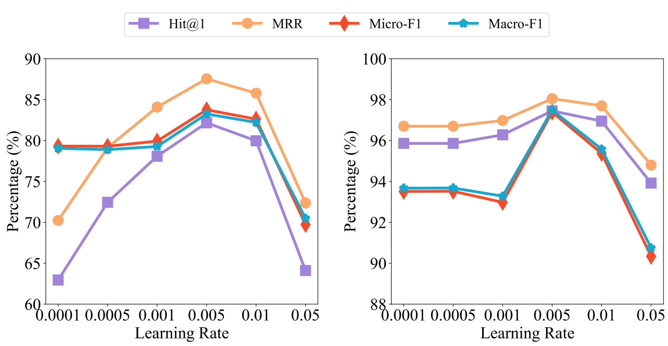
Appendix

A. Removed Subsection "Parameter Analysis"

1) Impact of Different Learning Rate: The impact of different learning rates on the proposed model is summarized in Fig. 1. It reveals that if the learning rate is too large, the model may overshoot the optimal parameters and become unstable, leading to non-convergence issues. Conversely, if the learning rate is too small, the model may get trapped in suboptimal solutions. Specifically, for the link prediction and classification tasks, the proposed model's performance significantly degrades when the learning rate exceeds 5×10^{-3}, while degradation occurs when the learning rate is lower than 5×10^{-3}. Hence, a learning rate of approximately 5×10^{-3} is selected to achieve optimal model performance. In conclusion, the learning rate is a crucial hyperparameter that significantly affects the performance of the proposed model ADMH-ER. Careful tuning of the learning rate is essential to ensure efficient convergence and achieve optimal results.



(a) DB-FB-YAGO (b) Douban-Baidu

Fig. 1. Performance comparison with different values of lr.

B. Removed Figure "Details of the neighborhood-based KG semantic network"

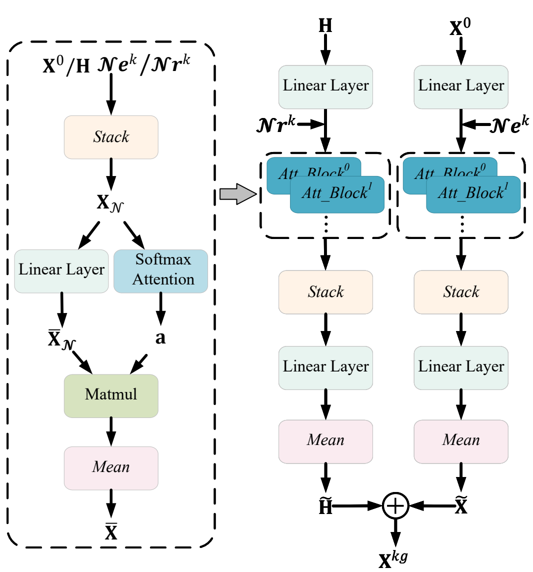


Fig. 2. Details of the neighborhood-based KG semantic network.

Algorithm 1 Training process of ADMH-ER

Require:
A union of multiple MMKGs $\mathcal{KG}=\{\mathcal{E},\mathcal{R},\mathcal{C},\mathcal{P},\mathcal{A},\mathcal{V},\mathcal{T}_r,\mathcal{T}_a\}=\cup_{i=0}^{n-1}\{\mathcal{KG}_i\}$, and a to-be-disambiguated entity set $\mathcal{E}'\subset\mathcal{E}$;
Initialize:
Positive samples P_s and negative samples N_s;
Sets of entities and relations appertaining the k-hop neighborhood of entities: $\mathcal{N}e^k$ and $\mathcal{N}r^k$;
Learned parameters Θ in NBKSN, GAT, the linear transformations for the image, text, and relation & attribute modalities $\{W_m\}_{m\in\{\mathcal{P},\mathcal{C},ra\}}$, ADMH, and HMLS;
Divide $\mathcal{E}'$ into B non-overlapping mini-batches $\{\mathcal{E}'_{(i)}\}$ and $\mathcal{KG}$ into B sub-MMKGs $\{\mathcal{KG}_{(i)}\}$, $i\in\{0,1,\cdots,B-1\}$;

repeat
  for $\mathcal{E}'_{(i)}$ and $\mathcal{KG}_{(i)}$ in each mini-batch **do**
    for each entity e_i in $\mathcal{E}'_{(i)}$ **do**
      Input $\mathbf{X}$, $\mathbf{H}$, $\mathcal{N}e_i^k$ and $\mathcal{N}r_i^k$ into NBKSN to obtain $\mathbf{X}_i^{kg}$ (Eqs. (2)-(7));
      Input $\mathcal{KG}_{(i)}$ and $\mathcal{N}e_i^1$ into GAT to obtain $\mathbf{X}_i^{hg}$;
      Input $\mathcal{P}_i$, $\mathcal{C}_i$, and $\{ar_1,\ldots,ar_l\}$ into ViT, LaBSE, and TF-IDF with $\{W_m\}_{m\in\{\mathcal{P},\mathcal{C},ra\}}$ to achieve $\{\mathbf{X}_i^m\}_{m\in\{\mathcal{P},\mathcal{C},ra\}}$;
      Fuse $\{\mathbf{X}_i^m\}_{m\in\{kg,ra,\mathcal{P},\mathcal{C}\}}$ using an MWH layer to obtain $\mathbf{X}_i^{\mu}$;
      Feed $\mathbf{X}_i^{\mu}$ into the ALMA sub-layer to achieve $\hat{\mathbf{X}}_i$ (Eqs. (8)-(10));
      Feed $\hat{\mathbf{X}}_i$ and $\mathbf{X}_i^{hg}$ into the AGN sub-layer to get $\mathbf{X}_i^{\nu}$ (Eqs. (11)-(15));
      Obtain the final multi-modal hybrid representation $\mathbf{X}_i^f$ using MWH to splice $\mathbf{X}_i^{hg}$, $\mathbf{X}_i^{\mu}$, and $\mathbf{X}_i^{\nu}$;
      Sample the positive example e_j from P_s and the negative example e_k from N_s for e_i, then obtain $\{\mathbf{X}_j^m\}_{m\in\{kg,ra,\mathcal{P},\mathcal{C},f\}}$ and $\{\mathbf{X}_k^m\}_{m\in\{kg,ra,\mathcal{P},\mathcal{C},f\}}$ for e_j and e_k in the same way;
    end for
    Calculate the MSELoss $\mathcal{L}_{link(i)}$, the CrossEntropyLoss $\mathcal{L}_{class(i)}$, and the contrastive loss $\mathcal{L}_{contrast(i)}$ using P_s and N_s (Eqs. (16)-(21));
    Minimize the final loss $\mathcal{L}$ in Eq. (22);
  end for
until the final epoch
Ensure: Θ and $\mathbf{X}^f$.

C. Pseudocode and a consistent example

1) Pseudocode: In this subsection, we briefly describe the algorithm for the proposed model ADMH-ER, as shown in Algorithm 1.

We begin by initializing several data elements. These include positive and negative samples P_s and N_s, as well as the sets of entities and relations in the k-hop neighborhood $\mathcal{N}e^k$ and $\mathcal{N}r^k$. Moreover, we initialize the parameters Θ for the various modules in ADMH-ER. To ensure efficient processing, the entities $\mathcal{E}'$ and the union of multiple MMKGs $\mathcal{KG}$ are

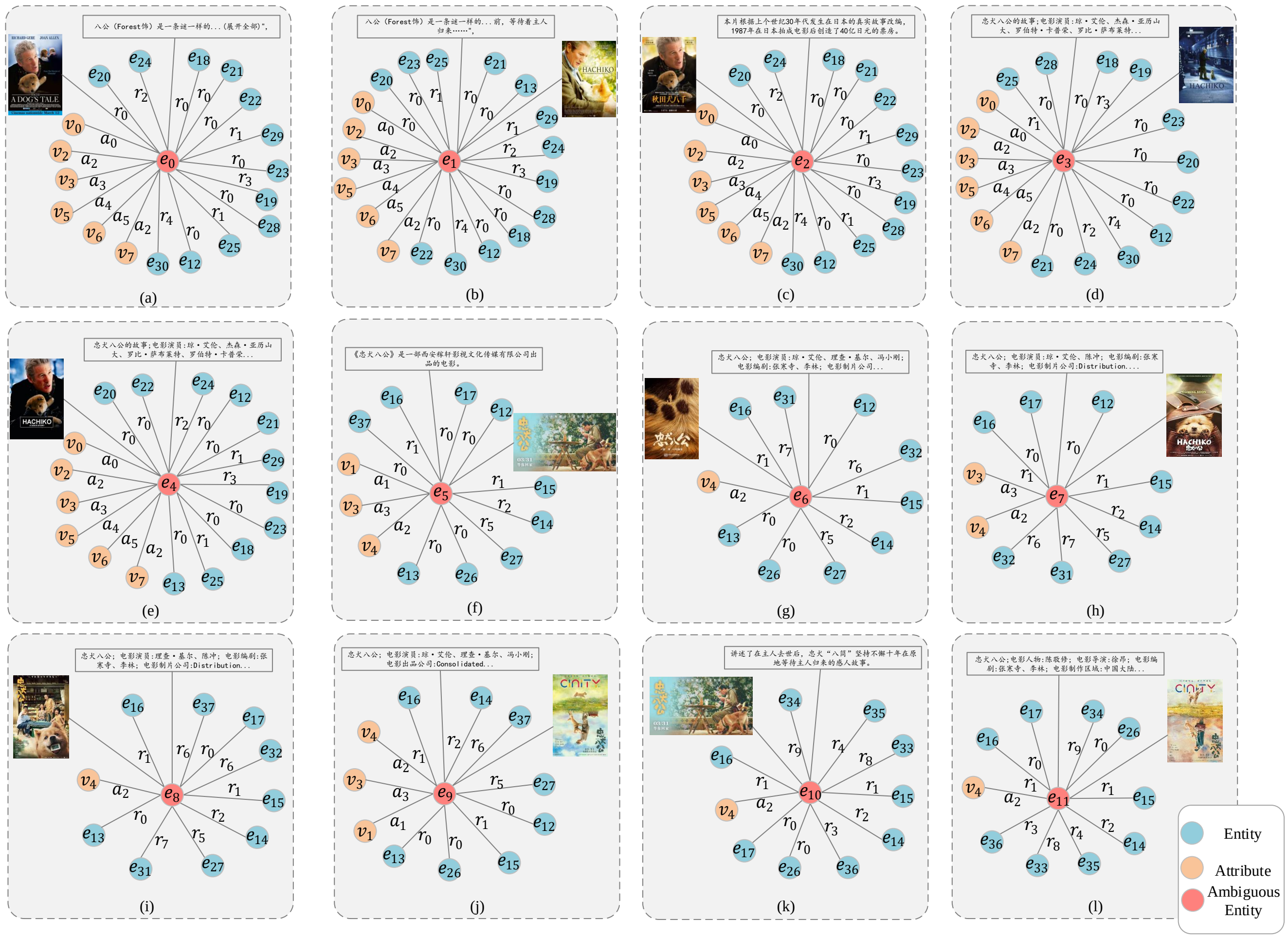


Fig. 3. Case visualization: Visualizing multi-modal information including single-hop structures, text, and images of ambiguous entities.

divided into B mini-batches, i.e., $\{\mathcal{E}'_{(i)}\}$ and $\{\mathcal{KG}_{(i)}\}$, where $i \in \{0, 1, \cdots, B-1\}$.

For each mini-batch, ADMH-ER processes each entity e_i in $\mathcal{E}'_{(i)}$ through three stages in Algorithm 1. (1) The first stage involves generating multi-modal knowledge representations. Specifically, in lines 4 and 5, NBKSN and GAT are used to generate KG and homogeneous graph structure representations, denoted as $\mathbf{X}_i^{kg}$ and $\mathbf{X}_i^{hg}$, respectively. Simultaneously, in line 6, the image, text, and relation & attribute modality representations for e_i, denoted as $\{\mathbf{X}_i^m\}_{m\in\{\mathcal{P},\mathcal{C},ra\}}$, are generated by inputting $\mathcal{P}_i$, $\mathcal{C}_i$, and $\{ar_1, \ldots, ar_l\}$ into ViT, LaBSE, and TF-IDF with $\{W_m\}_{m\in\{\mathcal{P},\mathcal{C},ra\}}$, respectively. (2) The second stage is to obtain multi-modal hybrid representations. Specifically, in line 7, the representations $\{\mathbf{X}_i^m\}_{m\in\{kg,ra,\mathcal{P},\mathcal{C}\}}$ obtained at the first stage are first fused through an MWH layer to obtain the preliminary multi-modal hybrid representation $\mathbf{X}_i^{\mu}$. Then, we feed $\mathbf{X}_i^{\mu}$ into an ALMA sub-layer to refine the high-quality multi-modal information $\mathbf{X}_i^{\mu}$ in line 8. Next, in line 9, $\mathbf{X}_i^{\mu}$ is passed through an AGN sub-layer to filter out noise, resulting in $\mathbf{X}_i^{\nu}$. Ultimately, in line 10, we splice $\mathbf{X}_i^{hg}$, $\mathbf{X}_i^{\mu}$, and $\mathbf{X}_i^{\nu}$ using the MWH layer to produce the final multi-modal hybrid representation $\mathbf{X}_i^{f}$ for each entity. (3) The third stage focuses on training the model based on the representations obtained in the previous stages. The process minimizes a combination of MSELoss $\mathcal{L}_{link(i)}$, CrossEntropyLoss $\mathcal{L}_{class(i)}$, and contrastive loss $\mathcal{L}_{contrast(i)}$ using P_s and N_s, in lines 13 and 14. This training technique improves the generalization ability of the proposed model, enhances feature learning, and it also guides both multi-modal knowledge encoding and adaptive denoising.

2) A Consistent Example: We employ the case study available online[1] to describe the process outlined in this pseudocode. Specifically, in the "$Case\_study/Preprocessing\_data/assessments$" folder, we begin by initializing several data elements. For the mini-batch "$xxx0$", positive and negative samples, P_s and N_s, are generated and stored in the "$xxx0.class$" and "$xxx0.link$" files for the two downstream tasks. The entity sub-set $\mathcal{E}'_{(0)}$ and corresponding sub-KG $\mathcal{KG}_{(0)}$ are generated and stored in the "$xxx0.id$" and "$xxx0.triples$" files. The 1-hop neighborhood $\mathcal{N}e^1$ of entities is stored in the "$xxx0.adj$" file. Other elements, such as sets of entities and relations in the k-hop neighborhood $\mathcal{N}e^k$ and $\mathcal{N}r^k$, are randomly sampled during training and cannot be preprocessed.

After initializing these data elements, we feed them into

[1]https://anonymous.4open.science/r/ADMH-ER-6649/

the ADMH-ER model and train their parameters based on two downstream tasks and a self-supervised learning task (i.e., MSELoss $\mathcal{L}_{link(0)}$, CrossEntropyLoss $\mathcal{L}_{class(0)}$, and contrastive loss $\mathcal{L}_{contrast(0)}$ using P_s and N_s). During training, the representations from key steps in the final epoch are saved, with the details as follows:

(1) NBKSN and GAT are utilized to process triples from "$Case\_study/Triples$" and adjacency matrices from "$Case\_study/Preprocessing\_data/assessments$", generating graph structural representations $\mathbf{X}_i^{kg}$ and $\mathbf{X}_i^{hg}$, which are stored as "$e\_r\_embed.pt$" and "$e\_e\_embed.pt$" in the "$Case\_study/results$" folder.

(2) Pre-trained models TF-IDF, ViT, and LaBSE are used to embed data from "$Case\_study/Preprocessing\_data/text$", "$Case\_study/Images$", and "$Case\_study/Text$" respectively, with the pre-processed information stored in the "$Case\_study/Preprocessing\_data/embH5$" folder. Based on these pre-processing information, we use the learned transformations $\{W_m\}_{m\in\{\mathcal{P},\mathcal{C},ra\}}$ to generate the corresponding representations $\{\mathbf{X}_i^m\}_{m\in\{\mathcal{P},\mathcal{C},ra\}}$, which are stored as "$images.pt$", "$text.pt$", and "$attributes.pt$" in the "$Case\_study/results$" folder.

(3) The ALMA sub-layer generates $\mathbf{X}_i^{\mu}$, which is stored as "$Case\_study/results/ALMA.pt$".

(4) The AGN sub-layer generates $\mathbf{X}_i^{\nu}$, which is stored as "$Case\_study/results/AGN.pt$".

(5) The final multi-modal hybrid representation $\mathbf{X}_i^{f}$ is obtained using the MWH layer and is stored as the "$Case\_study/results/final\_emb.pt$" file.

(6) Based on the saved representations from key steps in the final epoch, stored in the "$Case\_study/results$" folder, we generate the heatmaps presented in the manuscript.

D. Case Visualization

As shown in Figure 3, we present the multi-modal information of ambiguous entities, including single-hop structures, text, and images. For these ambiguous entities, $e_0 \sim e_4$ refers to the original version, while $e_5 \sim e_{11}$ corresponds to the remake.