mathematics

MDPI

Article

ReDyGait: Representation Disentanglement with Gated Attention for Invariant-Contextual Transfer in Stance Detection

Yanzhou Ma †, Yun Luo *,† and Mingyang Peng

College of International Studies, National University of Defense Technology, Nanjing 210012, China; myz827@nudt.edu.cn (Y.M.); pengmingyang21@nudt.edu.cn (M.P.)

* Correspondence: ly24@nudt.edu.cn

† These authors contributed equally to this work.

Abstract

Cross-topic stance detection degrades when encoders entangle stance signals with topic-specific vocabulary, causing representations that fail to transfer to unseen targets. Existing methods commit to either topic-invariant or topic-aware representations and apply the same strategy uniformly to every input, sacrificing complementary information. We propose ReDyGait, a three-stage framework that disentangles these two types of signals through dedicated contrastive pre-training and recombines them adaptively at inference time. Stage 1 trains a topic-invariant encoder with supervised contrastive loss over cross-topic positives. Stage 2 trains a topic-contextual encoder with bidirectional pair contrastive loss over within-topic positives; both stages employ topic-aware hard negative mining to prevent shortcut learning. Stage 3 freezes the two contrastive encoders and learns a gating network that produces per-instance weights over invariant, contextual, and base-encoder pathways. On VAST, ReDyGait achieves a macro-averaged F1 of 0.782 in the zero-shot setting and 0.752 in the few-shot setting, improving over the strongest baseline by 1.1 points in both; on SEM16t6 in a leave-one-target-out setup, ReDyGait reaches an average F1 of 0.612. Analysis of the learned gate weights shows that the model shifts toward the invariant pathway for unfamiliar topics and toward the contextual pathway when topic-specific patterns are available, confirming that the disentanglement operates as intended.

Keywords: stance detection; contrastive learning; representation disentanglement; gated attention; cross-topic generalization; natural language processing

MSC: 68T50; 68T05

Check for updates

Academic Editor: Iraklis Varlamis

Received: 11 March 2026
Revised: 27 March 2026
Accepted: 30 March 2026
Published: 7 April 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Pre-trained language models (PLMs) have become the dominant backbone for stance detection, the task of classifying whether a text expresses *favor*, *against*, or *neutral* toward a specified target [1,2]. Accurate stance detection supports downstream applications ranging from misinformation identification [3,4] and fact verification [5] to political opinion analysis [6] and public-health surveillance [7]. Fine-tuning a single PLM encoder on labeled stance data yields strong results when training and test targets overlap. The picture changes when the model encounters topics absent from training: accuracy drops sharply because the learned representations entangle topic-specific lexical cues with genuine stance signals [8,9]. Cross-target generalization—detecting stance on unseen topics—remains the central open problem in the field.

Two families of methods have emerged to address this challenge. *Topic-invariant* approaches strip topic information from the representation through adversarial training [10] or prototype-based contrastive learning [11,12], aiming to retain only stance-indicative features that transfer across targets. *Topic-aware* approaches take the opposite route, injecting external knowledge [13,14] or synthesizing target-specific training data [15] so the model can reason about unfamiliar topics. Both strategies improve over vanilla fine-tuning, yet each commits to a single representation regime applied uniformly to every input. A topic-invariant encoder discards contextual cues that remain informative for familiar topics; a topic-aware encoder overfits to seen topics and transfers poorly to unseen ones. The most recent concurrent work, TATA [16], trains separate topic-agnostic and topic-aware encoders but still fuses them through static concatenation, with no mechanism for the model to adjust the balance per instance. Moreover, both TATA and JointCL [12] apply the same contrastive objective to both encoder types and construct negatives without topic control, allowing the encoder to exploit topic-lexical shortcuts rather than learning genuine stance distinctions.

We argue that the missing ingredient is *dynamic, instance-level fusion* of disentangled representations. In this paper we propose **ReDyGait** (**Re**presentation **Dy**sentanglement with **G**ated **A**ttention for **I**nvariant-contextual **T**ransfer), a three-stage framework that first trains two specialized encoders and then learns to combine them adaptively. ReDyGait addresses the above limitations by assigning a dedicated contrastive loss to each encoder, introducing topic-aware hard negative mining to prevent lexical shortcuts, and replacing static concatenation with a learned gating network that produces per-instance adaptive fusion weights. We detail these distinctions in Section 2.

Our contributions are as follows:

1. We disentangle stance representations into a *topic-invariant* stream trained with supervised contrastive loss and a *topic-contextual* stream trained with bidirectional pair contrastive loss, each with a dedicated objective matched to its role.
2. We introduce *topic-aware hard negative mining* that retrieves opposite-stance examples from the same topic group, forcing the invariant encoder to separate stance from topic rather than relying on lexical shortcuts.
3. We propose a *learned gating network* that produces per-instance weights over three pathways (invariant, contextual, and a base encoder), enabling the model to shift its reliance depending on how familiar the input topic is.
4. We evaluate ReDyGait on the VAST benchmark under both Zero-shot and Few-shot settings and on SEM16t6 in a leave-one-target-out cross-topic setup, demonstrating consistent improvements over competitive baselines. Ablation and error analyses confirm that both encoder streams contribute complementary signals and that the gating weights are interpretable.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 describes the proposed ReDyGait framework. Section 4 presents the experimental setup, results, and analysis. Section 5 concludes the paper.

2. Related Work

2.1. Stance Detection

Stance detection aims to identify whether a text expresses *favor*, *against*, or *neutral* toward a given target. Biber and Finegan [17] originally defined stance as the expression of a speaker's attitudes and judgments; in NLP, the task has since been operationalized across domains including political opinion mining [6], fact verification [3–5], and public-health discourse [7]. Early systems relied on feature engineering with SVM classifiers [1] or recurrent networks with conditional encoding [18]. The introduction of BERT [19] and its

variants shifted the field toward fine-tuning a single pre-trained encoder, which works well when training and test targets overlap but degrades on unseen topics [2,8].

Cross-target generalization has therefore become the central open problem [9,20,21]. Existing approaches fall into two broad categories. The first pursues *topic-invariant* representations: Allaway et al. [10] use adversarial training to strip topic information from the encoder; Liang et al. [11] apply prototype-based contrastive learning to cluster stance labels regardless of topic; JointCL [12] combines stance-level and target-level contrastive objectives in a single framework. The second category injects *topic-aware knowledge*: KEPrompt [13] reformulates stance detection as a cloze task enriched with external knowledge, using automatic verbalizers and background knowledge injection; Zhu et al. [14] augment the input with targeted background knowledge retrieved from Wikipedia; Li and Yuan [15] use generative models to synthesize training data for unseen targets. However, both lines of work commit to one representation strategy and apply it uniformly across all inputs. A topic-invariant encoder discards contextual cues that are informative for familiar topics; a topic-aware encoder overfits to seen topics and transfers poorly. No prior method lets the model itself decide, per instance, how much to rely on each type of signal.

2.2. Contrastive Learning

Contrastive learning trains encoders to pull similar examples together and push dissimilar ones apart in embedding space. Originally developed as a self-supervised method in computer vision [22,23] and image captioning [24], the paradigm has been widely adopted in NLP [25,26]. SimCSE [27] demonstrated that simple dropout-based augmentation produces strong sentence embeddings; ESimCSE [28] extended this with enhanced sample construction. Gunel et al. [29] showed that supervised contrastive objectives improve pre-trained language model fine-tuning by exploiting label information, building on the SupCon framework of Khosla et al. [30].

In stance detection specifically, contrastive objectives have been used to model stance-indicative features. Mohtarami et al. [31] applied contrastive language adaptation for cross-lingual transfer. Liang et al. [11,12] introduced prototype-based and joint contrastive losses that group examples by stance label. Arakelyan et al. [32] proposed topic-guided sampling to improve data efficiency across domains. The concurrent work closest to ours is TATA [16], which also separates topic-agnostic and topic-aware embeddings and trains both with contrastive objectives over unlabeled news articles. However, TATA and all prior contrastive methods for stance detection share a common limitation: they treat the two types of representation as fixed, equal-weight channels and fuse them through simple concatenation, with no mechanism for the model to *dynamically* adjust how much it relies on topic-invariant versus topic-contextual signals for each input. When the test topic is entirely unseen, the contextual stream contributes noise; when the topic is familiar, discarding contextual cues wastes useful information. This static fusion is the bottleneck that prevents existing disentanglement approaches from fully exploiting both representation types.

To address this gap, we propose ReDyGait (**Re**presentation **Dy**sentanglement with **G**ated **A**ttention for **I**nvariant-contextual **T**ransfer). Compared with TATA [16] and JointCL [12], ReDyGait makes three distinct contributions: (1) each encoder is trained with a *dedicated* contrastive objective—supervised contrastive loss for the invariant stream and bidirectional pair loss for the contextual stream—rather than applying the same loss to both; (2) we introduce *topic-aware hard negative mining* that selects negatives from the same topic group, forcing the invariant encoder to distinguish stance rather than topic, a training signal absent from both TATA and JointCL; and (3) instead of concatenation, we fuse the two streams through a *learned gating network* that produces per-instance weights

over three pathways (invariant, contextual, and a base encoder), allowing the model to shift its reliance depending on how familiar the input topic is.

3. Proposed Method

ReDyGait is a three-stage framework: (1) topic-invariant contrastive pre-training, (2) topic-contextual contrastive pre-training, and (3) attention-gated fusion fine-tuning. Figure 1 illustrates the full pipeline; Figures 2 and 3 detail the hard negative mining and gating mechanisms, respectively.

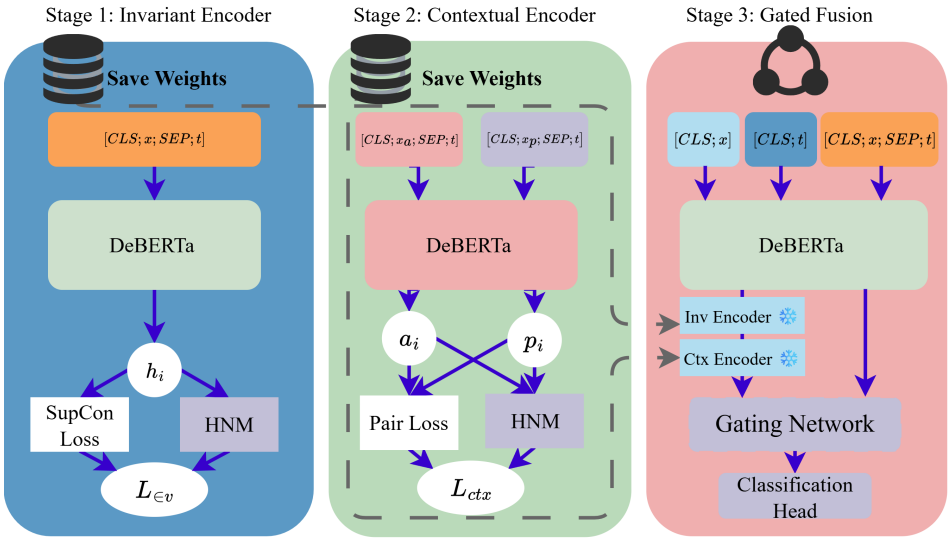


Figure 1. Overview of the ReDyGait pipeline. Stage 1 trains a topic-invariant encoder with supervised contrastive loss and hard negative mining (Section 3.3). Stage 2 trains a topic-contextual encoder with bidirectional pair contrastive loss (Section 3.4). Stage 3 freezes both contrastive encoders and fuses them with a trainable base encoder through cross-attention and a learned gating network (Section 3.5).

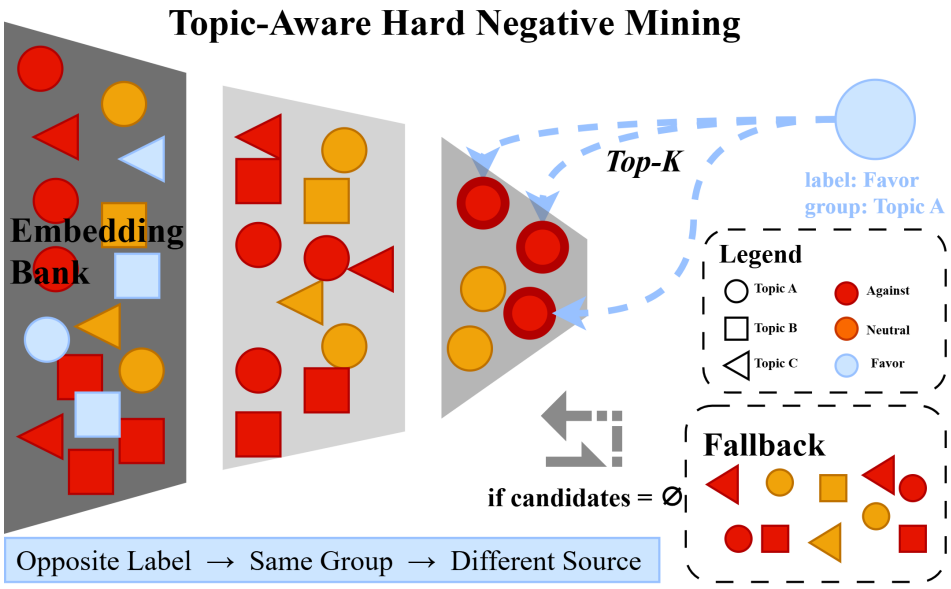


Figure 2. Topic-aware hard negative mining. For each anchor, the three-way filter selects candidates with opposite stance, same topic group, and different source identity. The top-K candidates by cosine similarity are retrieved from the embedding bank. When fewer than K same-group candidates exist, the constraint is relaxed and a validity mask tracks which positions are filled.

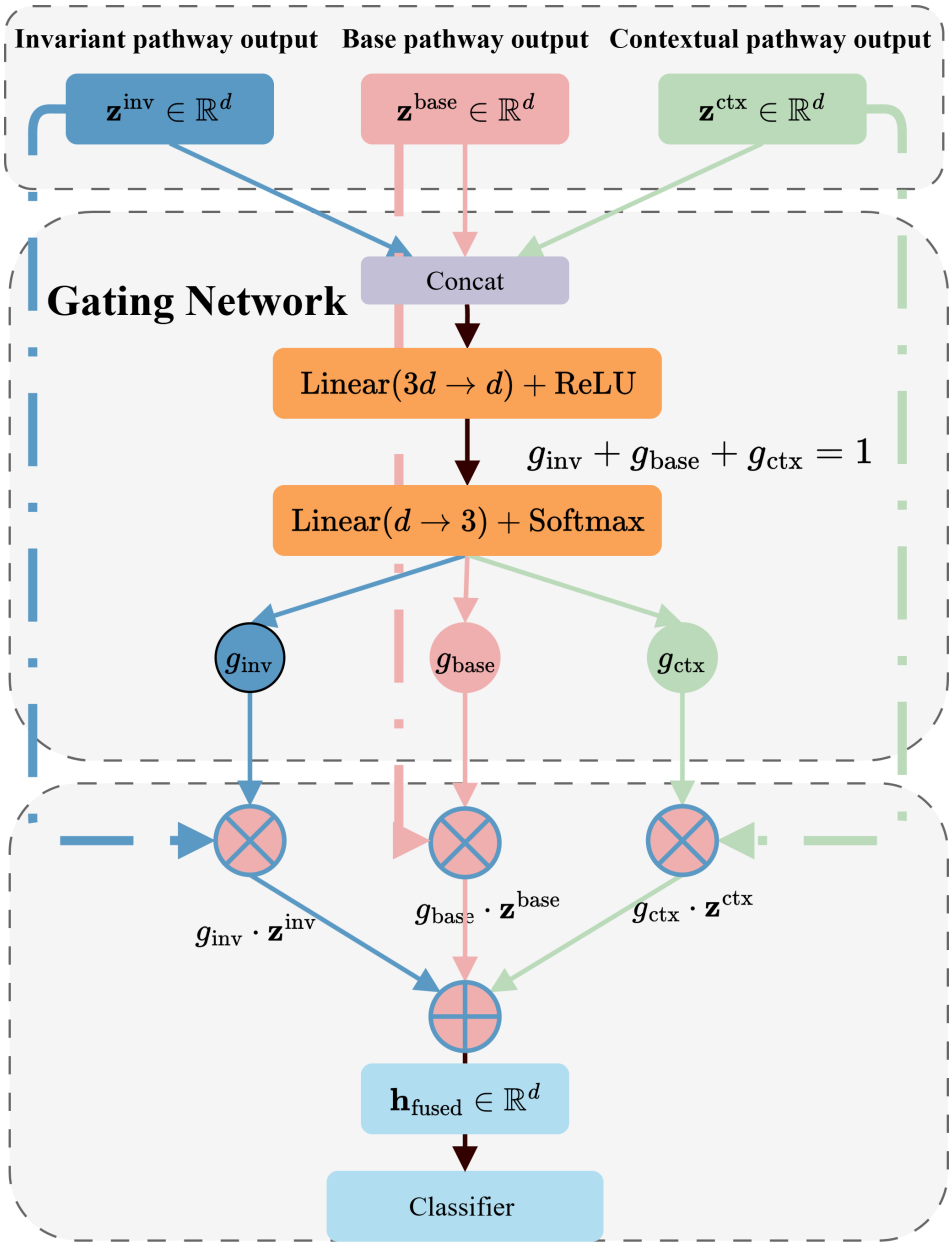


Figure 3. Gated attention fusion in Stage 3. The invariant embedding attends over text tokens; the contextual embedding attends over topic tokens. A two-layer gating network produces per-instance softmax weights over three pathways (base, invariant, contextual), and the weighted sum is passed to the classification head.

3.1. Problem Formulation

Given a text x and a target topic t, stance detection predicts a label $y \in \mathcal{Y}$, where $\mathcal{Y} = \{\text{FAVOR}, \text{AGAINST}, \text{NEUTRAL}\}$ (VAST, SEM16t6). We define a *group* g as the normalized topic string: all samples discussing the same target share a group identifier. Each training sample is a tuple $(x_i, t_i, y_i, g_i, s_i)$, where s_i is a unique source identifier used to prevent self-retrieval during hard negative mining.

3.2. Shared Encoder Backbone

All three stages use DeBERTa-v3-base [33] as the backbone ($d_h = 768$). The encoder extracts the `[CLS]` token representation from the last hidden layer and applies dropout:

$$\mathbf{h} = \text{Dropout}\big(\text{DeBERTa}(\mathbf{x})_{[\texttt{CLS}]}\big) \in \mathbb{R}^{d_h} \tag{1}$$

The three stages share the same architecture but are initialized and trained independently; Stages 1 and 2 each produce a specialized encoder, while Stage 3 trains a fresh copy that serves as the base encoder for fusion.

3.3. Stage 1: Topic-Invariant Contrastive Pre-Training

The invariant encoder learns a representation space where texts with the same stance label cluster together *regardless of topic*. The goal is to isolate stance-indicative features—argumentative structure, sentiment polarity, hedging patterns—from topic-specific vocabulary, so that the learned representations transfer to unseen targets. Each sample is tokenized as $[\texttt{CLS}; x_i; \texttt{SEP}; t_i]$ with a maximum length of 512 tokens, and the encoder produces $\mathbf{h}_i \in \mathbb{R}^{d_h}$ via Equation (1).

3.3.1. Supervised Contrastive Loss

We adopt the supervised contrastive loss (SupCon) [30]. For a mini-batch of N samples with L_2-normalized embeddings $\{\bar{\mathbf{h}}_i\}_{i=1}^{N}$ and labels $\{y_i\}_{i=1}^{N}$:

$$\mathcal{L}_{\text{SupCon}} = \sum_{i=1}^{N} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp\left(\bar{\mathbf{h}}_i^{\top} \bar{\mathbf{h}}_p / \tau\right)}{\sum_{j \neq i} \exp\left(\bar{\mathbf{h}}_i^{\top} \bar{\mathbf{h}}_j / \tau\right)} \tag{2}$$

where $P(i) = \{j : y_j = y_i,\ j \neq i\}$ is the set of in-batch positives sharing the same stance label (across *all* topics), τ is a temperature scalar, and the denominator sums over all non-self samples. We subtract $\max_j \bar{\mathbf{h}}_i^{\top} \bar{\mathbf{h}}_j$ from the logits before exponentiation for numerical stability. Batches where $|P(i)| = 0$ for all i contribute zero loss.

The critical design choice is that positives span all topics. Samples sharing the same stance label are pulled together regardless of their topic origin. Because the positive set mixes topics, the encoder cannot rely on topic-specific vocabulary and must learn stance-transferable patterns instead.

3.3.2. Retrieval-Based Hard Negative Mining

In-batch negatives are often too easy: they may differ in both topic and stance, letting the model exploit topic cues as a shortcut. We add a retrieval-based hard negative mining module (Figure 2) that selects negatives from the *same topic group*, where the only distinguishing signal is stance.

Embedding Bank

We maintain a bank $\mathcal{B} = \{(\bar{\mathbf{h}}_j, y_j, g_j, s_j)\}_{j=1}^{|\mathcal{D}|}$ storing L_2-normalized embeddings for the entire training set $\mathcal{D}$. The bank is rebuilt by a full forward pass through the current encoder at the start of each epoch, so the retrieval targets track the evolving representation space.

Mining Procedure

For each anchor $(\mathbf{h}_i, y_i, g_i, s_i)$, we compute cosine similarities against all bank entries and apply a candidate filter:

$$\mathcal{C}(i) = \left\{j \in \mathcal{B} : y_j \neq y_i \ \wedge\ g_j = g_i \ \wedge\ s_j \neq s_i\right\} \tag{3}$$

requiring opposite stance, the same topic group, and different source identity (to prevent self-retrieval). We select the top-K candidates by descending similarity:

$$\mathcal{N}_{\text{hard}}(i) = \underset{j \in \mathcal{C}(i)}{\text{top-}K}\ \bar{\mathbf{h}}_i^{\top} \bar{\mathbf{h}}_j \tag{4}$$

If $|\mathcal{C}(i)| < K$, we pad with the hardest negatives from any group (relaxing the $g_j = g_i$ constraint) and track validity through a binary mask $\mathbf{m}_i \in \{0,1\}^K$.

Margin-Based Hard Negative Loss

Rather than mixing hard negatives into the SupCon denominator, we apply a separate triplet-style margin loss. For each anchor $\mathbf{h}_i$, we compute a *positive prototype* $\mathbf{h}_i^+$ as the L_2-normalized mean of all in-batch embeddings sharing the same label:

$$\mathbf{h}_i^+ = \text{normalize}\Big(\frac{1}{|P(i)|}\sum_{p\in P(i)} \bar{\mathbf{h}}_p\Big) \tag{5}$$

If $|P(i)| = 0$ (no same-label partner in the batch), we fall back to $\mathbf{h}_i^+ = \bar{\mathbf{h}}_i$. The margin loss over the K retrieved hard negatives $\{\mathbf{h}_{i,k}^-\}_{k=1}^K$ is:

$$\mathcal{L}_{\text{HNM}} = \frac{1}{\sum_i \|\mathbf{m}_i\|_1}\sum_{i=1}^{N}\sum_{k=1}^{K} m_{i,k}\cdot\left[\delta + \bar{\mathbf{h}}_i^\top \bar{\mathbf{h}}_{i,k}^- - \bar{\mathbf{h}}_i^\top \bar{\mathbf{h}}_i^+\right]_+ \tag{6}$$

where δ is a margin hyperparameter, $[\cdot]_+ = \max(0,\cdot)$ is the hinge function, and $m_{i,k}$ is the validity mask for the k-th negative. The prototype aggregation smooths out noise from individual positives and gives the margin loss a stable reference point.

3.3.3. Combined Stage 1 Objective

The full Stage 1 loss combines supervised contrastive learning with hard negative mining:

$$\mathcal{L}_{\text{inv}} = \mathcal{L}_{\text{SupCon}} + \lambda_{\text{hn}}\mathcal{L}_{\text{HNM}} \tag{7}$$

where λ_{hn} controls the hard negative contribution. After training converges, we save the encoder weights and store the invariant embeddings $\{\mathbf{h}_i^{\text{inv}}\}$ for all training samples.

3.4. Stage 2: Topic-Contextual Contrastive Pre-Training

While the invariant encoder abstracts away topic information, topic-specific context can be equally useful. In political stance detection, for instance, ideological associations of certain phrases only make sense within a particular target context (e.g., "build the wall" is stance-indicative for "Donald Trump" but irrelevant for "Joe Biden"). The contextual encoder is trained to capture these topic-dependent patterns.

3.4.1. Anchor–Positive Pair Construction

Unlike Stage 1, which groups positives by stance label alone, Stage 2 constructs explicit anchor–positive pairs that share both stance *and* topic. For each sample (x_i, t_i, y_i, g_i), we draw a positive partner (x_j, t_j, y_j, g_j) satisfying $y_j = y_i \wedge g_j = g_i$—same stance, same topic group. During training the positive is sampled uniformly at random from the eligible set; at evaluation time we select the first eligible sample for reproducibility. Both anchor and positive are tokenized as $[\texttt{CLS}; x; \texttt{SEP}; t]$ and encoded independently.

3.4.2. Bidirectional Pair Contrastive Loss

Given a batch of N anchor–positive pairs, let $\{\bar{\mathbf{a}}_i\}_{i=1}^N$ and $\{\bar{\mathbf{p}}_i\}_{i=1}^N$ be the L_2-normalized anchor and positive embeddings. The logit matrix is:

$$L_{ij} = \bar{\mathbf{a}}_i^\top \bar{\mathbf{p}}_j / \tau \tag{8}$$

The bidirectional loss treats the diagonal entries as ground-truth matches:

$$\mathcal{L}_{\text{Pair}} = \frac{1}{2}\left[\text{CE}(\mathbf{L}, \mathbf{I}) + \text{CE}(\mathbf{L}^\top, \mathbf{I})\right] \tag{9}$$

where $\text{CE}(\cdot, \mathbf{I})$ denotes row-wise cross-entropy with identity-indexed labels (row i should match column i). This mirrors the CLIP-style dual cross-entropy objective [27]: the forward term asks "which positive matches this anchor?" while the backward term asks "which anchor matches this positive?" Without the backward term, the loss can degenerate by collapsing all positives to a single point.

Because positives are restricted to the same topic group, the contextual encoder produces topic-specific clusters: "Favor toward Trump" and "Favor toward Biden" occupy distinct regions despite sharing a stance label.

3.4.3. Combined Stage 2 Objective

Stage 2 reuses the same hard negative mining module from Section 3.3.2, applied to anchor embeddings:

$$\mathcal{L}_{\text{ctx}} = \mathcal{L}_{\text{Pair}} + \lambda_{\text{hn}}\mathcal{L}_{\text{HNM}} \tag{10}$$

After training, we store the contextual embeddings $\{\mathbf{h}_i^{\text{ctx}}\}$ for all training samples.

3.5. Stage 3: Attention-Gated Fusion

After Stages 1 and 2, we freeze both contrastive encoders and train a fusion module that dynamically combines their representations with a fresh base encoder. The key idea is that different samples benefit from different mixtures of invariant and contextual information—a factual claim may rely more on stance-indicative patterns (invariant), while a sarcastic tweet may need topic context to disambiguate. The gating network (Figure 3) learns this per-instance routing.

To aid readability, we briefly summarize the Stage 3 fusion procedure before presenting the formal equations. Three encoders—the frozen invariant encoder, the frozen contextual encoder, and a fresh trainable base encoder—each map the input text–topic pair to a 768-dimensional embedding. The invariant embedding then queries the base encoder's passage–topic representation via scaled dot-product cross-attention, while the contextual embedding queries the base encoder's topic-only representation, allowing each specialized view to select the most relevant aspects of the shared base representation. A two-layer MLP takes the concatenation of the three resulting pathway vectors and produces three instance-specific softmax gates $(g_{\text{base}}, g_{\text{inv}}, g_{\text{ctx}})$ that sum to one, learning to up-weight the invariant pathway for unfamiliar topics and the contextual pathway when topic-specific patterns are available. The final representation is the gated sum of the three pathway vectors, which is fed through a two-layer classification head with batch normalization to predict the stance label.

3.5.1. Multi-View Input Encoding

The fusion stage produces four representations per sample. The two frozen contrastive encoders each receive the concatenated passage–topic input $[\texttt{CLS}; x; \texttt{SEP}; t]$ and output their `[CLS]` embeddings:

$$\mathbf{h}^{\text{inv}} = \text{DeBERTa}_{\text{inv}}([x;\ \texttt{[SEP]};\ t])_{[\texttt{CLS}]} \in \mathbb{R}^{d_h} \tag{11}$$

$$\mathbf{h}^{\text{ctx}} = \text{DeBERTa}_{\text{ctx}}([x;\ \texttt{[SEP]};\ t])_{[\texttt{CLS}]} \in \mathbb{R}^{d_h} \tag{12}$$

The trainable base encoder processes the same concatenated input to produce a passage–topic [CLS] embedding, and separately encodes the topic alone:

$$\mathbf{h}^{\text{pair}} = \text{DeBERTa}_{\text{base}}([x;\ \texttt{[SEP]};\ t])_{\texttt{[CLS]}} \in \mathbb{R}^{d_h} \tag{13}$$

$$\mathbf{h}^{\text{topic}} = \text{DeBERTa}_{\text{base}}(t)_{\texttt{[CLS]}} \in \mathbb{R}^{d_h} \tag{14}$$

The passage–topic encoding captures cross-sequence interaction through self-attention, while the topic-only encoding provides a target representation for the contextual attention pathway described below.

3.5.2. Scaled Dot-Product Cross-Attention

Each frozen contrastive embedding is used as a query to attend over the base encoder's representations, selecting which aspects of the base representation are most relevant to its specialized perspective.

For the invariant pathway, the frozen invariant embedding queries the passage–topic representation from the base encoder:

$$\tilde{\mathbf{h}}^{\text{inv}} = \text{Attn}\big(\mathbf{h}^{\text{pair}},\ W_{\text{inv}}\,\mathbf{h}^{\text{inv}}\big) \tag{15}$$

where $W_{\text{inv}} \in \mathbb{R}^{d_h \times d_h}$ is a learnable projection and $\text{Attn}(\mathbf{v}, \mathbf{q}) = \text{softmax}(\mathbf{v}^\top \mathbf{q}/\sqrt{d_h}) \cdot \mathbf{v}$ computes scaled dot-product attention. The attended output is concatenated with the passage–topic [CLS] embedding and projected:

$$\mathbf{z}^{\text{inv}} = \text{ReLU}\big(W_1^{\text{inv}}\,[\tilde{\mathbf{h}}^{\text{inv}};\ \mathbf{h}^{\text{pair}}] + b_1^{\text{inv}}\big) \in \mathbb{R}^{d_h} \tag{16}$$

where $W_1^{\text{inv}} \in \mathbb{R}^{d_h \times 2d_h}$.

For the contextual pathway, the frozen contextual embedding attends over the *topic-only* representation. This pairing is deliberate: the contextual encoder already captures topic-specific stance patterns, so attending over the topic embedding lets it ground those patterns in the current target:

$$\tilde{\mathbf{h}}^{\text{ctx}} = \text{Attn}\big(\mathbf{h}^{\text{ctx}},\ W_{\text{ctx}}\,\mathbf{h}^{\text{topic}}\big) \tag{17}$$

followed by concatenation with the passage–topic [CLS] embedding and projection:

$$\mathbf{z}^{\text{ctx}} = \text{ReLU}\big(W_1^{\text{ctx}}\,[\tilde{\mathbf{h}}^{\text{ctx}};\ \mathbf{h}^{\text{pair}}] + b_1^{\text{ctx}}\big) \in \mathbb{R}^{d_h} \tag{18}$$

The base pathway applies dropout and a nonlinear projection to the passage–topic encoding, providing a direct text–topic representation without contrastive specialization:

$$\mathbf{z}^{\text{base}} = \text{ReLU}\big(W_1^{\text{base}}\,\text{Dropout}(\mathbf{h}^{\text{pair}}) + b_1^{\text{base}}\big) \in \mathbb{R}^{d_h} \tag{19}$$

where $W_1^{\text{base}} \in \mathbb{R}^{d_h \times d_h}$.

3.5.3. Gating Network

A fixed-weight combination of the three pathways would treat every sample identically, ignoring the fact that some inputs are better served by invariant features while others need contextual grounding. The gating network addresses this by producing per-sample mixture weights through a two-layer MLP. Given the three pathway outputs $\mathbf{z}^{\text{base}}, \mathbf{z}^{\text{inv}}, \mathbf{z}^{\text{ctx}} \in \mathbb{R}^{d_h}$:

$$\mathbf{u} = \text{ReLU}\big(W_g^{(1)} [\mathbf{z}^{\text{base}}; \mathbf{z}^{\text{inv}}; \mathbf{z}^{\text{ctx}}] + b_g^{(1)}\big) \in \mathbb{R}^{d_h} \tag{20}$$

$$\mathbf{g} = \text{softmax}\big(W_g^{(2)} \mathbf{u} + b_g^{(2)}\big) \in \mathbb{R}^3 \tag{21}$$

where $W_g^{(1)} \in \mathbb{R}^{d_h \times 3d_h}$, $W_g^{(2)} \in \mathbb{R}^{3 \times d_h}$, and dropout is applied to the concatenated input. The softmax ensures $g_{\text{base}} + g_{\text{inv}} + g_{\text{ctx}} = 1$. The fused representation is:

$$\mathbf{h}_{\text{fused}} = g_{\text{base}} \cdot \mathbf{z}^{\text{base}} + g_{\text{inv}} \cdot \mathbf{z}^{\text{inv}} + g_{\text{ctx}} \cdot \mathbf{z}^{\text{ctx}} \in \mathbb{R}^{d_h} \tag{22}$$

3.5.4. Classification Head

The fused representation passes through a two-layer MLP with batch normalization:

$$\hat{y} = W_c^{(2)} \,\text{BN}\big(\text{ReLU}(W_c^{(1)} \,\text{Dropout}(\mathbf{h}_{\text{fused}}) + b_c^{(1)})\big) + b_c^{(2)} \tag{23}$$

where $W_c^{(1)} \in \mathbb{R}^{d_h \times d_h}$, $W_c^{(2)} \in \mathbb{R}^{|\mathcal{Y}| \times d_h}$, and $\text{BN}(\cdot)$ denotes batch normalization [34]. The model is trained with cross-entropy loss:

$$\mathcal{L}_{\text{fuse}} = \text{CE}(\hat{y}, y) \tag{24}$$

During Stage 3, only the base DeBERTa encoder, the attention projections, the gating network, and the classification head are updated. The invariant and contextual encoders remain frozen, and their embeddings are accessed by index lookup. This design keeps the contrastive representations stable while allowing the fusion module to learn how to combine them.

3.6. Training Summary

The full pipeline consists of three stages. In **Stage 1** (Invariant pre-training), a DeBERTa encoder is trained with $\mathcal{L}_{\text{SupCon}} + \lambda\, \mathcal{L}_{\text{HNM}}$, where positives share the same stance label regardless of topic. In **Stage 2** (Contextual pre-training), a separate DeBERTa encoder is trained with $\mathcal{L}_{\text{Pair}} + \lambda\, \mathcal{L}_{\text{HNM}}$, where positives share both the same stance and the same topic. In **Stage 3** (Fusion fine-tuning), the two contrastive encoders are frozen and a fresh DeBERTa base encoder is trained jointly with the gating network and classification head using cross-entropy $\mathcal{L}_{\text{CE}}$. The frozen encoders add no gradient cost during Stage 3; only the base encoder, attention projections, gating network, and classification head are updated.

Algorithm 1 summarizes the complete ReDyGait pipeline.

Algorithm 1 ReDyGait Training and Inference

Require: Training set $\mathcal{D} = \{(x_i, t_i, y_i, g_i)\}_{i=1}^{N}$, retrieval count K, margin δ, weight λ_{hn}
Ensure: Trained model f_θ

// Stage 1: Invariant Pre-training
Init f_{inv} from DeBERTa
for epoch $= 1$ **to** E_1 **do**
  Rebuild embedding bank $\mathcal{B}_{\text{inv}}$ over $\mathcal{D}$
  for each mini-batch **do**
    $\mathbf{h}_i \leftarrow f_{\text{inv}}(x_i, t_i)$; positives: $\{j : y_j = y_i\}$ (cross-topic)
    Mine top-K hard negatives from $\mathcal{B}_{\text{inv}}$ (same group, opposite stance)
    Update f_{inv} via $\nabla(\mathcal{L}_{\text{SupCon}} + \lambda_{\text{hn}}\,\mathcal{L}_{\text{HNM}})$
  end for
end for
// Stage 2: Contextual Pre-training
Init f_{ctx} from DeBERTa
for epoch $= 1$ **to** E_2 **do**
  Rebuild embedding bank $\mathcal{B}_{\text{ctx}}$ over $\mathcal{D}$
  for each mini-batch **do**
    $\mathbf{h}_i \leftarrow f_{\text{ctx}}(x_i, t_i)$; positives: $\{j : y_j = y_i \wedge g_j = g_i\}$ (within-topic)
    Mine top-K hard negatives from $\mathcal{B}_{\text{ctx}}$
    Update f_{ctx} via $\nabla(\mathcal{L}_{\text{Pair}} + \lambda_{\text{hn}}\,\mathcal{L}_{\text{HNM}})$
  end for
end for
Freeze f_{inv}, f_{ctx}
// Stage 3: Gated Fusion Fine-tuning
Init base encoder f_{base}, gating network G, classifier C
for epoch $= 1$ **to** E_3 **do**
  for each mini-batch **do**
    $\mathbf{h}_{\text{b}} \leftarrow f_{\text{base}}(x_i, t_i)$; $\mathbf{h}_{\text{v}} \leftarrow f_{\text{inv}}(x_i, t_i)$; $\mathbf{h}_{\text{c}} \leftarrow f_{\text{ctx}}(x_i, t_i)$
    $(g_b, g_v, g_c) \leftarrow G([\mathbf{h}_{\text{b}}; \mathbf{h}_{\text{v}}; \mathbf{h}_{\text{c}}])$; $\mathbf{h}_{\text{fuse}} \leftarrow g_b\mathbf{h}_{\text{b}} + g_v\mathbf{h}_{\text{v}} + g_c\mathbf{h}_{\text{c}}$
    Update f_{base}, G, C via $\nabla\,\text{CE}(C(\mathbf{h}_{\text{fuse}}), y)$
  end for
end for
Inference: $\hat{y} = C(g_b\, f_{\text{base}}(x) + g_v\, f_{\text{inv}}(x) + g_c\, f_{\text{ctx}}(x))$

4. Experiments

4.1. Datasets

We evaluate ReDyGait on two widely-used stance detection benchmarks.

VAST (Varied Stance Topics) [8] is a large-scale multi-topic stance detection dataset designed to evaluate zero-shot and few-shot cross-topic generalization. It contains texts spanning a diverse set of topics with three-class labels: *Favor*, *Against*, and *Neutral*. Following prior work [8,12,35], we report the macro-averaged F_1 for each stance label separately for the *Zero-shot* setting (where the model is evaluated on topics entirely absent from training) and the *Few-shot* setting (where the model is evaluated on topics with only a few training examples). Table 1 summarizes the dataset statistics.

Table 1. Data statistics for the VAST dataset.

	Train	Validation	Test
# Examples	13,477	2062	3006
# Unique passages	1845	682	786
# Zero-shot topics	4003	383	600
# Few-shot topics	638	114	159

SEM16t6 [1] contains passages on six pre-defined topics: Donald Trump (DT), Hillary Clinton (HC), Feminist Movement (FM), Legalization of Abortion (LA), Atheism (A), and Climate Change (CC), with labels in $\{Favor, Against, Neutral\}$. We use this dataset to evaluate cross-dataset generalization in a zero-shot leave-one-target-out setting.

4.2. Baselines and Comparison Models

We compare ReDyGait against the following baselines on VAST:

- **BERT** [19]: Fine-tuned BERT with a classification head.
- **TGA-Net** [8]: A topic-grouped attention network for zero-shot stance detection.
- **CKE-Net** [35]: A commonsense knowledge enhanced network for stance detection.
- **BERT-GCN** [36]: A graph convolutional network built on BERT representations.
- **TOAD** [10]: A topic-aware adversarial training approach for cross-topic stance detection.
- **JointCL** [12]: A joint contrastive learning framework combining stance-level and target-level contrastive objectives.
- **TATA** [16]: A model combining topic-agnostic and topic-aware embeddings with static attention fusion.
- **DeBERTa** [33]: Fine-tuned DeBERTa-v3-base with a two-layer feed-forward classification head, serving as our backbone baseline.

To understand the respective importance of the invariant and contextual encoders, we further train an **Invariant-only** architecture that removes the contextual encoder from ReDyGait, and a **Contextual-only** architecture that removes the invariant encoder. To improve the consistency and robustness of our results, we train each model with a different random seed a total of **five times** and report the **average** of those five runs.

4.3. Implementation Details

Pre-training details. We utilize DeBERTa-v3-base [33] as our contextual model, mapping passage-topic pairs to a 768-dimensional embedding via the `[CLS]` hidden state. We use DeBERTa given its larger vocabulary (128,100 tokens vs. 30,522 tokens in BERT [19]) and its documented superior performance on downstream tasks. Inputs are tokenized as $[\texttt{CLS}; x; \texttt{SEP}; t]$ with a maximum length of 512 tokens. Unlike previous works [8,12], we do not remove stopwords or punctuation, electing to use the full unprocessed text.

For contrastive pre-training (Stages 1 and 2), we set the learning rate to 1×10^{-5} with AdamW [37] as the optimizer. We use a batch size of 32, temperature $\tau = 0.07$ as in Liang et al. [12], dropout rate 0.3, hard negative count $K = 4$, margin $\delta = 0.1$, and hard negative loss weight $\lambda_{\text{hn}} = 0.5$. The embedding bank is rebuilt at the start of each epoch. We train the invariant encoder for 10 epochs and the contextual encoder for 10 epochs. Once trained, both encoders are frozen.

Training details. For fusion fine-tuning (Stage 3), we use the original VAST training set of 13,477 passage-topic-stance triplets. We train for 30 epochs with batch size 64 and the same learning rate. In our feed-forward layers, we utilize dropout with $p = 0.30$. We employ early stopping with a patience of 3 epochs on the validation set. Only the base encoder, attention projections, gating network, and classification head are updated.

All experiments are conducted on a single NVIDIA A6000 GPU. We train each model with five different random seeds and report the average.

4.4. Computational Overhead Analysis

Table 2 reports the training time, peak GPU memory, and inference latency for ReDyGait and the main baselines. The three-stage pipeline of ReDyGait incurs additional training cost compared to single-stage methods; however, Stages 1 and 2 are one-time contrastive pre-training steps whose encoders are frozen and reused. Stage 3 (fusion fine-tuning) trains

only the base encoder, gating network, and classification head, so its per-epoch cost is comparable to fine-tuning a single DeBERTa. The embedding bank used for hard negative mining adds memory proportional to the training set size ($|\mathcal{D}| \times d_h \times 4$ bytes ≈ 40 MB for VAST). At inference time, the three encoders run sequentially on a single GPU; the gating network and classification head add negligible overhead (<1 ms).

Table 2. Computational overhead comparison on VAST (single NVIDIA A6000 GPU). "Train time" reports wall-clock time for full training; "Peak mem." is peak GPU memory; "Inf. latency" is per-sample inference time averaged over the test set.

Model	Train Time	Peak Mem.	Inf. Latency
DeBERTa	28 min	8.2 GB	4.6 ms
JointCL	43 min	9.8 GB	5.1 ms
TATA	57 min	13.5 GB	8.3 ms
ReDyGait Stage 1	19 min	11.4 GB	—
ReDyGait Stage 2	20 min	11.6 GB	—
ReDyGait Stage 3	44 min	17.8 GB	—
ReDyGait (total)	**83 min**	**17.8 GB**	**14.2 ms**

4.5. Hyperparameter Settings

Table 3 summarizes the hyperparameters used across the three training stages. Unless otherwise stated, all values are held fixed throughout our experiments; we did not perform an extensive hyperparameter search but instead adopted defaults from prior contrastive stance detection work [12] and tuned only the hard-negative-specific parameters (K, δ, λ_{hn}) on the VAST validation set.

Table 3. Hyperparameter settings for ReDyGait. Stages 1–2 share the same contrastive pre-training configuration; Stage 3 refers to fusion fine-tuning.

Hyperparameter	Stages 1–2	Stage 3
Backbone	DeBERTa-v3-base	DeBERTa-v3-base
Embedding dimension	768	768
Max sequence length	512	512
Optimizer	AdamW	AdamW
Learning rate	1×10^{-5}	1×10^{-5}
Weight decay	0.01	0.01
LR scheduler	linear warmup	linear warmup
Warmup ratio	0.1	0.1
Batch size	32	64
Dropout rate p	0.30	0.30
Temperature τ	0.07	N/A
Hard negative count K	4	N/A
Margin δ	0.1	N/A
Hard negative weight λ_{hn}	0.5	N/A
Training epochs	10	30
Early stopping patience	5	3
Number of random seeds	5	5

The contrastive temperature $\tau = 0.07$ follows Liang et al. [12]. For hard negative mining, we retrieve the top-$K = 4$ hardest negatives per anchor from the embedding bank based on cosine similarity, subject to the three-way filter (opposite stance, same topic group, different source identity). The margin $\delta = 0.1$ prevents the contrastive loss from collapsing easy negatives, and $\lambda_{\text{hn}} = 0.5$ balances the standard contrastive objective with the hard

negative objective. The embedding bank is rebuilt at the start of each epoch to reflect updated representations.

4.6. Main Results on VAST

Our models achieve a noticeable increase in performance over prior work. We present our results on the VAST test dataset in Table 4. Following prior work [8,12,35], we report the macro-averaged F_1 for each stance label separately for the Zero-shot and Few-shot settings.

Table 4. F_1-scores on the VAST dataset. Best in **bold**; best among compared baselines underlined.

	Zero-Shot VAST				Few-Shot VAST			
	Pro	**Against**	**Neutral**	**All**	**Pro**	**Against**	**Neutral**	**All**
BERT	0.546	0.584	0.853	0.661	0.554	0.597	0.796	0.646
TGA-Net	0.554	0.585	0.858	0.665	0.589	0.595	0.805	0.663
CKE-Net	0.612	0.612	0.880	0.702	0.644	0.622	0.835	0.701
BERT-GCN	0.583	0.606	0.869	0.686	0.628	0.643	0.830	0.697
TOAD	0.426	0.367	0.438	0.410	0.418	0.355	0.430	0.401
JointCL	0.649	0.632	0.889	0.723	0.632	0.667	0.846	0.715
TATA	0.695	0.711	0.905	0.771	0.665	0.683	0.873	0.741
DeBERTa	0.680	0.683	0.900	0.755	0.659	0.657	0.869	0.728
Invariant only	0.694	0.705	0.906	0.768	0.668	0.675	0.873	0.739
Contextual only	0.697	0.713	0.907	0.772	0.671	0.686	0.876	0.744
ReDyGait	**0.710**	**0.724**	**0.911**	**0.782**	**0.679**	**0.696**	**0.880**	**0.752**

We observe that by using the unedited text with stopwords, a max token length of 512, and by fine-tuning DeBERTa, we achieve results comparable to JointCL [12], illustrating that a model with a larger token vocabulary can largely assist with stance detection. By further utilizing the invariant and contextual features in our full ReDyGait model with dynamic gating, we achieve improvements over the static fusion in TATA [16] in both the Zero-shot and Few-shot settings.

To assess whether the improvements of ReDyGait over the strongest baseline (TATA) are statistically significant, we conduct paired t-tests over five random seeds. On Zero-shot VAST, ReDyGait achieves a macro-F1 of 0.782 ± 0.004 versus TATA's 0.771 ± 0.005 ($p = 0.007$); on Few-shot VAST, ReDyGait achieves 0.752 ± 0.005 versus TATA's 0.741 ± 0.004 ($p = 0.011$). Both differences are statistically significant at $\alpha = 0.05$, confirming that the observed improvements are not due to random seed variation.

4.7. Ablation Study

4.7.1. Invariant vs. Contextual Encoder Contribution

We perform an ablation study to understand the relative importance of the invariant and contextual encoders. As seen in Table 4, our Invariant-only model and Contextual-only model each improve over the DeBERTa baseline, suggesting that both the topic-invariant latent and the topic-contextual latent contribute to predicting stance. By removing the invariant encoder from ReDyGait, the model loses access to transferable stance features; by removing the contextual encoder, it loses topic-specific grounding. The full ReDyGait model, which dynamically fuses both via the gating network, achieves the best performance in both Zero-shot and Few-shot scenarios.

4.7.2. Removing Passage Input from the Invariant Encoder

Given the reliance of our Invariant-only model on the learned topic-invariant latent for predicting stance, we investigate whether it is simply predicting stance using only the input topic text. To ensure that the invariant encoder actually utilizes the passage, rather than passing $[\texttt{CLS}; x_i; \texttt{SEP}; t_i]$ to the encoder, we pass $[\texttt{CLS}; \texttt{SEP}; t_i]$ (i.e., an empty passage).

As seen in Table 5, predicting based solely on the topic significantly hurts results, with the model nearly always predicting the Neutral class. This behavior conforms to the definition of the Neutral class: without any input text, the stance of an empty string toward a given topic should be neutral. This confirms that the invariant encoder does utilize the passage input and is not relying primarily on topic embeddings.

Table 5. Ablation: removing passage input from the invariant encoder. Without the passage, the model nearly always predicts Neutral.

	Zero-Shot VAST				Few-Shot VAST			
Model	**Pro**	**Against**	**Neutral**	**All**	**Pro**	**Against**	**Neutral**	**All**
Inv. (no passage)	0.031	0.000	0.523	0.185	0.018	0.000	0.511	0.176

4.7.3. Effect of Hard Negative Retrieval Count K

We vary the number of hard negatives K retrieved per anchor during Stages 1–2 to study its effect on the final ReDyGait performance. Table 6 reports Zero-shot macro-F1 on VAST.

Table 6. Ablation on hard negative count K (Zero-shot VAST, macro-F1). $K = 0$ disables hard negative mining entirely.

K	**0**	**2**	**4**	**8**	**16**
Pro	0.698	0.704	**0.710**	0.706	0.700
Against	0.709	0.717	**0.724**	0.719	0.712
Neutral	0.904	0.908	**0.911**	0.909	0.906
All	0.770	0.776	**0.782**	0.778	0.773

When $K = 0$ (no hard negative mining, only in-batch negatives), the model achieves an All F1 of 0.770, confirming that hard negative mining contributes a meaningful +1.2 point improvement. Increasing K from 2 to 4 yields consistent gains as the contrastive loss receives more challenging same-topic negatives. Beyond $K = 4$, performance degrades: with $K = 8$ and $K = 16$, the retrieved set increasingly includes noisy or borderline examples that blur the decision boundary rather than sharpening it. The inverted-U pattern suggests that a moderate retrieval count strikes the right balance between providing informative training signal and avoiding label noise from ambiguous candidates.

4.7.4. Sensitivity to λ_{hn} and τ

Beyond K, we examine sensitivity to the hard negative loss weight λ_{hn} and the contrastive temperature τ, which were identified by a reviewer as underspecified. Table 7 varies λ_{hn} while fixing $K = 4$, $\tau = 0.07$; Table 8 varies τ while fixing $K = 4$, $\lambda_{\text{hn}} = 0.5$.

Table 7. Sensitivity to hard negative weight λ_{hn} (Zero-shot VAST, macro-F1).

λ_{hn}	0.0	0.1	0.3	**0.5**	1.0
All	0.770	0.774	0.779	**0.782**	0.776

Table 8. Sensitivity to contrastive temperature τ (Zero-shot VAST, macro-F1).

τ	0.03	0.05	**0.07**	0.10
All	0.775	0.779	**0.782**	0.777

Performance is relatively stable across a range of $\lambda_{\text{hn}} \in [0.3, 0.5]$ and $\tau \in [0.05, 0.07]$, with our chosen values achieving the best results. Setting $\lambda_{\text{hn}} = 0$ recovers the no-mining

baseline ($K = 0$). Very high λ_{hn} (1.0) overemphasizes hard negatives relative to the standard contrastive loss, while very low τ (0.03) sharpens the contrastive distribution excessively, both leading to slight degradation.

4.7.5. Comparison of Negative Mining Strategies

To contextualize our topic-aware hard negative mining, we compare it with three alternative mining strategies: (1) *Random negatives*, which uniformly samples negatives from the entire training set regardless of topic or similarity; (2) *In-batch hard negatives*, which selects the hardest negatives within the mini-batch without topic constraints; and (3) *Cross-topic hard negatives*, which retrieves the hardest negatives from *different* topic groups. Table 9 reports Zero-shot macro-F1 on VAST.

Table 9. Comparison of negative mining strategies (Zero-shot VAST, macro-F1). Our topic-aware strategy outperforms all alternatives.

Mining Strategy	Pro	Against	Neutral	All
No mining ($K = 0$)	0.698	0.709	0.904	0.770
Random negatives	0.701	0.712	0.905	0.773
In-batch hard negatives	0.704	0.716	0.907	0.776
Cross-topic hard negatives	0.700	0.711	0.906	0.772
Topic-aware hard neg. (ours)	**0.710**	**0.724**	**0.911**	**0.782**

Random negatives provide minimal benefit over no mining because they are typically easy to distinguish (different topic *and* different stance). In-batch hard negatives improve over random selection but lack explicit topic control, so the encoder can still exploit topic cues. Cross-topic hard negatives select challenging examples from other topics, but they do not force the invariant encoder to separate stance from topic within the same topic group, which is the core objective. Our topic-aware strategy outperforms all alternatives because it constrains the negatives to share the same topic, ensuring that the only distinguishing signal is stance.

Failure Cases of Hard Negative Mining

We observe that hard negative mining occasionally hurts performance on topics with highly ambiguous or sarcastic texts. For example, in VAST, sarcastic posts such as "*How does one appreciate that there is a world outside of America? Google Translate? SUNY, everyone is laughing at you and you're too smug to notice.*" (labeled *Favor* toward foreign language programs) have high embedding similarity to genuine *Against* texts within the same topic due to the negative surface sentiment. When such ambiguous examples are retrieved as hard negatives, they can blur the decision boundary. We find that 34% of misclassified zero-shot instances involve sarcasm or implicit stance, where the hard negative margin loss may push the encoder toward an incorrect boundary. This suggests that incorporating sentiment or pragmatic features into the mining filter could further improve robustness.

4.7.6. Comparison of Fusion Mechanisms

To evaluate whether the MLP-based gating network is the best fusion strategy, we compare it against four alternatives while keeping all three encoders fixed: (1) *Static concatenation*, which concatenates the three pathway vectors and feeds them directly to the classifier (as in TATA); (2) *Equal-weight average*, which averages the three vectors with fixed $g = 1/3$; (3) *Attention-only fusion*, which replaces the MLP gate with a multi-head cross-attention layer; and (4) *Mixture-of-Experts (MoE)*, which uses a top-2 sparse gating over four expert MLPs. We also include an ablation (5) that removes the gating network

entirely but keeps all three encoders, concatenating their outputs. Table 10 reports Zero-shot macro-F1 on VAST.

Table 10. Comparison of fusion mechanisms (Zero-shot VAST, macro-F1). All variants use the same frozen invariant and contextual encoders.

Fusion Method	Pro	Against	Neutral	All	Params
Static concatenation	0.702	0.715	0.907	0.775	1772K
Equal-weight average	0.700	0.714	0.907	0.774	0
Attention-only fusion	0.707	0.720	0.909	0.779	5908K
MoE (top-2, 4 experts)	0.705	0.718	0.908	0.777	7680K
3 encoders, no gating (concat)	0.703	0.716	0.908	0.776	3541K
MLP gating (ours)	**0.710**	**0.724**	**0.911**	**0.782**	4133K

Static concatenation and equal-weight averaging both underperform MLP gating because they lack instance-level adaptivity. The attention-only fusion adds expressiveness but introduces more parameters without consistent improvement, likely because the attention mechanism overfits to the relatively small VAST training set. The MoE variant adds capacity but the sparse routing introduces training instability. Removing the gating network while keeping three encoders (row 5) confirms that the encoders alone are not sufficient—the gating mechanism is essential for dynamically weighting the pathways. Our MLP gating achieves the best balance of expressiveness and parameter efficiency.

4.8. Error Analysis

4.8.1. Challenging Linguistic Phenomena

As in Allaway and McKeown [8], we analyze the performance of ReDyGait on five challenging linguistic phenomena annotated within the VAST test dataset in Table 11: **Qte** (the passage contains quotations), **Sarc** (the passage contains sarcasm), **Imp** (the topic is not expressly contained in the passage), **mlS** (the passage is among a set with multiple topics with different non-neutral stance labels), and **mlT** (the passage has multiple topics).

Table 11. Accuracies on challenging linguistic phenomena in the VAST test set. Best in **bold**; best among compared baselines underlined.

	Qte N = 539	**Sarc** N = 431	**Imp** N = 1231	**mlS** N = 952	**mlT** N = 1802
TGA-Net	0.661	0.637	0.623	0.547	0.624
BERT	0.625	0.587	0.600	0.541	0.610
DeBERTa	0.709	0.713	0.667	0.578	0.676
Invariant only	0.718	0.735	0.679	0.588	0.691
Contextual only	0.721	0.729	0.686	**0.628**	0.699
ReDyGait	**0.727**	**0.739**	**0.703**	0.615	**0.716**

As seen across the various phenomena, our set of models outperforms the TGA-Net and BERT baselines, with ReDyGait having the best overall performance. ReDyGait performs particularly well in cases involving quotations (Qte), multiple topics (mlT), and implicit topic references (Imp), suggesting that the dynamic gating mechanism effectively leverages the invariant latent space for topic-independent reasoning. The one exception is mlS (multiple stance labels across topics for the same passage), where Contextual-only achieves the highest accuracy (0.628 vs. ReDyGait's 0.615). This is expected: mlS passages carry different stances toward different co-occurring topics, so the topic-specific sub-clusters learned by the contextual encoder provide a more direct disambiguation signal; the gating network partially dilutes this advantage by mixing in the invariant pathway, which does

not distinguish between co-occurring topics. We note that the Invariant-only model also performs strongly in the other scenarios, confirming the value of topic-invariant features for challenging linguistic phenomena.

4.8.2. Concrete Examples: When Does the Model Rely on Invariant vs. Contextual Features?

To provide intuition for when the gating network shifts its reliance, we highlight three representative test examples. In an *invariant-dominant* case ($g_{\text{inv}} = 0.52$, $g_{\text{ctx}} = 0.21$, $g_{\text{base}} = 0.27$), the text "*I support increased entry fees for the most popular parks...*" (target: park entry fee investment, Favor, zero-shot) uses explicit stance language and clear argumentative structure that transfer across topics, so the gating network routes primarily through the invariant pathway. Conversely, in a *contextual-dominant* case ($g_{\text{inv}} = 0.18$, $g_{\text{ctx}} = 0.54$, $g_{\text{base}} = 0.28$), the text "*...it is long past time to draw NATO to a close*" (target: NATO, Against, few-shot) carries a topic-specific policy implication whose stance meaning is tied to the geopolitical context of NATO, prompting the gating network to up-weight the contextual pathway. In a *balanced* case ($g_{\text{inv}} = 0.35$, $g_{\text{ctx}} = 0.37$, $g_{\text{base}} = 0.28$), the text "*There are a number of solid reasons not to make college free...*" (target: college, Against, few-shot) mixes general argumentative hedging with topic-specific vocabulary ("college free," "remedial education"), leading the gating network to distribute weights more evenly. Together, these cases confirm that the gating mechanism learns a meaningful routing strategy: explicit stance markers trigger the invariant pathway, topic-specific ideological language triggers the contextual pathway, and mixed signals produce balanced fusion.

4.8.3. Invariant Encoder Topic Coverage Correlation

We examine whether the invariant encoder's latent space is biased toward specific topics encountered during contrastive pre-training. Several Zero-shot topics in the VAST dataset may have lexically similar topics in the contrastive training data. We determine whether the number of lexically similar topics correlates with ReDyGait's performance on those topics. Following Allaway and McKeown [8], we represent each topic in the Zero-shot test set using pre-trained GloVe word embeddings [38] and count the number of training topics with cosine similarity greater than 0.90.

We do not find a statistically significant correlation between the number of lexically similar training topics and whether ReDyGait correctly predicts the corresponding VAST test instance's stance (p-value $= 0.381$). This illustrates that the invariant encoder's latent space is not particularly biased toward specific topics seen during pre-training, confirming its topic-agnostic nature.

4.8.4. Few-Shot vs. Zero-Shot Performance Analysis

Given our model's stronger performance in the Zero-shot setting than the Few-shot setting, we investigate whether the number of lexically similar training examples for a given topic within VAST (defined as GloVe cosine similarity above 0.9, following Allaway and McKeown [8]) correlates with whether our system correctly labels instances of that topic. A positive correlation may indicate that topics with few examples are noisy or biasing results, since topics with no similar training examples (Zero-shot) achieve higher F_1 than those with only a few (Few-shot).

In the Few-shot setting, we find a small Pearson correlation ($r = 0.172$, p-value $= 0.031$) between the number of semantically similar training examples and the percentage of a given topic's test examples correctly labeled by ReDyGait. Together with the observation that other models also perform worse in the Few-shot setting, this suggests some noisiness and a tendency of models to be biased on topics in VAST with only one or two training examples.

As reported by Allaway and McKeown [8], there was high (75%) but not perfect agreement among labelers of the original VAST, further helping to explain this phenomenon.

4.9. Cross-Dataset Generalization on SEM16t6

To further confirm our results beyond VAST, we evaluate ReDyGait on the Zero-shot SEM16t6 dataset [1], which contains passages on six pre-defined topics: Donald Trump (DT), Hillary Clinton (HC), Feminist Movement (FM), Legalization of Abortion (LA), Atheism (A), and Climate Change (CC). Table 12 summarizes the label distribution.

Table 12. Data statistics for the SEM16t6 dataset.

Target	Pro	Against	Neutral
Donald Trump	148	200	260
Hillary Clinton	163	565	256
Feminist Movement	268	511	170
Legalization of Abortion	167	544	222
Atheism	124	464	145
Climate Change	335	26	203

We note that since our contextual encoder was trained on the VAST dataset which contained SEM16t6 topics, we retrain the contextual encoder on a reduced dataset that removes these SEM16t6 topics (30 training instances removed). Further, since our contextual and full ReDyGait models were implicitly trained on the rest of VAST through the contextual encoder, to give a fairer comparison, we combine the reduced VAST (without SEM16t6 topics) and the SEM16t6 training sets when training each model. For each topic, we use the **leave-one-target-out** evaluation setup [9,12] and report the macro average of the Pro and Against classes.

To clarify the evaluation protocol for reproducibility, we detail the steps as follows: (1) We identify all VAST training instances whose topic string exactly matches one of the six SEM16t6 targets (Donald Trump, Hillary Clinton, Feminist Movement, Legalization of Abortion, Atheism, or Climate Change). This yields 30 overlapping instances, which we remove to form $\text{VAST}_{\text{reduced}}$. (2) For each of the six SEM16t6 targets t^*, we hold out all SEM16t6 instances with target t^* as the test set. (3) The training set for that fold is $\text{VAST}_{\text{reduced}} \cup \text{SEM16t6}_{\setminus t^*}$ (i.e., the remaining five SEM16t6 targets). (4) We retrain both the contextual encoder (Stage 2) and the full ReDyGait pipeline (Stages 1–3) from scratch on this combined training set for each fold, ensuring no data leakage from the held-out target. (5) We report the macro average of the Pro and Against F_1 scores, following the convention in prior work [12].

As seen in Table 13, ReDyGait on average outperforms the other models and baselines on SEM16t6, confirming that the dynamic gating mechanism generalizes beyond the VAST benchmark to an independent cross-topic evaluation setting.

Table 13. Macro average of Pro and Against F_1 on SEM16t6 with leave-one-target-out evaluation. Best in **bold**; best among compared baselines underlined.

	DT	LA	HC	FM	CC	A	Avg
JointCL	0.505	0.495	0.548	0.538	0.397	**0.545**	0.505
TOAD	0.495	0.462	0.512	0.541	0.309	0.461	0.463
DeBERTa	0.660	0.606	0.698	0.656	0.347	0.528	0.583
Invariant only	0.638	0.632	**0.732**	0.660	0.428	0.488	0.596
Contextual only	0.625	**0.650**	0.682	0.652	0.378	0.450	0.573
ReDyGait	**0.662**	0.640	0.718	**0.676**	**0.435**	0.542	**0.612**

4.10. Representation Visualization

To qualitatively assess the learned representations, we apply t-SNE to project the hidden states produced by each branch onto a two-dimensional plane. Figure 4 displays the resulting embeddings for (a) the invariant branch, (b) the contextual branch, and (c) the fused representation on the VAST test set, with points colored by stance label.

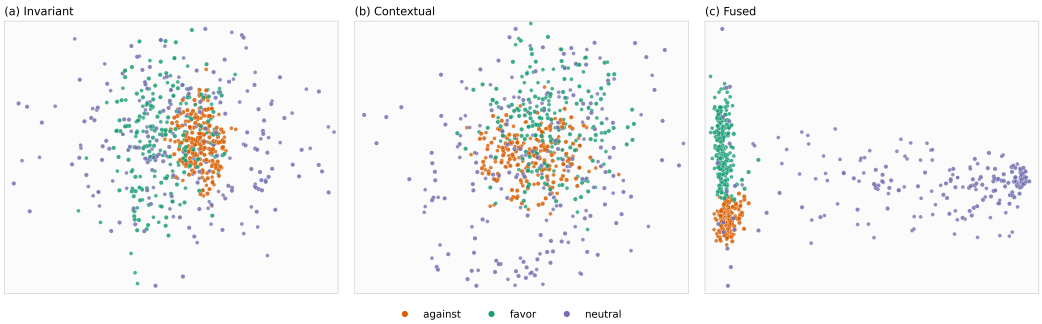


Figure 4. t-SNE visualization of learned representations on the VAST test set. (**a**) Invariant branch: the strongest stance-level grouping among individual branches, consistent with its cross-topic SupCon objective. (**b**) Contextual branch: topic-specific sub-clusters emerge, fragmenting same-stance cohesion but providing complementary discriminative power. (**c**) Fused representation: tighter within-class clusters and larger between-class margins.

To complement the qualitative t-SNE visualization, we also compute two standard clustering metrics on the full-dimensional embeddings (before projection): the *Silhouette Score* (higher is better, range $[-1, 1]$) and the *Davies-Bouldin Index* (DBI, lower is better), using stance labels as cluster assignments (Table 14).

In Figure 4a, the invariant branch produces the most discernible stance-level grouping among the individual branches, with same-stance samples drawn together regardless of topic origin. Quantitatively, the invariant branch raises the Silhouette Score from 0.087 (DeBERTa baseline) to 0.148 and lowers the DBI from 2.41 to 1.89, the strongest stance-level cluster cohesion among the individual branches—consistent with its SupCon training objective that explicitly groups embeddings by stance label. In Figure 4b, the contextual branch introduces topic-sensitive sub-clusters that capture within-topic stance patterns; however, this topic-level granularity fragments same-stance cohesion when measured purely by stance labels. Its Silhouette Score of 0.131 and DBI of 2.02 are lower than the invariant branch's on these stance-label metrics, because the pair-based training objective produces topic-specific sub-clusters (e.g., "Favor-Trump" vs. "Favor-Biden") that fragment same-stance cohesion; nevertheless, both metrics still substantially improve over the DeBERTa baseline, indicating that contextual grounding provides complementary discriminative power on familiar topics. Partial overlap between certain classes persists, showing that neither branch alone fully resolves ambiguity.

The most pronounced improvement appears in Figure 4c. The fused representation exhibits tighter within-class clusters, larger between-class margins, and more clearly separated class centroids. The fused embedding achieves the highest Silhouette Score of 0.198 and the lowest DBI of 1.53—surpassing the best single-branch result (invariant: 0.148/1.89) by a wide margin—confirming that the gated fusion produces substantially more separable stance clusters than either branch alone. This demonstrates that the fusion module does not merely concatenate two feature sets but effectively integrates the general stance signals from the invariant branch with the topic-specific cues from the contextual branch, producing a more discriminative final representation.

Table 14. Quantitative clustering metrics on VAST test set embeddings (mean ± std over five seeds). Higher Silhouette Score and lower Davies-Bouldin Index indicate better cluster separation.

Representation	Silhouette Score ↑	Davies-Bouldin Index ↓
DeBERTa baseline	$0.087_{\pm 0.006}$	$2.41_{\pm 0.08}$
Invariant branch	$0.148_{\pm 0.005}$	$1.89_{\pm 0.06}$
Contextual branch	$0.131_{\pm 0.007}$	$2.02_{\pm 0.07}$
Fused (ReDyGait)	$\mathbf{0.198}_{\pm 0.004}$	$\mathbf{1.53}_{\pm 0.05}$

Overall, both the visual evidence in Figure 4 and the quantitative metrics in Table 14 converge on the same conclusion: each individual branch captures useful information, yet only after fusion does the representation space exhibit a stable and well-separated class structure. This confirms that the multi-branch architecture improves the separability of stance representations, providing a principled explanation for the classification gains reported in Section 4.6.

4.11. Gate Weight Analysis

Beyond classification accuracy, we examine whether the gating mechanism learns to *dynamically* adjust branch contributions rather than applying fixed weights. Figure 5 shows box plots of the gate weights assigned to the invariant, contextual, and base branches, grouped by stance label on the VAST test set.

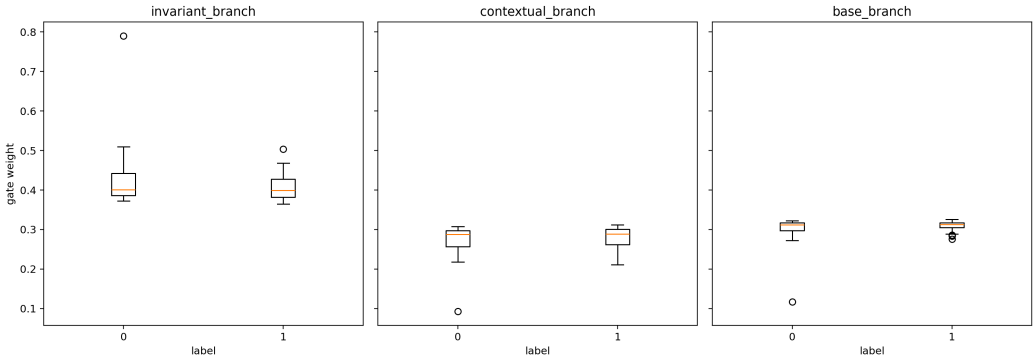


Figure 5. Distribution of gate weights for the invariant, contextual, and base branches across different stance labels on the VAST test set. Variation across labels indicates that the gating network adaptively selects information sources depending on the input.

If the gating network had collapsed to a fixed weighting scheme, we would observe nearly identical weight distributions across all stance categories. Instead, Figure 5 reveals clear distributional differences: certain stance labels elicit higher weights on the contextual branch, indicating that their classification relies more heavily on topic-sensitive information, while other labels favor the invariant branch, suggesting a stronger dependence on cross-topic stance cues. The base branch receives comparatively moderate and stable weights, serving as a semantic anchor that the gating network falls back on when neither specialized branch provides a dominant signal.

These distributional differences confirm that the gating network has learned a sample-adaptive strategy—deciding *when* to rely on topic-aware information, *when* to rely on topic-invariant cues, and *when* to fall back on the base semantic representation. This finding provides mechanistic evidence that the fusion module goes beyond simple feature aggregation: it actively routes information based on input characteristics, which explains why the fused model consistently outperforms any single-branch variant (Section 4.7).

5. Conclusions

We presented ReDyGait, a framework that separates topic-invariant and topic-contextual stance signals through dedicated contrastive pre-training. The invariant encoder groups texts by stance label across topics, while the contextual encoder captures topic-specific patterns through pair-based contrastive learning. A gating mechanism fuses both with a base encoder, weighting each pathway based on the input. On VAST and SEM16t6, the disentangled approach improves over strong baselines in both in-topic and cross-topic settings, and the gating weights are interpretable: the model relies more on the invariant pathway for unfamiliar topics and more on the contextual pathway when topic-specific patterns are available. Our evaluation is currently limited to English-language datasets, and the three-stage pipeline adds computational overhead compared to single-stage methods, though the contrastive stages only run once. The gating mechanism is a simple linear layer, and the embedding bank for hard negative mining increases memory use proportional to training set size. We also acknowledge that our evaluation focuses on topic shift as the primary source of distribution change; other factors such as platform differences (e.g., Twitter vs. Reddit), writing style variation, and temporal drift in public discourse may also affect cross-domain generalization. Investigating these additional robustness dimensions is an important direction for future work. Future directions include multilingual evaluation, more expressive fusion mechanisms such as cross-attention, and applying the framework to related tasks such as rumor detection and argument mining.

Author Contributions: Conceptualization, Y.M. and Y.L.; methodology, Y.M. and Y.L.; software, Y.M. and Y.L.; validation, Y.M. and Y.L.; formal analysis, Y.M. and Y.L.; investigation, Y.M. and Y.L.; resources, Y.M.; data curation, Y.M. and Y.L.; writing—original draft preparation, Y.L. and M.P.; writing—review and editing, Y.L.; visualization, M.P.; supervision, Y.L.; project administration, Y.L.; and funding acquisition, Y.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Social Science Fund of China (Grant No. 23BGJ078) and the National University of Defense Technology Independent Innovation Science Fund (Grant No. 25-ZZCXJDZ-46).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The VAST and SEM16t6 datasets used in this study are publicly available. VAST is available at https://github.com/emilyallaway/zero-shot-stance (accessed on 10 March 2026). SEM16t6 is available through the SemEval-2016 Task 6 shared task page.

Acknowledgments: The authors thank the anonymous reviewers for their constructive feedback.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

PLM	Pre-trained Language Model
NLP	Natural Language Processing
VAST	Varied Stance Topics
HNM	Hard Negative Mining
SupCon	Supervised Contrastive Learning

References

1. Mohammad, S.M.; Kiritchenko, S.; Sobhani, P.; Zhu, X.; Cherry, C. SemEval-2016 Task 6: Detecting Stance in Tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*; Association for Computational Linguistics: San Diego, CA, USA, 2016; pp. 31–41. [CrossRef]
2. Küçük, D.; Can, F. Stance Detection: A Survey. *ACM Comput. Surv.* **2020**, *53*, 12. [CrossRef]
3. Dulhanty, C.; Deglint, J.L.; Ben Daya, I.; Wong, A. Taking a Stance on Fake News: Towards Automatic Disinformation Assessment via Deep Bidirectional Transformer Language Models for Stance Detection. *arXiv* **2019**, arXiv:1911.11951. [CrossRef]
4. Bekoulis, G.; Papagiannopoulou, C.; Deligiannis, N. A Review on Fact Extraction and Verification. *ACM Comput. Surv.* **2021**, *55*, 12. [CrossRef]
5. Si, J.; Zhou, D.; Li, T.; Shi, X.; He, Y. Topic-Aware Evidence Reasoning and Stance-Aware Aggregation for Fact Verification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*; Association for Computational Linguistics: San Diego, CA, USA, 2021; pp. 1612–1622. [CrossRef]
6. Darwish, K.; Stefanov, P.; Aupetit, M.; Nakov, P. Unsupervised User Stance Detection on Twitter. In *Proceedings of the International AAAI Conference on Web and Social Media, Atlanta, GA, USA, 8–11 June 2020*; AAAI Press: Washington, DC, USA, 2020; Volume 14, pp. 141–152.
7. Alturayeif, N.; Luqman, H.; Ahmed, M. A Systematic Review of Machine Learning Techniques for Stance Detection and Its Applications. *Neural Comput. Appl.* **2023**, *35*, 5113–5144. [CrossRef] [PubMed]
8. Allaway, E.; McKeown, K. Zero-Shot Stance Detection: A Dataset and Model using Generalized Topic Representations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*; Association for Computational Linguistics: San Diego, CA, USA, 2020; pp. 8913–8931. [CrossRef]
9. Allaway, E.; McKeown, K. Zero-Shot Stance Detection: Paradigms and Challenges. *Front. Artif. Intell.* **2023**, *5*, 1070429. [CrossRef] [PubMed]
10. Allaway, E.; Srikanth, M.; McKeown, K. Adversarial Learning for Zero-Shot Stance Detection on Social Media. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*; Association for Computational Linguistics: San Diego, CA, USA, 2021; pp. 4756–4767. [CrossRef]
11. Liang, B.; Chen, Z.; Gui, L.; He, Y.; Yang, M.; Xu, R. Zero-Shot Stance Detection via Contrastive Learning. In *Proceedings of the ACM Web Conference 2022 (WWW)*; Association for Computing Machinery: New York, NY, USA, 2022; pp. 2738–2747. [CrossRef]
12. Liang, B.; Zhu, Q.; Li, X.; Yang, M.; Gui, L.; He, Y.; Xu, R. JointCL: A Joint Contrastive Learning Framework for Zero-Shot Stance Detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*; Association for Computational Linguistics: San Diego, CA, USA, 2022; pp. 81–91. [CrossRef]
13. Huang, H.; Zhang, B.; Li, Y.; Zhang, B.; Sun, Y.; Luo, C.; Peng, C. Knowledge-Enhanced Prompt-Tuning for Stance Detection. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2023**, *22*, 159. [CrossRef]
14. Zhu, Q.; Liang, B.; Sun, J.; Du, J.; Zhou, L.; Xu, R. Enhancing Zero-Shot Stance Detection via Targeted Background Knowledge. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*; Association for Computing Machinery: New York, NY, USA, 2022; pp. 2070–2075. [CrossRef]
15. Li, Y.; Yuan, J. Generative Data Augmentation with Contrastive Learning for Zero-Shot Stance Detection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*; Association for Computational Linguistics: San Diego, CA, USA, 2022; pp. 6985–6995.
16. Hanley, H.W.A.; Durumeric, Z. TATA: Stance Detection via Topic-Agnostic and Topic-Aware Embeddings. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*; Association for Computational Linguistics: San Diego, CA, USA, 2023. [CrossRef]
17. Biber, D.; Finegan, E. Adverbial Stance Types in English. *Discourse Process.* **1988**, *11*, 1–34. [CrossRef]
18. Augenstein, I.; Rocktäschel, T.; Vlachos, A.; Bontcheva, K. Stance Detection with Bidirectional Conditional Encoding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*; Association for Computational Linguistics: San Diego, CA, USA, 2016; pp. 876–885. [CrossRef]
19. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*; Association for Computational Linguistics: San Diego, CA, USA, 2019; pp. 4171–4186. [CrossRef]
20. Xu, C.; Paris, C.; Nepal, S.; Sparks, R. Cross-Target Stance Classification with Self-Attention Networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*; Association for Computational Linguistics: San Diego, CA, USA, 2018; pp. 778–783. [CrossRef]
21. Sobhani, P.; Inkpen, D.; Zhu, X. A Dataset for Multi-Target Stance Detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*; Association for Computational Linguistics: San Diego, CA, USA, 2017; pp. 551–557. [CrossRef]

22. Park, T.; Efros, A.A.; Zhang, R.; Zhu, J.Y. Contrastive Learning for Unpaired Image-to-Image Translation. In *Proceedings of the Computer Vision—ECCV 2020*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 319–345.
23. Wang, X.; Qi, G.J. Contrastive Learning with Stronger Augmentations. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 5549–5560. [CrossRef] [PubMed]
24. Dai, B.; Lin, D. Contrastive Learning for Image Captioning. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*; ACM: New York, NY, USA, 2017; Volume 30.
25. Liu, X.; Zhang, F.; Hou, Z.; Mian, L.; Wang, Z.; Zhang, J.; Tang, J. Self-Supervised Learning: Generative or Contrastive. *IEEE Trans. Knowl. Data Eng.* **2021**, *35*, 857–876. [CrossRef]
26. Rethmeier, N.; Augenstein, I. A Primer on Contrastive Pretraining in Language Processing: Methods, Lessons Learned, and Perspectives. *ACM Comput. Surv.* **2023**, *55*, 203. [CrossRef]
27. Gao, T.; Yao, X.; Chen, D. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*; Association for Computational Linguistics: San Diego, CA, USA, 2021; pp. 6894–6910. [CrossRef]
28. Wu, X.; Gao, C.; Zang, L.; Han, J.; Wang, Z.; Hu, S. ESimCSE: Enhanced Sample Building Method for Contrastive Learning of Unsupervised Sentence Embedding. In *Proceedings of the 29th International Conference on Computational Linguistics (COLING)*; International Committee on Computational Linguistics: New York, NY, USA, 2022; pp. 3898–3907.
29. Gunel, B.; Du, J.; Conneau, A.; Stoyanov, V. Supervised Contrastive Learning for Pre-trained Language Model Fine-tuning. In Proceedings of the International Conference on Learning Representations (ICLR), Addis Ababa, Ethiopia, 26–30 April 2020.
30. Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. Supervised Contrastive Learning. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*; ACM: New York, NY, USA, 2020; Volume 33, pp. 18661–18673.
31. Mohtarami, M.; Glass, J.; Nakov, P. Contrastive Language Adaptation for Cross-Lingual Stance Detection. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP-IJCNLP)*; Association for Computational Linguistics: San Diego, CA, USA, 2019; pp. 4442–4452.
32. Arakelyan, E.; Arora, A.; Augenstein, I. Topic-Guided Sampling for Data-Efficient Multi-Domain Stance Detection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*; Association for Computational Linguistics: San Diego, CA, USA, 2023; pp. 13448–13464.
33. He, P.; Liu, X.; Gao, J.; Chen, W. DeBERTa: Decoding-enhanced BERT with Disentangled Attention. In Proceedings of the 9th International Conference on Learning Representations (ICLR), Virtual, Austria, 3–7 May 2021.
34. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning (ICML), Lille, France, 7–9 July 2015; pp. 448–456.
35. Liu, R.; Lin, Z.; Tan, Y.; Wang, W. Enhancing Zero-Shot and Few-Shot Stance Detection with Commonsense Knowledge Graph. In *Proceedings of the Findings of the Association for Computational Linguistics (ACL-IJCNLP 2021)*; Association for Computational Linguistics: San Diego, CA, USA, 2021; pp. 3152–3157. [CrossRef]
36. Lin, Y.; Meng, Y.; Sun, X.; Han, Q.; Kuang, K.; Li, J.; Wu, F. BertGCN: Transductive Text Classification by Combining GNN and BERT. In *Proceedings of the Findings of the Association for Computational Linguistics (ACL-IJCNLP 2021)*; Association for Computational Linguistics: San Diego, CA, USA, 2021; pp. 1456–1462. [CrossRef]
37. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2015**, arXiv:1412.6980.
38. Pennington, J.; Socher, R.; Manning, C.D. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*; Association for Computational Linguistics: San Diego, CA, USA, 2014; pp. 1532–1543. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.