi(n))\rangle + \|L^{\top}\psi(n)\|^2 - \|L^{\top}\psi(p)\|^2$, an in-projected-space inner product margin plus a projected item-norm difference, reported per dataset in Table 17. The projected embedding on top of the DPT-tuned model improves hard accuracy on 10 of 11 datasets, and on 8 of those the tuned inner product margin is 1.8 to 5.9 times larger than the base, with the item-norm term staying small in magnitude relative to the inner product margin. The remaining two, Polis Bowling Green and Remesh Foreign, also gain hard accuracy but with only a marginal increase in the mean inner product margin (ratios 1.1 and 1.3 respectively), indicating the improvement there is in where the margin lands rather than its mean. The takeaway is that the same rank-20 budget extracts a stronger preference signal once the base directions have been cleaned by DPT.

Table 17: Per-dataset mean of the in-projected-space inner product margin $2\langle L^\top\psi(a), L^\top(\psi(p) - \psi(n))\rangle$ and the projected item-norm term $\|L^\top\psi(n)\|^2 - \|L^\top\psi(p)\|^2$ on the dataset's share of the 875 hard triplets, computed under the rank-20 metric probe fit on either the base or the tuned encoder (single seed). The ratio column is tuned divided by base on the inner product margin.

	Inner product margin			Item-norm difference	
Dataset	**Base**	**Tuned**	**Ratio**	**Base**	**Tuned**
GSC Abortion (gen)	+1.30	+2.38	1.8	−0.25	−0.39
GSC Abortion (val)	+1.96	+3.47	1.8	−0.54	−0.61
GSC Chatbot	+0.11	+0.64	5.9	+0.02	+0.10
Polis Seattle	+0.93	+1.94	2.1	+0.08	+0.17
Polis Bowling Green	+1.17	+1.34	1.1	−0.40	−0.09
Polis Brexit	+0.56	+2.17	3.9	−0.02	−0.42
Polis Canadian	+1.16	+5.11	4.4	−0.02	−0.93
Polis UBI	+1.11	+2.00	1.8	−0.18	−0.08
Remesh Campus	+1.79	+3.28	1.8	−0.27	−0.08
Remesh Foreign	+2.17	+2.71	1.3	−0.71	−0.50
Remesh RTA	+0.82	+1.66	2.0	−0.15	+0.07

Two practical readings follow. First, DPT and the per-topic probe are complementary contributions, not redundant. DPT shifts the global encoder geometry to suppress the nuisance subspace, the per-topic projected embedding extracts the dataset-specific preference subspace, and the two compose with no measurable cost on natural data and a consistent gain on hard triplets. Second, val-selection on natural data is not an unalloyed good for downstream evaluation under distribution shift. When the candidate directions a fit can choose from include some that correlate with preference, val-selection can pull them in, and the model will look fine on its own held-out split while degrading on any distribution where the nuisance correlation flips.

F Full Model Comparison

Table 18 is the complete version of the abridged main-results table (Table 4 in §6), covering all 25 embedding models evaluated in this work.

The 25 baselines are: ST5-Base, ST5-Large, ST5-XL [Ni et al., 2022]; all-mpnet-base-v2, all-MiniLM-L6, all-MiniLM-L12, all-distilroberta, NLI-mpnet, Paraphrase-MiniLM [Reimers and Gurevych, 2019]; e5-large-v2 [Wang et al., 2022]; BGE-large-en-v1.5 [Xiao et al., 2024]; GTE-large [Li et al., 2023]; mxbai-embed-large-v1 [Lee et al., 2024]; Arctic-embed-l [Merrick et al., 2024]; Stella-en-1.5B [Zhang et al., 2024]; Qwen2-1.5B-instruct [Yang et al., 2024]; OpenAI text-embedding-3-small and text-embedding-3-large [OpenAI, 2024]; voyage-3 [Voyage AI, 2024]; voyage-3-large [Voyage AI, 2025]; voyage-4, voyage-4-lite, voyage-4-large [Voyage AI, 2026]; StanceAware-SBERT [Ghafouri et al., 2024]; and BGE-SparseCL (arguana) [Xu et al., 2024].

F.1 Baseline encoding format

Many baselines were trained as asymmetric retrieval encoders, expecting the query side and the passage side of an input to be formatted differently. Treating these encoders symmetrically (passing both anchor and candidate through plain `encode(text)`) consistently under-represents what they can do. We therefore apply each model's intended query/passage convention, treating the participant's anchor text as the query and the candidate statement as the passage. The conventions below are sourced from each model card or registered `config_sentence_transformers.json`, and in all cases cosine similarity is computed between the query-side and passage-side embeddings.

- **Symmetric (no prefixes):** ST5-XL/Large/Base, all-mpnet-base, all-MiniLM-{L6,L12}, all-distilroberta, NLI-mpnet, Paraphrase-MiniLM, GTE-large [Li et al., 2023] (the older non-instruct variant), OpenAI text-embedding-3-{small, large}, StanceAware-SBERT.
- **e5-large-v2:** `"query: "` prefix on anchors, `"passage: "` prefix on candidates.
- **BGE-large-en-v1.5, MxBAI-embed-large-v1:** `"Represent this sentence for searching relevant passages: "` prefix on anchors, plain on candidates.

Table 18: Triplet accuracy (%) across all 11 evaluation datasets, full comparison across 25 models. Best result in each column in **bold**. Abridged version in the body as Table 4.

	GSC			Remesh			Polis					
Model	AbG	AbV	Chat	Camp	Frgn	RTA	Seatl	BG	Brxt	Can	UBI	**Mean**
ST5-XL + DPT (ours)	**81.8**	**76.4**	65.9	**67.5**	**65.6**	69.5	**69.8**	**65.1**	**68.4**	**62.5**	61.9	**68.6**
ST5-XL	81.1	70.5	**69.7**	65.8	61.1	69.9	60.7	57.6	56.4	55.5	69.2	65.2
OAI text-emb-3-large	79.6	74.5	56.1	63.9	61.1	**71.4**	58.1	57.7	51.6	50.0	68.1	62.9
ST5-Large	75.5	65.5	57.2	64.6	59.9	69.2	58.8	58.1	49.9	56.8	**72.2**	62.5
Voyage-4-large	69.9	69.0	63.7	60.0	57.9	69.0	55.4	58.3	50.3	48.3	63.6	60.5
e5-large-v2	63.9	63.6	51.8	65.1	59.4	70.5	53.5	57.1	56.2	57.5	66.1	60.4
Voyage-4	72.3	66.6	62.1	58.4	56.7	67.0	54.9	57.4	50.8	50.1	64.5	60.1
ST5-Base	68.6	62.5	47.1	63.7	59.0	66.6	58.4	56.9	51.1	56.0	68.8	59.9
NLI-mpnet	56.1	66.9	47.6	65.6	61.0	69.5	58.3	56.9	52.8	55.5	68.7	59.9
MxBAI-embed-large	63.9	62.4	55.1	62.7	59.9	69.2	54.2	56.2	50.5	54.5	70.0	59.9
GTE-large	67.6	65.6	59.5	60.2	56.3	67.7	54.7	55.5	46.2	51.6	69.5	59.5
OAI text-emb-3-small	63.8	65.3	51.9	62.1	58.2	70.8	56.2	57.0	50.3	50.4	66.0	59.3
Voyage-3-large	68.3	61.0	60.6	57.2	57.5	67.6	58.5	57.6	50.7	49.8	63.5	59.3
Voyage-4-lite	70.1	64.5	61.3	58.2	56.8	65.4	53.3	57.0	51.2	49.3	65.5	59.3
BGE-large-en	64.7	59.5	56.5	62.5	57.6	68.0	54.2	55.5	53.2	52.9	66.3	59.2
StanceAware-SBERT	58.5	58.2	36.4	64.8	59.7	68.2	68.1	56.1	52.5	52.3	68.3	58.5
Paraphrase-MiniLM	58.2	67.8	51.7	59.7	58.3	67.5	51.7	56.4	53.3	52.4	64.7	58.3
all-mpnet-base	57.1	55.6	60.4	58.5	57.0	69.2	54.4	56.4	47.1	52.4	68.0	57.8
Arctic-embed-l	56.4	62.4	58.5	57.6	56.2	67.5	53.4	57.6	50.2	47.3	64.2	57.4
all-distilroberta	61.8	55.8	53.4	59.5	57.5	67.6	53.7	55.4	50.5	48.5	65.8	57.2
BGE-SparseCL (arguana)	62.4	61.7	42.7	59.2	57.2	66.1	50.1	56.1	57.5	52.6	63.0	57.2
Voyage-3	63.8	50.4	57.5	58.0	54.9	65.2	53.0	55.5	50.6	45.8	65.2	56.4
all-MiniLM-L12	57.8	57.1	47.6	59.2	57.8	67.2	52.0	55.3	49.9	48.8	64.1	56.1
all-MiniLM-L6	55.4	52.3	48.9	60.2	57.6	67.1	53.6	55.1	48.4	48.1	65.4	55.7
Qwen2-1.5B-instruct	59.6	50.8	49.7	56.5	54.9	58.0	48.5	50.9	49.4	50.5	62.7	53.8
Stella-en-1.5B	43.5	50.2	65.3	55.0	55.1	54.0	53.4	49.1	47.4	57.0	49.7	52.7

- **Snowflake Arctic-embed-l-v2.0:** `"query: "` prefix on anchors, plain on candidates. The v2 redesign uses E5-style prefixes rather than the v1 BGE prompt.
- **BGE-SparseCL (arguana):** plain encoding on both sides, matching the authors' own evaluation script. The released checkpoint ships without an SBERT `modules.json`; we verified that the default sentence-transformers wrapper applies mean pooling, matching the paper's training-time `--pooler_type avg`.
- **Voyage:** the API exposes an `input_type` parameter; we pass `"query"` for anchors and `"document"` for candidates.
- **Instruction-tuned (GTE-Qwen2-1.5B-instruct, Stella-en-1.5B-v5):** task-tailored instruction wrapped in each model's standard `"Instruct: <task>\nQuery: "` template on anchors, plain text on candidates.

Instruction-tuned prompt selection. The registered prompts in each model's `config_sentence_transformers.json` target web-search retrieval (e.g., *"Given a web search query, retrieve relevant passages that answer the query"*), which does not match a sentence-to-sentence preference task. Qwen was trained with diverse task instructions and supports arbitrary task strings. Stella was trained primarily on two registered prompts, `s2p_query` for sentence-to-passage retrieval and `s2s_query` for sentence-to-sentence similarity, with the model card recommending those two for general use. We swap in a task-tailored instruction, *"Given an opinion statement on a contested social or political issue, retrieve other statements that share the same stance and underlying values,"* using the template described above.

For Stella, we verified that this choice is benign on natural data: the task-tailored instruction yields a 52.7% macro-mean, the registered `s2s_query` prompt yields 52.8% (a 0.1 pp difference), and the `s2p_query` prompt yields 49.4% (worse, as expected for the web-search framing). For Qwen, the task-tailored instruction and the registered web-search instruction give identical 53.8% macro-mean. The Table 18 rows for these two models report the task-tailored configuration.

G Prompts

Three LLM prompts define the training-data pipeline: GPT-4o-mini filters the issue pool, Claude Sonnet generates 5 diverse opinions per surviving issue, and GPT-4o rewrites a sampled anchor into

a hard triplet. The same rewrite prompt is used to produce the held-out hard eval triplets from real participant anchors (§4.3). The robustness experiments in Appendices E.5 and E.6 use three further rewrite prompts, given at the end of this section. Placeholders in braces (`{issue_text}`, `{anchor}`) are substituted at call time. For reproducibility, exact model identifiers and sampling settings are:

- **Issue filter:** `gpt-4o-mini`, temperature 0, max tokens 5.
- **Opinion generation:** `claude-sonnet-4-20250514`, temperature 1.0, max tokens 1024.
- **Hard-triplet rewrite:** `gpt-4o`, temperature 0.7, max tokens 400.
- **Rewrite variants (Appendices E.5 and E.6):** `gpt-4o` (prompts B and C, mirrored prompt), `gpt-4o-mini` and `claude-sonnet-4-6` (original rewrite prompt); temperature 0.7, max tokens 400.

Issue filter (GPT-4o-mini, one call per candidate issue).

```
You are deciding whether a debate topic is a political, policy, or
social issue suitable for studying public opinion and preference
diversity.

ACCEPT if:
- It is a political, policy, economic, or social issue
- People's positions reflect their values, ideology, or worldview
  (not just personal taste)
- It is the kind of topic debated in legislatures, newspapers, public
  surveys, or town halls
- There is a clear spectrum of positions (e.g., pro-regulation vs.
  free-market)

REJECT if:
- It is purely factual or scientific with a clear answer
- It is abstract philosophy with no policy relevance
- It is a niche technical question that only specialists would engage
  with
- It is about personal lifestyle preferences rather than public affairs
- It has near-universal consensus
- It is nonsensical or too vague

Topic: "{issue_text}"

Respond with exactly one word: ACCEPT or REJECT
```

Opinion generation (Claude Sonnet, one call per issue).

```
Topic: {issue_text}

Write 5 standalone opinions on this topic from 5 different people.
These should read like things people wrote in a forum - NOT like
they are answering a question.

The 5 opinions should be clearly distinct and evenly spaced across
the full opinion spectrum:
1. Strongly supportive - fully committed to this position
2. Moderately supportive - in favor but with reservations
3. Genuinely ambivalent - sees valid points on both sides
4. Moderately opposed - against but acknowledges some merit
5. Strongly opposed - firmly against

Important:
- Do not start with Yes, No, I agree, I disagree or any
  response-like framing
- Each opinion should stand alone as a statement of belief
- Make the difference between adjacent positions (e.g., 1 vs 2,
  4 vs 5) clear and meaningful
- Keep them short and natural - average ~25 words, some shorter,
  some longer

{"opinions": ["op1", "op2", "op3", "op4", "op5"]}
```

Hard-triplet rewrite (GPT-4o, one call per sampled anchor).

```
You are given a person's opinion statement on a political or social
topic.

Original statement: "{anchor}"

Generate two rewritten versions that test whether a model understands
values vs. word overlap:

1. SEMANTIC DISTRACTOR (tricks models that rely on word overlap):
   - Keep nearly IDENTICAL wording and sentence structure as the
     original
   - Only flip the conclusion/stance to be the opposite
   - Goal: MAXIMIZE word overlap while having opposite meaning
   - Example: "Renewable energy must be prioritized above all else"
     -> "Renewable energy is worth considering, but should NOT be
     prioritized above all else"

2. PREFERENCE MATCH (requires understanding values, not words):
   - Express the SAME stance and values as the original
   - But use completely DIFFERENT vocabulary, framing, and sentence
     structure
   - Goal: MINIMIZE word overlap while preserving the underlying
     position
   - Example: "Renewable energy must be prioritized above all else"
     -> "Fossil fuel dependence is our greatest threat - government
     must act decisively on clean power"

Respond with valid JSON only:
{"semantic_distractor": "...", "preference_match": "..."}
```

Normal-correlation rewrite (GPT-4o, one call per sampled anchor).

```
You are given a person's opinion statement on a political or social
topic.

Original statement: "{anchor}"

Generate two rewritten versions that exhibit the natural correlation
between stance and wording, where same stance tends to share words
and opposite stance tends to use different words:

1. PREFERENCE MATCH (high word overlap with the anchor, same stance):
   - Reuse most of the anchor's vocabulary and content words
   - Preserve the original stance and values
   - BUT it must be a genuine paraphrase, NOT a verbatim or near-
     verbatim copy of the anchor. Change at least 3-5 words to
     synonyms, rearrange clauses, or restructure the sentence.
     Otherwise the rewrite is invalid.
   - Goal: MAXIMIZE word overlap while still meaningfully rewording
   - Example: "Renewable energy must be prioritized above all else"
     -> "We must put renewable energy ahead of every other concern"

2. SEMANTIC DISTRACTOR (low word overlap with the anchor, opposite
   stance):
   - Use completely DIFFERENT vocabulary, framing, and sentence
     structure
   - Express the OPPOSITE stance on the same underlying issue
   - Goal: MINIMIZE word overlap while having opposite meaning
   - Example: "Renewable energy must be prioritized above all else"
     -> "Cheap, reliable fossil fuels are what people actually
     need. Chasing wind and solar comes at too great a cost."

Respond with valid JSON only:
{"preference_match": "...", "semantic_distractor": "..."}
```

Rewrite prompt B (Appendix E.6).

```
Below is an opinion someone expressed about a social or political
issue.

Opinion: "{anchor}"

Rewrite it twice:

(a) PREFERENCE MATCH - a statement this person would also agree
    with. It must take the same side of the issue and reflect the
    same underlying values, but share as few words, phrases, and
    grammatical patterns with the original as possible. Change the
    framing entirely.

(b) SEMANTIC DISTRACTOR - a statement this person would reject.
    Reuse the original's words, phrasing, and sentence structure
    almost verbatim, changing only what is needed to reverse the
    position it takes.

For example, given "School uniforms should be mandatory in all
public schools":
(a) "Letting kids dress identically removes distractions and levels
    the social playing field - every district ought to adopt this
    policy."
(b) "School uniforms should not be mandatory in any public school."

Respond with valid JSON only:
{"preference_match": "...", "semantic_distractor": "..."}
```

Rewrite prompt C (Appendix E.6).

```
You are building a benchmark that separates stance understanding
from lexical similarity.

Statement: "{anchor}"

Produce:
1. "semantic_distractor": flip the statement's stance while copying
   its vocabulary and syntax as closely as possible (maximal surface
   overlap, opposite position).
2. "preference_match": keep the statement's stance and values intact
   while expressing them in completely new words and structure
   (minimal surface overlap, same position).

Answer with a single JSON object containing exactly those two keys.
```

Mirrored rewrite for easy controls (Appendix E.5).

```
You are given a person's opinion statement on a political or social
topic.

Original statement: "{anchor}"

Generate two rewritten versions:

1. SEMANTIC DISTRACTOR (opposite stance, different words):
   - Express the OPPOSITE stance on the same underlying issue
   - Use completely DIFFERENT vocabulary, framing, and sentence
     structure
   - Goal: MINIMIZE word overlap while having opposite meaning
   - Example: "Renewable energy must be prioritized above all else"
     -> "Cheap, reliable fossil fuels are what people actually
     need - chasing wind and solar comes at too great a cost."

2. PREFERENCE MATCH (same stance, same words):
   - Keep nearly IDENTICAL wording and sentence structure as the
     original
   - Preserve the SAME stance and values, changing only a word or two
   - Goal: MAXIMIZE word overlap while keeping the same meaning
```

```
   - Example: "Renewable energy must be prioritized above all else"
     -> "Renewable energy must be prioritized above everything else"

Respond with valid JSON only:
{"semantic_distractor": "...", "preference_match": "..."}
```

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The four contributions listed in §1 are each supported by formal results (Theorem 1) and experiments (§6, §7) on 11 datasets. Limitations are discussed in §8.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The Discussion (§8) includes a dedicated Limitations and future work paragraph addressing evaluation scope, the gap between universal and per-topic methods, and responsible use caveats.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Theorem 1 states its assumptions in §5.1 and the full proof is in Appendix C. A proof sketch is provided in the main text.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: All prompts are in Appendix G, hyperparameters and per-encoder selected configurations in Appendix D, dataset URLs in Appendix B, and exact model identifiers and sampling settings are provided. All evaluation data is publicly available, and the code, synthetic training data, and trained adapters are released (see the next item).

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code, the synthetic training data, and the trained LoRA adapters are available at https://github.com/cartgr/Embeddings-for-Preferences, and the DPT-tuned encoder is released at https://huggingface.co/cartgr/embeddings-for-preferences-st5-xl. All evaluation datasets are publicly available; URLs are provided in Appendix B.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Training details including LoRA configuration, optimizer grid, selected hyperparameters, and data splits are in §5.2 and Appendix D. Per-topic scorer details are in §7.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Table 3 and per-topic scorer results report mean $\pm$ std over 5 seeds. Appendix E.8 runs statistical tests for our main comparison. Appendix E.11 provides bootstrap CIs and Wilcoxon tests for clustering comparisons.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix D specifies that all experiments were run on a single NVIDIA A100 GPU and that each DPT training seed takes less than 20 minutes.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research uses publicly available deliberation datasets and does not involve deception or personally identifiable information. Responsible use caveats are discussed in §8.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The Discussion (§8) describes positive applications (deliberation design, viewpoint diversity) and warns against using preference embeddings as a basis for binding decisions.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The released artifact is a lightweight LoRA adapter for existing public encoders, not a general-purpose language model. All evaluation data is already publicly available.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All datasets and models are cited with URLs in Appendix B, which also lists all licenses (Apache-2.0, MIT, CC-BY-4.0, AGPL-3.0). All permit research use.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The released code includes the DPT-tuned LoRA adapters and the synthetic hard-triplet generation pipeline, accompanied by a README. Prompts and generation details are in Appendix G.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: No new data collection was conducted. All datasets were previously collected and publicly released by other researchers.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: No new human subjects research was conducted; see above.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: LLMs are used to generate the synthetic training data (§5.2). Exact model identifiers, prompts, and sampling settings are in Appendix G.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.