T^{3}RD: Test-Time Training for Rumor Detection on Social Media

Huaiwen Zhang
Inner Mongolia University
huaiwen.zhang@imu.edu.cn

Xinxin Liu
Inner Mongolia University
xinxin.liu@mail.imu.edu.cn

Qing Yang
Inner Mongolia University
yangqing@mail.imu.edu.cn

Yang Yang
Inner Mongolia University
yangyang@mail.imu.edu.cn

Fan Qi*
Tianjin University of Technology
fanqi@mail.tjut.edu.cn.

Shengsheng Qian
Institute of Automation, Chinese Academy of Sciences
shengsheng.qian@nlpr.ia.ac.cn

Changsheng Xu
Institute of Automation, Chinese Academy of Sciences
csxu@nlpr.ia.ac.cn

ABSTRACT

With the increasing number of news uploaded to the internet daily, rumor detection has garnered significant attention in recent years. Existing rumor detection methods excel on familiar topics with sufficient training data (high resource) collected from the same domain. However, when facing emergent events or rumors propagated in different languages, the performance of these models is significantly degraded, due to the lack of training data and prior knowledge (low resource). To tackle this challenge, we introduce the Test-Time Training for Rumor Detection (T^{3}RD) to enhance the performance of rumor detection models on low-resource datasets. Specifically, we introduce self-supervised learning (SSL) as an auxiliary task in the test-time training. It consists of global and local contrastive learning, in which the global contrastive learning focuses on obtaining invariant graph representations and the local one focuses on acquiring invariant node representations. We employ the auxiliary SSL tasks for both the training and test-time training phases to mine the intrinsic traits of test samples and further calibrate the trained model for these test samples. To mitigate the risk of distribution distortion in test-time training, we introduce feature alignment constraints aimed at achieving a balanced synergy between the knowledge derived from the training set and the test samples. The experiments conducted on the two widely used cross-domain datasets demonstrate that the proposed model achieves a new state-of-the-art in performance. Our code is available at https://github.com/social-rumors/T3RD.

CCS CONCEPTS

• **Information systems** → **Social networks**.

*corresponding author

WWW '24, May 13–17, 2024, Singapore, Singapore

ACM ISBN 979-8-4007-0171-9/24/05…$15.00
https://doi.org/10.1145/3589334.3645443

KEYWORDS

Fake News Detection, Test-time Training, Graph Neural Network

ACM Reference Format:
Huaiwen Zhang, Xinxin Liu, Qing Yang, Yang Yang, Fan Qi, Shengsheng Qian, and Changsheng Xu. 2024. T^{3}RD: Test-Time Training for Rumor Detection on Social Media. In *Proceedings of the ACM Web Conference 2024 (WWW '24), May 13–17, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3589334.3645443

1 INTRODUCTION

With the development of the Internet, social media platforms have become a convenient channel for users to access information, express opinions, and communicate. More and more people are keen to participate in discussions about the hot topic on social media and express their opinions. Due to the limited domain expertise and relevant data on the emergent events, many rumors [40] appear. For instance, during the COVID-19 pandemic, a false rumor claiming that "5G technology was spreading the virus" rapidly spread through social media, causing enormous damage, including the burning of multiple base stations and posing a severe threat to public safety.

Existing rumor detection methods [2, 22, 24, 30, 36], generally follow the conventional training-test paradigm of deep learning. The rumor detection model is trained to discern correlations between labels and latent input features based on the training set and then leverages this learned knowledge to classify test samples. Recent approaches [19, 20] model the spreading structure of rumors as graphs to capture structural features. For example, Bian et al. [2] construct propagation and dispersion graphs for rumors and aggregate neighbor features based on reply or retweet relations. Tian et al. [26] jointly model both user and comment propagation networks. However, these approaches are based on a hypothesis that requires the training datasets and test sets to maintain the same distribution. In practice, the daily release of diverse claims across domains by real-world social media platforms, coupled with the urgency of emergent events, complicates the timely acquisition of adequate labeled data. The gap between the source and target domains results in an unsatisfactory detection performance. As shown in Fig. 1, the rumor detection model is trained on the source

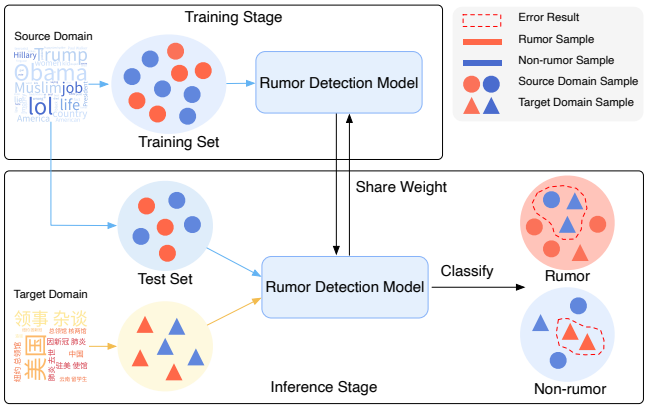

Figure 1: A toy example to illustrate the gap between source and target domain. The rumor detection models trained on the source domain may not adapt well to the target domain. More error-classifying results are observed when directly inferring the target domain samples on the model.

domain, in which the content of training samples is English. When the newly emergent domain is different from the source domain, the model trained on the source domain shows poor performance in the target domain. Directly training the detection model on the source domain and evaluating the newly emergent domain may show unsatisfactory performance. Some studies[12, 34] try to migrate this challenge by involving some test data in training to learn the knowledge of the target domain.

Test-Time Training (TTT) [16, 25] is the method that can mitigate the distribution shift between source and target domain. The essence of TTT is to design an auxiliary task for both the training and test-time training phases to mine the intrinsic traits of target domain samples and further calibrate the trained model. Considering the unique characteristics exhibited by individual samples from the target domain, TTT has great promise to enhance generalization capabilities. However, most TTT approaches are designed for image classification [1, 16, 25] and are not well suited for rumor detection. In rumor detection, limited expertise restricts the acquisition of labeled data in the target domain. The data sampled from the target domain is too limited to represent the distribution of the whole test set, resulting in a severe distortion of the learned feature space during test-time training.

To address these challenges, we present Rumor Detection Test-Time Training (T^3RD), a novel test-time training framework aimed to enhancing the efficacy of rumor detection in emergency scenarios. Our approach revolves around leveraging graph structures to emulate social media conversations while employing a multi-level self-supervised contrastive learning for test-time training on graphs. In order to mitigate the distribution distortion in test-time training, we introduce a features alignment that strives to strike a delicate balance between the knowledge derived from the training set and the distinctive features obtained from test samples. Empirical evaluations conducted on various benchmarks demonstrate the superior performance of our proposed method compared to state-of-the-art approaches. Our contributions are summarized as follows:

- We explore the novel task of test-time training on rumor detection tasks, which is capable of maintaining the training knowledge and effectively capturing the unique characteristics of the test samples.
- We propose a novel model T^3RD for test-time training in rumor detection, in which the auxiliary multi-level self-supervised learning tasks are designed to enhance the generalization capabilities of the model in low-resource scenarios.
- Extensive experiments conducted on two widely used cross-domain datasets demonstrate the effectiveness of our proposed method.

2 RELATED WORK

Rumor Detection. The initial studies [3, 5, 15, 31] in rumor detection primarily concentrate on developing a supervised classifier based on post contents, user profiles, and propagation patterns. With the development of deep learning, neural networks such as convolutional neural networks [32], recurrent neural networks [17, 20, 21, 37], attention mechanisms [7], and transformer models [4, 35, 38], graph neural networks [2, 13] are employed. However, in real-world scenarios, new events from different domains and languages are constantly emerging. Existing rumor detection models face a significant challenge since no sufficient labeled data is available in target domains, i.e., low-resource regimes [10, 12]. In this paper, we introduce a novel test-time training framework for rumor detection, aiming to improve the generalization capabilities of the model in low-resource scenarios.

Test-Time Training. The primary objective of test time training is to adapt models based on test samples in the presence of distributional shifts. For example, Sun et al. [25] introduce the self-supervision test-time training to enhance a model's generalization ability under distribution shift. This is achieved by having the model solve a self-supervised task for test samples. Experimental results in the image domain [1, 6, 25] demonstrate the effectiveness of this framework in reducing the performance gap between training and test sets. There have also been endeavors to integrate test-time training with meta-learning [1], as well as its application in reinforcement learning [8]. Besides these applications, test-time training is also being investigated in the graph domain. For example, Wang et al. [29] present the GT3 test-time training framework to narrow the performance disparity between training and test datasets. In this work, we explore the novel task of test-time training on rumor detection and propose multi-level self-supervised learning tasks to maintain the training knowledge while adopting the underlying traits of test samples from low-resource domains.

3 PRELIMINARY

We mainly focus on the rumor detection task in the low-resource regime. We define the source domain dataset with high-resource as $\mathbb{D}^s = \{E_i^s, y_i^s\}_{i=1}^{N^s}$, where N^s is the total number of events. Each event $E_i^s = \{c_i, R(c_i)\}$ consists of the source claim c_i, its responsive posts $R(c_i)$, and $y_i^s \in \{\text{rumor, non-rumor}\}$ is the label of E_i^s. The responsive posts $R(c_i) = \{c_i, r_1^i, \ldots, r_{|E_i|-1}^i\}$, where r_j^i is the j-th responsive text, and $|E_i| - 1$ is the total number of responsive posts.

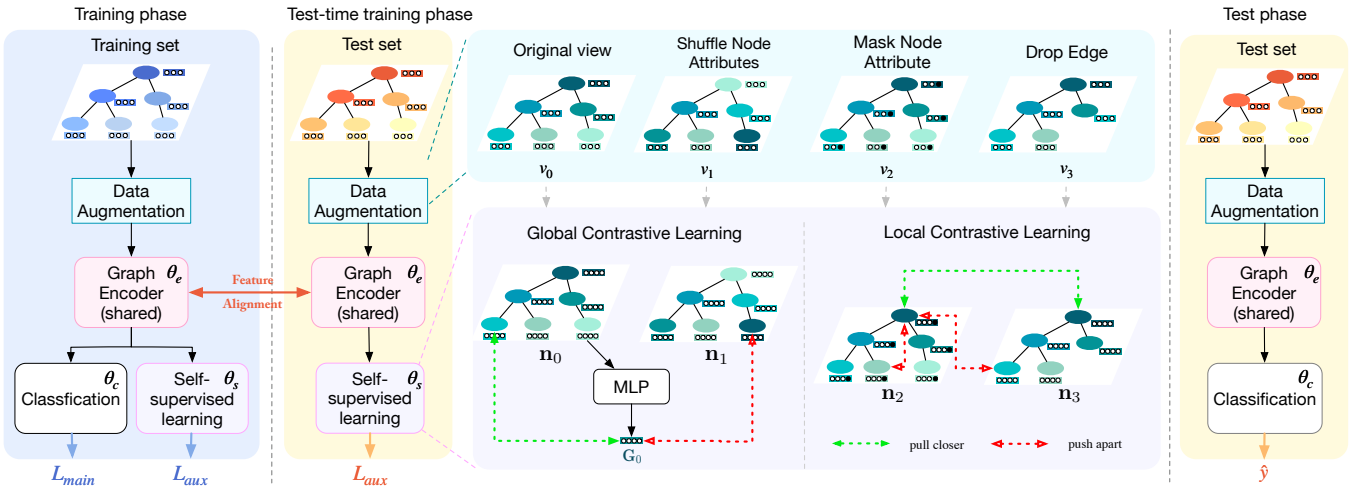

Figure 2: The overall architecture of T^{3}RD. The shared graph encoder extracts the feature representations of the input graph. The blue L_{main} and L_{aux} represent the losses obtained during the training phase for the main and auxiliary tasks, while the orange L_{aux} denotes the losses during the test-time training phase. We further align the feature spaces of the shared graph encoder between the training and test sets via feature alignment. $\hat{y}$ denotes the predicted values during the testing phase.

For low-resource regimes, we define a much smaller labeled target domain dataset $\mathbb{D}^{lt} = \{E_i^{lt}, y_i^{lt}\}_{i=1}^{N^{lt}}$ for training, where $N^{lt} \ll N^s$. The target domain dataset for testing is denoted as $\mathbb{D}^{ut} = \{E_i^{ut}, y^{ut}\}_{i=1}^{N^{ut}}$.

Given an event $E_i = \{c_i, R(c_i)\}$, we can build the propagation graph as $\mathcal{G}^i = \langle \mathcal{V}^i, \mathcal{E}^i \rangle$, where $\mathcal{V}^i = \{c_i, r_1^i, \ldots, r_{|E_i|-1}^i\}$ is the node, and $\mathcal{E}^i = \{e_{ht}^i \mid h, t \in \mathcal{V}^i\}$ denotes the set of edges, from the responded node h to its follow-up comment node t. The adjacency matrix $\boldsymbol{A}^i \in \{0,1\}^{|E_i|\times|E_i|}$ is defined as follows:

$$a_{th}^i = \begin{cases} 1, & \text{if node } t \text{ comment to the node } h. \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

$\boldsymbol{X}^i = [\boldsymbol{x}_0^{i\top}, \boldsymbol{x}_1^{i\top}, \ldots, \boldsymbol{x}_{|E_i|-1}^{i\top}]^\top$ is the feature matrix that is extracted from the posts in event E_i. $\mathbf{x}_0^i$ is the feature of the claim c_i and $\mathbf{x}_j^i$ represents the feature of replay r_j^i. The target of rumor detection is to train a classification model $f(\mathcal{G}) \rightarrow y$ that is used to predict whether the event is a rumor.

4 METHODOLOGY

In this section, we present the proposed architecture of Test-Time Training for Rumor Detection (T^{3}RD). Rumor detection serves as the main task of T^{3}RD (Sec. 4.1). Here, we define the loss function for rumor detection as L_{main} . Following the principle of test-time training [16], we elaborately design the auxiliary tasks that effectively capture the information of social media conversations. Specifically, we introduce self-supervised learning (SSL) (Sec. 4.2) as the auxiliary task to capture the node-graph and node-node information. For simplicity, we denote the objective of the auxiliary SSL task as L_{aux}. Furthermore, to ensure that the model trained in the main task is capable of adapting to the test samples, we introduce a feature alignment task (Sec. 4.3) to mitigate the risk of distribution distortion [16] in test-time training. The objective of feature alignment is defined as L_{algin}. The overall architecture of our rumor detection approach is illustrated in Fig. 2.

4.1 The Main Rumor Detection Task

We adopt the Bi-directional Graph Convolutional Networks [2] (Bi-GCN) as the rumor detection backbone of our framework. Given the event $E = \{c, R(c)\}$, and its adjacency matrix $\boldsymbol{A}$ and feature matrix $\boldsymbol{X}$, we first obtain propagation and dispersion features through Top-Down graph convolutional Networks (TD-GCN) and Bottom-Up graph convolutional Networks (BU-GCN), respectively. Note that the TD-GCN and BU-GCN share the same feature matrix $\boldsymbol{X}$, but the adjacency matrix $\boldsymbol{A}_{TD} = \boldsymbol{A}$ and $\boldsymbol{A}_{BU} = \boldsymbol{A}^\top$.

The Shared Graph Encoder. We adopt 2-layer graph convolutional networks (GCN) to capture the propagation features of the event:

$$\begin{aligned} \boldsymbol{H}_{TD}^1 &= \text{ReLU}(\hat{\boldsymbol{A}}_{TD} \cdot \boldsymbol{X} \cdot \boldsymbol{W}_{TD}^1) \\ \boldsymbol{H}_{TD}^2 &= \text{ReLU}(\hat{\boldsymbol{A}}_{TD} \cdot \boldsymbol{H}_{TD}^1 \cdot \boldsymbol{W}_{TD}^2) \end{aligned} \tag{2}$$

where the $\hat{\mathbf{A}}_{TD} = \tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}$ is the normalized adjacency matrix. $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_N$ for the self-connection. $\tilde{\mathbf{D}}_{ii} = \Sigma_j \tilde{\mathbf{A}}_{ij}$ which is the degree of the i-th node. W_{TD}^* are the trainable weighted matrix. Similar to the TD-GCN, the same procedure described in Eq. 2 is applied in the BU-GCN to capture the dispersion features of the event and ultimately obtain the output node representation $\boldsymbol{H}_{BU}^2$. We aggregate the output of TD-GCN and BU-GCN separately using mean pooling (MEAN(·)) to obtain representations for propagation and dispersion, and then generate the final feature representation $\boldsymbol{S}_i$ for the event E_i by concatenating the two representations:

$$\boldsymbol{S}_i = \text{concat}(\text{MEAN}(\boldsymbol{H}_{TD}^2), \text{MEAN}(\boldsymbol{H}_{BU}^2)) \tag{3}$$

In short, we formulate the graph encoding process as $\boldsymbol{S}_i = g(\mathcal{G}_i; \boldsymbol{\theta}_e)$, where $\boldsymbol{\theta}_e = \{\boldsymbol{W}_{TD}^1, \boldsymbol{W}_{TD}^2, \boldsymbol{W}_{BU}^1, \boldsymbol{W}_{BU}^2\}$ is the parameters.

The Training Objective. The predicted label $\hat{y}_i = f(S_i; \theta_c)$ is obtained by inputting the feature representations S_i into several full connection layers with parameter θ_c and a softmax function: . Given the high-resource dataset $\mathbb{D}^s$ and the low-resource annotated dataset $\mathbb{D}^{lt}$, we train the model to minimize the cross-entropy loss between the prediction $\hat{y}_i$ and the ground-truth label y_i:

$$L_{CE}(\mathcal{G}, y; \theta_e, \theta_c) = -\frac{1}{N}\sum_{i=1}^{N}\log(f(g(\mathcal{G}_i; \theta_e); \theta_c)) \tag{4}$$

where $N = N^s + N^{lt}$, and p_i is the probability of the correct prediction.

To further utilize the annotated label, a supervised contrastive learning (SCL) objective is introduced, which enhances the discriminative power of rumor representations S_i of the training events by clustering samples of the same class while separating those from different classes. Specifically, the clustering contrastive loss for the i-th event is defined as:

$$\mathrm{I}_{SCL}(S_i) = \sum_{j\neq i}^{N^s} \mathbb{1}_{[y_i=y_j]} \log \frac{\exp\left(S_i^\top S_j/\tau\right)}{\sum_{k\neq i}^{N^s}\exp\left(S_i^\top S_k/\tau\right)} \tag{5}$$

where $\mathbb{1}$ is the indicator, and τ is the temperature parameter. Thus, we define the SCL loss as:

$$L_{SCL}(\mathcal{G}, y; \theta_e) = -\frac{1}{N^s}\sum_{i=1}^{N^s}\mathrm{MEAN}\left(\mathrm{I}_{SCL}(S_i)\right) \tag{6}$$

Finally, we train the main task as the combination of the cross-entropy and supervised contrastive learning objectives:

$$L_{\mathrm{main}}(\mathcal{G}, y; \theta_e, \theta_c) = L_{CE}(\mathcal{G}, y; \theta_e, \theta_c) + L_{SCL}(\mathcal{G}, y; \theta_e) \tag{7}$$

4.2 The Auxiliary Self-supervised Contrastive Learning Tasks

The key to the test-time training lies in an appropriate auxiliary self-supervised learning task [16, 25]. Inspired by the success of multi-view contrastive learning in graph [9, 29], we adopt two self-supervised contrastive learning tasks, namely, global and local contrastive learning, to capture the intrinsic traits of social media conversations.

Global Contrastive Learning. The objective of global contrastive learning (GCL) is to help the node representation capture graph structure information from the entirety of the social media conversation graph. In essence, GCL operates by maximizing the mutual information between the node-level representations and the global-level graph representation [23, 28]. As shown in Fig. 2, given the original rumor propagation graph, we first generate different views of it with data augmentation. For GCL, we employ two views: the original view v_0, where the structure and attribute of the graph remain unaltered; the shuffled node attributes view v_1, where the structure of the graph remains unaltered, while the attributes of nodes are randomly shuffled. The two corresponding node representations, $\boldsymbol{n}_0$ and $\boldsymbol{n}_1$ can be obtained by inputting the two views into the shared graph encoder. Following DGI [28], a global graph representation $G_0 = \mathrm{MLP}(\boldsymbol{n}_0; \theta_g)$ is summarized by employing a multilayer perceptron with parameter θ_d to process the node representation matrix $\boldsymbol{n}_0$, which is extracted from the original view v_0. The positive pairs of GCL $\{\langle \boldsymbol{n}_{0,j}, G_0\rangle\}_{j=1}^{|E|}$ consist of the node representation $\boldsymbol{n}_{0,j}$ of the original view v_0, and the graph representation G_0 of the view v_0, while the negative pairs $\{\langle \boldsymbol{n}_{1,j}, G_0\rangle\}_{j=1}^{|E|}$ are composed of the node representation $\boldsymbol{n}_{1,j}$ from the view v_1 and the graph representation G_0. The distance between the node representation and the graph representation is measured as $d(\boldsymbol{n}_{i,j}, G_i) = \mathrm{Sigmoid}(\boldsymbol{n}_{i,j} * G_i)$, where the $*$ is the inner product. Since the positive pairs should have higher scores, and the negative pairs should have lower scores, the formulation of the objective function for GCL is delineated as follows:

$$L_{GCL}(\mathcal{G}; \theta_e, \theta_g) = -\frac{1}{2|E|}\sum_{j=1}^{|E|}(\log d(\boldsymbol{n}_{0,j}, G_0) + \log(1 - d(\boldsymbol{n}_{1,j}, G_0))) \tag{8}$$

Local Contrastive Learning. We further introduce the local contrastive learning (LCL) task to learn a robust node representation which able to tolerate slight perturbations of attributes and structures in social media conversations. The objective of the LCL is to fine-tune the shared graph encoder, enabling it to accurately discern if two nodes, observed from distinct views, represent the same node within the original graph. Given a rumor propagation structure input graph $\mathcal{G}_i$, as shown in Fig. 2, two views of the graph can be generated through data augmentation. Specifically, we mask some attributes of nodes in view v_2, and drop some edges in view v_3. Both operations are not introduce significant alterations to the input graph. The node representations $\boldsymbol{n}_2$ and $\boldsymbol{n}_3$ are then generated by the shared graph encoder, with the views v_2 and v_3 as the input graphs, respectively. The positive paris of LCL $\{\langle \boldsymbol{n}_{2,j}, \boldsymbol{n}_{3,j}\rangle\}_{j=1}^{|E|}$ consists of the same nodes in two augmented views. The negative paris are $\{\langle \boldsymbol{n}_{2,i}, \boldsymbol{n}_{3,j}\rangle\}_{i\neq j}^{|E|}$, $\{\langle \boldsymbol{n}_{2,i}, \boldsymbol{n}_{2,j}\rangle\}_{i\neq j}^{|E|}$. Inspired by InfoNCE [27, 39], we define the objective for a positive node pair $(\boldsymbol{n}_{2,i}, \boldsymbol{n}_{3,i})$ as follows:

$$\mathrm{I}_{LCL}\left(\boldsymbol{n}_{2,i}, \boldsymbol{n}_{3,i}\right) = \log \frac{exp\left(\boldsymbol{F}_{2,i}^\top \boldsymbol{F}_{3,i}/\tau\right)}{\sum_{j\neq i}\left(exp\left(\boldsymbol{F}_{2,i}^\top \boldsymbol{F}_{3,j}/\tau\right) + exp\left(\boldsymbol{F}_{2,i}^\top \boldsymbol{F}_{2,j}/\tau\right)\right)} \tag{9}$$

where $\boldsymbol{F}_{ij} = \mathrm{MLP}(\boldsymbol{n}_{ij}; \theta_l)$ is a two-layer perceptron, and θ_l is the parameter. The objective function for LCL can be defined as follows:

$$L_{LCL}(\mathcal{G}; \theta_e, \theta_l) = -\frac{1}{2|E|}\sum_{i=1}^{|E|}\left(\mathrm{I}_{LCL}\left(\boldsymbol{n}_{2,i}, \boldsymbol{n}_{3,i}\right) + \mathrm{I}_{LCL}\left(\boldsymbol{n}_{3,i}, \boldsymbol{n}_{2,i}\right)\right) \tag{10}$$

The Training Objective. The total loss function of the auxiliary self-supervised contrastive task is a weighted combination of GCL and LCL losses:

$$L_{aux}(\mathcal{G}; \theta_e, \theta_s) = L_{GCL}(\mathcal{G}; \theta_e, \theta_g) + \beta L_{LCL}(\mathcal{G}; \theta_e, \theta_l) \tag{11}$$

where β is the parameter that balances global and local contrastive learning, and $\theta_s = \{\theta_g, \theta_l\}$ is the parameter of the discriminator and the two-layer perceptron.

4.3 Feature Alignment

Without any constraints on feature distribution, test-time training may lead to the shared graph encoder excessively adapting to the auxiliary self-supervised learning task. This overfitting can degrade, rather than enhance performance on the primary task [16].

Table 1: Statistics of the datasets in this paper.

Statistics	Source Twitter	Target Weibo-COVID19	Source Weibo	Target Twitter-COVID19
# of events	1154	399	4649	400
# of tree nodes	60409	26687	1956449	406185
# of non-rumors	579	146	2336	148
# of rumors	575	253	2313	252
Avg.# of posts/tree	52	67	420	1015
Language	English	Chinese	Chinese	English

The problem becomes even more precarious in rumor detection, since rumor samples not only exhibit significant variations in their post attributes but also in the structure of the rumor propagation graphs. To alleviate this problem, we propose two constraints for the learning of the shared graph encoder to ensure that the feature distribution of test events stays close to that of the training domain.

Contrastive Feature Alignment. In the low-resource setting, we have a small annotated target domain dataset $\mathbb{D}^{lt}$ for training. For an event E_i^{lt} from the target data, we can align the features of source domain and target domain samples by employing contrastive learning, i.e., pushing samples from the same category in the source and target domains closer than samples from different categories. We define the label-aware contrastive loss for the i-th target event as:

$$\mathrm{I}_{TCL}(\boldsymbol{S}_j^{lt}) = \sum_{j=1}^{N^{lt}} \mathbb{1}_{\left[y_i = y_j^{lt}\right]} \log \frac{\exp\left(\boldsymbol{S}_i^{\top} \boldsymbol{S}_j^{lt} / \tau\right)}{\sum_{k=1}^{N^{lt}} \exp\left(\boldsymbol{S}_i^{\top} \boldsymbol{S}_k^{lt} / \tau\right)} \tag{12}$$

The objective function for contrastive feature alignment can be defined as:

$$L_{TCL}(\mathcal{G}, y; \boldsymbol{\theta}_e) = -\frac{1}{N^s} \sum_{i=1}^{N^s} \mathrm{MEAN}\left(\mathrm{I}_{TCL}(\mathcal{G}, y; \boldsymbol{\theta}_e)\right) \tag{13}$$

Distribution Alignment. After training the model with the source domain dataset $\mathbb{D}^s = \{E_i^s, y_i^s\}_{i=1}^{N^s}$, we can obtain the embedding $\boldsymbol{S}_i^s$ of each source domain sample E_i^s. Thus, the mean μ_s and covariance matrix Σ_s of the source domain samples can be calculated:

$$\mu^s = \frac{1}{N^s} \sum_{i=1}^{N^s} \boldsymbol{S}_i^s, \quad \Sigma^s = \frac{1}{N^s - 1} \sum_{i=1}^{N^s} (\boldsymbol{S}_i^s - \mu_s)(\boldsymbol{S}_i^s - \mu_s)^{\top} \tag{14}$$

In the test-time training process, given the event E_i^{ut} from the test dataset $\mathbb{D}^{ut}$, we can compute distribution statistics μ_i^{ut} and Σ_i^{ut} among the corresponding graph views $v_0, \cdots, v_3$ of the graph $\mathcal{G}_i^{ut}$ of event E_i^{ut}. The distribution alignment aims to enforce the distribution statistics of the test event to closely align with those of the training graph samples. The objective L_{da} can be defined as:

$$L_{da}(\mathcal{G}; \boldsymbol{\theta}_e) = \left\|\mu^s - \mu^{ut}\right\|_2^2 + \left\|\Sigma^s - \Sigma^{ut}\right\|_F^2. \tag{15}$$

where $\|\cdot\|_2$ is the Euclidean norm and $\|\cdot\|_F$ is the Frobenius norm.

4.4 The Test-time Training Framework

The Training Phase. During the training phase, we train all the parameters for both the rumor detection task (main task), the auxiliary self-supervised contrastive learning task (SSL task), and the feature alignment with the training set $\mathbb{D}^T$. Note that in the low-resource scenarios, the training set $\mathbb{D}^T$ consists of a source domain dataset $\mathbb{D}^s$ and a much smaller target domain dataset $\mathbb{D}^{lt}$. In the zero-short scenario, only the source domain dataset $\mathbb{D}^s$ is available. During training, the framework is optimized by minimizing a weighted sum of the L_{main}, L_{TCL}, and L_{aux}:

$$\min_{\boldsymbol{\theta}_e, \boldsymbol{\theta}_c, \boldsymbol{\theta}_s} L_{main}\left(\mathcal{G}, y; \boldsymbol{\theta}_e, \boldsymbol{\theta}_c\right) + \alpha L_{aux}\left(\mathcal{G}, y; \boldsymbol{\theta}_e, \boldsymbol{\theta}_s\right) + L_{TCL}(\mathcal{G}, y; \boldsymbol{\theta}_e) \tag{16}$$

where α is a parameter to balance the rumor detection task and the self-supervised task. By minimizing the overall training loss, we obtain the optimal parameters $\boldsymbol{\theta}_e^*$, $\boldsymbol{\theta}_c^*$ and $\boldsymbol{\theta}_s^*$ for the shared graph encoder, the rumor classification model, and the self-supervised learning model.

The Test-time Training Phase. Given the test event E^{ut} from $\mathbb{D}^{ut}$, the test-time training process fine-tunes the learned model with the auxiliary self-supervised contrastive learning task and feature alignment task by minimizing a weighted combination of losses L_{aux} and L_{DA}:

$$\min_{\boldsymbol{\theta}_e, \boldsymbol{\theta}_s} L_{aux}\left(\mathcal{G}; \boldsymbol{\theta}_e^*, \boldsymbol{\theta}_s^*\right) + \gamma L_{DA}(\mathcal{G}; \boldsymbol{\theta}_e^*) \tag{17}$$

where γ is the parameter to balance the self-supervised learning task and features alignment task. The optimal parameters $\boldsymbol{\theta}_e^{*'}$ and $\boldsymbol{\theta}_s^{*'}$ are obtained by minimizing the overall test-time training loss.

The Test Phase. In the testing phase, we predict the labels of E^{ut} with the optimal parameters $\boldsymbol{\theta}_e^{*'}$, $\boldsymbol{\theta}_c^*$:

$$\hat{y} = f(g(\mathcal{G}; \boldsymbol{\theta}_e^{*\prime}); \boldsymbol{\theta}_c^*) \tag{18}$$

5 EXPERIMENT

5.1 Datasets

We evaluate the proposed model with two widely-used sets [12, 14, 24] of real-world cross-domain rumor datasets . The first set encompasses the English Twitter dataset [18] and the Chinese Weibo-COVID19 dataset [12]. The second set comprises the Chinese Weibo dataset [17] and the English Twitter-COVID19 dataset [12]. These cross-domain datasets are annotated with two binary labels: Non-rumor and Rumor. Detailed statistics for both sets of cross-domain datasets are provided in Tab. 1.

5.2 Baseline and Evaluation Metrics

We compare our model with the CNN-based model (CNN) [33], RNN-based model (RNN) [17], tree-structured recursive model (RvNN) [20], transformer-based model (PLAN) [11], GCN-based model (BiGCN) [2], adversarial contrastive learning framework built on top of BiGCN (ACLR-BiGCN) [12], and zero-shot rumor detection model based on prompt learning (RPL) [14].

We employ commonly-used metrics to evaluate the effectiveness of our proposed method. The accuracy (ACC) measures the probability of correctly predicting the samples. The F1-score provides distinct scores for positive (RF1), negative (NF1), and macro-average (Mac-F1). These metrics span a range from 0 to 1, with higher values signifying superior performance. We conduct 5-fold cross-validation on the target datasets. The best performances are highlighted in **bold**.

Table 2: The experimental results on the Target (Source) domain.

Target (Source)	Weibo-COVID19 (Twitter)				Twitter-COVID19 (Weibo)			
Models	ACC.	Mac-F1	RF1	NF1	ACC.	Mac-F1	RF1	NF1
CNN	0.445	0.402	0.476	0.328	0.498	0.389	0.528	0.249
RNN	0.463	0.414	0.498	0.329	0.510	0.388	0.533	0.243
RvNN	0.514	0.482	0.538	0.426	0.540	0.391	0.534	0.247
PLAN	0.532	0.496	0.578	0.414	0.573	0.423	0.549	0.298
BiGCN	0.569	0.508	0.586	0.429	0.616	0.415	0.577	0.252
ACLR-BiGCN	0.873	0.861	0.896	0.827	0.765	0.686	**0.766**	0.605
T^3RD	**0.896**	**0.880**	**0.917**	**0.843**	**0.781**	**0.696**	0.706	**0.685**

Table 3: Experimental results with different language settings.

Target (Source)	Cantonese (English)		Arabic (English)	
Models	ACC.	Mac-F1	ACC.	Mac-F1
BiGCN	0.538	0.504	0.586	0.487
ACLR-BiGCN	0.653	0.617	0.671	0.579
T^3RD	**0.739**	**0.637**	**0.730**	**0.583**

5.3 Overall Performance

Low-resource Rumor Detection. In Tab. 2, we evaluate our proposed method on Weibo-COVID19 and Twitter-COVID19 test sets. We note that the baselines in the first group exhibit poor performance, likely due to their sole reliance on sequential information while neglecting the propagation structure. Other baselines leverage the structural properties of information propagation on social media, affirming the significance of propagation structure representations within our framework. In the second group of structure-based baselines, PLAN and BiGCN outperform RvNN, attributed to the feature vector architecture and tree structures. The third group focuses on cross-domain rumor detection, with ACLR aligning the source and target domains through supervised contrast [12]. In contrast, our model T^3RD employs test-time training to further extract additional information from the test data. In contrast, our proposed T^3RD achieves state-of-the-art performance on the Weibo-COVID19 and Twitter-COVID19 datasets and makes a 2.3% and 1.6% improvement in accuracy scores, respectively. These results further emphasize the efficacy of test-time training in improving cross-domain rumor detection by extracting information from test data. In tab. 3, we evaluate the proposed method in more different language settings. We use the Cantonese-COVID19 dataset and the Arabic-COVID19 dataset [14] as the target data, and the English Twitter dataset as the source data. Our proposed T^3RD achieves the best performance in both Cantonese-COVID19 and Arabic-COVID19 datasets, with an 8.6% and 5.9% improvement in accuracy, respectively, over the state-of-the-art ACLR-BiGCN model. These results demonstrate the effectiveness of our proposed method for cross-domain and cross-language rumor detection tasks.

Zero-short Rumor Detection. We further carry out experiments on zero-shot target domain rumor detection, where the model is trained on source domain data and tested on target domain data. The experimental results are shown in Tab. 4. In comparison with the state-of-the-art, our proposed T^3RD method achieves a 5.2% improvement in accuracy score on the Weibo-COVID19 dataset. On the Twitter-COVID19 dataset, T^3RD also achieve a competitive performance, with a 0.8% improvement in accuracy over the state-of-the-art RPL model. These results demonstrate the generalization capabilities of our proposed method for zero-shot rumor detection.

5.4 Ablation Study

Effectiveness of T^3RD on different backbones. In Tab. 5, we investigate the effectiveness of test-time training on different backbones. We observe that the performance of RvNN, PLAN, and BiGCN is significantly improved by incorporating test-time training, demonstrating the effectiveness of our proposed method in enhancing the generalization capabilities of the model.

Effectiveness of test-time training. In Fig. 3a, we investigate the effectiveness of test-time training on rumor detection. Specifically, in the T^3RD-w/o-TTT model, we remove the test-time training phase from T^3RD but retain the self-supervised learning in the training phase. In the T^3RD-w/o-(TTT&SSL) model, we remove the test-time training phase and all the self-supervised learning from T^3RD. We adopt Weibo as the source domain dataset and Twitter-COVID19 as the target domain dataset. As depicted in Fig. 3a, performance exhibits a gradual decline, indicating the efficacy of self-supervised learning in rumor detection, while underscoring the importance of test-time training in information extraction from the test set.

Effectiveness of components in T^3RD. We utilize Twitter as the source domain dataset and Weibo-COVID19 as the target domain dataset to evaluate the effectiveness of the components of our T^3RD. As shown in Fig. 3b, we evaluate the contributions of supervised, global, and local contrastive learning, as well as the contrastive feature alignment and distribution alignment within the T^3RD framework, by sequentially omitting each component. These five variants of T^3RD are denoted as T^3RD-w/o-SCL, T^3RD-w/o-GCL, T^3RD-w/o-LCL, T^3RD-w/o-TCL, and T^3RD-w/o-DA, respectively. The results reveal that excluding any of these elements leads to a deterioration in performance, highlighting the essential contribution of each component to the efficacy of rumor detection.

Table 4: Experimental results of zero-shot rumor detection.

Target (Source)	Weibo-COVID19 (Twitter)				Twitter-COVID19 (Weibo)			
Models	ACC.	Mac-F1	RF1	NF1	ACC.	Mac-F1	RF1	NF1
BiGCN	0.615	0.524	0.729	0.319	0.545	0.529	0.511	0.547
ACLR-BiGCN	0.721	0.685	0.788	0.582	0.676	0.642	0.739	0.545
RPL	0.745	0.719	0.804	0.634	0.727	0.697	0.793	**0.601**
T^{3}RD	**0.797**	**0.788**	**0.832**	**0.743**	**0.735**	**0.701**	**0.808**	0.593

Table 5: Experimental results of T^{3}RD with different backbones.

Target (Source)	Weibo-COVID19 (Twitter)			
Models	ACC.	Mac-F1	RF1	NF1
RvNN	0.514	0.482	0.538	0.426
+ ACLR	0.778	0.716	0.843	0.589
+ T^{3}RD	0.801	0.786	0.841	0.731
PLAN	0.532	0.496	0.578	0.414
+ ACLR	0.824	0.769	0.842	0.696
+ T^{3}RD	0.851	0.840	0.869	0.810
BiGCN	0.569	0.508	0.586	0.429
+ ACLR	0.873	0.861	0.896	0.827
+ T^{3}RD	**0.896**	**0.880**	**0.917**	**0.843**

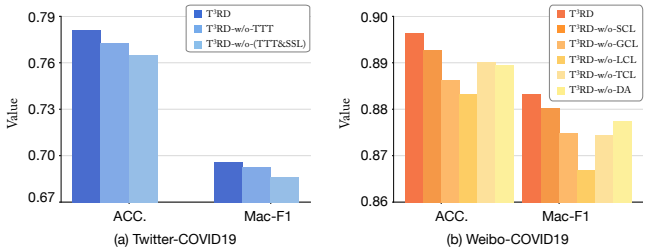

Figure 3: The ablation study of components in T^{3}RD.

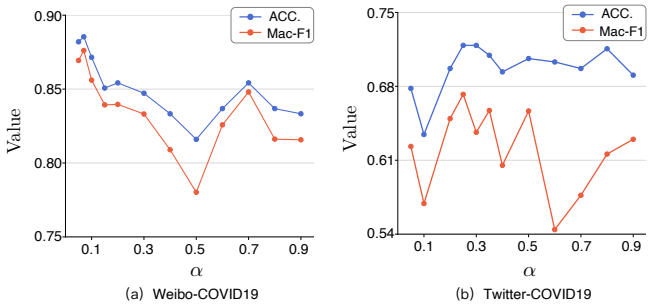

Figure 4: Parameter analysis for the α on Weibo-COVID19 and Twitter-COVID19.

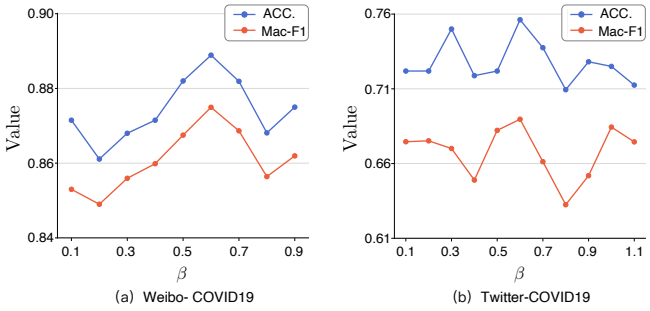

Figure 5: Parameter analysis for the β on Weibo-COVID19 and Twitter-COVID19.

5.5 Hyper-parameter Analysis.

Hyper-parameter α. To investigate the influence of the hyperparameter α in Eq. 16 on performance (Accuracy and Macro F1 score), we conduct a qualitative analysis within the T^{3}RD architecture. For the target data, Weibo-COVID19, we use Twitter as the source data (in Fig. 4a). In terms of Twitter-COVID19, we use Weibo as the source data (in Fig. 4b). The horizontal axis represents the values of α. We observe that when Weibo-COVID19 is the target dataset, the optimal result is achieved at $\alpha = 0.07$, and for Twitter-COVID19, the best result is obtained at $\alpha = 0.25$. From the overall trend, as the α value increases, the performance exhibits a certain degree of fluctuation. We believe this occurs because the model, while optimizing the representation distribution, compromises the mapping relationship with labels.

Hyper-parameter β. To examine the impact of the hyper-parameter β in Eq. 11 on performance metrics, we perform a qualitative analysis within the T^{3}RD framework, illustrated in Fig. 5. The horizontal axis represents the β values. According to the results, we observe that the optimal outcome is achieved at $\beta = 0.6$ when Weibo-COVID19 is the target dataset, while $\beta = 0.6$ yields the best result when Twitter-COVID19 is the target dataset.

Hyper-parameter γ. To study the effectiveness of hyper-parameter γ in our Eq. 17, we conduct qualitative analysis under T^{3}RD architecture in Fig. 6. The horizontal axis denotes the value of γ. According to the results, we observe that the optimal outcome is achieved at $\gamma = 0.7$ when Weibo-COVID19 is the target dataset, while $\gamma = 0.5$ yields the best result when Twitter-COVID19 is the target dataset.

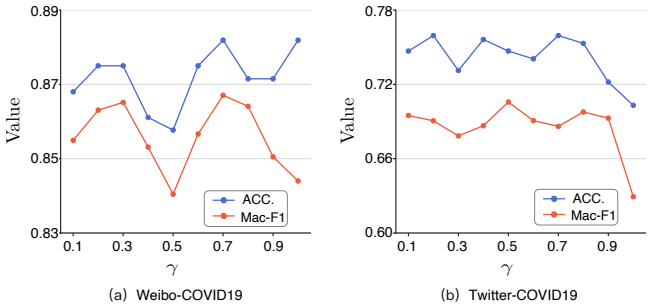

Figure 6: Parameter analysis for the γ on Weibo-COVID19 and Twitter-COVID19.

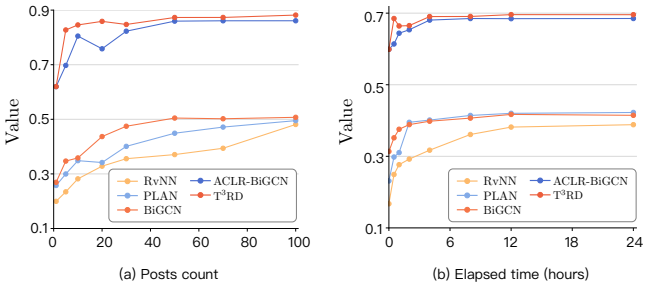

Figure 7: Early detection performance is assessed at various checkpoints based on the count of posts (or elapsed time) on both the Weibo-COVID19 (a) and Twitter-COVID19 (b) datasets.

5.6 Early Detection

Early detection aims to identify rumors at the initial stages of dissemination, which is crucial for mitigating their societal effects. This capability is thus an essential metric for assessing rumor detection models. To facilitate an early detection task, we follow the existing settings [2, 35] and introduce a sequence of detection "delays" or checkpoints, defined by the volume of reply posts or the elapsed time since the original post. For model evaluation, we consider only the data available up to each checkpoint, with performance measured by the macro F1 scores at these intervals.

As illustrated in Fig. 7, we make a comparison of the performance between our method with RvNN, PLAN, BiGCN, and ACLR at various checkpoints. The proposed T^{3}RD outperforms other approaches throughout the entire lifecycle and achieves a relatively high Macro F1 score at an early stage. Our method requires about 20 posts on Weibo-COVID19 and 4 hours on Twitter-COVID19 to achieve stable performance, while the state-of-the-art method ACLR requires 50 posts to achieve a similar level of performance. This demonstrates the outstanding early detection capability of our method. Additionally, early-stage performance tends to exhibit more or less fluctuation. This is due to the increase in semantic and structural information as statements propagate, resulting in a corresponding increase in noise.

Table 6: Experimental results of Cross-domain Rumor Detection.

Target	Weibo-COVID19		Twitter-COVID19	
Models	ACC.	Mac-F1	ACC.	Mac-F1
ACLR (cross-domain)	0.884	0.855	0.737	0.623
T^{3}RD (cross-domain)	**0.900**	**0.889**	0.771	0.689
T^{3}RD	0.896	0.880	**0.781**	**0.696**

5.7 Cross-domain Rumor Detection

The default setting for our main experiments is cross-domain and cross-language, the results of which are shown in Tab. 2. In this subsection, we exclusively conduct experiments within the cross-domain setting. For the target dataset, Weibo-COVID19, the comprehensively annotated Weibo dataset serves as the source. Conversely, for Twitter-COVID19, the Twitter dataset is utilized as the source. The results are shown in Tab. 6. With Weibo as the source data, our model improves the 0.4% Accuracy and 0.9% Macro F1 on the Weibo-COVID19 dataset, indicating our superior ability in cross-domain rumor detection on Weibo. On the other hand, performance on Twitter-COVID19, with Twitter as the source, is comparatively lower. This discrepancy may stem from the lesser event diversity in the Twitter dataset compared to Weibo, where our model attains an Accuracy of approximately 77.1% and a Macro F1 score of 68.9% across response ranking variants. The T^{3}RD approach not only addresses the challenge of data scarcity in rumor detection but also reduces the dependence on extensively annotated domain- and language-specific datasets.

6 CONCLUSION AND FUTURE WORK

In this work, we introduce a novel test-time training framework for rumor detection, aimed at reducing the performance gap between training and test datasets. This framework incorporates a test-time training phase that leverages self-supervised learning techniques to derive insights from the test data. Specifically, global contrastive learning is employed to derive graph structures that enhance node representations, while local contrastive learning is dedicated to refining robust node representations. Together, these elements facilitate a comprehensive understanding of the test set, thereby improving the framework's overall generalization in detecting rumors. Comprehensive experimental results on two widely used datasets demonstrate the superiority of the proposed method. In future work, we will explore the design of more effective auxiliary tasks for test-time training on rumor detection.

Acknowledgement This work is supported by National Natural Science Foundation of China (No. 62206137, 62206200, 62106262, 62036012 and 62276257), Inner Mongolia Natural Science Foundation (No.2022MS06025), Program for Young Talents of Science and Technology in Universities of Inner Mongolia Autonomous Region (No.NJYT23105), Beijing Natural Science Foundation (No.JQ23018), Tianjin Natural Science Foundation (No.22JCQNJC00940).

REFERENCES

[1] Alexander Bartler, Andre Bühler, Felix Wiewel, Mario Döbler, and Bin Yang. 2022. MT3: Meta Test-Time Training for Self-Supervised Test-Time Adaption. In *International Conference on Artificial Intelligence and Statistics, AISTATS 2022, 28-30 March 2022, Virtual Event (Proceedings of Machine Learning Research, Vol. 151)*. PMLR, 3080–3090.

[2] Tian Bian, Xi Xiao, Tingyang Xu, Peilin Zhao, Wenbing Huang, Yu Rong, and Junzhou Huang. 2020. Rumor Detection on Social Media with Bi-Directional Graph Convolutional Networks. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press, 549–556.

[3] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2011. Information credibility on twitter. In *Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28 - April 1, 2011*. ACM, 675–684.

[4] Guowei Fan, Tao Wu, and Yamei Lei. 2022. Research on Rumor Detection Based on RoBERTa-BiGRU Model. In *International Conference on Artificial Intelligence and Security*. Springer, 194–204.

[5] Adrien Friggeri, Lada A. Adamic, Dean Eckles, and Justin Cheng. 2014. Rumor Cascades. In *Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM 2014, Ann Arbor, Michigan, USA, June 1-4, 2014*. The AAAI Press.

[6] Yang Fu, Sifei Liu, Umar Iqbal, Shalini De Mello, Humphrey Shi, and Jan Kautz. 2021. Learning to Track Instances without Video Annotations. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, 8680–8689.

[7] Han Guo, Juan Cao, Yazi Zhang, Junbo Guo, and Jintao Li. 2018. Rumor Detection with Hierarchical Social Attention Network. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*. ACM, 943–951.

[8] Nicklas Hansen, Rishabh Jangir, Yu Sun, Guillem Alenyà, Pieter Abbeel, Alexei A. Efros, Lerrel Pinto, and Xiaolong Wang. 2021. Self-Supervised Policy Adaptation during Deployment. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

[9] Kaveh Hassani and Amir Hosein Khasahmadi. 2020. Contrastive Multi-View Representation Learning on Graphs. In *Proceedings of International Conference on Machine Learning*. 3451–3461.

[10] Maria Janicka, Maria Pszona, and Aleksander Wawer. 2019. Cross-Domain Failures of Fake News Detection. *Computación y Sistemas* 23, 3 (2019).

[11] Ling Min Serena Khoo, Hai Leong Chieu, Zhong Qian, and Jing Jiang. 2020. Interpretable Rumor Detection in Microblogs by Attending to User Interactions. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press, 8783–8790.

[12] Hongzhan Lin, Jing Ma, Liangliang Chen, Zhiwei Yang, Mingfei Cheng, and Guang Chen. 2022. Detect Rumors in Microblog Posts for Low-Resource Domains via Adversarial Contrastive Learning. In *Findings of the Association for Computational Linguistics: NAACL 2022, Seattle, WA, United States, July 10-15, 2022*. Association for Computational Linguistics, 2543–2556.

[13] Hongzhan Lin, Jing Ma, Mingfei Cheng, Zhiwei Yang, Liangliang Chen, and Guang Chen. 2021. Rumor Detection on Twitter with Claim-Guided Hierarchical Graph Attention Networks. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. Association for Computational Linguistics, 10035–10047.

[14] Hongzhan Lin, Pengyao Yi, Jing Ma, Haiyun Jiang, Ziyang Luo, Shuming Shi, and Ruifang Liu. 2023. Zero-Shot Rumor Detection with Propagation Structure via Prompt Learning. In *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*. AAAI Press, 5213–5221.

[15] Xiaomo Liu, Armineh Nourbakhsh, Quanzhi Li, Rui Fang, and Sameena Shah. 2015. Real-time Rumor Debunking on Twitter. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015*. ACM, 1867–1870.

[16] Yuejiang Liu, Parth Kothari, Bastien van Delft, Baptiste Bellot-Gurlet, Taylor Mordan, and Alexandre Alahi. 2021. TTT++: When Does Self-Supervised Test-Time Training Fail or Thrive?. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*. 21808–21820.

[17] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J. Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting Rumors from Microblogs with Recurrent Neural Networks. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*. IJCAI/AAAI Press, 3818–3824.

[18] Jing Ma, Wei Gao, and Kam-Fai Wong. 2017. Detect Rumors in Microblog Posts Using Propagation Structure via Kernel Learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*. Association for Computational Linguistics, 708–717.

[19] Jing Ma, Wei Gao, and Kam-Fai Wong. 2018. Rumor Detection on Twitter with Tree-structured Recursive Neural Networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*. Association for Computational Linguistics, 1980–1989.

[20] Jing Ma, Wei Gao, and Kam-Fai Wong. 2018. Rumor Detection on Twitter with Tree-structured Recursive Neural Networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*. Association for Computational Linguistics, 1980–1989.

[21] Shiwen Ni, Jiawen Li, and Hung-Yu kao. 2021. HAT4RD: Hierarchical Adversarial Training for Rumor Detection in Social Media. *Sensors (Basel, Switzerland)* 22 (2021).

[22] Xing Su, Jian Yang, Jia Wu, and Yuchen Zhang. 2023. Mining User-aware Multi-relations for Fake News Detection in Large Scale Online Social Networks. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, WSDM 2023, Singapore, 27 February 2023 - 3 March 2023*. ACM, 51–59.

[23] Fan-Yun Sun, Jordan Hoffman, Vikas Verma, and Jian Tang. 2019. InfoGraph: Unsupervised and Semi-supervised Graph-Level Representation Learning via Mutual Information Maximization. In *International Conference on Learning Representations*.

[24] Tiening Sun, Zhong Qian, Sujun Dong, Peifeng Li, and Qiaoming Zhu. 2022. Rumor Detection on Social Media with Graph Adversarial Contrastive Learning. In *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*. ACM, 2789–2797.

[25] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. 2020. Test-Time Training with Self-Supervision for Generalization under Distribution Shifts. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 9229–9248.

[26] Lin Tian, Xiuzhen Zhang, and Jey Han Lau. 2022. DUCK: Rumour Detection on Social Media by Modelling User and Comment Propagation Networks. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*. Association for Computational Linguistics, 4939–4949.

[27] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation Learning with Contrastive Predictive Coding. *CoRR* abs/1807.03748 (2018).

[28] Petar Velickovic, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, and R. Devon Hjelm. 2019. Deep Graph Infomax. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

[29] Yiqi Wang, Chaozhuo Li, Wei Jin, Rui Li, Jianan Zhao, Jiliang Tang, and Xing Xie. 2022. Test-Time Training for Graph Neural Networks. *CoRR* abs/2210.08813 (2022).

[30] Weizhi Xu, Junfei Wu, Qiang Liu, Shu Wu, and Liang Wang. 2022. Evidence-aware Fake News Detection with Graph Neural Networks. In *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*. ACM, 2501–2510.

[31] Fan Yang, Yang Liu, Xiaohui Yu, and Min Yang. 2012. Automatic detection of rumor on sina weibo. In *Proceedings of the ACM SIGKDD workshop on mining data semantics*. 1–7.

[32] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2017. A Convolutional Approach for Misinformation Identification. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*. ijcai.org, 3901–3907.

[33] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2017. A Convolutional Approach for Misinformation Identification. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*. ijcai.org, 3901–3907.

[34] Zhenrui Yue, Huimin Zeng, Yang Zhang, Lanyu Shang, and Dong Wang. 2023. MetaAdapt: Domain Adaptive Few-Shot Misinformation Detection via Meta Learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*. Association for Computational Linguistics, 5223–5239.

[35] Li Yuechen, Qian Lingfei, and Ma Jing. 2022. Early Detection of Rumors Based on BERT Model. In *AI and Analytics for Public Health: Proceedings of the 2020 INFORMS International Conference on Service Science*. Springer, 261–268.

[36] Huaiwen Zhang, Shengsheng Qian, Quan Fang, and Changsheng Xu. 2021. Multimodal Disentangled Domain Adaption for Social Media Event Rumor Detection. *IEEE Transactions on Multimedia* 23 (2021), 4441–4454.

[37] Huaiwen Zhang, Shengsheng Qian, Quan Fang, and Changsheng Xu. 2022. Multi-Modal Meta Multi-Task Learning for Social Media Rumor Detection. *IEEE Transactions on Multimedia* 24 (2022), 1449–1459.

[38] Taozheng Zhang and Shuaidong Hu. 2023. Unsupervised Cross-Domain Rumor Detection from Multiple Sources Based on RoBERTa and Multi-CNN. In *Proceedings of the 2023 7th International Conference on Deep Learning Technologies*. 85–90.

[39] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. 2021. Graph Contrastive Learning with Adaptive Augmentation. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*. ACM / IW3C2, 2069–2080.

[40] Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. 2018. Detection and Resolution of Rumours in Social Media: A Survey. *ACM Comput. Surv.* 51, 2, Article 32 (feb 2018), 36 pages.