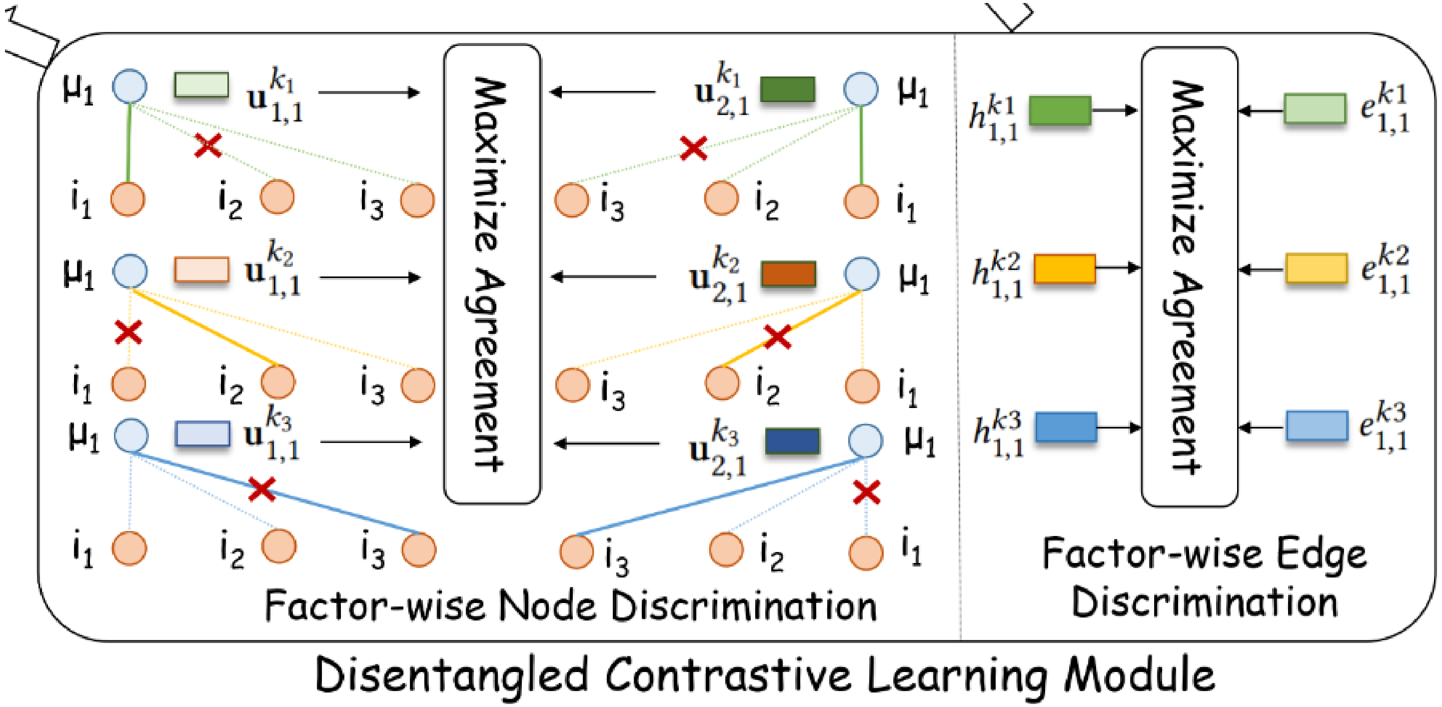
对比学习相关论文

Leibniz-Institut
für Sozialwissenschaften

Reported by Chenghong Li

Method

本文通过随机删除边，来构造子图进行对比学习



$$\mathcal{L}_{fnd}^{user} = -\mathbb{E}_{\mathcal{K}\times\mathcal{U}}\left[\log D\left(\mathrm{u}_{1,i}^{k}, \mathrm{u}_{2,i}^{k}\right)\right] + \mathbb{E}_{\mathcal{K}\times\mathcal{U}\times\mathcal{U}'}\left[\log D\left(\mathrm{u}_{1,i}^{k}, \mathrm{u}_{2,i'}^{k}\right)\right] \quad (11)$$

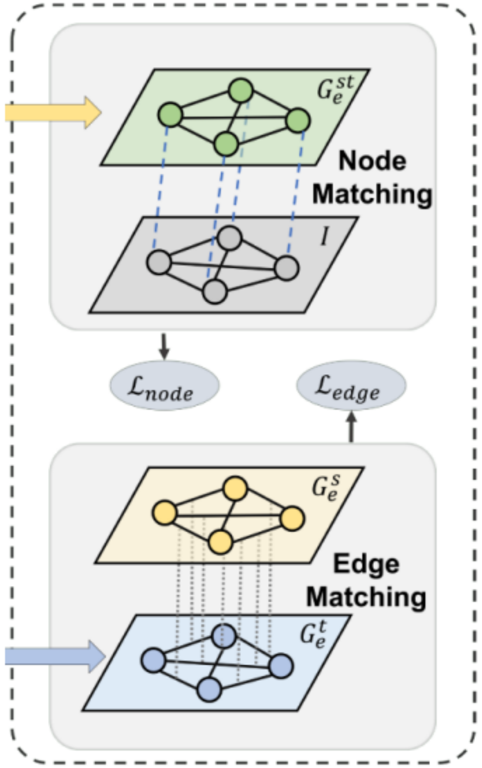
$$D(\mathrm{a}, \mathrm{b}) = \sigma\left(\mathrm{a}^{\top}\mathrm{W}\mathrm{b}\right) \qquad \mathcal{L}_{fnd} = \mathcal{L}_{fnd}^{user} + \mathcal{L}_{fnd}^{item}$$

$$\mathcal{L}_{fed} = -\mathbb{E}_{\mathcal{K}\times\mathcal{E}}\left[\log D\left(\mathrm{h}_{i,j}^{k}, \mathrm{e}_{i,j}^{k}\right)\right] + \mathbb{E}_{\mathcal{K}\times\mathcal{E}\times\mathcal{E}'}\left[\log D\left(\mathrm{h}_{i,j}^{k}, \mathrm{e}_{i',j'}^{k}\right)\right] \quad (12)$$

$$\mathcal{L}_{\text{sup}} = \frac{1}{|\mathcal{T}|}\sum_{(i,j)\in\mathcal{T}}\left(\hat{r}_{ij} - r_{ij}\right)^{2}, \quad (13)$$

$$\mathcal{L} = \mathcal{L}_{\text{sup}} + \lambda_1\mathcal{L}_{fnd} + \lambda_2\mathcal{L}_{fed} \quad (14)$$

2023_IJCAI_SemiGNN-PPI Self-Ensembling Multi-Graph Neural Network for Efficient and Generalizable Protein-Protein Interaction Prediction



$$\mathcal{L}_{node} = ||\text{diag}(\text{Adj}(G_e^{st})) - \text{diag}(I)||_2, \quad (9)$$

$$\mathcal{L}_{edge} = ||\text{Adj}(G_e^s) - \text{Adj}(G_e^t)||_2, \quad (8)$$

$$\mathcal{L} = \mathcal{L}_{sup} + \lambda_{con}\mathcal{L}_{con} + \lambda_{edge}\mathcal{L}_{edge} + \lambda_{node}\mathcal{L}_{node}, \quad (10)$$

2023_WSDM_A Multimodal Framework for the Identification of VaccineCritical Memes on Twitter

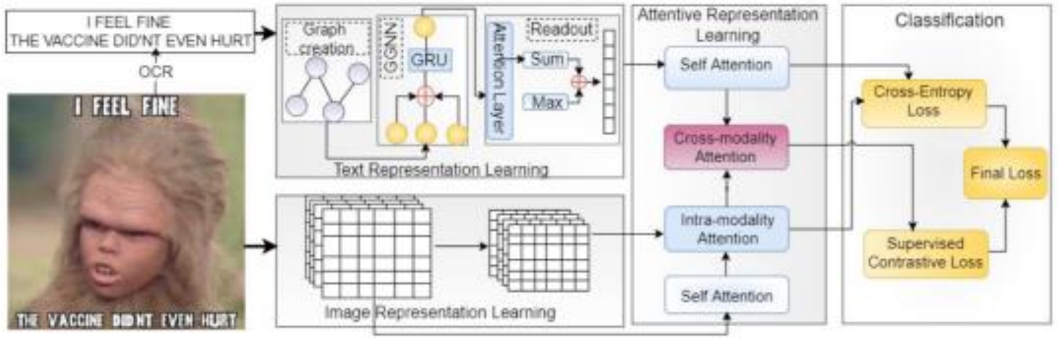
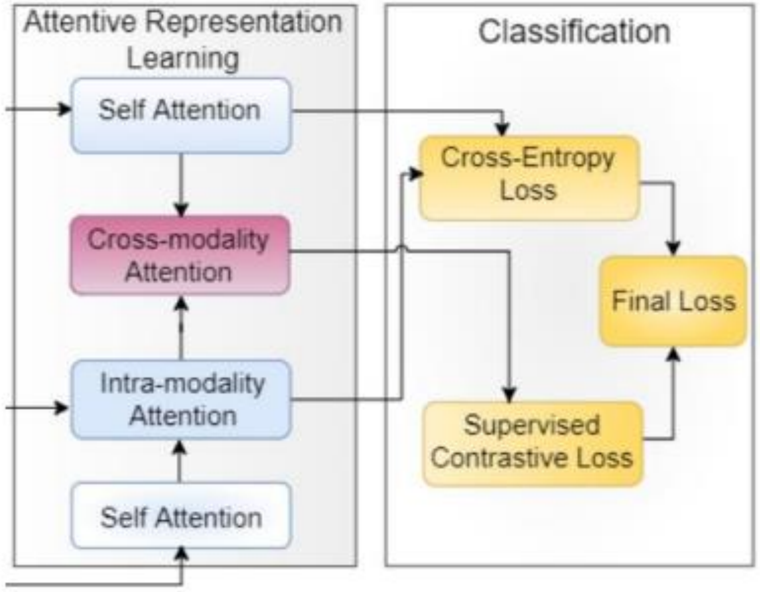


Figure 2: Overall architecture of the proposed multimodal framework.



Classification

$$L_{SCL} = \sum_{i=1}^{N} \frac{-1}{N_{\hat{y}_i} - 1} \sum_{j=1}^{N} 1_{i \neq j} \cdot 1_{\hat{y}_i = \hat{y}_j} \cdot \log\left(\frac{exp(z_i) \cdot (z_j)/\tau)}{\sum_{k=1}^{N} exp(z_i) \cdot (z_k)/\tau)}\right) \tag{13}$$

$$L_{CE} = -\sum_{c=1}^{c} y \log(\hat{y}) \tag{14}$$

$$L = (1 - \lambda)L_{CE} + \lambda L_{SCL} \tag{15}$$

2023_WSDM_GOOD-D: On Unsupervised Graph Out-Of-Distribution Detection

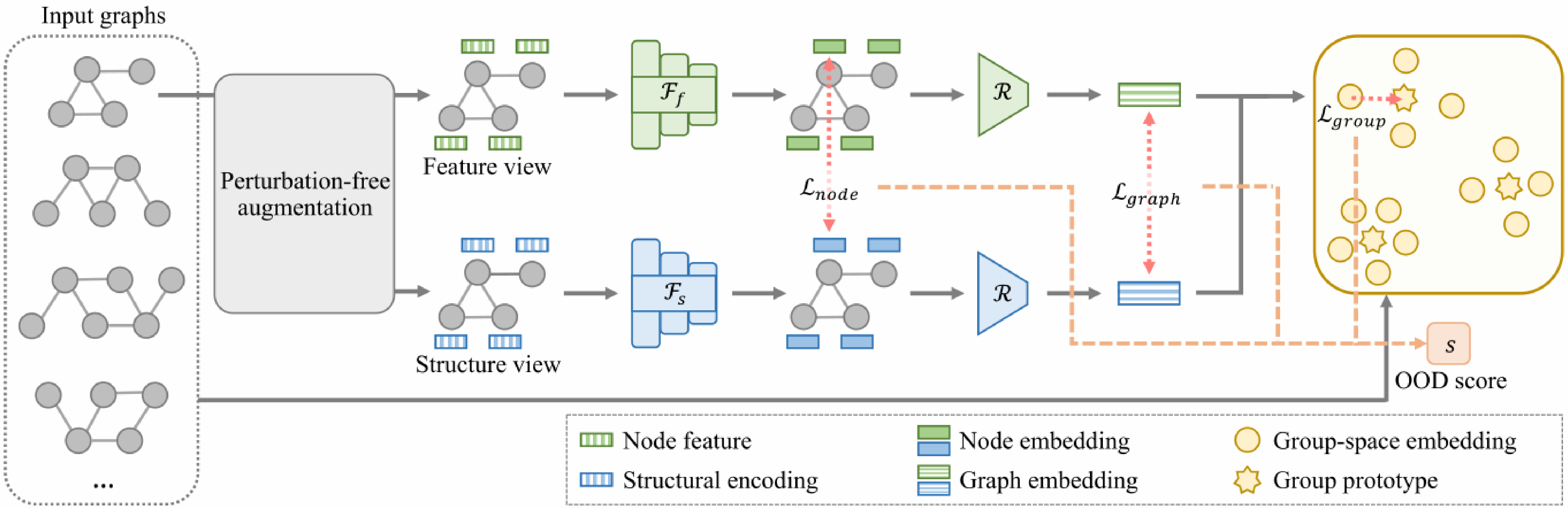


Figure 2: An overall illustration of the proposed method GOOD-D.

$$\mathcal{L}_{node} = \frac{1}{|\mathcal{B}|}\sum_{G_j\in\mathcal{B}}\frac{1}{2|\mathcal{V}_{G_j}|}\sum_{v_i\in\mathcal{V}_{G_j}}\left[\ell(\mathbf{z}_i^{(f)},\mathbf{z}_i^{(s)})+\ell(\mathbf{z}_i^{(s)},\mathbf{z}_i^{(f)})\right],$$

$$\ell(\mathbf{z}_i^{(f)},\mathbf{z}_i^{(s)}) = -\log\frac{e^{\mathrm{sim}(\mathbf{z}_i^{(f)},\mathbf{z}_i^{(s)})/\tau}}{\sum_{v_k\in\mathcal{V}_{G_j}\setminus v_i}e^{\mathrm{sim}(\mathbf{z}_i^{(f)},\mathbf{z}_k^{(s)})/\tau}}, \quad (5)$$

$$\mathcal{L}_{graph} = \frac{1}{2|\mathcal{B}|}\sum_{G_i\in\mathcal{B}}\left[\ell(\mathbf{z}_{G_i}^{(f)},\mathbf{z}_{G_i}^{(s)})+\ell(\mathbf{z}_{G_i}^{(s)},\mathbf{z}_{G_i}^{(f)})\right],$$

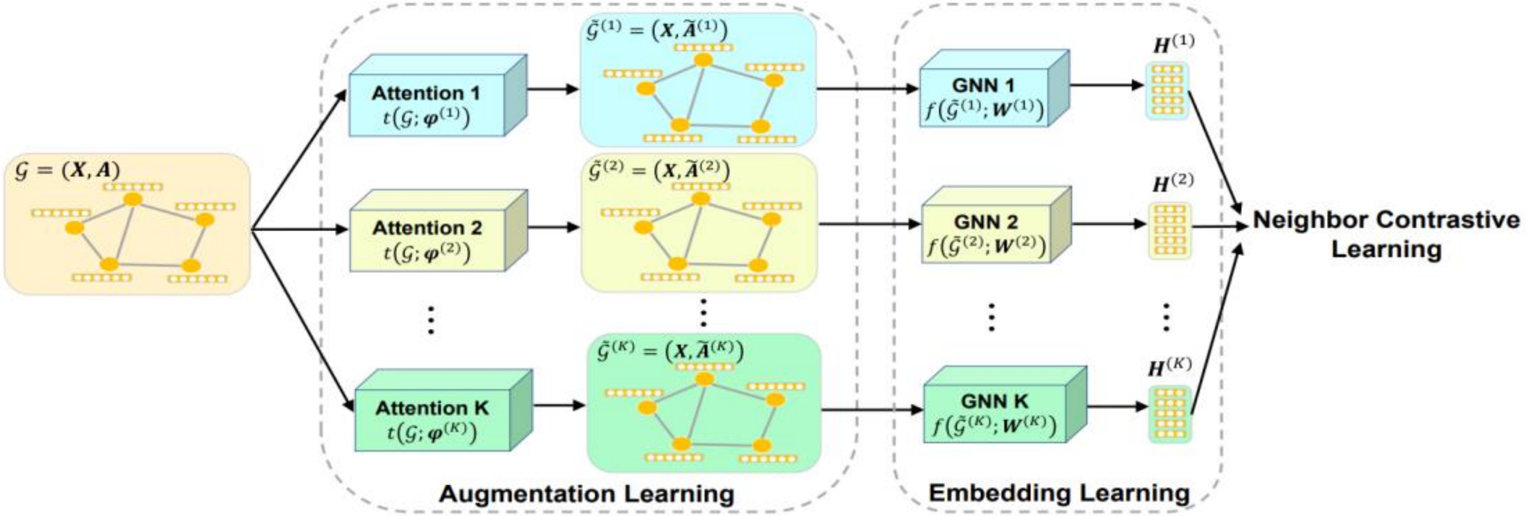
$$\ell(\mathbf{z}_{G_i}^{(f)},\mathbf{z}_{G_i}^{(s)}) = -\log\frac{e^{\mathrm{sim}(\mathbf{z}_{G_i}^{(f)},\mathbf{z}_{G_i}^{(s)})/\tau}}{\sum_{G_j\in\mathcal{B}\setminus G_i}e^{\mathrm{sim}(\mathbf{z}_{G_i}^{(f)},\mathbf{z}_{G_j}^{(s)})/\tau}}, \quad (6)$$

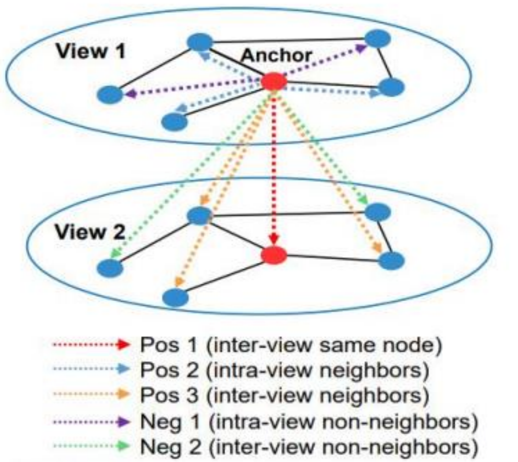
concatenate $\mathbf{h}_{G_i}^{(f)}$ with $\mathbf{h}_{G_i}^{(s)} \rightarrow \mathbf{z}_{G_i} \xrightarrow{\text{K-means}} C = \{\mathbf{c}_i\}_{i=1}^{K}$

$$\mathcal{L}_{group} = -\frac{1}{|\mathcal{B}|}\sum_{G_i\in\mathcal{B}}\log\frac{e^{\mathrm{sim}(\mathbf{z}_{G_i},\mathbf{c}_j)/\tau_j}}{\sum_{c_k\in C\setminus c_i}e^{\mathrm{sim}(\mathbf{z}_{G_i},\mathbf{c}_k)/\tau_k}}, \quad (7)$$

2023_AAAI_Neighbor Contrastive Learning on Learnable Graph Augmentation

Method





(c) Neighbor Contrastive Loss (Ours)

GNN encoder $f(\cdot\,; W^{(k)})$

$$h_i^{(k)} = \mathrm{ELU}\left(\sum_{v_j \in \mathcal{N}_i \cup \{v_i\}} \tilde{A}_{ij}^{(k)} W^{(k)} x_j\right) \tag{2}$$

$$h_i = ||_{k=1}^{K} h_i^{(k)} \tag{3}$$

$$\ell(H^{(1)}, H^{(2)}) = \frac{1}{2N} \sum_{i=1}^{N} \left[\ell(h_i^{(1)}) + \ell(h_i^{(2)})\right] \tag{5}$$

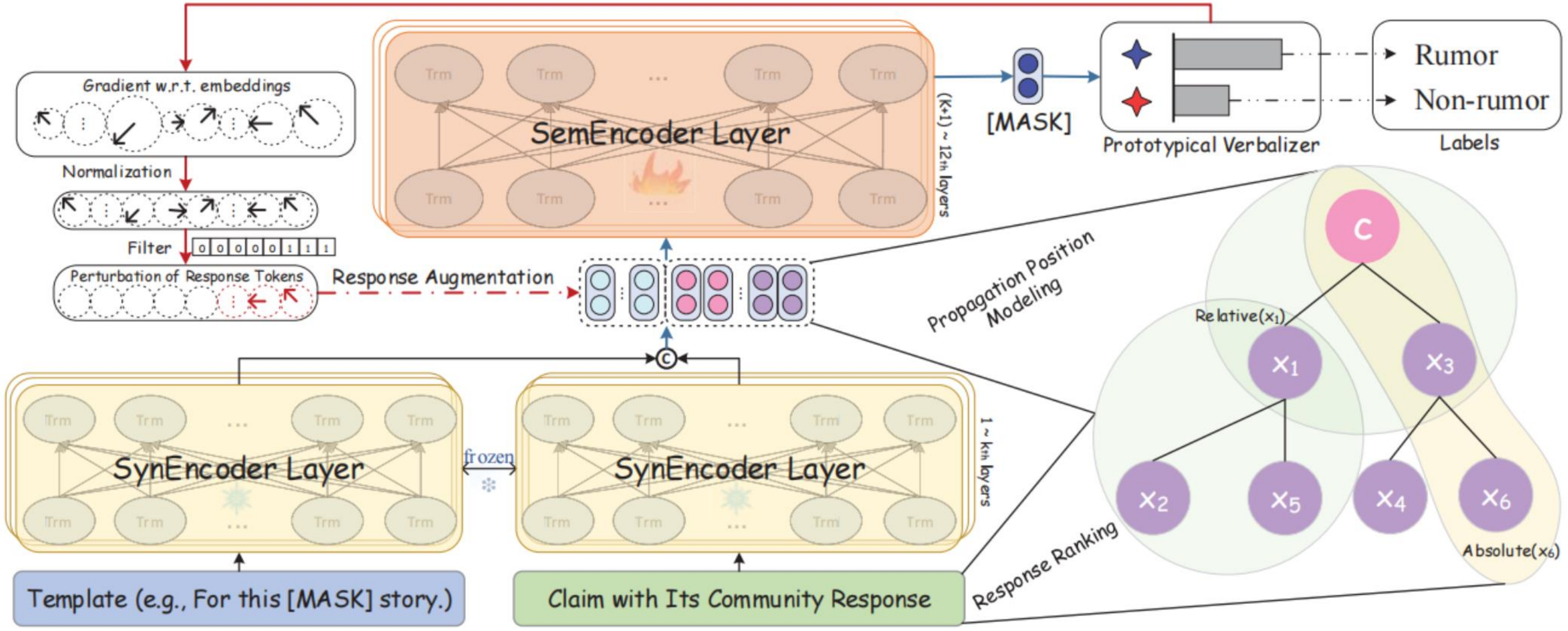
$$\mathcal{L} = \frac{1}{K} \sum_{k=1, k \neq l}^{K} \ell(H^{(k)}, H^{(l)}) \tag{6}$$

$$\ell(h_i^{(1)}) = -\log \frac{\left(e^{\theta(h_i^{(1)}, h_i^{(2)})/\tau} + \sum_{v_j \in \mathcal{N}_i}\left(e^{\theta(h_i^{(1)}, h_j^{(1)})/\tau} + e^{\theta(h_i^{(1)}, h_j^{(2)})/\tau}\right)\right) / (2|\mathcal{N}_i| + 1)}{e^{\theta(h_i^{(1)}, h_i^{(2)})/\tau} + \sum_{j \neq i}\left(\underline{e^{\theta(h_i^{(1)}, h_j^{(1)})/\tau}} + \underline{e^{\theta(h_i^{(1)}, h_j^{(2)})/\tau}}\right)} \tag{4}$$

$$\underline{\sum_{j \neq i} e^{\theta(h_i^{(1)}, h_j^{(1)})/\tau}} = \underbrace{\sum_{v_j \in \mathcal{N}_i} e^{\theta(h_i^{(1)}, h_j^{(1)})/\tau}}_{\text{intra-view pos}} + \underbrace{\sum_{v_j \notin \mathcal{N}_i} e^{\theta(h_i^{(1)}, h_j^{(1)})/\tau}}_{\text{intra-view neg}}$$

$$\underline{\sum_{j \neq i} e^{\theta(h_i^{(1)}, h_j^{(2)})/\tau}} = \underbrace{\sum_{v_j \in \mathcal{N}_i} e^{\theta(h_i^{(1)}, h_j^{(2)})/\tau}}_{\text{inter-view pos}} + \underbrace{\sum_{v_j \notin \mathcal{N}_i} e^{\theta(h_i^{(1)}, h_j^{(2)})/\tau}}_{\text{inter-view neg}}$$

2023_AAAI_Zero-Shot Rumor Detection with Propagation Structure via Prompt Learning



Given the *[MASK]* token representation H_i^m of a training example C_i, we minimize a prototypical loss that is the negative log-likelihood:

$$\mathcal{L}_{proto} = -log \frac{e^{\mathcal{S}(H_i^m, l_y)}}{\sum_{y'} e^{\mathcal{S}(H_i^m, l_{y'})}} \quad (6)$$

$$\mathcal{L}_{con} = -\frac{1}{B_{y_i} - 1} \sum_j \mathbb{1}_{[i \neq j]} \mathbb{1}_{[y_i = y_j]} log \frac{e^{\mathcal{S}(H_i^m, H_j^m)}}{\sum_{j'} \mathbb{1}_{[i \neq j']} e^{\mathcal{S}(H_i^m, H_{j'}^m)}} \quad (7)$$

2022_CIKM_Multi-level Contrastive Learning Framework for Sequential Recommendation

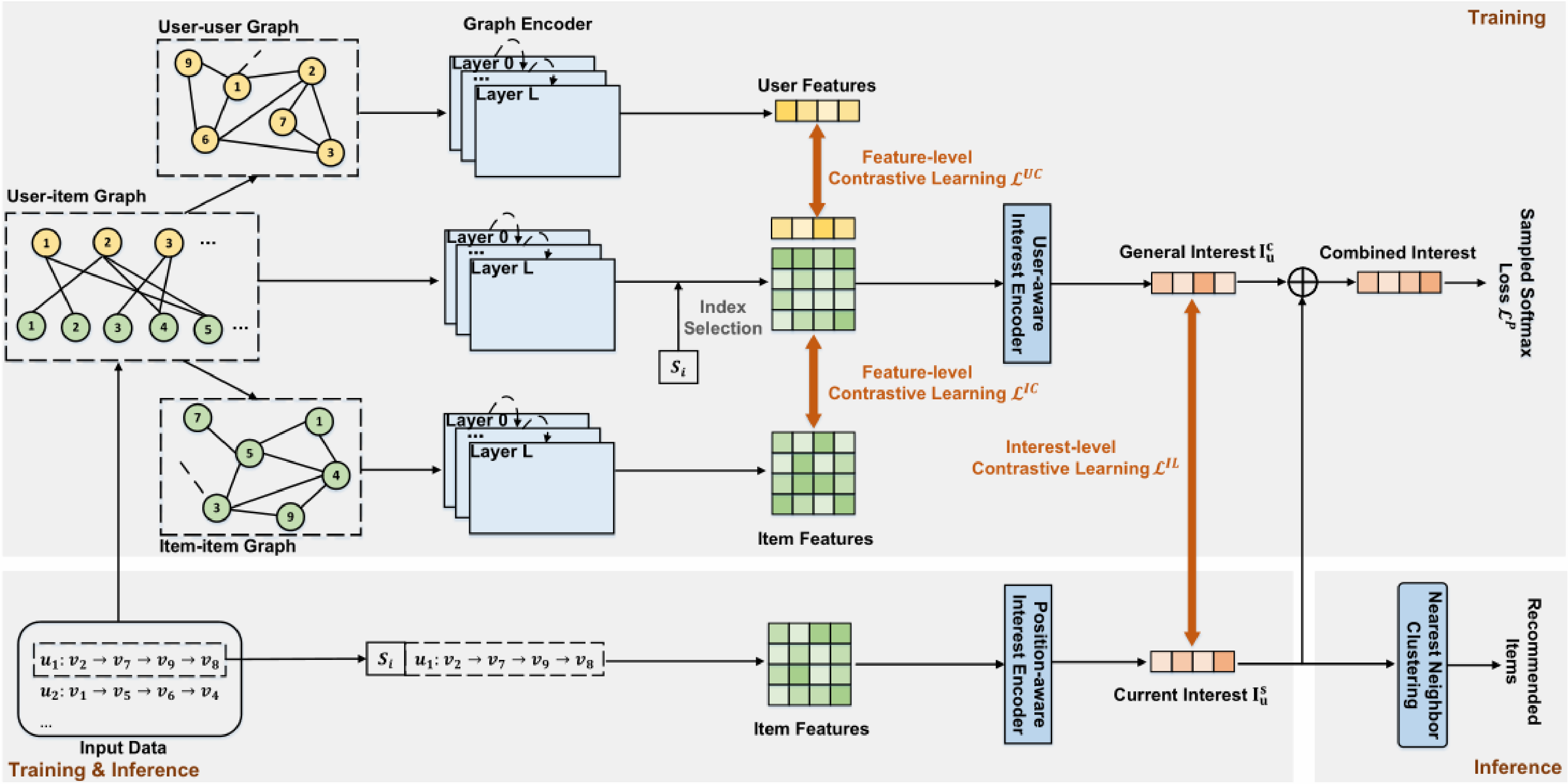


Figure 2: An overview of the proposed framework. ⊕ denotes the element-wise summation.

$$\mathbf{T}^{I,s} = \left(\mathbf{W}_2^p \sigma(\mathbf{W}_1^p \mathbf{I}_u^s + \mathbf{b}_1^p) + \mathbf{b}_2^p\right),$$

$$\mathbf{T}^{I,c} = \left(\mathbf{W}_2^p \sigma(\mathbf{W}_1^p \mathbf{I}_u^c + \mathbf{b}_1^p) + \mathbf{b}_2^p\right),$$

$$\mathcal{L}^{IL} = \sum_{i=1} -\log \frac{\Psi\left(\mathbf{T}_i^{I,s}, \mathbf{T}_i^{I,c}\right)}{\sum_j \Psi\left(\mathbf{T}_i^{I,s}, \mathbf{T}_j^{I,c}\right) + \sum_{j \neq i} \Psi\left(\mathbf{T}_i^{I,s}, \mathbf{T}_j^{I,s}\right)},$$

2022_TNNLS_Prototypical Graph Contrastive Learning

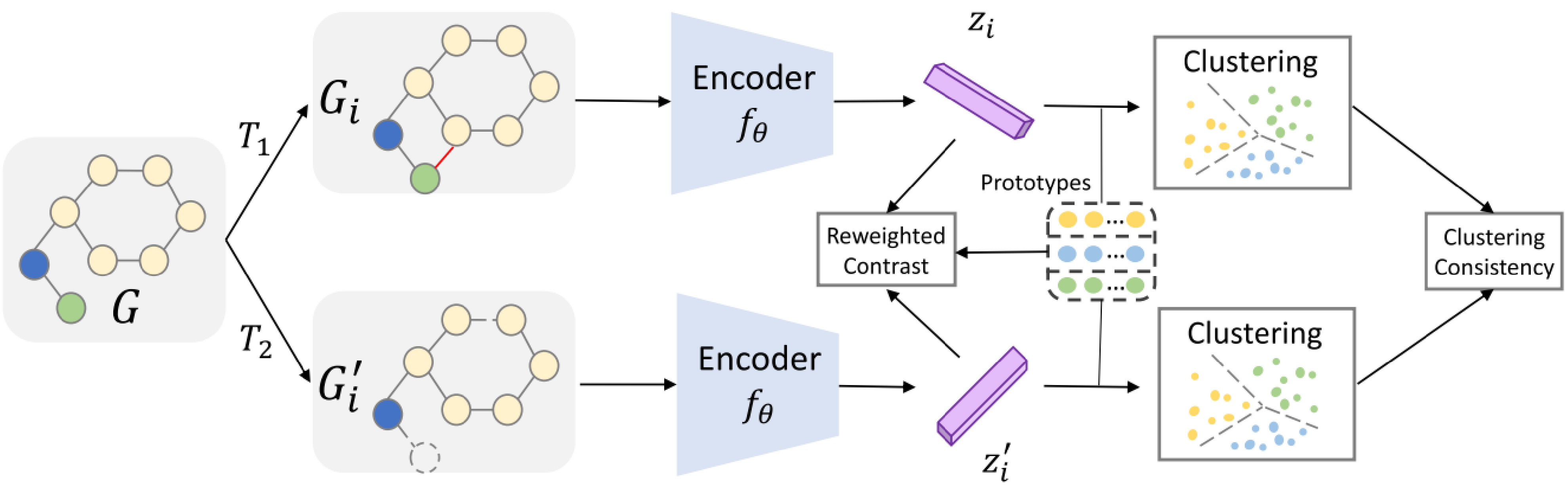
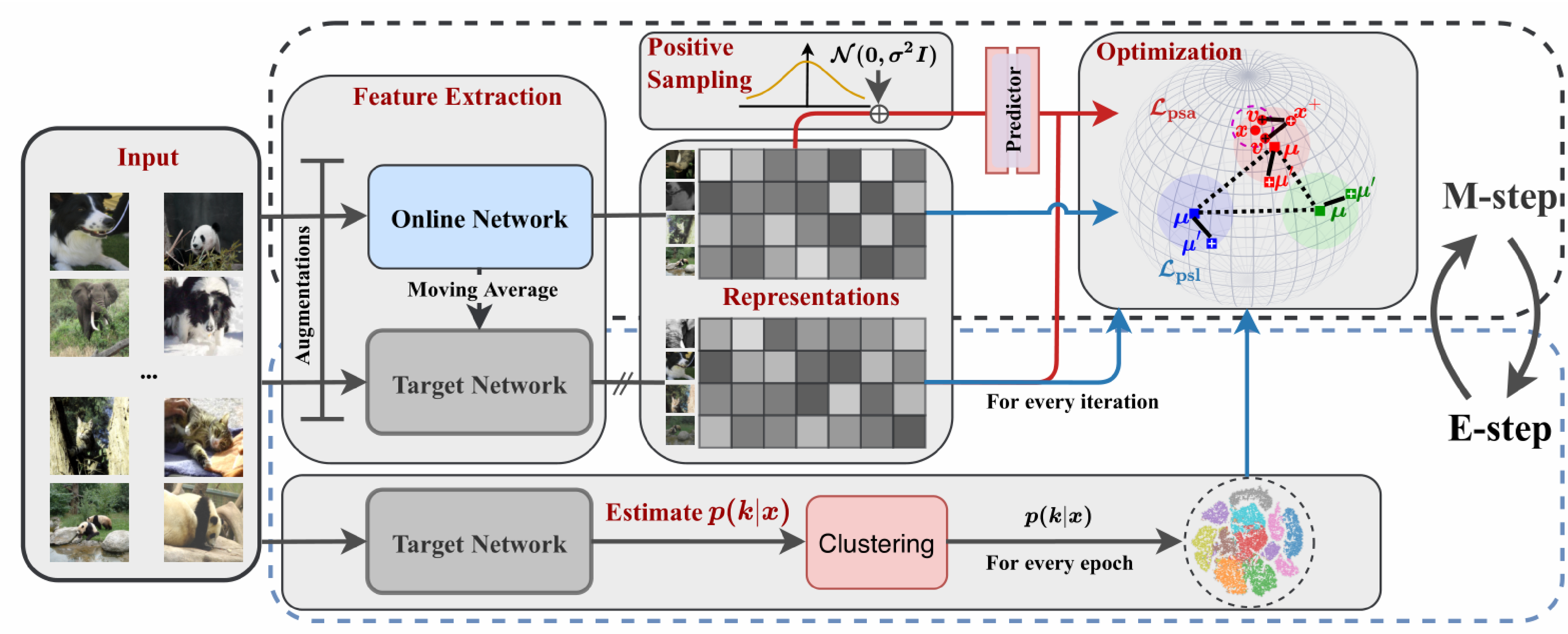


Fig. 2. Overview of PGCL. Two graph data augmentations T_1 and T_2 are applied to the input graph G. Then, two graph views G_i and G_i' are fed into the shared encoder f_θ (including GNNs and a projection head) to extract the graph representations z_i and z_i'. We perform the online clustering via assigning the representations of samples within a batch to prototype vectors (cluster centroids). The representations are learned via encouraging the clustering consistency between correlated views (Section IV-A) and a reweighted contrastive objective (Section IV-B), where prototype vectors are also updated along with the encoder parameters by backpropagation.

$$z_i,\ z_i' = f_\theta(G_i),\ f_\theta(G_i')$$

$$\mathcal{L}_{\textbf{InfoNCE}} = -\sum_{i=1}^{n} \log \frac{\exp\left(z_i \cdot z_i'/\tau\right)}{\exp\left(z_i \cdot z_i'/\tau\right) + \sum_{j=1, j\neq i}^{2N} \exp\left(z_i \cdot z_j/\tau\right)} \quad (3)$$

2022_TPAMI_Learning Representation for Clustering via Prototype Scattering and Positive Sampling



$$\mathcal{L}_{\text{psl}} = \frac{1}{K}\sum_{k=1}^{K} -\log \frac{\exp\left(\frac{\mu_k^{\top}\mu_k'}{\tau}\right)}{\exp\left(\frac{\mu_k^{\top}\mu_k'}{\tau}\right) + \sum_{\substack{j=1\\ j\neq k}}^{K} \exp\left(\frac{\mu_k^{\top}\mu_j}{\tau}\right)} \approx \underbrace{\frac{1}{K}\sum_{k=1}^{K} -\frac{\mu_k^{\top}\mu_k'}{\tau}}_{\text{prototypical alignment}} + \underbrace{\frac{1}{K}\sum_{k=1}^{K} \log \sum_{\substack{j=1\\ j\neq k}}^{K} \exp\left(\frac{\mu_k^{\top}\mu_j}{\tau}\right)}_{\text{prototypical uniformity}}$$

2022_AAAI_Contrast and Generation Make BART a Good Dialogue Emotion Recognizer

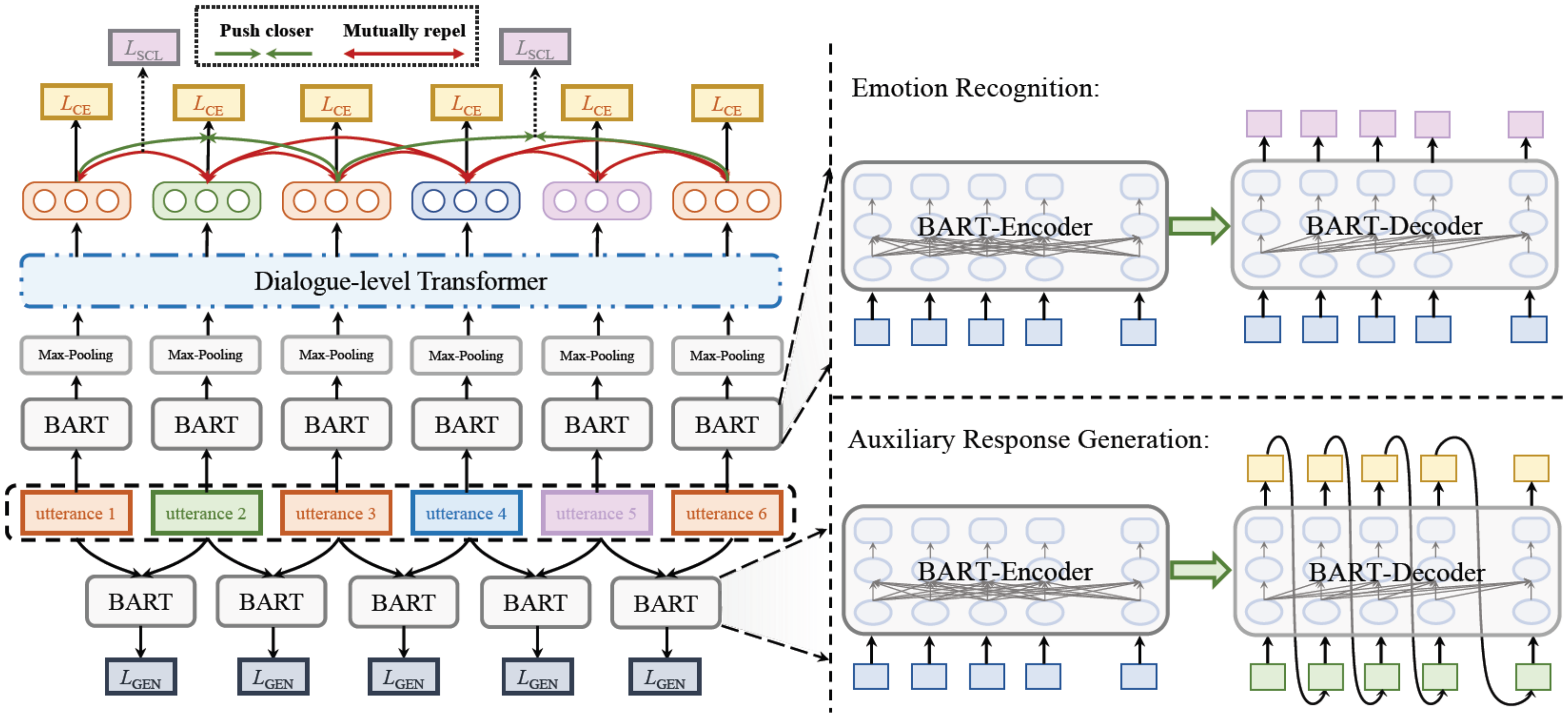


Figure 2: The overall framework of CoG-BART. The utterance is fed into BART for N utterances in a batch to get its hidden state. The representation of the utterance obtained after max-pooling the hidden state of each utterance is fed to the upper-level dialogue-level Transformer for modeling context dependencies. The obtained context-dependent utterance representations are utilized to compute the cross-entropy loss and supervised contrastive loss. In addition, the two adjacent utterance pairs are used for the auxiliary response generation.

$$X = [H_{d\text{-}win}, \overline{H}_{d\text{-}win}], \quad (10)$$

$$\mathcal{L}_{\mathrm{SCL}} = \sum_{i\in I} \frac{-1}{|P(i)|} \sum_{p\in P(i)} \mathrm{SIM}(p, i), \quad (11)$$

$$\mathrm{SIM}(p, i) = \log \frac{\exp((X_i \cdot X_p)/\tau)}{\sum_{a\in A(i)} \exp(X_i \cdot X_a/\tau)}, \quad (12)$$

2022_EMNLP_Supervised Prototypical Contrastive Learning for Emotion Recognition in Conversation

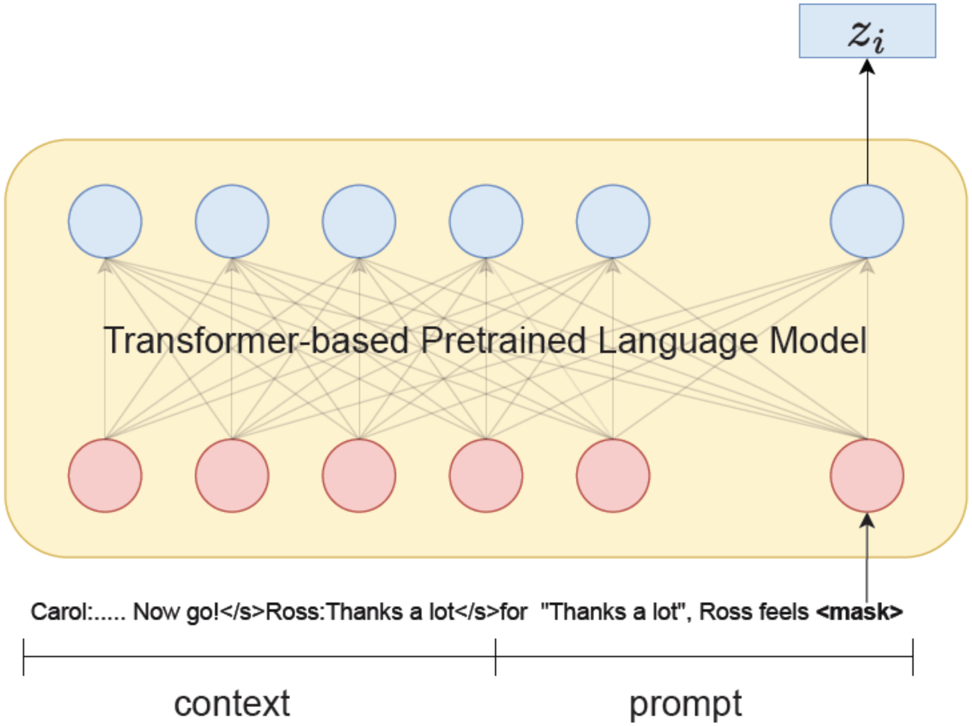


Figure 3: The architecture of our prompt-based context encoder.

$$\mathcal{P}_{sup}(i) = \sum_{z_p \in P(i)} \mathcal{F}(z_i, z_p) \tag{6}$$

$$\mathcal{L}_i^{sup} = -\log \frac{1}{|P(i)|} \frac{\mathcal{P}_{sup}(i)}{\mathcal{N}_{sup}(i)} \tag{7}$$

$$\mathcal{L}_i^{spcl} = -\log \left(\frac{1}{|P(i)|+1} \cdot \frac{\mathcal{P}_{spcl}(i)}{\mathcal{N}_{spcl}(i)} \right) \tag{12}$$

$$\mathcal{L}^{spcl} = \sum_{i=1}^{N} \mathcal{L}_i^{spcl} \tag{13}$$

2022_WWW_Rumor Detection on Social Media with Graph Adversarial Contrastive Learning

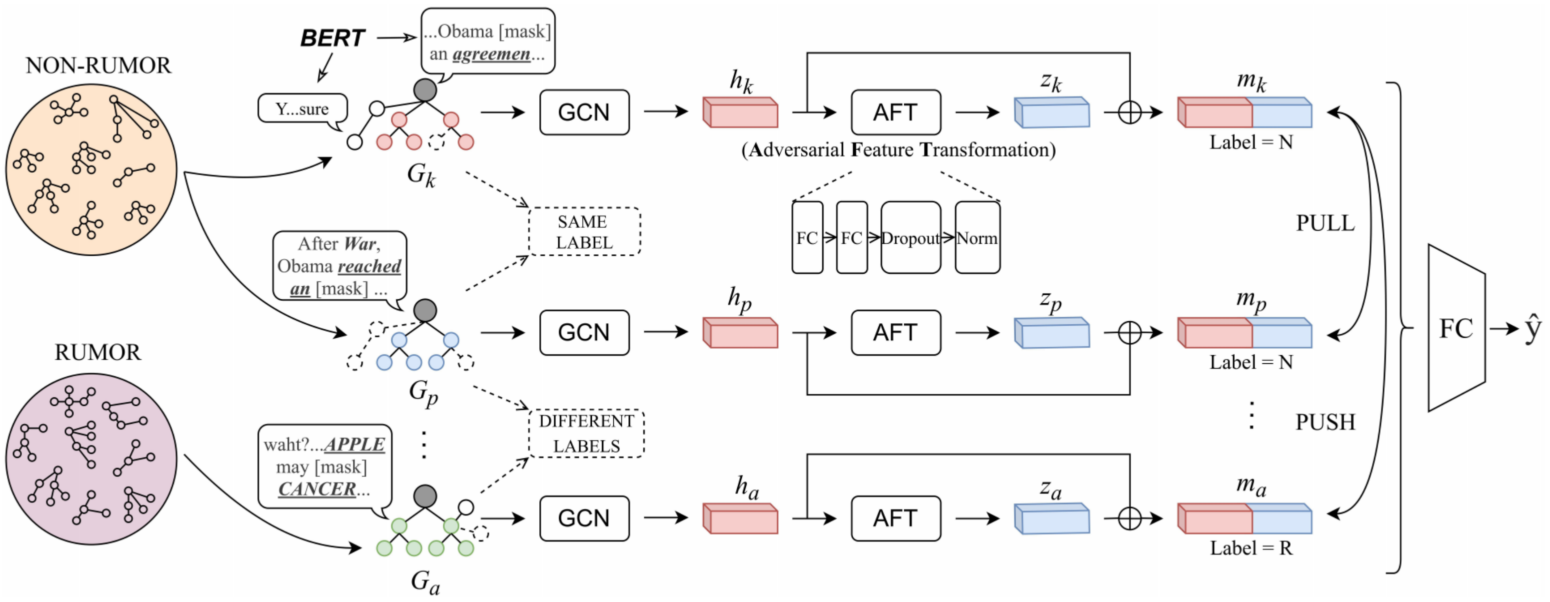


Figure 3: Overview of our GACL rumor detection model. Given an input batch of data, we split it into two clusters referring to rumors and non-rumors. Then, the various types of data augmentation strategies are applied to generate the perturbed rumor trees such as G_k, G_p and G_a (that are just generic elements and used as examples). Next, the representation of the rumor trees is calculated using BERT and GCN to obtain a 64-dimensional feature vector h. Finally, h and the adversarial feature z generated by the AFT module are concatenated together for subsequent contrastive training and classification.

$$\mathcal{L}_{sup} = -\sum_{k \in K} log \left\{ \frac{1}{|P(k)|} \sum_{p \in P(k)} \frac{exp(sim(m_k, m_p)\tau)}{\sum_{a \in A(k)} exp(sim(m_k, m_a)\tau)} \right\}, \quad (10)$$

2022_SIGIR_Knowledge Graph Contrastive Learning for Recommendation

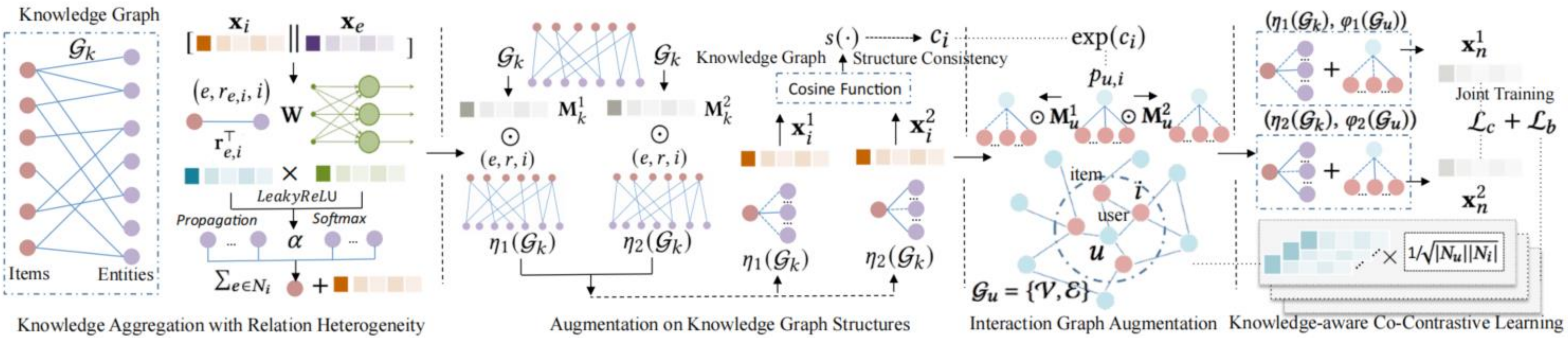


Figure 3: The overall architecture of our proposed KGCL. Knowledge-aware co-contrastive learning with augmentation functions on both knowledge graph $\eta(\cdot)$ and user-item interaction graph $\varphi(\cdot)$. Our contrastive objective $\mathcal{L}_c$ is jointly optimized with main embedding space shared by the knowledge graph aggregation and graph-based CF encoder.

$$\mathcal{L}_c = \sum_{n \in \mathcal{V}} -\log \frac{\exp(s(\mathbf{x}_n^1, \mathbf{x}_n^2)/\tau)}{\sum_{n' \in \mathcal{V}, n' \neq n} \exp(s(\mathbf{x}_n^1, \mathbf{x}_{n'}^2)/\tau)} \tag{8}$$

2022_KDD_Label-enhanced Prototypical Network with Contrastive Learning for Multi-label Few-shot Aspect Category Detection

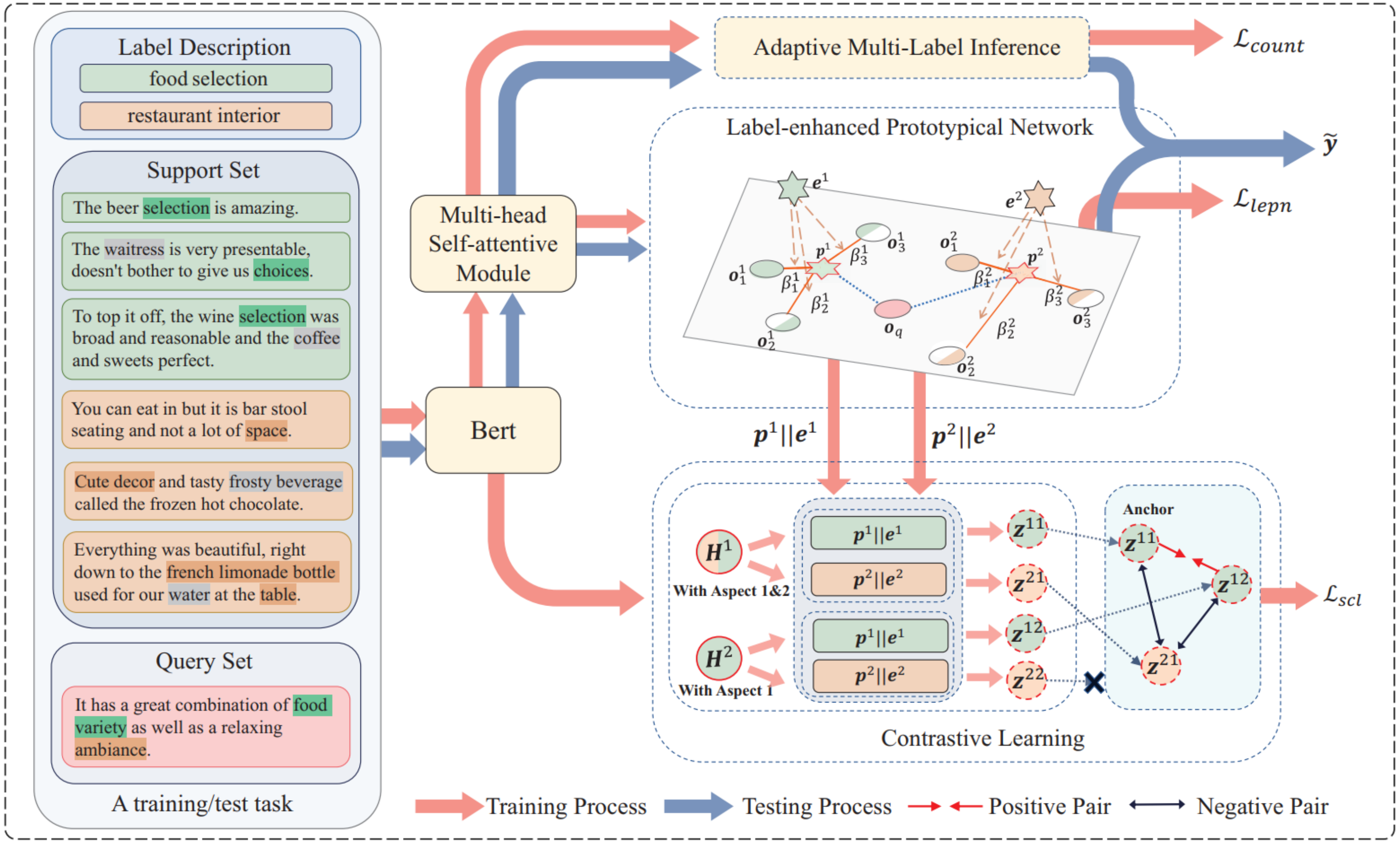


Figure 2: Illustration of our proposed method LPN.

Integrating with Contrastive Learning

$$P = \{p^1, p^2, ..., p^N\}$$

$$E = \{e^1, e^2, ..., e^N\}$$

$$\{a^1, a^2, ..., a^N\}, \text{ where } a^i = [p^i || e^i] \in \mathbb{R}^{2d}$$

$$z^i = g^i H^T, \qquad (10)$$

$$g^i = \text{softmax}((W_a a^i + b_a)^T H), \qquad (11)$$

$$Z = \{z^{ij} \in \mathbb{R}^d | i \in \{1, 2, ..., N\}, j \in \{1, 2, ..., N_t\}\}$$

$$Y = \{y^{ij} \in \{0, 1\} | i \in \{1, 2, ..., N\}, j \in \{1, 2, ..., N_t\}\}$$

$$\mathcal{L}_{scl}^{ij} = -\frac{1}{|\Lambda^{ij}|} \sum_{z^{ik} \in \Lambda^{ij}} \log \frac{\exp(z^{ij} \cdot z^{ik} / \tau)}{\sum_{z^* \in \Gamma^{ij}} \exp(z^{ij} \cdot z^* / \tau)}, \qquad (12)$$

$$\mathcal{L}_{scl} = \frac{1}{|I|} \sum_{z^{ij} \in I} \mathcal{L}_{scl}^{ij}. \qquad (13)$$

2022_SIGIR_Multi-level Cross-view Contrastive Learning for Knowledge-aware Recommender System

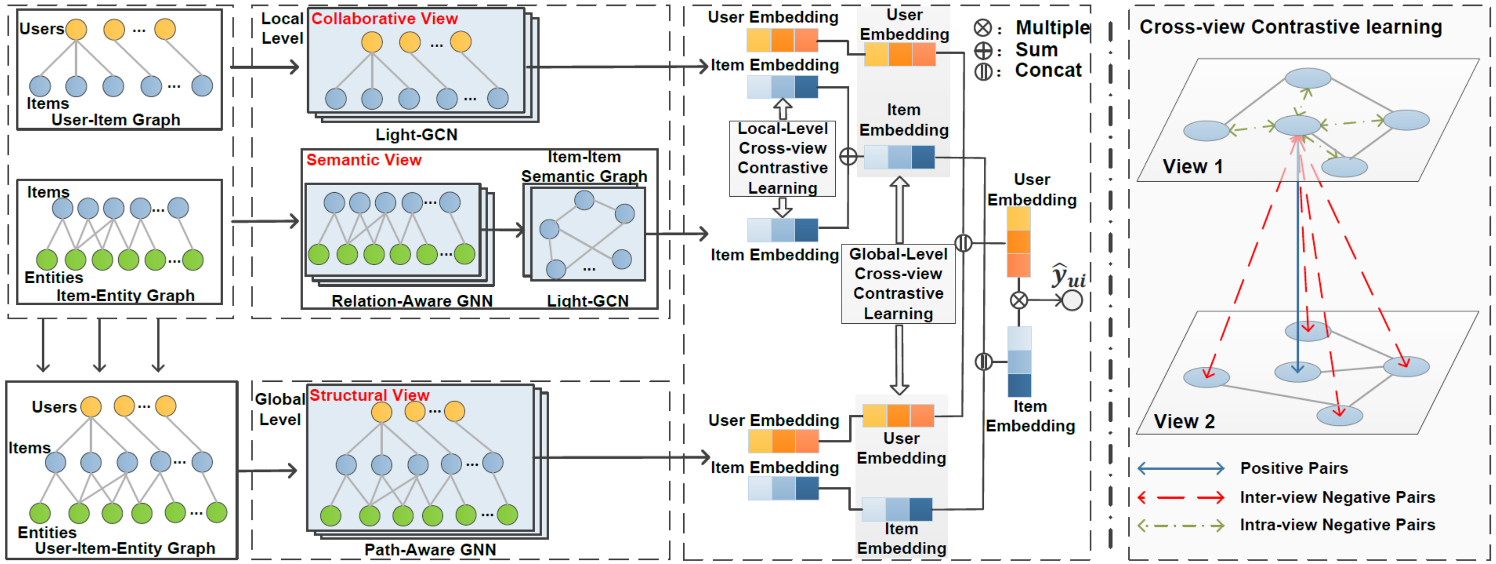


Figure 2: Illustration of the proposed MCCLK model. The left subfigure shows model framework of MCCLK; and the right subfigure presents the details of cross-view contrastive learning mechanism. Best viewed in color.

$$\mathcal{L}^{local} = -\log \frac{e^{s\left(z_i^s\_p, z_i^c\_p\right)/\tau}}{\underbrace{e^{s\left(z_i^s\_p, z_i^c\_p\right)/\tau}}_{\text{positive pair}} + \underbrace{\sum_{k\neq i} e^{s\left(z_i^s\_p, z_k^s\_p\right)/\tau}}_{\text{intra-view negative pairs}} + \underbrace{\sum_{k\neq i} e^{s\left(z_i^s\_p, z_k^c\_p\right)/\tau}}_{\text{inter-view negative pairs}}}, \qquad (10)$$

$$\mathcal{L}_i^{g} = -\log \frac{e^{s\left(z_i^g\_p, z_i^l\_p\right)/\tau}}{\underbrace{e^{s\left(z_i^g\_p, z_i^l\_p\right)/\tau}}_{\text{positive pair}} + \underbrace{\sum_{k\neq i} e^{s\left(z_i^g\_p, z_k^g\_p\right)/\tau}}_{\text{intra-view negative pairs}} + \underbrace{\sum_{k\neq i} e^{s\left(z_i^g\_p, z_k^l\_p\right)/\tau}}_{\text{inter-view negative pairs}}},$$

$$\mathcal{L}_i^{l} = -\log \frac{e^{s\left(z_i^l\_p, z_i^g\_p\right)/\tau}}{\underbrace{e^{s\left(z_i^l\_p, z_i^g\_p\right)/\tau}}_{\text{positive pair}} + \underbrace{\sum_{k\neq i} e^{s\left(z_i^l\_p, z_k^l\_p\right)/\tau}}_{\text{intra-view negative pairs}} + \underbrace{\sum_{k\neq i} e^{s\left(z_i^l\_p, z_k^g\_p\right)/\tau}}_{\text{inter-view negative pairs}}}, \qquad (15)$$

2022_NAACL_Detect Rumors in Microblog Posts for Low-Resource Domains via Adversarial Contrastive Learning

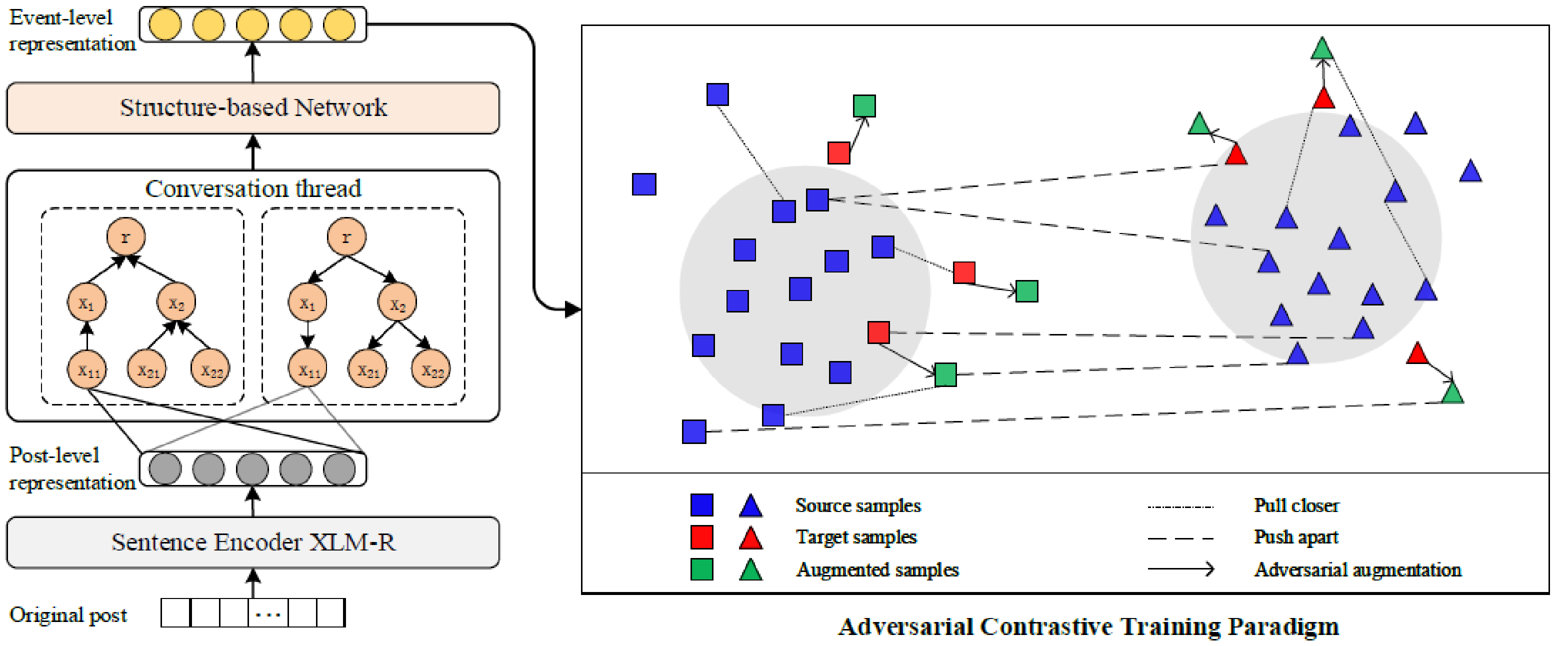


Figure 2: The overall architecture of our proposed method. For source and small target training data, we first obtain post-level representations after cross-lingual sentence encoding, then train the structure-based network with the adversarial contrastive objective. For target test data, we extract the event-level representations to detect rumors.

$$\mathcal{L}^s_{SCL} = -\frac{1}{N^s}\sum_{i=1}^{N^s}\frac{1}{N_{y_i^s}-1}\sum_{j=1}^{N^s}\mathbb{1}_{[i\neq j]}\mathbb{1}_{[y_i^s=y_j^s]} \; log\frac{exp(\mathrm{sim}(o_i^s,o_j^s)/\tau)}{\sum_{k=1}^{N^s}\mathbb{1}_{[i\neq k]}exp(\mathrm{sim}(o_i^s,o_k^s)/\tau)} \tag{5}$$

$$\mathcal{L}^t_{SCL} = -\frac{1}{N^t}\sum_{i=1}^{N^t}\frac{1}{N_{y_i^t}}\sum_{j=1}^{N^s}\mathbb{1}_{[y_i^t=y_j^s]} \; log\frac{exp(\mathrm{sim}(o_i^t,o_j^s)/\tau)}{\sum_{k=1}^{N^s}exp(\mathrm{sim}(o_i^t,o_k^s)/\tau)} \tag{6}$$

2022_WWW_Zero-Shot Stance Detection via Contrastive Learning

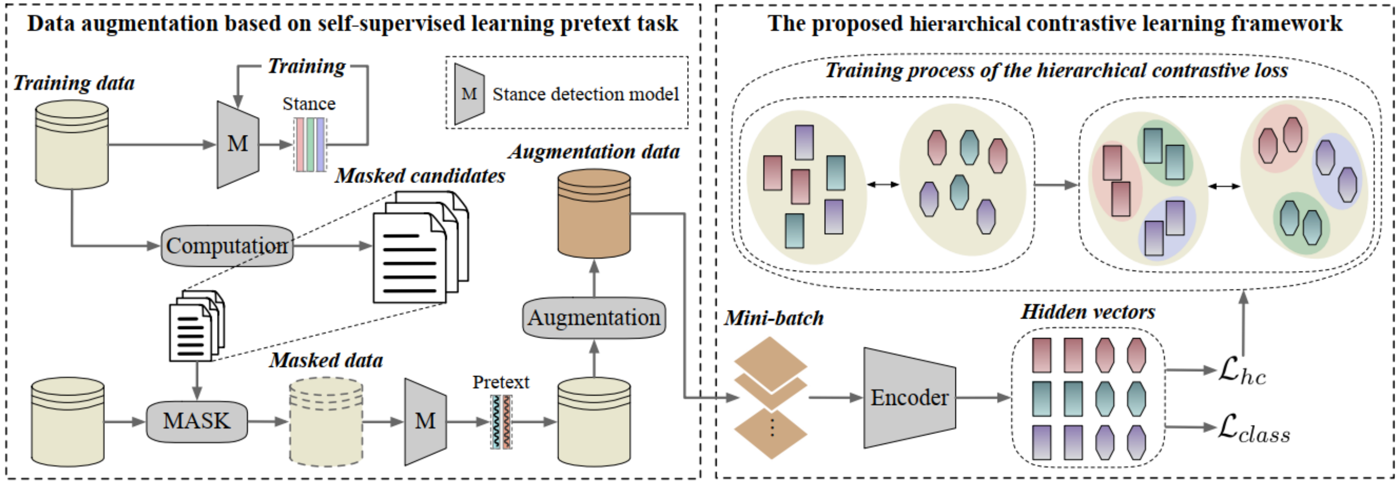


Figure 2: The architecture of the proposed PT-HCL framework. Shapes in gradient colors represent hidden vectors, and different shapes represent different labels of the surrogate supervised signal provided by the pretext task.

$$\mathcal{L}_{hc} = \frac{-1}{N_b} \sum_{\boldsymbol{h}_i \in \mathcal{B}} \ell(\boldsymbol{h}_i) \tag{3}$$

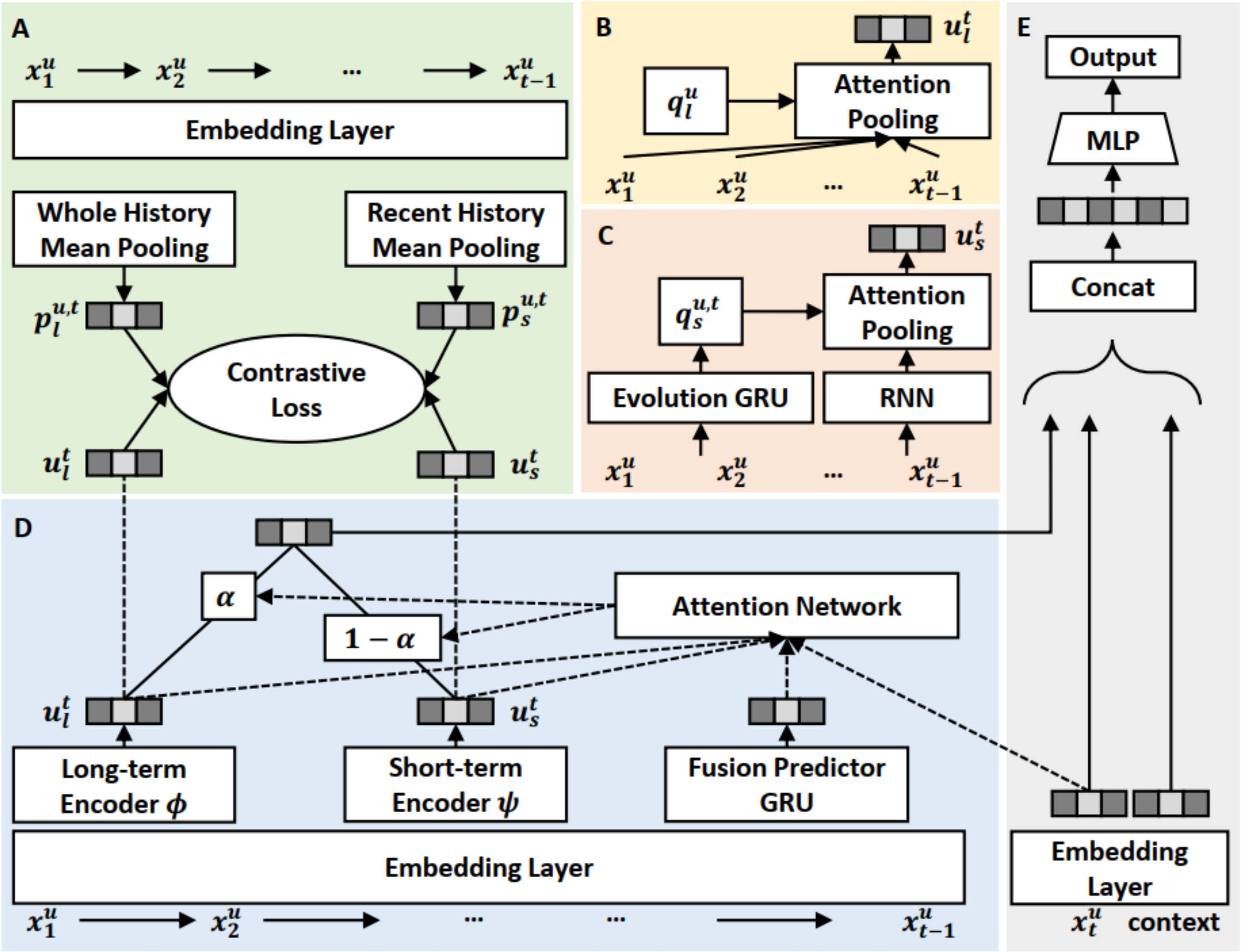
$$\ell(\boldsymbol{h}_i) = \log\Big(\frac{\sum_{j=1}^{N_b} \mathbb{1}_{[j \neq i]} \mathbb{1}_{[p^i = p^j]} f(\boldsymbol{h}_i, \boldsymbol{h}_j)}{\sum_{k=1}^{N_b} \mathbb{1}_{[k \neq i]} f(\boldsymbol{h}_i, \boldsymbol{h}_k)} \times \alpha \frac{\sum_{j=1}^{N_b} \mathbb{1}_{[j \neq i]} \mathbb{1}_{[p^i = p^j]} \mathbb{1}_{[y^i = y^j]} g(\boldsymbol{h}_i, \boldsymbol{h}_j)}{\sum_{k=1}^{N_b} \mathbb{1}_{[k \neq i]} \mathbb{1}_{[p^i = p^k]} g(\boldsymbol{h}_i, \boldsymbol{h}_k)} \Big) \tag{4}$$

$$f(\boldsymbol{u}, \boldsymbol{v}) = \exp(sim(\boldsymbol{u}, \boldsymbol{v})/\tau_p) \tag{5}$$

$$g(\boldsymbol{u}, \boldsymbol{v}) = \exp(sim(\boldsymbol{u}, \boldsymbol{v})/\tau_y) \tag{6}$$

where $\mathbb{1}_{[i=j]} \in \{0, 1\}$ is an indicator function evaluating to 1 iff $i = j$. $sim(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{u}^\top \boldsymbol{v}/\|\boldsymbol{u}\|\|\boldsymbol{v}\|$ denotes the cosine similarity between L_2 normalized vectors $\boldsymbol{u}$ and $\boldsymbol{v}$. α, τ_p, and τ_y are tuned hyperparameters to control the separable strength between positive and negative samples from augmentation labels and stance labels. Here, $\tau_p < \tau_y$. In this way, the novel contrastive learning scenario preferentially pulls together the clusters of points belonging to the same augmentation label in embedding space, and further slightly pulls together the clusters of points belonging to the same stance polarity in each separated embedding space. Simultaneously, pushing apart clusters of samples from different augmentation labels.

2022_WWW_Disentangling Long and Short-Term Interests for Recommendation



$$sim(\boldsymbol{u}_l^t, \boldsymbol{p}_l^{u,t}) > sim(\boldsymbol{u}_l^t, \boldsymbol{p}_s^{u,t}), \tag{19}$$

$$sim(\boldsymbol{p}_l^{u,t}, \boldsymbol{u}_l^t) > sim(\boldsymbol{p}_l^{u,t}, \boldsymbol{u}_s^t), \tag{20}$$

$$sim(\boldsymbol{u}_s^t, \boldsymbol{p}_s^{u,t}) > sim(\boldsymbol{u}_s^t, \boldsymbol{p}_l^{u,t}), \tag{21}$$

$$sim(\boldsymbol{p}_s^{u,t}, \boldsymbol{u}_s^t) > sim(\boldsymbol{p}_s^{u,t}, \boldsymbol{u}_l^t), \tag{22}$$

$$\mathcal{L}_{\text{con}}^{u,t} = f(\boldsymbol{u}_l, \boldsymbol{p}_l, \boldsymbol{p}_s) + f(\boldsymbol{p}_l, \boldsymbol{u}_l, \boldsymbol{u}_s) + f(\boldsymbol{u}_s, \boldsymbol{p}_s, \boldsymbol{p}_l) + f(\boldsymbol{p}_s, \boldsymbol{u}_s, \boldsymbol{u}_l) \tag{25}$$

2022_ACL_JointCL_ A Joint Contrastive Learning Framework for Zero-Shot Stance Detection

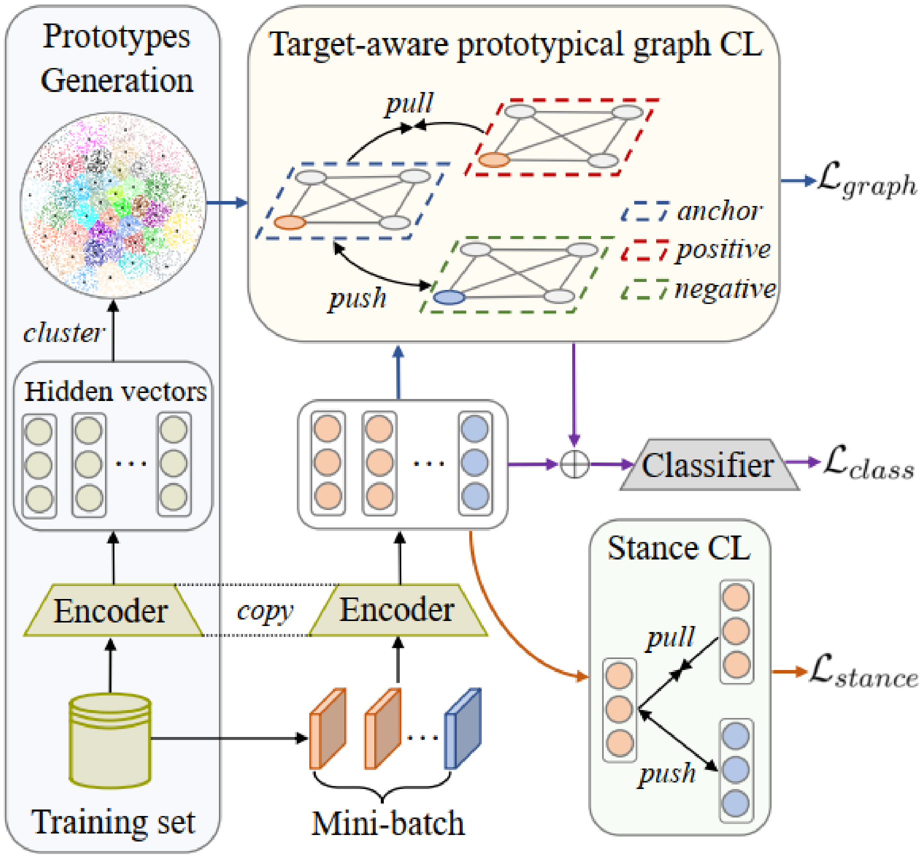


Figure 1: The architecture of our JointCL framework. $\oplus$ is vector concatenation. In the graphs, the gray ellipses denote prototypes, others denote hidden vectors. Vectors with the same color hold the same stance.

$$\mathcal{L}_{stance} = \frac{-1}{N_b} \sum_{\boldsymbol{h}_i \in \mathcal{B}} \ell^s(\boldsymbol{h}_i) \quad (2)$$

$$\ell^s(\boldsymbol{h}_i) = \log \frac{\sum_{j \in \mathcal{B} \setminus i} \mathbb{1}_{[y^i = y^j]} \exp(f(\boldsymbol{h}_i, \boldsymbol{h}_j)/\tau_s)}{\sum_{j \in \mathcal{B} \setminus i} \exp(f(\boldsymbol{h}_i, \boldsymbol{h}_j)/\tau_s)} \quad (3)$$

$$\mathcal{L}_{graph} = \frac{-1}{N_b} \sum_{\boldsymbol{\alpha}_i \in \mathcal{B}} \ell^g(\boldsymbol{\alpha}_i) \quad (6)$$

$$\ell^g(\boldsymbol{\alpha}_i) = \log \frac{\sum_{j \in \mathcal{B} \setminus i} \Phi(i,j) \exp(f(\boldsymbol{\alpha}_i, \boldsymbol{\alpha}_j)/\tau_g)}{\sum_{j \in \mathcal{B} \setminus i} \exp(f(\boldsymbol{\alpha}_i, \boldsymbol{\alpha}_j)/\tau_g)} \quad (7)$$

$$\Phi(i,j) = \begin{cases} 1 & \text{if } y^i = y^j \text{ and } p^i = p^j \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

2022_CVPR_Selective-Supervised Contrastive Learning with Noisy Labels

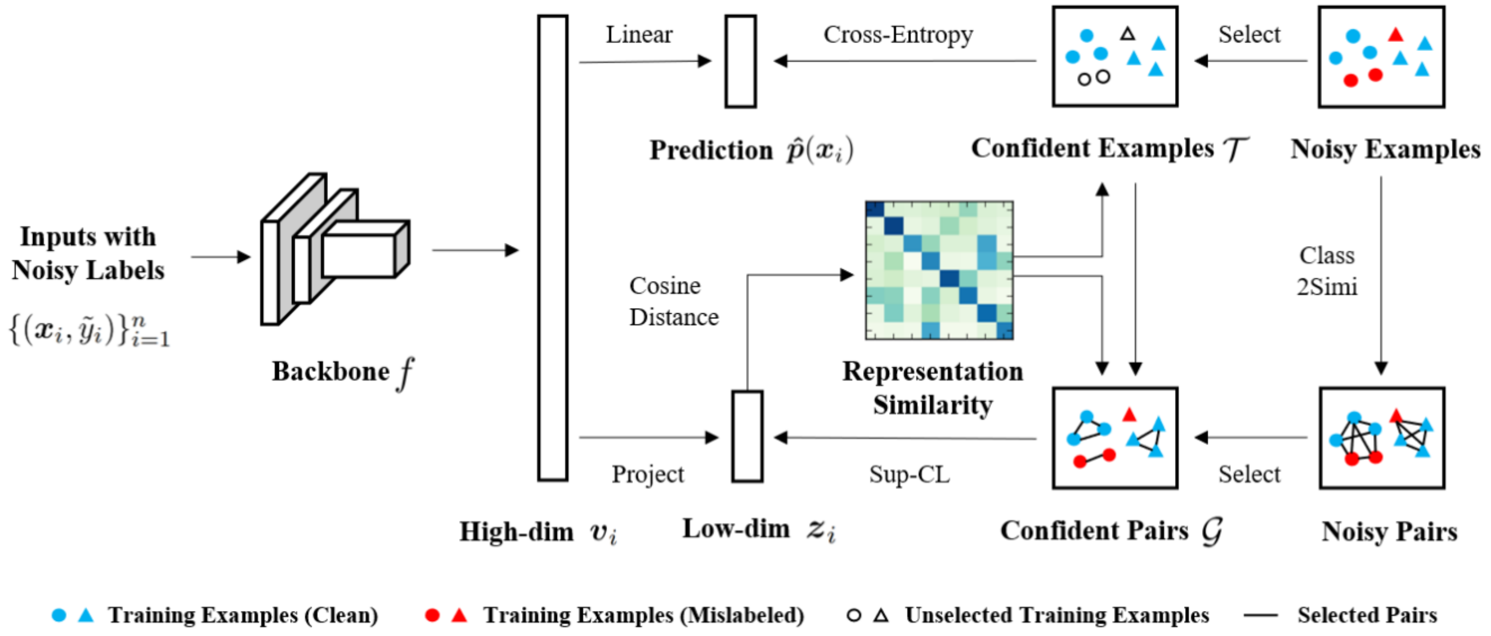


Figure 2. The illustration of the proposed Sel-CL, which progressively selects better confident pairs $\mathcal{G}$ for supervised contrastive learning based on the representation similarity. Without the noise rate prior, confident examples $\mathcal{T}$ are also obtained to help identify the pairs.

$$\mathcal{L} = \sum_{i \in I} \mathcal{L}_i(z_i) = \sum_{i \in I} \frac{-1}{|\mathcal{G}(i)|} \sum_{g \in \mathcal{G}(i)} \log \frac{\exp(z_i \cdot z_g / \tau)}{\sum_{a \in A(i)} \exp(z_i \cdot z_a / \tau)}, \tag{6}$$

$$\mathcal{L}_i^{\mathrm{MIX}}(z_i) = \lambda \mathcal{L}_a(z_i) + (1 - \lambda) \mathcal{L}_b(z_i), \tag{7}$$

2022_WWW_Intent Contrastive Learning for Sequential Recommendation

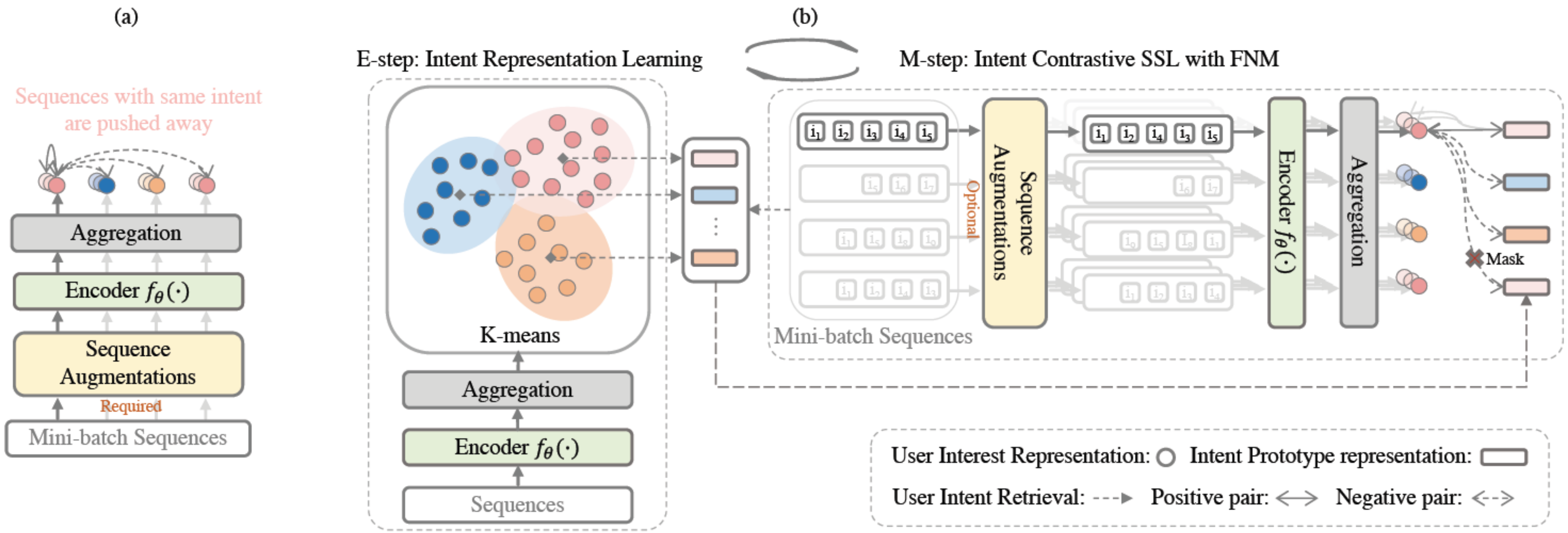


Figure 2: Overview of ICL. (a) An individual sequence level SSL for SR. (b) The proposed ICL for SR. It alternately performs intent representation learning and intent contrastive SSL with FNM within the generalized EM framework to maximizes mutual information (MIM) between a behavior sequence and its corresponding intent prototype.

$$\mathcal{L}_{\mathrm{ICL}} = \mathcal{L}_{\mathrm{ICL}}(\tilde{\mathbf{h}}_1^u, \mathbf{c}_u) + \mathcal{L}_{\mathrm{ICL}}(\tilde{\mathbf{h}}_2^u, \mathbf{c}_u), \quad (15)$$

and

$$\mathcal{L}_{\mathrm{ICL}}(\tilde{\mathbf{h}}_1^u, \mathbf{c}_u) = -\log \frac{\exp(\mathrm{sim}(\tilde{\mathbf{h}}_1^u, \mathbf{c}_u))}{\sum_{neg} \exp(\mathrm{sim}(\tilde{\mathbf{h}}_1^u, \mathbf{c}_{neg}))}, \quad (16)$$

$$\mathcal{L}_{\mathrm{ICL}}(\tilde{\mathbf{h}}_1^u, \mathbf{c}_u) = -\log \frac{\exp(\mathrm{sim}(\tilde{\mathbf{h}}_1^u, \mathbf{c}_u))}{\sum_{v=1}^{N} \mathbb{1}_{v \notin \mathcal{F}} \exp(\mathrm{sim}(\tilde{\mathbf{h}}_1, \mathbf{c}_v))}, \quad (17)$$

2022_WWW_Towards Unsupervised Deep Graph Structure Learning

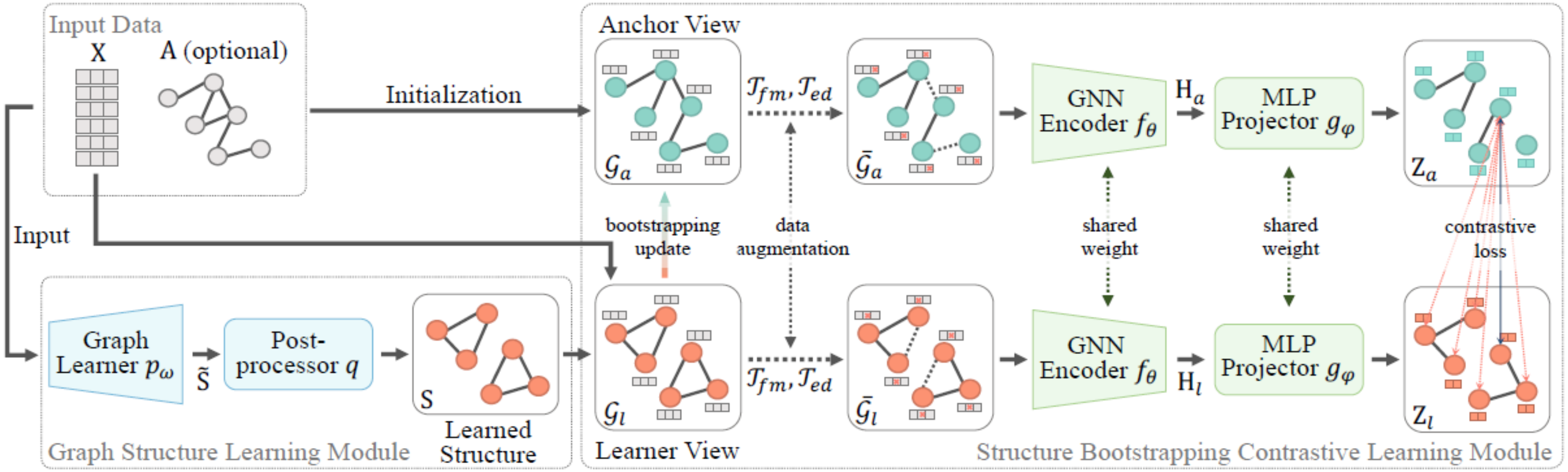


Figure 2: The overall pipeline of SUBLIME. In the graph structure learning module, the graph learner p_ω generates the sketched adjacency matrix $\tilde{S}$, and then the post processor q converts $\tilde{S}$ into the learned structure S. After that, the structure bootstrapping contrastive learning module optimizes S by maximizing the agreement between the learner view and anchor view.

$$\mathcal{L} = \frac{1}{2n} \sum_{i=1}^{n} \left[\ell(z_{l,i}, z_{a,i}) + \ell(z_{a,i}, z_{l,i}) \right],$$
$$\ell(z_{l,i}, z_{a,i}) = \log \frac{e^{\text{sim}(z_{l,i}, z_{a,i})/t}}{\sum_{k=1}^{n} e^{\text{sim}(z_{l,i}, z_{a,k})/t}}, \tag{14}$$

where $\text{sim}(\cdot,\cdot)$ is the cosine similarity function, and t is the temperature parameter. $\ell(z_{a,i}, z_{l,i})$ is computed following $\ell(z_{l,i}, z_{a,i})$.

2022_AAAI_Augmentation-Free Self-Supervised Learning on Graphs

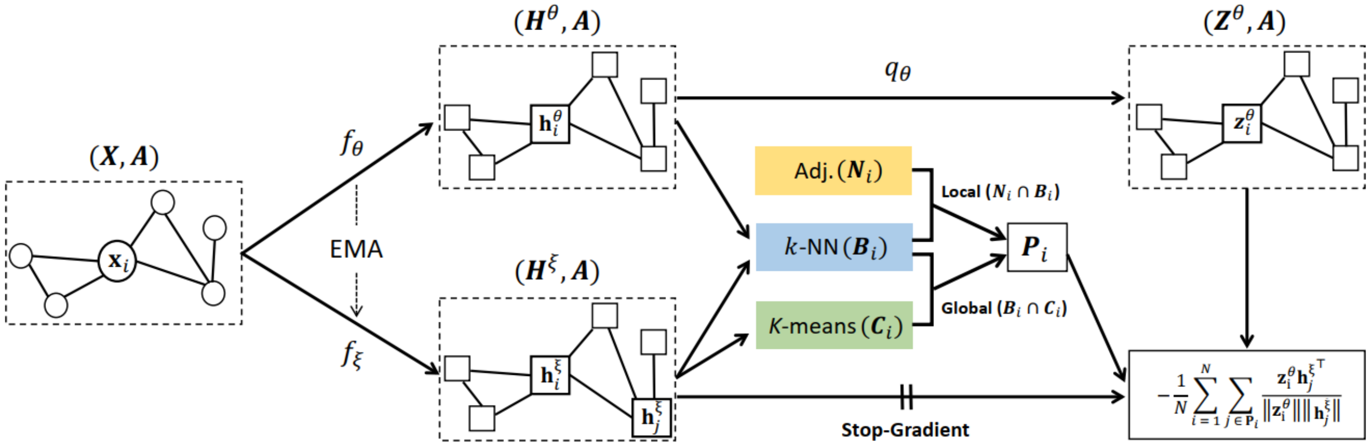


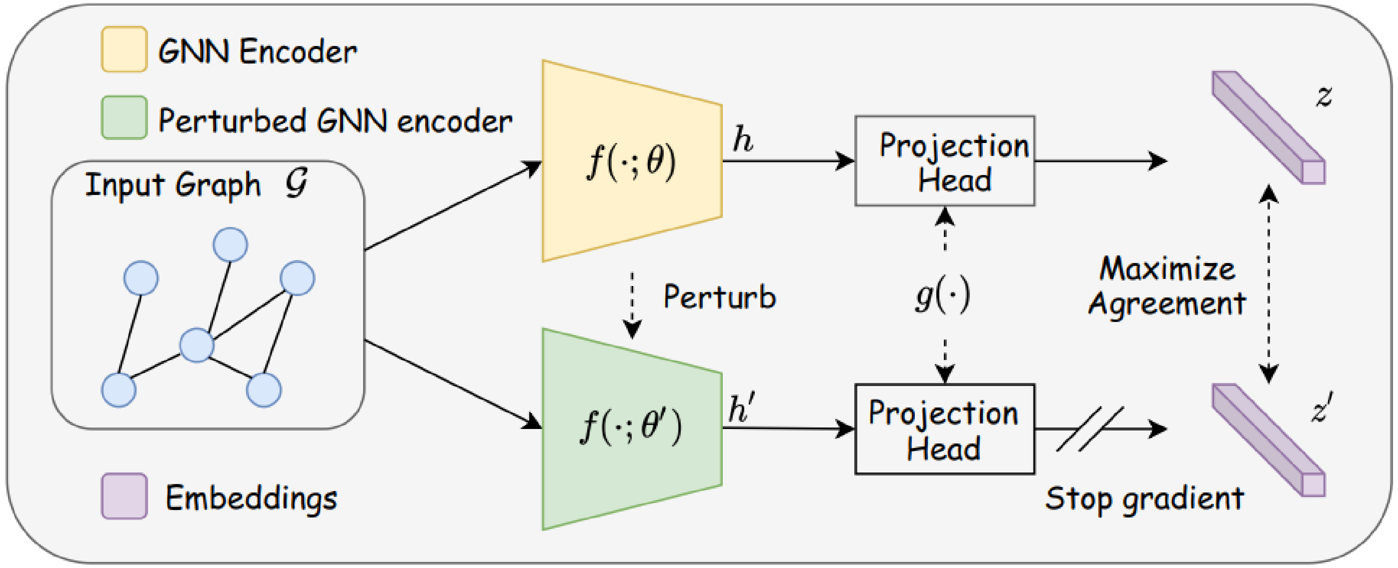
Figure 2: The overall architecture of AFGRL. Given a graph, f_θ and f_ξ generate node embeddings $\mathbf{H}^\theta$ and $\mathbf{H}^\xi$ both of which are used to obtain k-NNs for node v_i, i.e., $\mathbf{B}_i$. Combining it with $\mathbf{N}_i$, we obtain local positives, i.e., $\mathbf{B}_i \cap \mathbf{N}_i$. To obtain global positives for node v_i, K-means clustering is performed on $\mathbf{H}^\xi$, and the result $\mathbf{C}_i$ is combined with $\mathbf{B}_i$, i.e., $\mathbf{B}_i \cap \mathbf{C}_i$. Finally, we combine local and global positives to obtain real positives, i.e., $\mathbf{P}_i$. A predictor q_θ projects $\mathbf{H}^\theta$ to $\mathbf{Z}^\theta$, which is used to compute the final loss along with $\mathbf{H}^\xi$. Note that f_θ is updated via gradient descent of the loss, whereas f_ξ is updated via EMA of f_θ.

$$sim(v_i, v_j) = \frac{\mathbf{h}_i^\theta \cdot \mathbf{h}_j^\xi}{\|\mathbf{h}_i^\theta\| \|\mathbf{h}_j^\xi\|}, \forall v_j \in \mathcal{V} \tag{1}$$

$$\mathbf{P}_i = (\mathbf{B}_i \cap \mathbf{N}_i) \cup (\mathbf{B}_i \cap \mathbf{C}_i) \tag{2}$$

$$\mathcal{L}_{\theta,\xi} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{v_j \in \mathbf{P}_i} \frac{\mathbf{z}_i^\theta \mathbf{h}_j^{\xi\top}}{\left\|\mathbf{z}_i^\theta\right\| \left\|\mathbf{h}_j^\xi\right\|}, \tag{3}$$

2022_WWW_SimGRACE_A Simple Framework for Graph Contrastive Learning without Data Augmentation



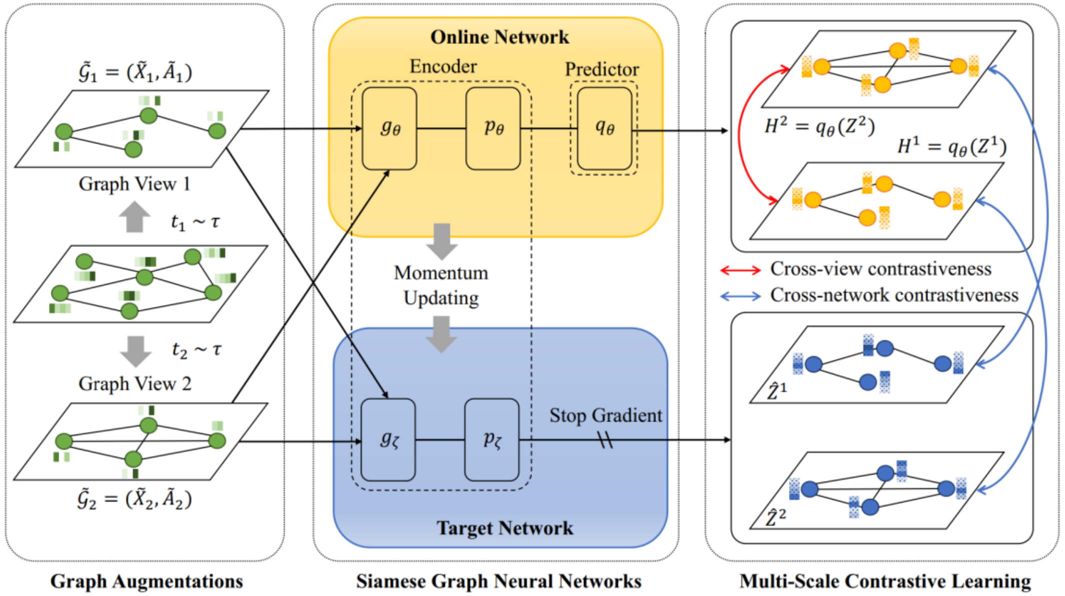
$$\mathbf{h} = f(\mathcal{G};\boldsymbol{\theta}), \mathbf{h}' = f(\mathcal{G};\boldsymbol{\theta}'). \tag{1}$$

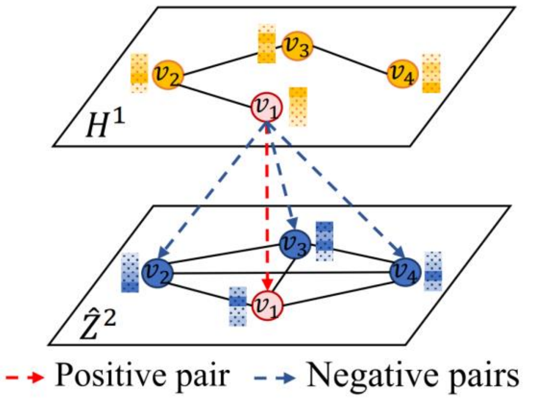
$$z = g(\mathbf{h}), z' = g(\mathbf{h}'). \tag{3}$$

$$\boldsymbol{\theta}_l' = \boldsymbol{\theta}_l + \eta \cdot \Delta\boldsymbol{\theta}_l; \quad \Delta\boldsymbol{\theta}_l \sim \mathcal{N}\left(0, \sigma_l^2\right), \tag{2}$$

$$\ell_n = -\log \frac{\exp\left(\operatorname{sim}\left(z_n, z_n'\right)\right)/\tau)}{\sum_{n'=1, n'\neq n}^{N} \exp\left(\operatorname{sim}\left(z_n, z_{n'}\right)/\tau\right)}, \tag{4}$$

2021_IJCAI_Multi-Scale Contrastive Siamese Networks for Self-Supervised Graph Representation Learning





(a) Cross-network contrastive-
ness.

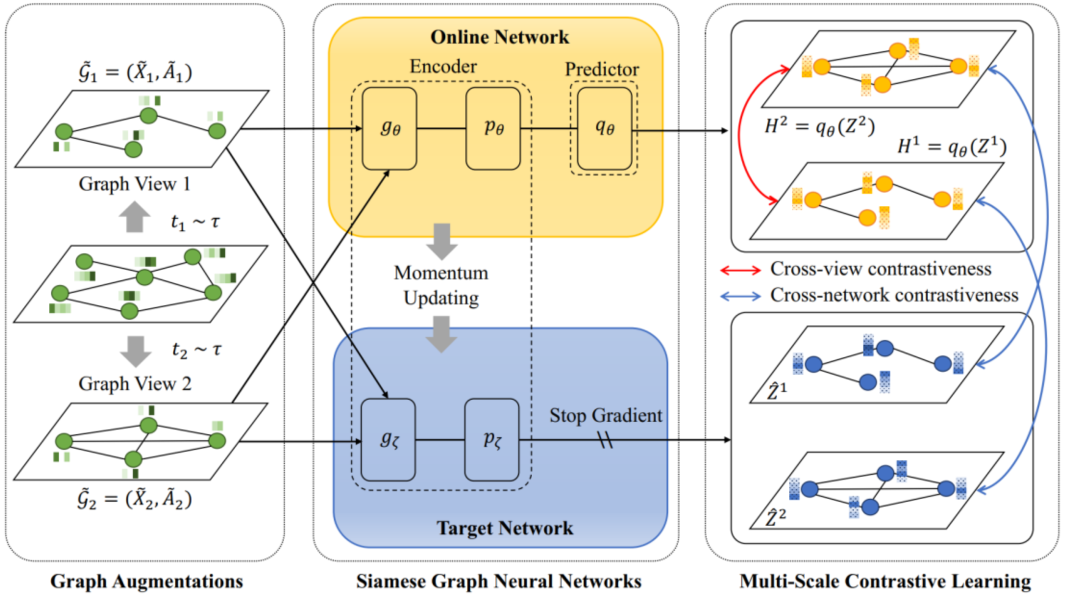
Cross-Network Contrastive Learning

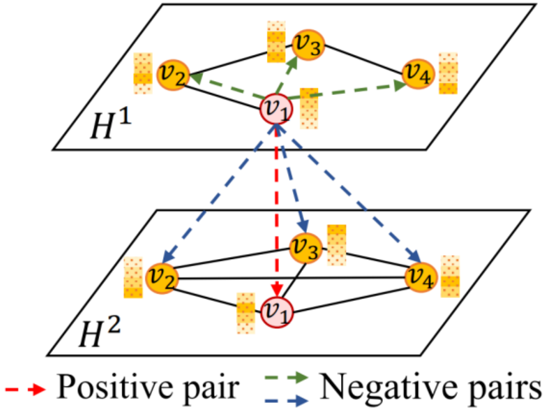
$$\mathcal{L}_{cn}^{1}(v_i) = -\log \frac{\exp(\text{sim}(h_{v_i}^{1}, \hat{z}_{v_i}^{2}))}{\sum_{j=1}^{N} \exp(\text{sim}(h_{v_i}^{1}, \hat{z}_{v_j}^{2}))},$$

$$\mathcal{L}_{cn}^{2}(v_i) = -\log \frac{\exp(\text{sim}(h_{v_i}^{2}, \hat{z}_{v_i}^{1}))}{\sum_{j=1}^{N} \exp(\text{sim}(h_{v_i}^{2}, \hat{z}_{v_j}^{1}))}.$$

$$\mathcal{L}_{cn} = \frac{1}{2N} \sum_{i=1}^{N} \left(\mathcal{L}_{cn}^{1}(v_i) + \mathcal{L}_{cn}^{2}(v_i) \right).$$

2021_IJCAI_Multi-Scale Contrastive Siamese Networks for Self-Supervised Graph Representation Learning





(b) Cross-view contrastiveness.

Cross-View Contrastive Learning

$$\mathcal{L}^1_{inter}(v_i) = -\log \frac{\exp(\mathrm{sim}(h^1_{v_i}, h^2_{v_i}))}{\sum_{j=1}^{N} \exp(\mathrm{sim}(h^1_{v_i}, h^2_{v_j}))}.$$

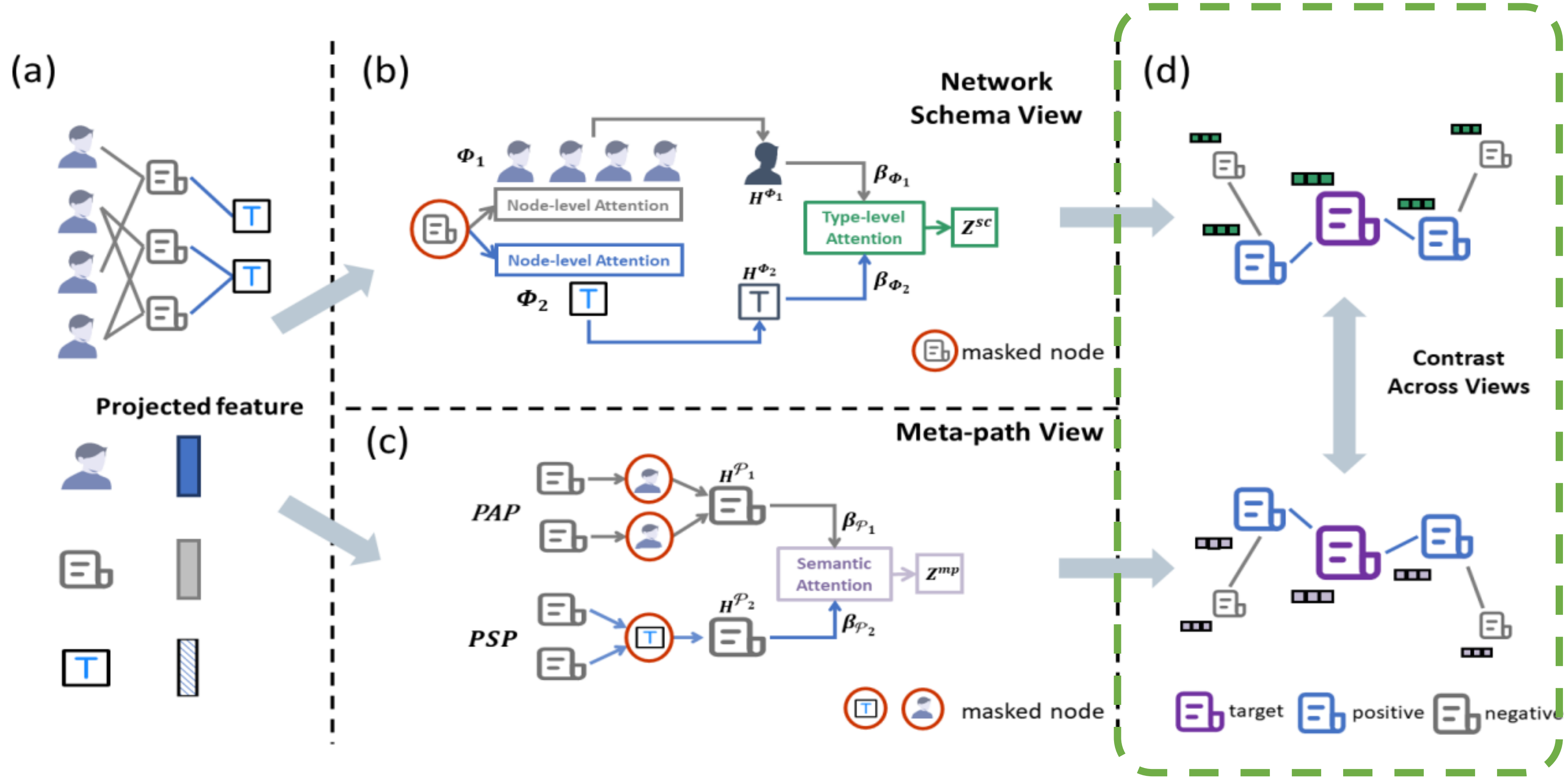
$$\mathcal{L}^1_{intra}(v_i) = -\log \frac{\exp(\mathrm{sim}(h^1_{v_i}, h^2_{v_i}))}{\exp(\mathrm{sim}(h^1_{v_i}, h^2_{v_i})) + \Phi},$$

$$\Phi = \sum_{j=1}^{N} \mathbb{1}_{i \neq j} \exp(\mathrm{sim}(h^1_{v_i}, h^1_{v_j})),$$

$$\mathcal{L}^k_{cv}(v_i) = \mathcal{L}^k_{intra}(v_i) + \mathcal{L}^k_{inter}(v_i), \quad k \in \{1, 2\}.$$

$$\mathcal{L}_{cv} = \frac{1}{2N} \sum_{i=1}^{N} \left(\mathcal{L}^1_{cv}(v_i) + \mathcal{L}^2_{cv}(v_i) \right),$$

2021_KDD_Self-supervised Heterogeneous Graph Neural Network with Co-contrastive Learning



$$\mathcal{L}_i^{sc} = -\log \frac{\sum_{j\in\mathbb{P}_i} exp\left(sim\left(z_i^{sc}\_proj, z_j^{mp}\_proj\right)/\tau\right)}{\sum_{k\in\{\mathbb{P}_i\bigcup\mathbb{N}_i\}} exp\left(sim\left(z_i^{sc}\_proj, z_k^{mp}\_proj\right)/\tau\right)}, \tag{11}$$

2021_NeurIPS_Adversarial Graph Augmentation to Improve Graph Contrastive Learning

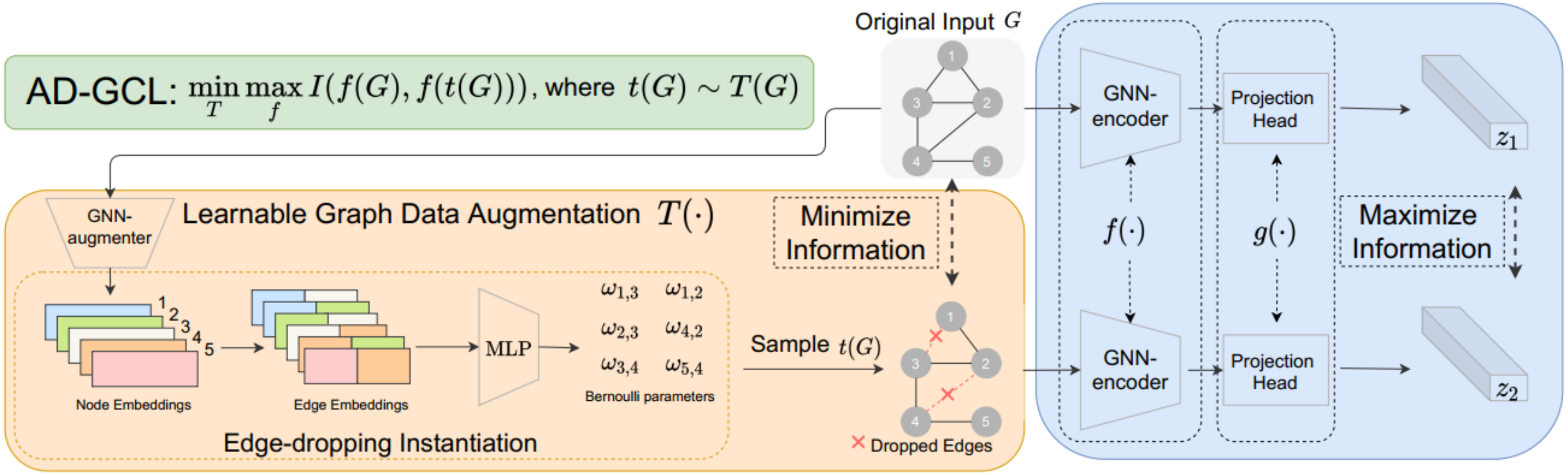


Figure 1: The AD-GCL principle and its instantiation based on learnable edge-dropping augmentation. AD-GCL contains two components for graph data encoding and graph data augmentation. The GNN encoder $f(\cdot)$ maximizes the mutual information between the original graph G and the augmented graph $t(G)$ while the GNN augmenter optimizes the augmentation $T(\cdot)$ to remove the information from the original graph. The instantiation of AD-GCL proposed in this work uses edge dropping: An edge e of G is randomly dropped according to Bernoulli(ω_e), where ω_e is parameterized by the GNN augmenter.

$$I(f_\Theta(G); f_\Theta(t(G))) \to \hat{I} = \frac{1}{m}\sum_{i=1}^{m} \log \frac{\exp(sim(z_{i,1}, z_{i,2}))}{\sum_{i'=1, i'\neq i}^{m} \exp(sim(z_{i,1}, z_{i',2}))} \tag{9}$$

2021_CIKM_Contrastive Curriculum Learning for Sequential User Behavior Modeling via Data Augmentation

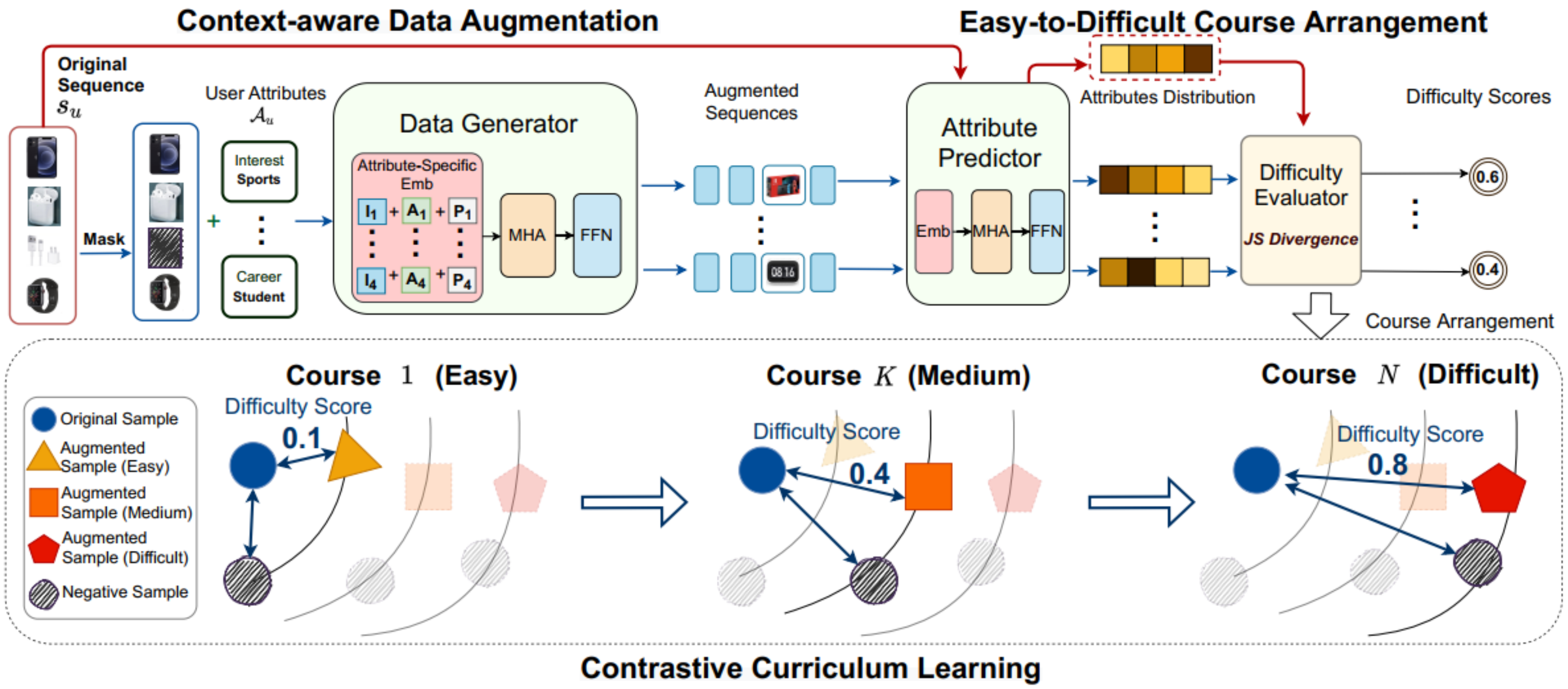


Figure 1: The overall architecture of our proposed approach. The distance of the blue left-right arrow represents the difficulty score between the original sample and the augmented samples at the lower part.

$$\mathcal{L} = -\lambda * \log \frac{\exp(\mathrm{sim}(\boldsymbol{v}_u \cdot \boldsymbol{v}_{z^+})/\tau)}{\exp(\mathrm{sim}(\boldsymbol{v}_u \cdot \boldsymbol{v}_{z^+})/\tau) + \sum_{z^- \in \mathcal{N}} \exp(\mathrm{sim}(\boldsymbol{v}_u \cdot \boldsymbol{v}_{z^-})/\tau)}, \quad (14)$$