



A survey of social network alignment methods based on graph representation learning

Yutong WU^{1*}, Feiyang LI^{2*}, Zhan SHI^{1,2}✉, Zhipeng TIAN², Wang ZHANG^{2,3}, Peng FANG^{1,2}, Renzhi XIAO^{1,4}, Fang WANG^{1,2}, Dan FENG^{1,2}

1. Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology, Wuhan 430074, China
2. School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China
3. Institute of Artificial Intelligence, Huazhong University of Science and Technology, Wuhan 430074, China
4. School of Software Engineering, Jiangxi University of Science and Technology, Nanchang 330013, China

Received September 12, 2024; accepted May 22, 2025

E-mail: zshi@hust.edu.cn. * These authors contributed equally to this work.

© The Author(s) 2025. This article is published with open access at link.springer.com and journal.hep.com.cn

Abstract

Social network alignment (SNA) aims to match corresponding users across different platforms, playing a critical role in cross-platform behavior analysis, personalized recommendations, security, and privacy protection. Traditional methods based on attribute and structural features face significant challenges due to the sparsity, heterogeneity, and dynamic nature of social networks, resulting in limited accuracy and efficiency. Recent advances in graph representation learning (GRL) provide promising solutions to these issues by leveraging deep learning to extract network features, effectively addressing sparsity, integrating heterogeneous data, and adapting to network dynamics. This paper presents a comprehensive survey of SNA methods based on GRL. We first introduce key definitions and outline a framework for SNA using GRL. Next, we systematically review state-of-the-art advancements in both static and dynamic networks, considering homogeneous and heterogeneous settings, including emerging approaches integrating large language models (LLMs). We further conduct an in-depth comparative analysis, highlighting the effectiveness of different GRL-based methods, with a particular emphasis on LLM-enhanced techniques. Finally, we discuss open challenges and outline potential future research directions in this rapidly evolving field.

Keywords

social network alignment; graph representation learning; heterogeneous social network; dynamic social networks; graph neural network; large language models

1 Introduction

With the rapid rise of Internet technologies and social networks, users are increasingly engaging across multiple platforms. Social networks now play a central role in communication, commerce, and entertainment. According to the 2023 Global Web Index report [1], the average user is active on more than six platforms a month, highlighting the need for efficient cross-platform identity resolution. With significant overlap in users and content, efficiently identifying similar users across networks has become a key research focus. **Social network alignment (SNA)**, also known as user identity linkage or anchor link prediction, aims to match the corresponding users on various networks. It addresses key challenges such as data fragmentation, user experience optimization, and security. In identity verification, SNA detects fraudulent activities by uncovering profile inconsistencies [2]. In privacy protection, it identifies data leaks while raising concerns about unintended violations [3]. In behavior analysis, SNA integrates overlapping communities to reveal shared

interests [4]. In addition, it improves cross-platform recommendations by aligning user preferences [5]. With its wide-ranging applications and inherent challenges, SNA continues to attract substantial interest from both academia and industry.

This paper reviews existing SNA research [6–8] and categorizes the methods into three types: attribute-based, structure-based, and graph representation learning-based. *Attribute-based methods* attempt to align users by matching explicit attributes such as usernames [9,10], locations [11], behaviors [12,13], and content [14,15]. However, these methods are highly dependent on user-provided data, which is often incomplete or anonymized, leading to significant accuracy and reliability issues. Furthermore, social networks exhibit inherent sparsity, and most users maintain only a few connections, resulting in limited relational data for effective alignment.

Structure-based methods focus on analyzing user interactions and network topology [16,17]. These approaches uncover hidden structural patterns across networks, reducing dependence on

individual attributes. However, they face significant challenges in heterogeneous networks, where node and edge types vary across platforms (e.g., LinkedIn, a professionally oriented platform, and Facebook, a personal interaction network). Furthermore, the dynamic nature of social networks requires frequent retraining, which introduces computational overhead and demands continuous model updates. In summary, due to the sparsity, heterogeneity, and dynamic nature of social networks, both attribute- and structure-based methods face limitations in maintaining alignment accuracy and efficiency.

Graph representation learning (GRL) offers a promising alternative [18–20]. These methods embed network nodes into a continuous vector space, where user similarity can be efficiently quantified based on geometric proximity, enabling more scalable and adaptive SNA solutions. To mitigate network sparsity, GRL transforms sparse, high-dimensional data into dense, low-dimensional embeddings, preserving node connections while revealing latent relationships. This significantly improves alignment accuracy, even in sparse networks [21–24]. To handle network heterogeneity, GRL embeds various types of nodes and edges in a shared vector space, effectively integrating multimodal information for cross-network matching [25–27]. To adapt to network dynamism, GRL uses temporal modeling techniques, enabling real-time updates of node representations and maintaining alignment accuracy over evolving networks [28–30]. In addition, GRL improves computational efficiency by simplifying similarity computations, outperforming traditional methods in large-scale SNA tasks. However, relying solely on GRL-based embeddings may limit precise one-to-one user alignment. To address this, recent research integrates large language models (LLMs) to refine network alignment by leveraging contextual semantics. LLMs improve similarity computations by interpreting node embeddings within linguistic and behavioral contexts, reducing ambiguity in user matching [31,32]. In summary, GRL-based SNA techniques effectively address sparsity, heterogeneity, and dynamism, offering superior alignment accuracy and computational efficiency. As a result, they have become the dominant approach in SNA research.

Many researchers have conducted comprehensive surveys of SNA. Shu et al. [6] focused on attribute- and structure-based methods, but did not explore GRL due to earlier technological limitations. Surveys such as [33,34] analyzed entity alignment (EA) within knowledge graphs but did not extend to SNA. Trung et al. [7] and Chen et al. [8] covered GRL-based SNA methods but largely ignored network heterogeneity and dynamism. In contrast, our review provides a novel and comprehensive perspective with the following key contributions.

- **Comprehensive coverage** A systematic review of GRL-based SNA methods, encompassing both foundational models and recent advances.
- **Addressing network heterogeneity and dynamism** A focused discussion on the underexplored challenges of heterogeneity and dynamic evolution in social networks.
- **Integration of LLMs** A pioneering analysis of the emerging

role of LLMs in enhancing GRL-based SNA techniques.

- **Proposing new directions** Insights into future research opportunities, including advanced multimodal fusion and novel alignment frameworks leveraging LLMs.

The paper is structured as follows. Section 2 defines key SNA concepts and introduces a GRL-based framework. Sections 3 and 4 examine GRL-based SNA methods for static and dynamic graphs, while analyzing the implications of network homogeneity and heterogeneity. Section 5 outlines the experimental setup and evaluates key models, including LLM-enhanced SNA models. Section 6 explores future research directions, with conclusions in Section 7.

■ 2 Overview

2.1 Definition

Following previous work [25,35,36], we first give the key definitions used throughout this survey.

Definition 1 (Social network) A social network is defined as a graph $G = (V, E)$, where $V = \{v_i | i = 1, \dots, N\}$ is the set of user nodes and $N = |V|$ denotes the total number of nodes. The set of edges is represented as $E = \{e = (v_i, v_j) | v_i, v_j \in V\}$, where (v_i, v_j) represents the relationship between v_i and v_j .

Definition 2 (Heterogeneous social network) A heterogeneous social network is defined as a graph $G = (V, E, T_V, T_E)$, where each node v and each edge e are associated with their mapping functions $\phi(v) : v \rightarrow T_V$ and $\varphi(e) : E \rightarrow T_E$, respectively. T_V and T_E denote the set of node types and edge types, where $|T_V| + |T_E| > 2$.

Definition 3 (Dynamic social network) A dynamic social network at time step t is defined as $G^t = (V^t, E^t)$, where $V^t = \{v_i^t | i = 1, \dots, N\}$ is the set of nodes at time step t and $N^t = |V^t|$ denotes the number of nodes. The set of edges is represented as $E^t = \{(v_i^t, v_j^t) | v_i^t, v_j^t \in V^t\}$, where (v_i^t, v_j^t) represents the relationship between users v_i^t and v_j^t at time step t .

Definition 4 (Anchor nodes and anchor links) Given two social networks $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. If $v_i \in V_1$ and $v_j \in V_2$ share the same identity in the real world, they are called anchor nodes. The pair (v_i, v_j) is called an anchor link.

Definition 5 (The SNA problem) Given two social networks $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, the goal is to learn a function F that predicts whether user $v_i \in V_1$ and user $v_j \in V_2$ belong to the same identity in the real world, i.e.,

$$F(v_i, v_j) = \begin{cases} 1, & \text{if } v_i \text{ and } v_j \text{ are the same identity,} \\ 0, & \text{otherwise.} \end{cases}$$

2.2 GRL framework for SNA

Figure 1 illustrates a typical framework for solving SNA tasks with GRL, comprising the following components:

- **Data collection** Collect user profiles, network structures and multimodal content (e.g., text, images) from different social

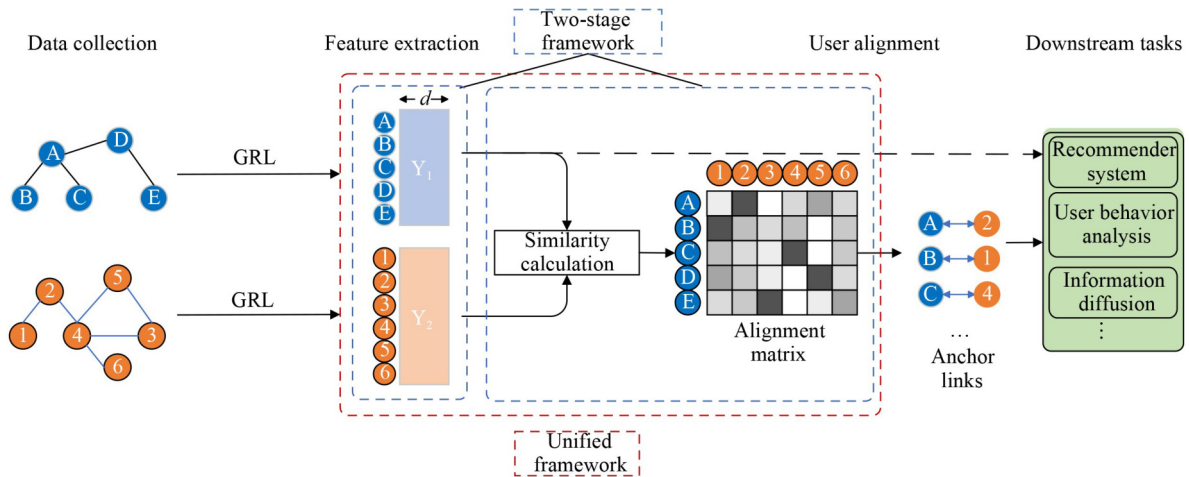


Fig. 1 GRL framework for solving SNA problems

platforms, followed by preprocessing to construct the input graph.

- **Feature extraction** Leverage GRL techniques to learn rich, low-dimensional representations that encode structural, semantic and temporal features for user identity distinction.
- **Similarity calculation** Compute user similarities based on learned embeddings and predefined similarity measures.
- **User alignment** Construct an alignment matrix that models cross-network correspondences, detecting and linking similar nodes to infer user identities.
- **Downstream tasks** Utilize alignment results for cross-platform recommendations, user behavior analysis, and information diffusion.

GRL plays a pivotal role in feature extraction, capturing behavioral patterns and social interactions to enhance user alignment. Furthermore, it efficiently adapts to dynamic and heterogeneous networks, handling real-time changes and multimodal data integration. Consequently, GRL significantly improves the accuracy, efficiency, and adaptability of SNA tasks, making it an essential component of social network analysis.

GRL-based SNA frameworks can be categorized into two types: two-stage and unified approaches, depending on whether GRL and user alignment are learned separately or jointly [37,38].

Two-stage frameworks treat GRL and user alignment as independent components. Initially, each network is embedded in a separate representation space, followed by a mapping function that aligns them in a common space. Although two-stage methods

provide greater flexibility in modeling heterogeneous networks, their independent training stages may introduce inconsistencies in learned representations.

Unified frameworks directly map the two networks to a shared space, simultaneously optimizing GRL and user alignment. This approach fully utilizes contextual information during alignment, ensuring coherence and completeness while minimizing information loss. However, jointly optimizing feature extraction and user alignment increases computational complexity, posing scalability challenges.

■ 3 GRL for static SNA

In this section, we categorize static GRL-based methods for SNA into two groups based on graph characteristics: homogeneous and heterogeneous static graphs. We review representative approaches in each category, highlighting their evolution and associated challenges. Table 1 summarizes the representative SNA methods discussed in this survey.

3.1 GRL for static homogeneous SNA

The development of GRL techniques for homogeneous SNA has progressed through distinct stages, driven by the increasing complexity of network structures and learning objectives. Early approaches based on matrix factorization pioneered low-dimensional graph embeddings by decomposing adjacency or similarity matrices. Shallow neural networks (SNNs) subsequently enhanced adaptability through random walk-based sampling and matrix decomposition. More recently, deep neural networks (DNNs) have advanced GRL by

Table 1 Representative GRL-based SNA methods

Graph type	Representative methods
Static homogeneous graph	REGAL [39], CONE-Align [40], PALE [41], FRUI-P [42], CENALP [43], SAUIL [44], GAlign [45], WAlign [46], HCNA [47], HackGAN [48], DANA [49], NAME [50]
Static heterogeneous graph	TransLink [51], MTransE [52], DPLink [53], TALP [37], INFUNE [54], LLMEA [31], ChatEA [32]
Dynamic homogeneous graph	DNA [55], DGA [56], DeepDSA [57], CTSA [58]
Dynamic heterogeneous graph	TEA-GNN [59], TREA [60], STEA [61], AGN [62]

leveraging iterative neighborhood aggregation to produce richer and more generalizable node representations. Table 2 summarizes representative GRL-based methods for static homogeneous SNA.

3.1.1 Matrix factorization-based methods

Matrix factorization methods are representative of early GRL techniques. These methods embed nodes into continuous vector spaces by factorizing adjacency or similarity matrices, effectively capturing the structural and relational properties of graphs [23,24,63,64].

REGAL [39] is a notable SNA method that applies matrix factorization to facilitate network alignment. It employs a two-stage framework: node representations encoding structural and attribute information are first extracted, followed by xNetMF-based embedding generation using node similarity measures. Alignment is then performed by matching embeddings with high similarity. However, REGAL exhibits limitations in preserving structural consistency across networks, as neighboring nodes may assume different roles in distinct graphs, leading to potential misalignments. To address this issue, CONE-Align [40] builds upon NetMF [23] to encourage neighboring nodes to have similar embeddings, thereby better preserving network topology. By aligning embedding subspaces across networks, CONE-Align improves node correspondence accuracy, particularly in noisy environments.

Despite their early success, matrix factorization-based methods face scalability challenges. Computationally intensive operations such as matrix inversion and eigenvector decomposition limit their applicability to large-scale or dynamically evolving networks.

3.1.2 SNN-based methods

SNN-based methods offer computational efficiency and scalability for large-scale networks while effectively preserving core structural properties. Representative methods include DeepWalk [21], Node2Vec [22], and LINE [65]. DeepWalk learns node embeddings

by applying truncated random walks combined with the Skip-Gram model to capture community structures. Node2Vec extends DeepWalk by introducing biased random walks to explore diverse structural patterns. LINE explicitly optimizes first-order and second-order proximities to preserve both direct and indirect neighborhood relationships. These methods have been widely applied in SNA tasks due to their balance between simplicity and effectiveness.

Building upon these techniques, PALE [41] adopts a two-stage framework (Fig. 2) for SNA. It first independently embeds two networks using LINE or DeepWalk to capture structural dependencies, followed by training a Multi-Layer Perceptron (MLP) to learn a mapping between embedding spaces guided by anchor nodes.

FRUI-P [42] removes the reliance on anchor supervision by employing an unsupervised approach. It generates node sequences through random walks, uses the Skip-Gram model to extract user-specific features, and establishes correspondences based on embedding similarity via a one-to-one matching strategy.

CENALP [43] integrates network alignment with link prediction within a unified framework. It constructs a compound network, applies biased random walks to learn cross-network node embeddings, and jointly optimizes alignment and link prediction, iteratively improving performance as more correspondences and missing links are discovered.

Despite their efficiency, SNN-based methods face inherent limitations in capturing complex relational patterns. Their shallow architectures hinder the modeling of hierarchical dependencies and long-range interactions, reducing their effectiveness in representing intricate social relationships.

3.1.3 DNN-based methods

DNNs, with their hierarchical architectures, excel at extracting complex, high-dimensional features by jointly modeling structural dependencies and node attributes [66–69]. Compared to shallow

Table 2 Static homogeneous GRL-based SNA methods

Method	Framework	Algorithm type	Core technique
REGAL [39]	Two-stage framework	Unsupervised	Matrix factorization (xNetMF)
CONE-Align [40]	Two-stage framework	Unsupervised	Matrix factorization (NetMF)
PALE [41]	Two-stage framework	Supervised	Shallow neural networks (e.g., LINE, DeepWalk), MLP-based mapping function
FRUI-P [42]	Two-stage framework	Unsupervised	Random walk, skip-Gram embedding
CENALP [43]	Unified framework	Supervised	Random walk, joint optimization of network alignment and link prediction
SAUIL [44]	Two-stage framework	Supervised	Siamese neural networks
GAlign [45]	Unified framework	Unsupervised	Multi-order graph convolutional networks, data augmentation
WAlign [46]	Unified framework	Unsupervised	Lightweight graph convolutional networks, Wasserstein adversarial training
HCNA [47]	Unified framework	Self-supervised	Multi-order hyperbolic graph convolutional networks
HackGAN [48]	Two-stage framework	Unsupervised	CycleGAN-based adversarial mapping, Wasserstein-Procrustes alignment
DANA [49]	Unified framework	Semi-supervised	Graph convolutional networks, domain-adversarial learning
NAME [50]	Unified framework	Supervised	Multi-embedding fusion (shallow, deep, community-aware), late fusion mechanism

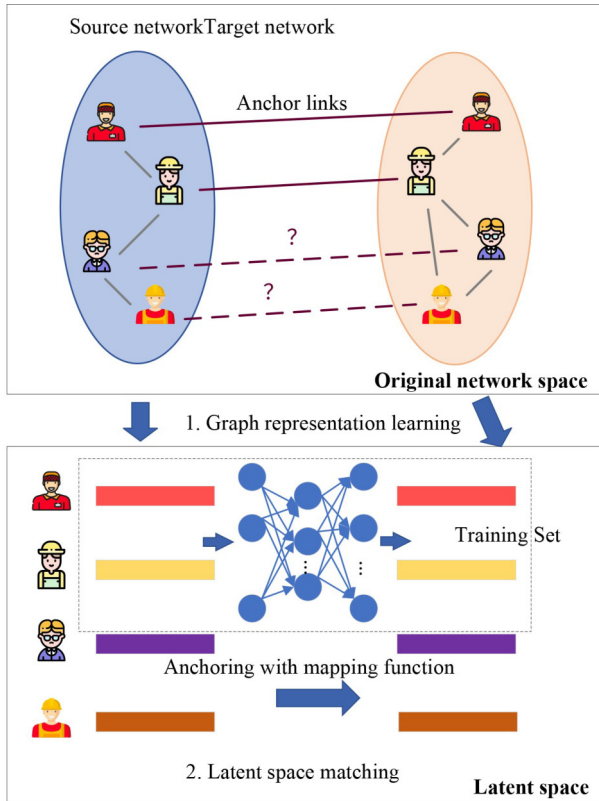


Fig. 2 The overview of PALE [41]

methods, DNNs capture more intricate and non-linear relationships, resulting in more expressive and generalizable node embeddings. Their powerful representation capabilities have made them widely adopted in various SNA tasks.

SAUIL [44] applies a siamese neural network with a bidirectional long- and short-term memory (BLSTM) subnetwork to generate deep feature representations from web browsing data. It processes pairs of web page sets through the BLSTM network and computes similarity scores. To address class imbalance in user identity linkage, SAUIL introduces a cost-sensitive loss function, mean distance false error (MDFE), which penalizes minority-class errors more heavily. Experiments on large-scale datasets demonstrate its robustness,

particularly under imbalanced conditions.

Among DNN-based approaches, graph convolutional networks (GCNs) [66] have achieved remarkable success by efficiently aggregating local and global neighborhood information. Through iterative message passing, GCNs enable nodes to capture richer structural contexts and higher-order relationships, crucial for accurate cross-network alignment.

GAlign [45] is an unsupervised alignment framework that leverages multi-order GCN embeddings to identify structurally and semantically similar nodes across networks. As illustrated in Fig. 3, GAlign employs GCN to encode structural and attribute information, introduces data augmentation to simulate real-world noise, and aggregates embeddings from multiple layers into an alignment matrix, which is iteratively refined for enhanced stability and accuracy.

WAlign [46] extends GCN-based models by incorporating a Wasserstein distance discriminator to improve alignment precision. Instead of relying solely on data augmentation, WAlign employs adversarial training, where a lightweight GCN variant (LGCN) is optimized against the discriminator to enhance robustness. Through iterative refinement, WAlign produces highly discriminative embeddings that support robust SNA performance even under noisy conditions.

Self-supervised learning (SSL) has significantly advanced network alignment by addressing annotation scarcity through contrastive regularization and automated view generation [70,71]. HCNA [47] incorporates hyperbolic representation learning with contrastive SSL to enhance alignment robustness. Compared to Euclidean embeddings, hyperbolic embeddings better capture hierarchical structures, while SSL components offer three advantages: (1) mitigating label dependence via cross-network view discrimination, (2) preserving alignment-invariant features through structural and attribute perturbations, and (3) enforcing consistent embedding spaces via graph-level contrastive objectives. HCNA generates multiple augmented views, including diffusion-based graphs, and employs multi-order hyperbolic GCNs to jointly capture network structures and node attributes. By performing contrastive learning at both node and graph levels, HCNA effectively aligns

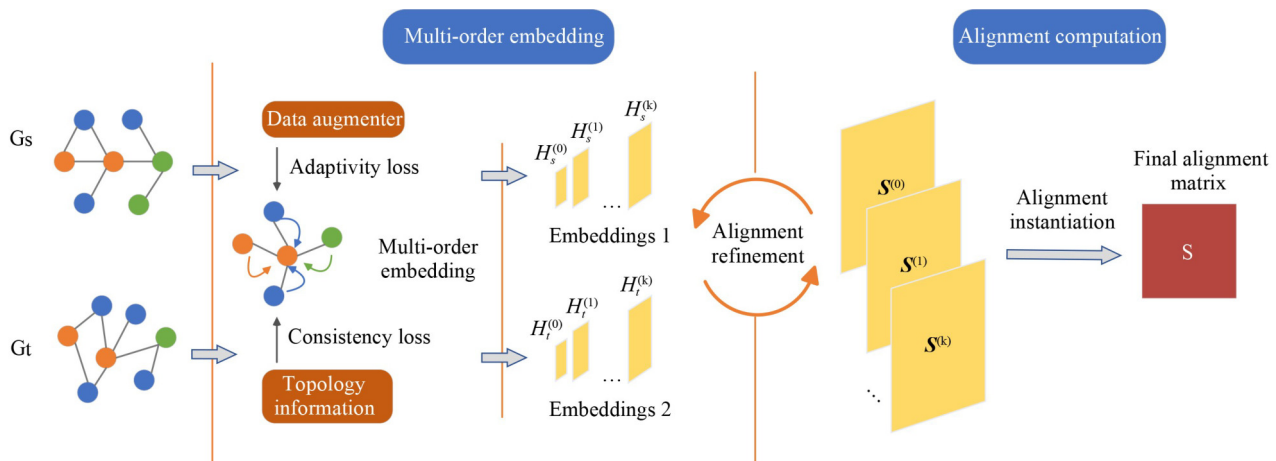


Fig. 3 The overview of GAlign [45]

similar nodes across different networks.

Adversarial learning further enhances deep graph representation learning, especially in unsupervised SNA scenarios, by improving robustness against noise, sparsity, and domain shifts. HackGAN [48] adopts a two-stage adversarial framework (Fig. 4). It first uses an unsupervised GNN to learn node embeddings and initializes mapping functions with a Wasserstein-Procrustes framework. Subsequently, mappings are refined via a cycle-consistent adversarial model combined with Sinkhorn distance losses. Final alignments are produced through collective correspondence assignment based on node distances.

DANA [49] extends adversarial learning by using domain adaptation principles. It integrates GCN to learn domain-invariant embeddings, facilitating robust alignment across heterogeneous networks. By jointly maximizing alignment likelihood and minimizing the ability of a domain classifier to distinguish between networks, DANA achieves superior cross-network generalization.

Although adversarial models enhance robustness, using a single embedding per node limits representational richness. To address this, NAME [50] proposes a multi-embedding framework that generates shallow, deep, and community-aware embeddings for each node. These embeddings are fused through a dynamic late fusion mechanism, enabling adaptation to diverse network structures and enhancing robustness against noise.

3.1.4 Limitations and discussion

In this section, we reviewed representative static homogeneous SNA methods developed through GRL. Matrix factorization approaches (e.g., REGAL, CONE-Align) efficiently capture structural properties but suffer from high computational complexity. Shallow neural networks (e.g., PALE, FRUI-P) improve scalability, but struggle to

model complex structural dependencies. Deep neural networks (e.g., GAlign, HCNA) further enhance feature representations by jointly modeling structure and attributes, significantly improving alignment accuracy. Building on these advances, adversarial learning-based methods (e.g., HackGAN, DANA) strengthen model robustness, particularly in unsupervised scenarios, while hyperbolic embeddings (e.g., HCNA) effectively capture hierarchical network structures. These methods are well suited for large-scale and noisy networks. However, DNN-based approaches still face challenges such as limited interpretability, heavy reliance on labeled data, and sensitivity to adversarial perturbations. Future work should aim to improve model transparency, reduce dependency on labeled data, and improve adversarial robustness to better support real-world SNA applications.

3.2 GRL for static heterogeneous SNA

Homogeneous SNA methods typically assume uniform network structures and consistent node attributes, limiting their ability to capture the diverse relational and semantic patterns inherent in real-world heterogeneous networks. Addressing network heterogeneity has thus become a central focus in advancing SNA research.

To tackle this challenge, various GRL techniques have been developed, evolving from translation-based embeddings to DNNs, and more recently to LLMs. Translation models embed nodes and their relations into continuous vector spaces, providing structured representations of heterogeneous networks. DNN-based methods further enhance modeling capacity by capturing intricate structural and attribute-level heterogeneity [72,73]. Building on these advances, LLM-driven approaches leverage contextualized, data-driven representations to model complex semantic dependencies and entity relationships, reflecting a shift towards scalable, knowledge-aware

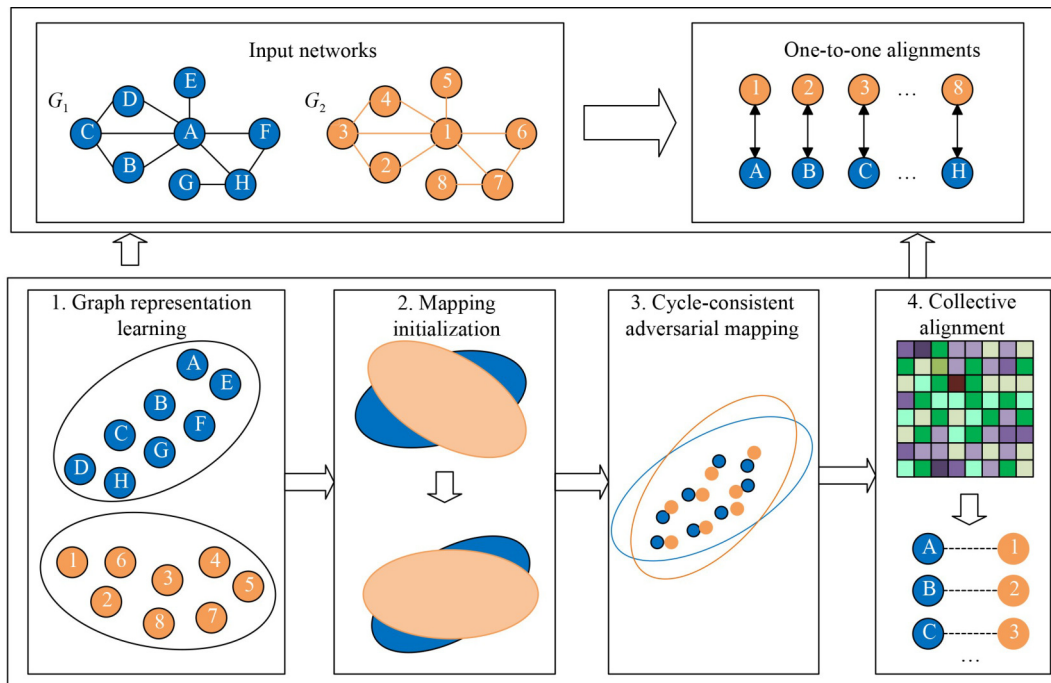


Fig. 4 The overview of HackGAN [48]

network alignment [74,75]. Table 3 summarizes representative GRL-based methods for static heterogeneous SNA.

3.2.1 Translation-based methods

Knowledge graphs (KGs) represent a prominent class of heterogeneous networks, where entities and relations form structured but semantically diverse graphs. Translation models, such as TransE [76], play a fundamental role in graph representation learning by embedding relational triples as vector translations from head entities to tail entities. With compact parameterization, simple structure, and strong interpretability, translation models effectively map structural and semantic relationships into low-dimensional spaces. Their scalability and efficiency have made them fundamental techniques for knowledge graph representation learning, being widely applicable to heterogeneous SNA tasks.

Inspired by translation models, Zhou et al. [51] proposed TransLink for heterogeneous social network alignment. TransLink abstracts network schemas and extracts interaction meta-paths to capture distinctive network features. It learns user and behavior embeddings separately for each network and maps them into a shared low-dimensional space using translation-based techniques. An iterative alignment mechanism progressively refines embeddings by incorporating high-confidence anchor links, enhancing both precision and recall. TransLink demonstrates strong robustness even with limited initial supervision.

MTransE [52] extends translation-based methods to multilingual knowledge graphs by independently encoding entities and relations in different languages and aligning them through axis calibration, translation vectors, and linear transformations. Unlike TransLink, which focuses on network alignment, MTransE targets multilingual entities and triple alignment to ensure semantic consistency across languages.

3.2.2 DNN-based methods

DNNs have demonstrated strong capabilities in heterogeneous SNA tasks by effectively capturing multi-source information and modeling complex structural dependencies. In graph representation learning, DNN-based methods generate expressive node embeddings that integrate various types of relational and attribute information across domains, enabling more accurate alignment.

Feng et al. [53] proposed DPLink, a two-stage deep learning model

for user identity linkage across online platforms based on heterogeneous mobility data. As illustrated in Fig. 5, DPLink employs a feature extractor composed of a location encoder and a trajectory encoder to learn key mobility features. To address domain shifts between platforms, it integrates a pretraining strategy and utilizes a coattention mechanism to enhance feature robustness under noisy conditions.

Li et al. [37] introduced TALP, a unified framework for predicting anchor links by combining type-specific and fusion-based representations. It employs a two-layer graph attention network (GAT): the first layer learns local type-aware embeddings through specific aggregation, while the second layer fuses these embeddings into global representations. By jointly considering type-specific and fused similarities, TALP improves anchor link prediction by ensuring both local consistency and global coherence.

INFUNE [54] enhances heterogeneous SNA by integrating network structure, user profiles, and content. As shown in Fig. 6, it consists of an information fusion module and a neighborhood enhancement module. The former uses an autoencoder to generate embeddings from multi-source data, while the latter refines them via GNNs based on neighborhood similarity. The final alignment is determined by a weighted combination of node and neighborhood similarities.

3.2.3 LLM-based methods

Although translation-based models and DNNs have achieved strong performance in SNA tasks, they predominantly rely on similarity-based GRL techniques to construct vector representations. Such methods often overlook the need for one-to-one semantic alignment among social network entities [31]. For instance, two users with distinct names but similar structural embeddings may be erroneously aligned, while entities with identical names but different meanings may be incorrectly matched. Leveraging extensive pretraining on diverse corpora, LLMs encode rich semantic and contextual knowledge, enabling them to resolve entity ambiguity by distinguishing conceptually similar entities and disambiguating homonymous entities [77–79]. Recent research has thus explored integrating LLMs with GRL-based EA techniques to enhance semantic reasoning capabilities.

Yang et al. [31] proposed LLMEA, an LLM-enhanced self-supervised model for EA. As illustrated in Fig. 7, LLMEA first

Table 3 Static heterogeneous GRL-based SNA methods

Method	Framework	Algorithm type	Core technique
TransLink [51]	Two-stage framework	Supervised	Translation model
MTransE [52]	Two-stage framework	Supervised	Translation model, cross-lingual embedding transformation
DPLink [53]	Two-stage framework	Supervised	Deep learning, co-attention mechanism
TALP [37]	Unified framework	Supervised	Graph attention network, type-aware embedding
INFUNE [54]	Unified framework	Semi-supervised	Heterogeneous graph neural networks
LLMEA [31]	Two-stage framework	Self-supervised	Large language models
ChatEA [32]	Two-stage framework	Self-supervised	Large language models

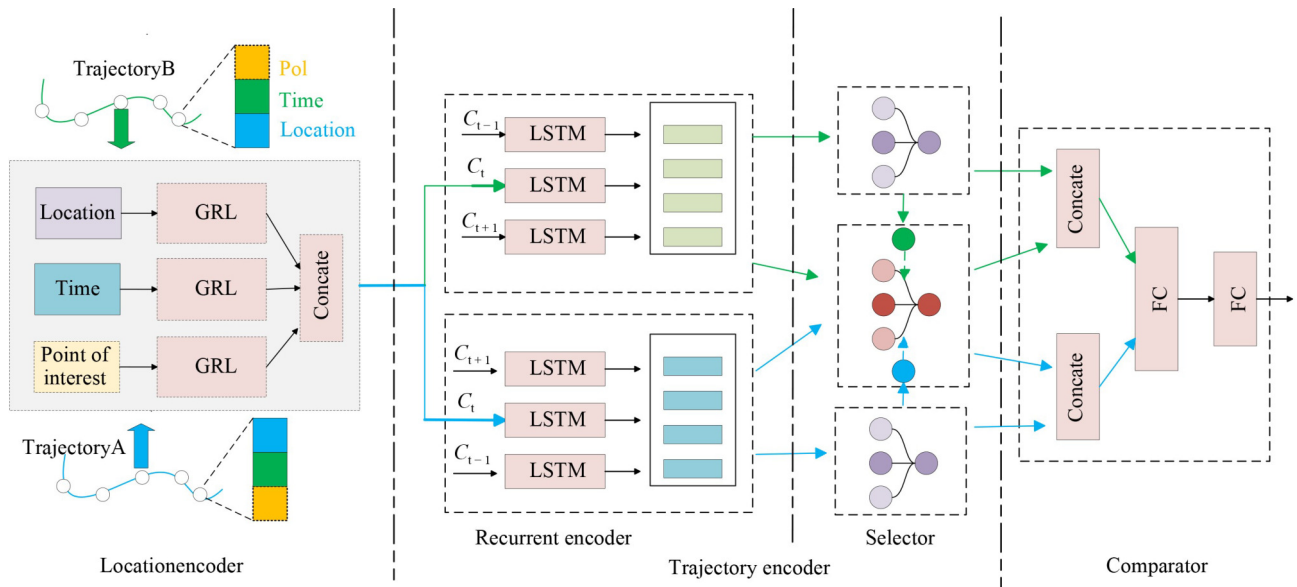


Fig. 5 The overview of DPLink [53]

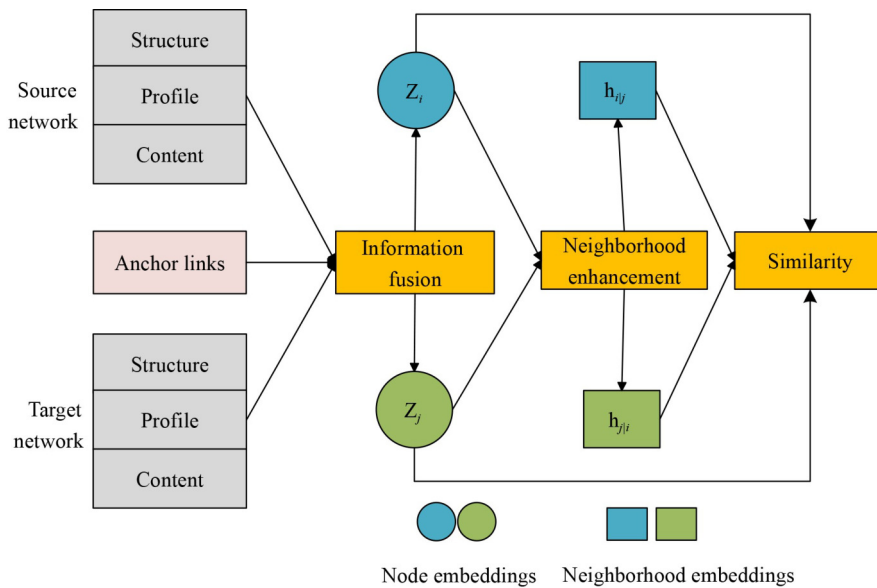


Fig. 6 The overview of INFUNE [54]

learns structural embeddings from KGs using a relation-aware graph attention network (RAGAT) to capture structural and relational similarities. It then generates candidate entities based on embedding similarities and refines them through edit distance calculations and virtual entity synthesis. Finally, the LLM formulates alignment decisions as multiple-choice questions, iteratively selecting the most likely aligned entity. By integrating structural knowledge with LLM-driven semantic reasoning, LLMEA significantly improves alignment accuracy.

Jiang et al. [32] introduced ChatEA, which further advances LLM-based alignment through dialogue-based reasoning. ChatEA begins with a KG-code translation module that converts KG structures into an LLM-interpretable format, enabling the use of extensive background knowledge. It follows a two-stage process: candidate

entity collection via preprocessed embeddings, followed by multi-dimensional dialogue-based evaluation to determine alignment probabilities. Unlike LLMEA, which focuses on embedding refinement and edit distance correction, ChatEA employs dialogue-driven inference to assess correspondences across multiple semantic dimensions. Its self-supervised framework allows iterative refinement based on feedback from the dialogue process, improving both transparency and adaptability in complex alignment scenarios.

3.2.4 Limitations and discussion

This section reviewed three major categories of static heterogeneous GRL-based SNA methods: translation models, DNNs, and LLMs. Translation-based approaches (e.g., TransLink, MTransE) provide interpretable alignment frameworks, but struggle to model complex semantic relationships. DNN-based methods (e.g., DPLink, TALP,

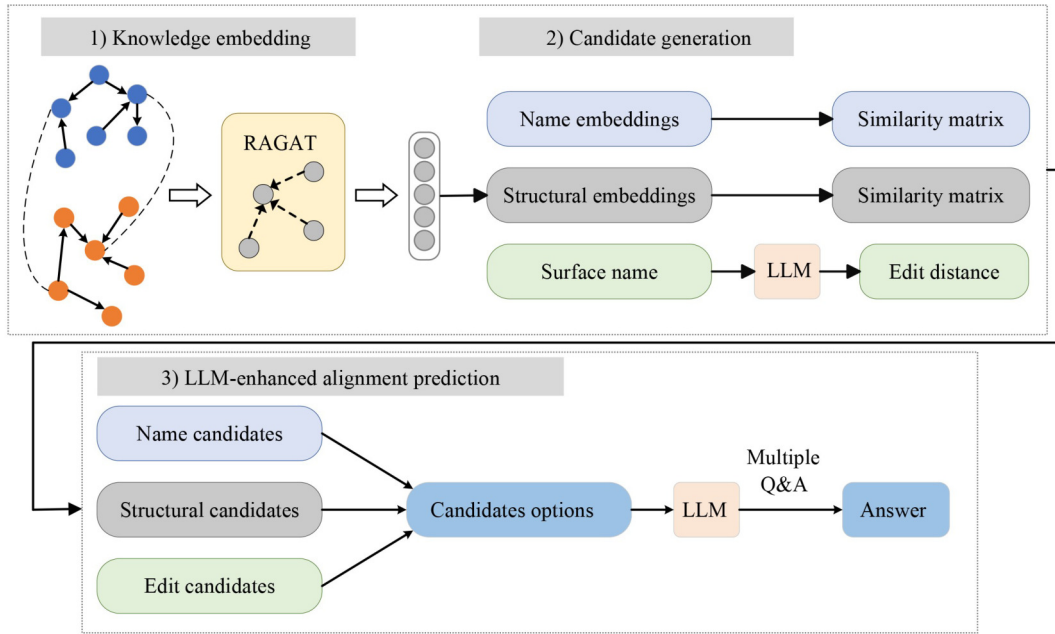


Fig. 7 The overview of LLMEA [31]

INFUNE) improve alignment by capturing intricate structural and type-specific features, but often overlook edge semantics and fine-grained relational nuances. LLM-based approaches (e.g., LLMEA, ChatEA) leverage extensive semantic knowledge and contextual reasoning, enhancing alignment robustness, but face challenges related to interpretability, context window limitations, and high computational overhead. Although LLM-based methods show great potential, future research should focus on improving model interpretability, optimizing computational efficiency, and exploring hybrid GRL-LLM frameworks to further enhance alignment accuracy in heterogeneous networks.

■ 4 GRL for dynamic SNA

Traditional SNA research predominantly focuses on static networks, which fail to capture the evolving nature of social interactions. In contrast, dynamic networks provide a more realistic representation by modeling temporal changes in social relationships. However, dynamics introduce additional challenges, including evolving user

preferences, topology variations, and changes in connectivity patterns. Table 4 summarizes representative dynamic GRL-based SNA methods.

4.1 GRL for dynamic homogeneous SNA

Sun et al. [55] proposed DNA, a pioneering method to align dynamic social networks by modeling temporal evolution. As illustrated in Fig. 8, DNA first applies random walk with restart (RWR) to estimate node proximities in two dynamic networks, followed by an LSTM autoencoder with consistency regularization to learn adaptive user embeddings. To ensure alignment, it constructs a common identity embedding subspace via an alternating optimization algorithm with theoretical convergence guarantees. However, relying on RWR transition matrices leads to high computational complexity.

Building on DNA, Sun et al. [56] introduced DGA, which incorporates a dynamic graph autoencoder to model evolving user embeddings. It employs attentive graph convolution to encode structural information, an LSTM unit for temporal dynamics, and a graph deconvolution to reconstruct network structures. DGA

Table 4 Dynamic GRL-based SNA methods

Method	Framework	Algorithm type	Core technique
DNA [55]	Two-stage framework	Supervised	Random walk with start, LSTM autoencoder
DGA [56]	Unified framework	Unsupervised	Dynamic graph autoencoder, attentive GCN, LSTM
DeepDSA [57]	Unified framework	Semi-supervised	GRU encoder, attention mechanism
CTSA [58]	Unified framework	Supervised	Parameter-shared GNN, self-attention mechanism
TEA-GNN [59]	Two-stage framework	Unsupervised	Time-aware attention mechanism
TREA [60]	Two-stage framework	Unsupervised	Temporal relational attention
STEA [61]	Two-stage framework	Unsupervised	Temporal information matching
AGN [62]	Two-stage framework	Supervised	Time-aware graph attention network

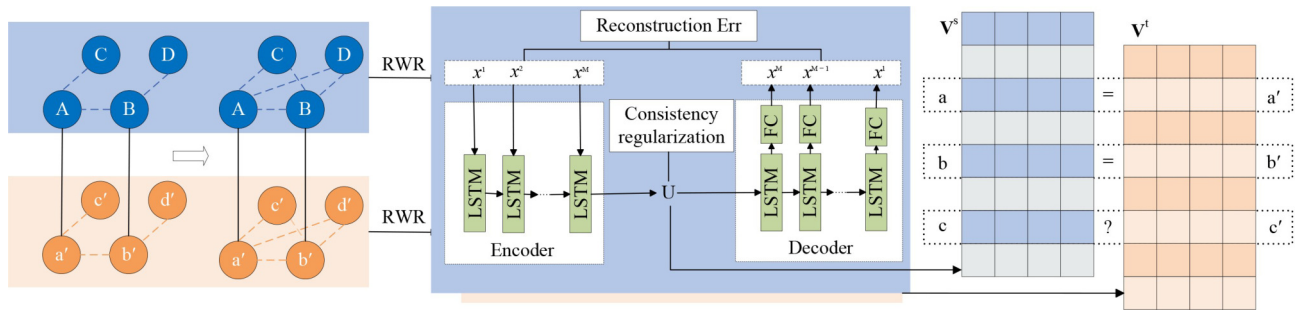


Fig. 8 The overview of DNA [55]

constructs a shared identity embedding subspace and applies an efficient alternating optimization algorithm to handle non-convexity, improving alignment performance across dynamic networks.

DeepDSA [57] is a deep learning-based approach that segments dynamic networks into multiple time slices, capturing both structural and attribute features. As shown in Fig. 9, it uses a GRU sequence model enhanced with attention mechanisms to integrate temporal representations. A semi-supervised spatial transformation maps initial embeddings into a target subspace for alignment. Empirical results demonstrate that DeepDSA outperforms existing methods by effectively integrating structural, attribute, and temporal information at scale.

While most dynamic SNA research focuses on aligning separate evolving networks, intra-network alignment across temporal snapshots remains underexplored. To address this, Liang et al. [58] proposed CTSA. As illustrated in Fig. 10, CTSA uses GNNs to encode nodes in each snapshot, incorporates self-attention to capture temporal dependencies, and introduces positional embeddings to encode sequential and structural information jointly. It refines node embeddings via feature propagation and optimizes alignment through a distance-based loss, significantly improving temporal alignment accuracy.

4.2 GRL for dynamic heterogeneous SNA

Real-world networks evolve through complex temporal and heterogeneous interactions, resulting in dynamic structural and

semantic patterns. The emergence of temporal knowledge graphs (TKGs), which incorporate timestamp information into relational triples, has highlighted the need for temporal reasoning in EA. However, most GRL-based alignment methods overlook temporal dynamics, limiting their ability to capture evolving relationships in large-scale heterogeneous graphs.

Xu et al. [59] first incorporated temporal information into GRL-based EA by proposing TEA-GNN. As illustrated in Fig. 11, TEA-GNN integrates relational and temporal information within a GNN framework using a time-aware attention mechanism that assigns weights based on both relationships and timestamps. Entities, relations, and timestamps are jointly encoded into a unified embedding space, while reverse relations and time intervals are modeled to preserve directional and temporal dependencies. In addition, the authors introduced four real-world TKG datasets (Section 5.1.2), which lay a solid foundation for future research on TKG alignment.

To handle emerging entities and evolving timestamps, Xu et al. [60] proposed TREA, which learns alignment-oriented TKG embeddings while dynamically accommodating new entities. TREA projects entities, relations, and timestamps into a shared space and uses neighborhood aggregation combined with temporal relational attention to capture semantic and temporal dependencies. A margin-based log-loss with a sequential time regularizer is employed to train embeddings while modeling unobserved timestamps.

Recognizing that temporal representations in TKGs often exhibit

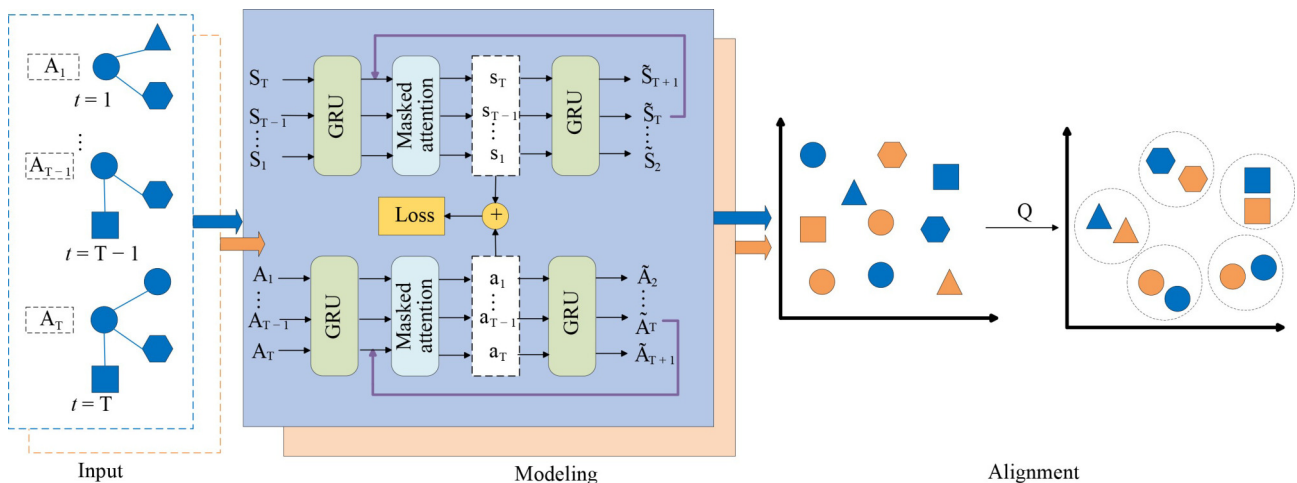


Fig. 9 The overview of DeepDSA [57]

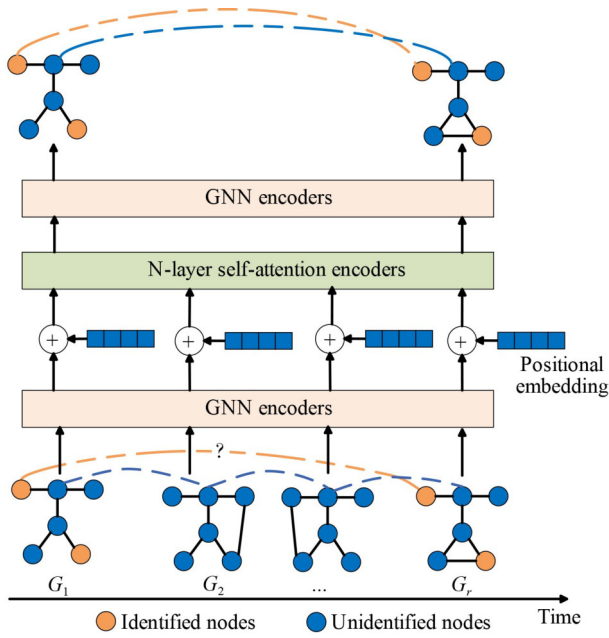


Fig. 10 The overview of CTSA [58]

uniformity, Cai et al. [61] introduced STEA. Instead of learning separate temporal embeddings, STEA captures entity representations using structural and relational information alone and applies a temporal information matching mechanism for alignment. It further employs an unsupervised approach to generate alignment seeds from temporal data, reducing reliance on labeled supervision.

Dynamic alignment methods must adapt to real-time network evolution. To this end, Li et al. [62] proposed AGN, an entity alignment method for temporal knowledge graphs based on an Adaptive Graph Network. AGN employs a time-aware GAT to model temporal and relational dynamics, and introduces adaptive relative error loss minimization (ARELM) and adaptive fine-tuning distance algorithm (AFDA) to optimize representations and alignment objectives in response to data changes.

4.3 Limitations and discussion

Dynamic GRL has significantly advanced SNA by enhancing alignment performance across evolving networks. These methods better model temporal dynamics and structural changes, enabling more robust and accurate alignment over time. However, challenges such as computational complexity, storage efficiency, and scalability remain. As networks grow, computational demands increase sharply, limiting real-time adaptability. Methods like DNA and DGA, while effective, involve complex optimization processes that hinder scalability. In addition, maintaining large volumes of temporal and structural data exacerbates storage challenges, emphasizing the need for more efficient storage and retrieval strategies.

Research on dynamic heterogeneous SNA, particularly for TKGs, remains limited. Many existing models rely on prealigned data, reducing their adaptability to emerging entities and timestamps. Frequent retraining further highlights the need for adaptive learning and self-supervised techniques to enhance scalability. Future work should focus on developing lightweight adaptive models capable of real-time updates, achieving a balance between computational efficiency and alignment accuracy for large-scale dynamic networks.

5 Experiments

Although several benchmarking studies have been conducted for SNA and EA tasks, the analysis and evaluation of GRL-based SNA methods still face significant challenges. Representative efforts include comparative studies on network alignment techniques [7,80] and benchmarking studies on embedding-based EA in knowledge graphs [81–83], which have provided valuable insights into evaluation protocols and dataset organization.

However, key limitations persist. First, existing evaluations often adopt different datasets and experimental settings, hindering fair and comprehensive comparisons among GRL-based methods. Second, many benchmarks focus either on traditional SNA approaches or exclusively on EA in knowledge graphs, without systematically covering both static and dynamic networks, as well as homogeneous

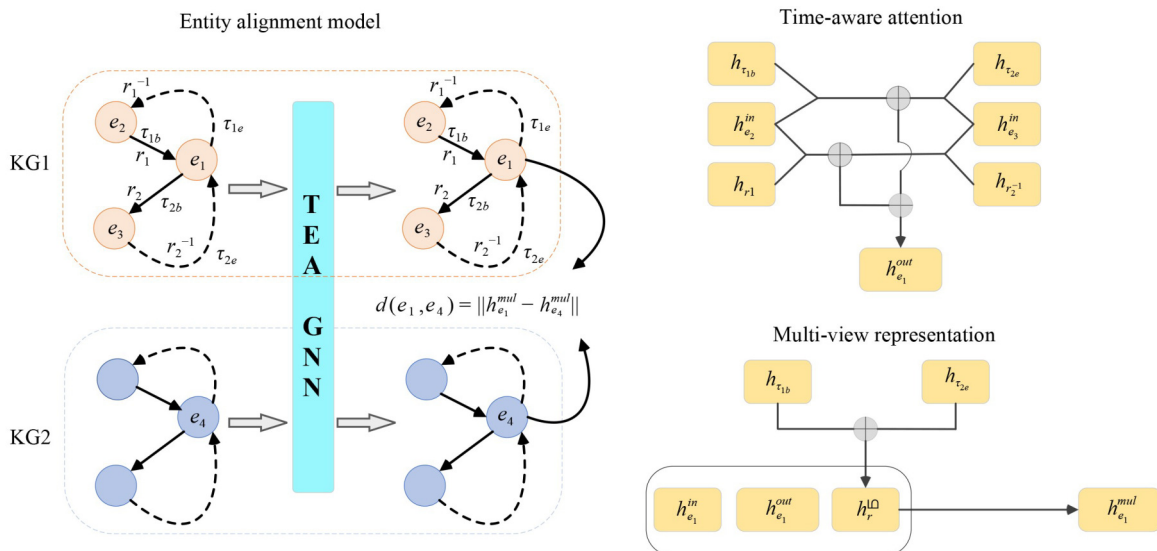


Fig. 11 The overview of TEA-GNN [59]

and heterogeneous settings. Third, systematic comparisons between GRL-based methods and emerging LLM-enhanced techniques remain lacking.

To address these gaps, we performed a unified experimental evaluation using a common set of datasets and consistent evaluation metrics. Our experiments comprehensively cover a wide range of representative SNA methods, including traditional GRL-based approaches and recent LLM-enhanced models, across different network types and alignment scenarios.

5.1 Experimental setups

5.1.1 Environment

The experiments were conducted on an Intel Xeon Gold 5320 CPU (2.20 GHz) with 500 GB RAM and an NVIDIA Tesla V100S-PCIE GPU (32 GB VRAM), running Ubuntu 20.04. The implementation was based on Python 3.10, PyTorch (v2.5.1), and the OpenAI library (v1.61.1). To ensure reproducibility, publicly available datasets and source code were utilized whenever possible. For models without open-source implementations, we reimplemented them by strictly adhering to the methodologies described in the original papers.

5.1.2 Datasets

To evaluate SNA methods under diverse structural and temporal conditions, a variety of benchmark datasets have been developed. These datasets differ significantly in network size, sparsity, heterogeneity, and temporal dynamics, presenting different challenges for alignment tasks. To provide a comprehensive overview, we categorize commonly used datasets into three groups, as summarized in Table 5: social networks, synthetic networks, and temporal networks.

Social networks. These datasets are widely used in social network analysis and consist of platforms such as Foursquare, Twitter, Flickr, Weibo and Douban [36–38,43,45–50,53,54,84–94]. Each experiment involves two networks for alignment.

Synthetic networks. To examine structural variation, we generated synthetic subnetworks by randomly partitioning a single network while preserving user nodes and modifying only the edge structure [38]. We also introduced noise to simulate structural perturbations and class imbalances [45,50]. Original networks used for synthetic data generation include Facebook [38–40] and brain networks (Bn) [45,50].

Temporal networks. Temporal datasets are used to evaluate models under dynamic conditions. For example, the DICEWS dataset, derived from ICEWS05-15 (2005–2015), includes two subsets: DICEWS-1K (1000 anchor links) and DICEWS-200 (200 anchor links) [59,60,95]. Furthermore, YAGO-WIKI50K-5K and YAGO-WIKI50K-1K, constructed from YAGO3 and Wikidata [59,60,96], contain 50000 entities with 5000 and 1000 anchor links, respectively.

5.1.3 Metrics

To assess the performance of SNA methods, we adopted three widely used metrics: Accuracy (Acc), Precision@10 (P@10), and Mean Average Precision (MAP).

Accuracy (Acc). Acc evaluates the proportion of correctly identified alignments:

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \quad (1)$$

where TP, TN, FP, and FN refer to true positives, true negatives, false positives, and false negatives, respectively.

Precision@10 (P@10). P@10 evaluates ranking performance by measuring the fraction of correct alignments among the top 10 retrieved candidates:

$$\text{P@10} = \frac{1}{n} \sum_{i=1}^n \mathbb{I}_i, \quad (2)$$

where \mathbb{I}_i is an indicator function that returns 1 if the ground truth is

Table 5 Common datasets for SNA

Type	Dataset	# Nodes	# Edges	# Anchors	Reference
Social networks	Foursquare-Twitter	17355 – 20417	132208 – 236882	1834	[36–38,49,53,84–90]
	Weibo-Douban	9576 – 10103	270780 – 527980	4752	[38,49,54,91–93]
	Douban Online-Offline	3906 – 1118	8164 – 1511	1118	[43,45–48,50]
	AllMovie-IMDB	6011 – 5713	124709 – 119073	5176	[45,47,50]
	Flickr-Myspace	6714 – 10733	7333 – 11081	267	[36,45,47,87,94]
	Flickr-Last.fm	12974 – 15436	16149 – 16319	452	[36,45,47,87,94]
Synthetic networks	Facebook	4039	88234	4039	[38–40]
	Bn	1781	9016	1781	[45,50]
Temporal networks	DICEWS-1K	9517 – 9537	307552 – 307553	1000	[59,60,95]
	DICEWS-200	9517 – 9537	307552 – 307553	200	[59,60,95]
	YAGO-WIKI50K-5K	49629 – 49222	221050 – 317814	5000	[59,60,96]
	YAGO-WIKI50K-1K	49629 – 49222	221050 – 317814	1000	[59,60,96]

within the top 10 candidates, and 0 otherwise, with n representing the total number of test cases.

Mean average precision (MAP). MAP measures ranking effectiveness by computing the mean of individual average precision scores:

$$MAP = \frac{1}{n} \sum_{i=1}^n \frac{1}{rank_i}, \quad (3)$$

where $rank_i$ denotes the position of the correct match in the ranked candidate list for the i th test case.

5.2 Main results

Table 6 presents the performance comparison of static homogeneous

SNA methods across multiple real-world datasets. The top three models for each dataset are highlighted in red, blue, and cyan, respectively.

DNN-based methods (e.g., NAME, HCNA, DANA) consistently outperform other approaches across most datasets. Their ability to capture high-order graph structures and latent inter-network dependencies through graph embeddings significantly enhances their performance. Notably, NAME achieves the highest P@10 on AllMovie-IMDB (0.942) and Weibo-Douban (0.627), demonstrating strong generalization capabilities across different network domains. Additionally, HCNA and DANA exhibit competitive results, particularly on datasets with complex structural dependencies, such as Douban Online-Offline and Flickr-Last.fm.

Table 6 Performance comparison of static homogeneous SNA methods

Type	Model	Foursquare-Twitter			Weibo-Douban			Douban Online-Offline		
		Acc	P@10	MAP	Acc	P@10	MAP	Acc	P@10	MAP
Matrix Factorization	REGAL	0.042	0.091	0.062	0.375	0.482	0.441	0.352	0.569	0.524
	CONE-Align	0.142	0.271	0.287	0.307	0.543	0.562	0.311	0.492	0.428
Shallow Neural Networks	PALE	0.041	0.184	0.164	0.231	0.414	0.423	0.184	0.478	0.353
	FRUI-P	0.136	0.292	0.314	0.226	0.364	0.371	0.352	0.576	0.494
	CENALP	0.141	0.246	0.227	0.234	0.316	0.283	0.223	0.384	0.297
Deep Neural Networks	SAUIL	0.294	0.391	0.343	0.311	0.522	0.487	0.291	0.672	0.698
	GAlign	0.102	0.367	0.314	0.296	0.484	0.447	0.432	0.770	0.544
	WAlign	0.187	0.342	0.284	0.342	0.587	0.462	0.342	0.681	0.601
	HCNA	0.143	0.396	0.342	0.376	0.542	0.563	0.521	0.776	0.623
	HackGAN	0.027	0.104	0.086	0.234	0.442	0.457	0.149	0.396	0.332
	DANA	0.252	0.451	0.413	0.231	0.472	0.387	0.281	0.522	0.328
NAME	0.132	0.439	0.348	0.351	0.627	0.588	0.498	0.835	0.662	

Type	Model	AllMovie-IMDB			Flickr-Myspace			Flickr-Last.fm		
		Acc	P@10	MAP	Acc	P@10	MAP	Acc	P@10	MAP
Matrix Factorization	REGAL	0.412	0.612	0.572	0.028	0.104	0.084	0.093	0.142	0.131
	CONE-Align	0.272	0.381	0.362	0.076	0.142	0.112	0.117	0.226	0.193
Shallow Neural Networks	PALE	0.673	0.746	0.732	0.003	0.009	0.005	0.004	0.114	0.084
	FRUI-P	0.625	0.841	0.737	0.144	0.322	0.246	0.174	0.362	0.393
	CENALP	0.467	0.824	0.857	0.106	0.231	0.193	0.052	0.175	0.198
Deep Neural Networks	SAUIL	0.307	0.483	0.528	0.232	0.384	0.405	0.146	0.294	0.256
	GAlign	0.785	0.871	0.817	0.114	0.322	0.186	0.141	0.262	0.233
	WAlign	0.457	0.722	0.674	0.097	0.304	0.263	0.122	0.253	0.274
	HCNA	0.764	0.884	0.821	0.162	0.254	0.228	0.183	0.304	0.276
	HackGAN	0.686	0.842	0.764	0.011	0.054	0.034	0.087	0.124	0.094
	DANA	0.324	0.561	0.492	0.182	0.374	0.385	0.216	0.394	0.326
NAME	0.824	0.942	0.886	0.127	0.299	0.324	0.154	0.286	0.255	

SNN-based methods (e.g., PALE, FRUI-P, CENALP) yield moderate results, primarily due to their dependence on explicit anchor node supervision. Their performance degrades substantially in low-resource settings, as evidenced by PALE’s notably low accuracy of 0.003 on Flickr-Myspace. However, FRUI-P outperforms matrix factorization methods on structured datasets such as Flickr-Last.fm, suggesting that its ability to exploit pairwise relationships plays a crucial role in network alignment.

Matrix factorization-based methods (e.g., REGAL, CONE-Align) perform reasonably well when networks share high structural similarity, as observed in Weibo-Douban (CONE-Align achieves a P@10 of 0.543). However, their effectiveness declines significantly when dealing with heterogeneous networks such as Foursquare-Twitter, where structural mismatches hinder alignment accuracy.

Datasets such as Foursquare-Twitter, Flickr-Myspace, and Flickr-Last.fm present considerable challenges due to their pronounced heterogeneity in network topology, user behavior, and attribute distributions. Performance degradation observed across multiple methods in these datasets highlights the necessity for more adaptive alignment frameworks capable of bridging cross-platform differences and handling structural inconsistencies.

Table 7 presents the performance of static heterogeneous SNA models on real-world datasets. DNN-based methods (e.g., TALP, DPLink, INFUNE) exhibit superior performance in most cases. TALP achieves the highest MAP scores on AllMovie-IMDB (0.836) and Flickr-Last.fm (0.423), demonstrating its strong ability to handle multimodal heterogeneity. Its hierarchical attention mechanism

integrates heterogeneous attributes through learnable semantic fusion layers, effectively capturing both local and global patterns, leading to improved alignment accuracy across diverse network structures.

Translation-based models (TransLink, MTransE) perform relatively poorly, particularly on datasets like Flickr-Last.fm, where MAP scores remain below 0.25. Their reliance on relation translatability in the embedding space makes them less effective in handling attribute semantic divergence (e.g., differing tag conventions) and structural heterogeneity (e.g., varied node degree distributions).

LLM-based methods demonstrate remarkable robustness across datasets. For example, on Weibo-Douban, ChatEA attains a MAP score of 0.671, outperforming TALP (0.497) by approximately 35%. This suggests that LLMs possess enhanced contextual semantic reasoning and cross-domain knowledge transfer capabilities, allowing them to adapt more effectively to diverse attribute schemas and interaction paradigms. These results indicate that LLM-based approaches hold significant promise for heterogeneous SNA, particularly in scenarios where textual and structural semantics play a crucial role.

Table 8 presents a comparative analysis of representative dynamic SNA methods. The results demonstrate that dynamic heterogeneous SNA methods (e.g., TEA-GNN, TREA, STEA, AGN) consistently surpass their homogeneous counterparts (e.g., DNA, DGA, DeepDSA, CSTA) across the DICEWS and YAGO-WIKI150K datasets.

Among these, STEA achieves state-of-the-art performance by

Table 7 Performance comparison of static heterogeneous SNA methods

Type	Model	Foursquare-Twitter			Weibo-Douban			Douban Online-Offline		
		Acc	P@10	MAP	Acc	P@10	MAP	Acc	P@10	MAP
Translation Model	TransLink	0.112	0.213	0.247	0.237	0.343	0.312	0.271	0.442	0.388
	MTransE	0.091	0.212	0.195	0.184	0.406	0.433	0.243	0.534	0.585
Deep Neural Networks	DPLink	0.282	0.791	0.443	0.261	0.422	0.391	0.211	0.582	0.458
	TALP	0.418	0.824	0.667	0.396	0.684	0.497	0.492	0.870	0.784
	INFUNE	0.086	0.246	0.228	0.228	0.549	0.498	0.213	0.459	0.427
Large Language Model	LLMEA	0.121	0.329	0.307	0.334	0.496	0.403	0.314	0.784	0.728
	ChatEA	0.281	0.484	0.413	0.401	0.788	0.671	0.548	0.873	0.841
Type	Model	AllMovie-IMDB			Flickr-Myspace			Flickr-Last.fm		
		Acc	P@10	MAP	Acc	P@10	MAP	Acc	P@10	MAP
Translation Model	TransLink	0.322	0.441	0.392	0.106	0.272	0.232	0.097	0.166	0.223
	MTransE	0.217	0.424	0.371	0.126	0.207	0.183	0.112	0.195	0.178
Deep Neural Networks	DPLink	0.354	0.745	0.722	0.102	0.344	0.325	0.186	0.324	0.277
	TALP	0.612	0.872	0.836	0.224	0.522	0.486	0.341	0.448	0.423
	INFUNE	0.237	0.493	0.477	0.116	0.217	0.197	0.132	0.225	0.218
Large Language Model	LLMEA	0.492	0.858	0.764	0.186	0.271	0.213	0.212	0.355	0.319
	ChatEA	0.476	0.798	0.749	0.346	0.592	0.525	0.246	0.452	0.385

Table 8 Performance comparison of dynamic SNA methods

Model	DICEWS-1K			DICEWS-200			YAGO-WIKI50K-5K			YAGO-WIKI50K-1K		
	Acc	P@10	MAP	Acc	P@10	MAP	Acc	P@10	MAP	Acc	P@10	MAP
DNA	0.134	0.329	0.293	0.117	0.368	0.272	0.317	0.493	0.367	0.213	0.372	0.278
DGA	0.253	0.386	0.266	0.274	0.427	0.304	0.372	0.565	0.493	0.297	0.531	0.476
DeepDSA	0.188	0.451	0.367	0.238	0.481	0.383	0.316	0.615	0.514	0.276	0.548	0.449
CTSA	0.145	0.269	0.229	0.127	0.301	0.258	0.219	0.381	0.324	0.213	0.346	0.283
TEA-GNN	0.864	0.927	0.898	0.892	0.935	0.921	0.881	0.952	0.914	0.742	0.901	0.792
TREA	0.907	0.962	0.941	0.915	0.971	0.928	0.937	0.982	0.947	0.829	0.918	0.893
STEA	0.939	0.972	0.943	0.931	0.977	0.942	0.963	0.991	0.969	0.948	0.973	0.957
AGN	0.921	0.969	0.938	0.924	0.972	0.935	0.895	0.966	0.938	0.932	0.989	0.968

integrating a lightweight GNN with a temporal alignment mechanism, effectively modeling temporal dependencies. It achieves the highest accuracy of 0.939 on DICEWS-1K and 0.963 on YAGO-WIKI50K-5K, highlighting the effectiveness of direct temporal encoding for dynamic EA. TREA, leveraging a temporal relational attention mechanism, captures temporal dependencies with high robustness, attaining superior MAP values of 0.941 on DICEWS-1K and 0.947 on YAGO-WIKI50K-5K. Meanwhile, AGN demonstrates remarkable adaptability to complex scenarios, benefiting from an adaptive temporal alignment framework, achieving P@10 of 0.989 and MAP of 0.968 on YAGO-WIKI50K-1K.

Overall, these heterogeneous approaches effectively overcome the limitations of conventional homogeneous methods in modeling dynamic network evolution. By integrating relational structures with temporal dynamics, they establish new benchmarks for temporal-aware EA.

5.3 Efficiency and scalability

Table 9 provides a comparative analysis of the computational efficiency of various SNA models across different datasets. To ensure a fair comparison, we selected the top-performing models from each category, as identified in Section 5.2. Due to their poor performance on temporal graphs, dynamic homogeneous models were excluded from the runtime analysis.

Among static homogeneous models, GAlign exhibits the lowest runtime across all datasets, requiring only 604.93 s on Weibo-Douban, 28.57 s on Douban Online-Offline, and 342.81 s on AllMovie-IMDB. Other efficient models include CONE-Align and FRUI-P, both of which maintain relatively low computational costs. In contrast, DNN-based methods such as HCNA and NAME incur significantly higher computational overhead, with HCNA reaching 11075.32 s on Weibo-Douban and 7314.75 s on AllMovie-IMDB. These results indicate that while DNN-based methods improve

Table 9 Runtime cost (in seconds)

Type	Model	Weibo-Douban	Douban Online-Offline	AllMovie-IMDB
Static homogeneous	CONE-Align	764.30	36.74	449.16
	FRUI-P	692.18	282.42	561.94
	GAlign	604.93	28.57	342.81
	HCNA	11075.32	3825.63	7314.75
	NAME	9685.07	796.37	5817.33
Static heterogeneous	MTransE	3687.18	659.41	1107.28
	TALP	4862.73	864.06	2789.25
	ChatEA	39744.46	25364.27	30617.85
Type	Model	DICEWS-1K	DICEWS-200	YAGO-WIKI50K-5K
Dynamic heterogeneous	TEA-GNN	1652.72	1767.94	5554.27
	TREA	273.67	268.93	3017.08
	STEA	619.18	631.49	4482.09
	AGN	2452.72	2467.94	6184.27

alignment accuracy, their high computational cost may limit their scalability for large-scale applications.

For static heterogeneous models, MTransE demonstrates the highest efficiency, achieving the lowest runtime across all datasets (3687.18 s on Weibo-Douban, 659.41 s on Douban Online-Offline, and 1107.28 s on AllMovie-IMDB). TALP follows closely, albeit with moderately higher computational costs. In contrast, LLM-based ChatEA exhibits an exceptionally high runtime, exceeding 39744 s on Weibo-Douban and 30617.85 s on AllMovie-IMDB. This substantial computational overhead poses challenges for real-time deployment without further optimization, highlighting the trade-off between LLM-enhanced alignment accuracy and efficiency.

Among dynamic heterogeneous models, TREA emerges as the most efficient, requiring only 273.67 s on DICEWS-1K and 268.93 s on DICEWS-200. Both TEA-GNN and STEA exhibit moderate computational efficiency, balancing temporal modeling accuracy with runtime performance. In contrast, AGN incurs the highest computational cost in this category, reaching 6184.27 s on YAGO-WIKI50K-5K. Although AGN’s adaptive optimization framework enhances alignment flexibility in evolving networks, its high computational demand may restrict its scalability in large-scale temporal settings.

To comprehensively evaluate the scalability of different SNA algorithms, we varied the number of nodes from 1000 to 1000000 using synthetic graphs generated with the Watts-Strogatz model [97] based on the Facebook and Bn datasets. Two key performance indicators were examined: runtime (measured on a logarithmic scale) and accuracy, as both metrics are highly sensitive to network size.

Figure 12 presents the runtime trends of different alignment methods as network size increases. Models exceeding the 37-hour execution limit were excluded from the results. As expected, all methods exhibit an increasing runtime with network expansion, but their scalability varies significantly. DNN-based models (HCNA, NAME) show poor scalability, failing to execute beyond 35000 nodes due to their high computational complexity. ChatEA (LLM-based method) incurs the highest computational overhead, requiring over 6000 s even at 1000 nodes, reflecting the substantial cost of LLM-based reasoning. Moderate runtime growth is observed in GAlign and MTransE, which remain computationally feasible up to 100000 nodes. In contrast, TREA and TEA-GNN demonstrate superior scalability, maintaining reasonable execution times even at 1000000 nodes. These results suggest that methods integrating temporal-aware optimizations are more scalable in large-scale dynamic networks.

As network size increases, all models experience accuracy degradation, primarily due to the growing complexity of node alignment in larger search spaces. Matrix factorization-based and SNN-based methods (CONE-Align, FRUI-P) exhibit a sharp accuracy decline, underscoring their limited adaptability to large-scale networks. Similarly, DNN-based models (GAlign, NAME) suffer noticeable performance drops, with NAME’s accuracy decreasing from 0.869 at 1000 nodes to 0.583 at 35000 nodes. LLM-based models, such as ChatEA, initially demonstrate strong alignment performance but degrade rapidly, reflecting the limitations

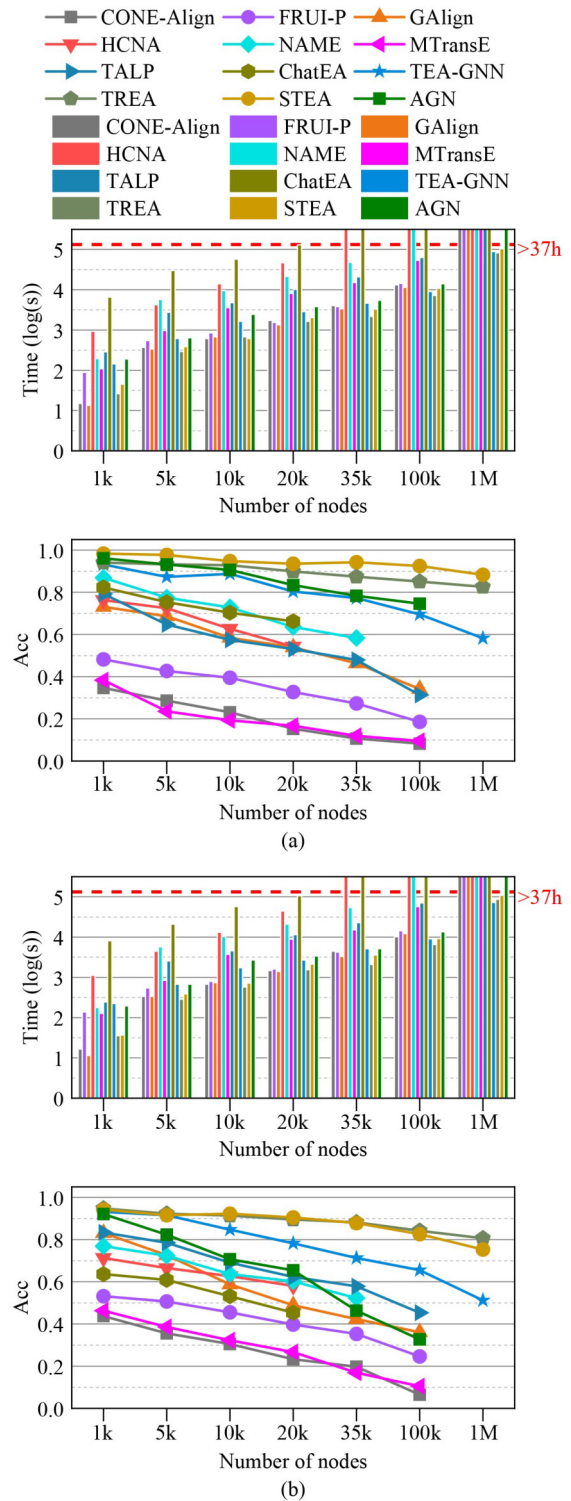


Fig. 12 Scalability of SNA methods. (a) Facebook; (b) Bn

of LLMs in handling large-scale alignment tasks efficiently. In contrast, TREA and STEA maintain superior accuracy, with STEA achieving 0.883 even at 1000000 nodes, highlighting its robustness, scalability, and effectiveness in large-scale scenarios.

5.4 Enhancing SNA with LLMs

In this study, we conducted a comprehensive empirical analysis of

LLMs for SNA. Specifically, we evaluated seven LLM variants: Qwen-7B [98], Qwen-14B [98], Qwen-32B [98], Llama2-7B [99], Llama2-13B [99], ERNIE-Speed-8K [100], and ERNIE-Speed-128K [101], integrated into three state-of-the-art alignment methods: NAME (for static homogeneous networks), TALP (for static heterogeneous networks), and STEA (for dynamic networks). Experiments were conducted on four widely used benchmark datasets: Douban Online-Offline, AllMovie-IMDB (static), and DICEWS-1K, YAGO-WIKI50K-5K (dynamic).

LLM performance evaluation. Figure 13 presents the performance of different LLM-enhanced alignment methods across various datasets. For clarity and conciseness, only Acc metrics are shown, as P@10 and MAP follow similar trends and exhibit comparable fluctuation patterns.

First, larger models consistently outperform smaller ones, demonstrating improved alignment capabilities. Qwen-32B surpasses Qwen-7B on Douban (0.529 vs. 0.501) and AllMovie-IMDB (0.858 vs. 0.828) in NAME tasks, while Llama2-13B outperforms Llama2-7B by 0.021 points on Douban-TALP. This trend is also observed in STEA and TALP, confirming that increasing model size enhances the ability to capture cross-platform semantic correlations and heterogeneous network structures.

Second, architectural differences impact performance. ERNIE-Speed-128K achieves state-of-the-art results, attaining 0.534 on Douban-TALP (+0.026 over Qwen-32B), likely due to its specialized pre-training strategies. Similarly, Qwen-32B outperforms Llama2-13B by 0.015 points on AllMovie-IMDB-NAME (0.858 vs. 0.843), suggesting that its hierarchical representation learning and parameter efficiency better handle long-tail EA. Furthermore, ERNIE’s context-aware attention mechanism improves TALP accuracy by 0.034 points when extending the context window from 8K to 128K, further

emphasizing the impact of architectural innovations.

Third, expanding the context window significantly enhances performance, particularly for complex network alignment tasks. ERNIE-Speed-128K improves NAME’s Douban score to 0.542 (+0.015 over 8K) by capturing cross-platform behavioral sequences, while boosting STEA on DICEWS-1K from 0.949 to 0.954 through global subgraph integration. Notably, extending ERNIE’s context from 8K to 128K yields a 0.022-point improvement in TALP on Douban, surpassing Qwen’s 0.013-point gain from model scaling (7B to 32B). This suggests that context expansion is a more cost-effective alternative to pure model scaling for semantic-intensive tasks requiring long-range dependency modeling.

Runtime and token consumption. To assess the practical feasibility of LLM-based SNA methods, we systematically quantified their runtime and computational resource demands. Although LLMs significantly improve alignment accuracy, their computational overhead remains a major challenge for real-world deployment.

As shown in Figs. 14 and 15, a single inference pass requires 16000 to 22000 seconds, which is 6.1 to 7.1 times longer than the No-LLM version, while consuming a substantial number of tokens (approximately 1,600 per instance). This high computational cost primarily stems from two factors: 1) Cross-network semantic reasoning via multilayer attention mechanisms, which enables long-range dependency modeling but significantly increases inference time. 2) Iterative optimization within ultra-high-dimensional parameter spaces, essential for heterogeneous feature fusion but

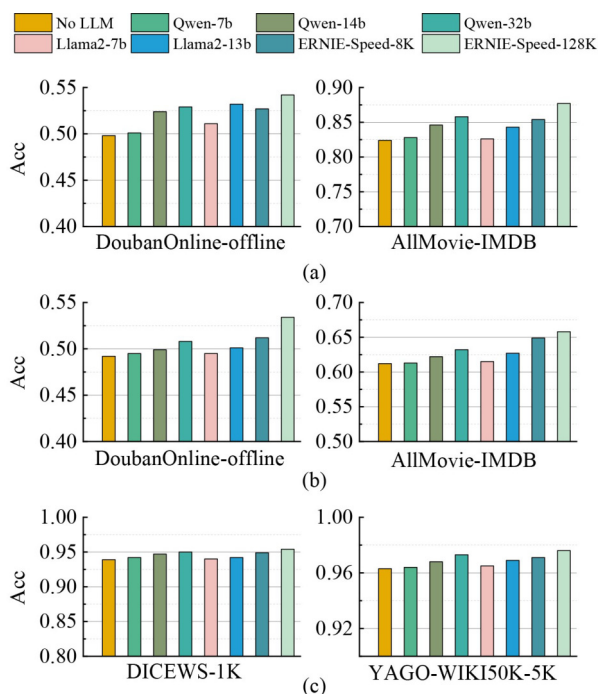


Fig. 13 Performance comparison of LLMs. (a) NAME; (b) TALP; (c) STEA

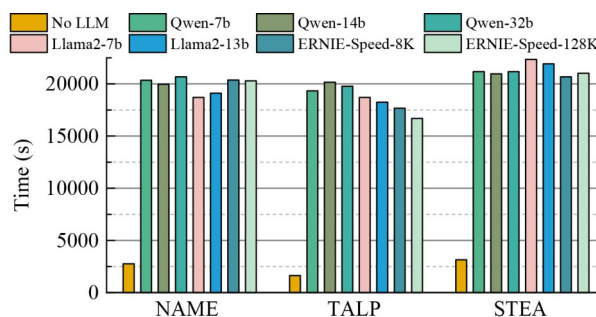


Fig. 14 LLM runtime comparison

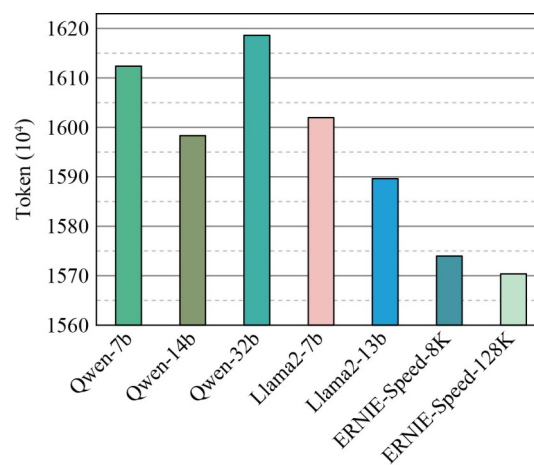


Fig. 15 LLM token consumption

computationally intensive.

To enable practical deployment of LLM-enhanced SNA methods, lightweight optimization techniques are essential. Potential strategies include dynamic quantization, hierarchical attention mechanisms, and adaptive inference pruning, which aim to retain the performance benefits of LLMs while significantly reducing computational costs. These optimizations would make LLM-based SNA approaches more scalable and feasible for real-world applications.

Effect of structural noise on LLMs. The resilience of SNA methods to structural noise is a critical evaluation factor, particularly in real-world scenarios where network data often contains incomplete or erroneous connections. Conventional alignment methods often rely on the assumption of cross-network topological consistency, where corresponding nodes share similar neighborhood structures across platforms. However, privacy constraints, data collection limitations, and platform-specific interaction patterns frequently introduce structural discrepancies, leading to missing or distorted connections. To assess the impact of such perturbations, we systematically introduce controlled structural noise by removing edges with probabilities ranging from 0% to 40%, evaluating the extent to which LLM-enhanced models maintain alignment performance under progressively degraded topological signals.

As shown in Fig. 16, all models experience performance degradation on both datasets (AllMovie-IMDB and Douban Online-Offline) as the edge removal probability increases from 0 to 0.4. This confirms that structural noise disrupts cross-network topological consistency, leading to reduced alignment accuracy. However, the degree of robustness varies across different LLM architectures and model scales. On AllMovie-IMDB, ERNIE-Speed-128K shows greater resilience, with accuracy decreasing from 0.877 to 0.467 (a 46.8% relative decline), while ERNIE-Speed-8K drops more sharply from 0.854 to 0.435 (a 49.1% relative decline). Similar trends are observed on Douban Online-Offline. These results suggest that models with larger context windows better capture global structural patterns and can more effectively infer missing connections, mitigating the impact of structural noise.

These findings underscore the importance of noise-aware pretraining in improving LLM robustness. Developing adaptive learning strategies and structural regularization techniques, such as graph-based adversarial training and contrastive learning for missing-edge recovery, could further enhance alignment performance in noisy environments, making LLM-based SNA methods more applicable to real-world heterogeneous networks.

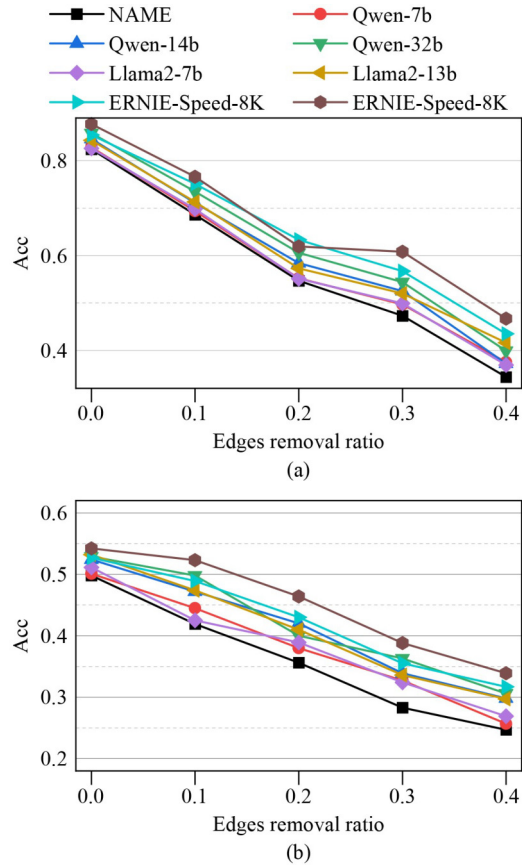


Fig. 16 Effect of structural noise on LLM. (a) AllMovie-IMDB; (b) Douban Online-Offline

5.5 Limitations and discussion

Table 10 provides an overview of different SNA methods, summarizing their strengths, limitations, and applicable scenarios.

DNN-based methods leverage GCNs and adversarial learning to effectively align heterogeneous networks by capturing intricate structural and attribute relationships. However, their high computational requirements necessitate GPU acceleration and optimized memory management for practical deployment.

Matrix factorization and shallow models remain well-suited for structured datasets with homogeneous attributes, offering computational efficiency ideal for resource-limited environments. However, they struggle to scale and adapt when handling complex or large-scale networks.

LLM-enhanced methods leverage advanced semantic reasoning

Table 10 Summary of SNA methods.

Method type	Strengths	Weaknesses	Applicable scenarios
Matrix factorization	Simple, interpretable, efficient for small networks	High cost, poor generalization, limited adaptability	Small, structured networks
Shallow neural networks	Scalable, robust to noise, moderate complexity	Limited features, struggles with heterogeneity, needs anchors	Moderate-sized networks with stable structures
Deep neural networks	Captures complexity, effective for heterogeneity	High cost, risk of overfitting, limited interpretability	Large-scale, heterogeneous networks
LLM-enhanced	Leverages semantic similarity, robust to noisy labels	High demand, pretraining needed, context limits	Networks with rich textual metadata

and cross-domain knowledge transfer, achieving outstanding performance across various datasets. However, their high computational demands for both training and inference pose significant challenges for real-world deployment. Recent efforts have been made to address the high computational cost of LLMs. For example, some methods [102–104] explore distilling reasoning capabilities from large models into smaller dense models, effectively reducing computational overhead while maintaining strong reasoning abilities. In addition, distributed optimization algorithms [105,106] have been proposed, which significantly reduce communication frequency between workers while maintaining or even improving performance compared to fully synchronous training.

In conclusion, the choice of an appropriate SNA method depends on network characteristics, available computational resources, and the alignment objectives. Although deep learning and LLM-enhanced models achieve state-of-the-art performance, their computational demands limit large-scale real-world deployment. In contrast, matrix factorization and shallow models remain viable solutions for structured networks with constrained resources. Future research should focus on improving the scalability, robustness, and adaptability to improve the practical applicability of SNA methods across diverse real-world scenarios.

■ 6 Future work

Although prior research on GRL-based SNA methods has achieved notable progress, significant opportunities remain for further exploration.

(1) Multimodal data integration for SNA. As social networks become increasingly complex, users generate multimodal data encompassing textual content, visual media, and interaction logs. Effectively integrating these heterogeneous sources is critical for improving SNA accuracy, robustness, and generalization [107]. To achieve effective multimodal fusion, alignment models must systematically incorporate semantic features from textual data, visual cues reflecting user preferences, and interaction patterns indicative of social behaviors. Leveraging complementary signals across modalities can enhance cross-platform user representations, leading to improved alignment precision. Furthermore, multimodal integration increases model adaptability, mitigating the impact of data sparsity, inconsistencies, and structural noise, thus enhancing SNA robustness in real-world scenarios.

(2) Explainability for SNA. Although many SNA models achieve high performance, they often lack transparency, making their decision processes difficult to interpret. To address this, Zhou et al. [36] proposed a framework combining adversarial matching with influence function theory to analyze alignment behavior and enhance interpretability.

Several techniques can further improve the explainability of SNA models. Feature attribution methods, such as shapley additive explanations (SHAP) [108,109], can highlight key structural or attribute features influencing alignment decisions, such as a user's degree, clustering coefficient, or profile similarity. Prototype-based interpretability can also be applied by identifying representative aligned and non-aligned node pairs [110], enabling explanation

through similarity to known alignment patterns. Incorporating these techniques into GRL-based SNA pipelines enhances model transparency, facilitates debugging, and improves trust and real-world deployability.

(3) Privacy preservation for SNA. With growing concerns over data privacy, achieving effective network alignment while protecting user information remains a key challenge. Privacy-preserving SNA methods enable cross-platform data integration without exposing sensitive content. For example, homomorphic encryption supports secure computations on encrypted data, enabling privacy-preserving model aggregation [111]. Differential privacy introduces noise to prevent data leakage, offering provable privacy guarantees [112]. Privacy set intersection protocols allow secure user identification across networks using cryptographic techniques [113].

However, these methods often face trade-offs between privacy and performance: homomorphic encryption is computationally intensive, and differential privacy may reduce alignment accuracy. Future work should explore lightweight encryption, adaptive privacy mechanisms, and federated learning to better balance security and efficiency, enabling scalable and privacy-aware SNA systems.

(4) LLMs for SNA. LLMs and SNA exhibit a mutually beneficial relationship: SNA can enhance LLM interpretability, while LLMs can significantly improve SNA effectiveness. However, the high computational cost of LLMs remains a major challenge. To enable scalable deployment of LLM-based SNA models, future research should focus on developing lightweight optimization techniques. For example, zeroth-order optimization eliminates backpropagation during fine-tuning, substantially reducing memory costs. Zhang et al. [114] proposed a backpropagation-free strategy to lower LLM fine-tuning overhead. Additionally, optimizing inference efficiency in multi-turn dialogues is crucial. Key-value (KV) cache reuse can minimize redundant computations and accelerate inference. Gao et al. [115] introduced CachedAttention, an attention mechanism that enables KV cache reuse, significantly improving inference speed. By adopting these strategies, future work can mitigate computational bottlenecks and unlock the full potential of LLMs in SNA applications.

■ 7 Conclusion

The application of GRL in SNA has received increasing attention from both academic and industrial communities. In this paper, we review and summarize GRL-based SNA methods from the perspective of static and dynamic graphs, considering both homogeneous and heterogeneous network characteristics. We also identify key challenges and promising future directions, including improving interpretability and exploring the potential of LLMs in SNA tasks. GRL-based SNA remains a vibrant and evolving research field, and we anticipate that new innovations and significant discoveries will continue to emerge in the future.

■ Acknowledgements

This work was supported by the National Key Research and Development Program of China (No. 2023YFB4502701), the National Natural Science Foundation of China (Grant Nos.

U22A2027, 82090044 and 62402187), the China Postdoctoral Science Foundation (Nos. GZB20240243 and 2024M751009), and the Postdoctoral Project of Hubei Province (No. 2024HBBHCXA024). The research was conducted at the Key Laboratory of Information Storage Systems, the Engineering Research Center for Data System and Technology, Huazhong University of Science and Technology, Wuhan, Hubei, China.

■ Competing interests

The authors declare that they have no competing interests or financial conflicts to disclose.

■ Open Access

This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made.

The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

■ References

- [1] Robson E, Hedges T. Behind the screens: social media trends in 2023. *Global Web Index*, 2023
- [2] Chatzakou D, Soler-Company J, Tsirikla T, Wanner L, Vrochidis S, Kompatsiaris I. User identity linkage in social media using linguistic and social interaction features. In: *Proceedings of the 12th ACM Conference on Web Science*. 2020, 295–304
- [3] Shao J, Wang Y, Shi B, Gao H, Shen H, Cheng X. Adversarial for social privacy: a poisoning strategy to degrade user identity linkage. 2022, arXiv preprint arXiv: 2209.00269
- [4] Jiang M, Cui P, Yuan N J, Xie X, Yang S. Little is much: bridging cross-platform behaviors through overlapped crowds. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. 2016, 13–19
- [5] Lee Y L, Zhou T, Yang K, Du Y, Pan L. Personalized recommender systems based on social relationships and historical behaviors. *Applied Mathematics and Computation*, 2023, 437: 127549
- [6] Shu K, Wang S, Tang J, Zafarani R, Liu H. User identity linkage across online social networks: a review. *ACM SIGKDD Explorations Newsletter*, 2016, 18(2): 5–17
- [7] Trung H T, Toan N T, Van Vinh T, Dat H T, Thang D C, Hung N Q V, Sattar A. A comparative study on network alignment techniques. *Expert Systems with Applications*, 2020, 140: 112883
- [8] Chen B, Chen X. A survey on user alignment across social networks. *Journal of Xihua University (Natural Science Edition)*, 2021, 40(4): 11–26
- [9] Mika P. Flink: semantic web technology for the extraction and analysis of social networks. *Journal of Web Semantics*, 2005, 3(2–3): 211–223
- [10] Zafarani R, Liu H. Connecting users across social media sites: a behavioral-modeling approach. In: *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2013, 41–49
- [11] Riederer C, Kim Y, Chaintreau A, Korula N, Lattanzi S. Linking users across domains with location data: theory and validation. In: *Proceedings of the 25th International Conference on World Wide Web*. 2016, 707–719
- [12] Kong X, Zhang J, Yu P S. Inferring anchor links across multiple heterogeneous social networks. In: *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*. 2013, 179–188
- [13] Liu S, Wang S, Zhu F, Zhang J, Krishnan R. HYDRA: large-scale social identity linkage via heterogeneous behavior modeling. In: *Proceedings of 2014 ACM SIGMOD International Conference on Management of Data*. 2014, 51–62
- [14] Goga O, Lei H, Parthasarathi S H K, Friedland G, Sommer R, Teixeira R. Exploiting innocuous activity for correlating users across sites. In: *Proceedings of the 22nd International Conference on World Wide Web*. 2013, 447–458
- [15] Narayanan A, Paskov H, Gong N Z, Bethencourt J, Stefanov E, Shin E C R, Song D. On the feasibility of internet-scale author identification. In: *Proceedings of 2012 IEEE Symposium on Security and Privacy*. 2012, 300–314
- [16] Zhang Z, Gu Q, Yue T, Su S. Identifying the same person across two similar social networks in a unified way: globally and locally. *Information Sciences*, 2017, 394–395: 53–67
- [17] Zhou X, Liang X, Zhang H, Ma Y. Cross-platform identification of anonymous identical users in multiple social media networks. *IEEE Transactions on Knowledge and Data Engineering*, 2016, 28(2): 411–424
- [18] Chen F, Wang Y C, Wang B, Kuo C C J. Graph representation learning: a survey. *APSIPA Transactions on Signal and Information Processing*, 2020, 9: e15
- [19] Li B, Pi D. Network representation learning: a systematic literature review. *Neural Computing and Applications*, 2020, 32(21): 16647–16679
- [20] Hoang V T, Jeon H J, You E S, Yoon Y, Jung S, Lee O J. Graph representation learning and its applications: a survey. *Sensors*, 2023, 23(8): 4168
- [21] Perozzi B, Al-Rfou R, Skiena S. DeepWalk: online learning of social representations. In: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2014, 701–710
- [22] Grover A, Leskovec J. node2vec: scalable feature learning for networks. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016, 855–864
- [23] Qiu J, Dong Y, Ma H, Li J, Wang K, Tang J. Network embedding as matrix factorization: unifying DeepWalk, LINE, PTE, and node2vec. In: *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*. 2018, 459–467
- [24] Zhang J, Dong Y, Wang Y, Tang J, Ding M. ProNE: fast and scalable network representation learning. In: *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. 2019, 4278–4284

- [25] Dong Y, Chawla N V, Swami A. metapath2vec: scalable representation learning for heterogeneous networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2017, 135–144
- [26] Liu X, Wu K, Liu B, Qian R. HNERec: scientific collaborator recommendation model based on heterogeneous network embedding. *Information Processing & Management*, 2023, 60(2): 103253
- [27] Liu J, Song L, Wang G, Shang X. Meta-HGT: metapath-aware HyperGraph transformer for heterogeneous information network embedding. *Neural Networks*, 2023, 157: 65–76
- [28] Nguyen G H, Lee J B, Rossi R A, Ahmed N K, Koh E, Kim S. Continuous-time dynamic network embeddings. In: Proceedings of the Web Conference 2018. 2018, 969–976
- [29] Zhang P, Yan Y, Li C, Wang S, Xie X, Song G, Kim S. Continual learning on dynamic graphs via parameter isolation. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2023, 601–611
- [30] Ji C, Zhao T, Sun Q, Fu X, Li J. Higher-order memory guided temporal random walk for dynamic heterogeneous network embedding. *Pattern Recognition*, 2023, 143: 109766
- [31] Yang L, Chen H, Wang X, Yang J, Wang F Y, Liu H. Two heads are better than one: integrating knowledge from knowledge graphs and large language models for entity alignment. 2024, arXiv preprint arXiv: 2401.16960
- [32] Jiang X, Shen Y, Shi Z, Xu C, Li W, Li Z, Guo J, Shen H, Wang Y. Unlocking the power of large language models for entity alignment. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2024, 7566–7583
- [33] Zhu B, Wang R, Wang J, Shao F, Wang K. A survey: knowledge graph entity alignment research based on graph embedding. *Artificial Intelligence Review*, 2024, 57(9): 229
- [34] Zeng K, Li C, Hou L, Li J, Feng L. A comprehensive survey of entity alignment for knowledge graphs. *AI Open*, 2021, 2: 1–13
- [35] Zhu D, Cui P, Zhang Z, Pei J, Zhu W. High-order proximity preserved embedding for dynamic networks. *IEEE Transactions on Knowledge and Data Engineering*, 2018, 30(11): 2134–2144
- [36] Zhou F, Wen Z, Trajcevski G, Zhang K, Zhong T, Liu F. Disentangled network alignment with matching explainability. In: Proceedings of the IEEE Conference on Computer Communications. 2019, 1360–1368
- [37] Li X, Shang Y, Cao Y, Li Y, Tan J, Liu Y. Type-aware anchor link prediction across heterogeneous networks based on graph attention network. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence. 2020, 147–155
- [38] Zheng C, Pan L, Wu P. JORA: weakly supervised user identity linkage via jointly learning to represent and align. *IEEE Transactions on Neural Networks and Learning Systems*, 2024, 35(3): 3900–3911
- [39] Heimann M, Shen H, Safavi T, Koutra D. REGAL: representation learning-based graph alignment. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 2018, 117–126
- [40] Chen X, Heimann M, Vahedian F, Koutra D. CONE-Align: consistent network alignment with proximity-preserving node embedding. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2020, 1985–1988
- [41] Man T, Shen H, Liu S, Jin X, Cheng X. Predict anchor links across social networks via an embedding approach. In: Proceedings of the 25th International Joint Conference on Artificial Intelligence. 2016, 1823–1829
- [42] Zhou X, Liang X, Du X, Zhao J. Structure based user identification across social networks. *IEEE Transactions on Knowledge and Data Engineering*, 2017, 30(6): 1178–1191
- [43] Du X, Yan J, Zha H. Joint link prediction and network alignment via cross-graph embedding. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence. 2019, 2251–2257
- [44] Qiao Y, Wu Y, Duo F, Lin W, Yang J. Siamese neural networks for user identity linkage through web browsing. *IEEE Transactions on Neural Networks and Learning Systems*, 2020, 31(8): 2741–2751
- [45] Trung H T, Van Vinh T, Tam N T, Yin H, Weidlich M, Hung N Q V. Adaptive network alignment with unsupervised and multi-order convolutional networks. In: Proceedings of the 36th International Conference on Data Engineering (ICDE). 2020, 85–96
- [46] Gao J, Huang X, Li J. Unsupervised graph alignment with Wasserstein distance discriminator. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021, 426–435
- [47] Saxena S, Chakraborty R, Chandra J. HCNA: hyperbolic contrastive learning framework for self-supervised network alignment. *Information Processing & Management*, 2022, 59(5): 103021
- [48] Yang L, Wang X, Zhang J, Yang J, Xu Y, Hou J, Xin K, Wang F Y. HackGAN: harmonious cross-network mapping using CycleGAN with Wasserstein-Procrustes learning for unsupervised network alignment. *IEEE Transactions on Computational Social Systems*, 2023, 10(2): 746–759
- [49] Hong H, Li X, Pan Y, Tsang I W. Domain-adversarial network alignment. *IEEE Transactions on Knowledge and Data Engineering*, 2022, 34(7): 3211–3224
- [50] Huynh T T, Duong C T, Nguyen T T, Van V T, Sattar A, Yin H, Nguyen Q V H. Network alignment with holistic embeddings. *IEEE Transactions on Knowledge and Data Engineering*, 2023, 35(2): 1881–1894
- [51] Zhou J, Fan J. Translink: user identity linkage across heterogeneous social networks via translating embeddings. In: Proceedings of the IEEE Conference on Computer Communications. 2019, 2116–2124
- [52] Chen M, Tian Y, Yang M, Zaniolo C. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence. 2017, 1511–1517
- [53] Feng J, Zhang M, Wang H, Yang Z, Zhang C, Li Y, Jin D. DPLink: user identity linkage via deep neural network from heterogeneous mobility data. In: Proceedings of the World Wide Web Conference. 2019, 459–469
- [54] Chen S, Wang J, Du X, Hu Y. A novel framework with information fusion and neighborhood enhancement for user identity linkage. In: Proceedings of the 24th European Conference on Artificial Intelligence. 2020, 1754–1761
- [55] Sun L, Zhang Z, Ji P, Wen J, Su S, Yu P S. DNA: dynamic social network alignment. In: Proceedings of 2019 IEEE International Conference on Big Data (Big Data). 2019, 1224–1231

- [56] Sun L, Zhang Z, Wang F, Ji P, Wen J, Su S, Philip S Y. Aligning dynamic social networks: an optimization over dynamic graph autoencoder. *IEEE Transactions on Knowledge and Data Engineering*, 2023, 35(6): 5597–5611
- [57] Wang F, Ji P, Sun L, Wei Q, Li G, Zhang Z. A deep learning based dynamic social network alignment method. *Acta Electronica Sinica*, 2022, 50(8): 1925–1936
- [58] Liang S, Tang S, Meng Z, Zhang Q. Cross-temporal snapshot alignment for dynamic networks. *IEEE Transactions on Knowledge and Data Engineering*, 2023, 35(3): 2406–2420
- [59] Xu C, Su F, Lehmann J. Time-aware graph neural network for entity alignment between temporal knowledge graphs. In: *Proceedings of 2021 Conference on Empirical Methods in Natural Language Processing*. 2021, 8999–9010
- [60] Xu C, Su F, Xiong B, Lehmann J. Time-aware entity alignment using temporal relational attention. In: *Proceedings of the ACM Web Conference 2022*. 2022, 788–797
- [61] Cai L, Mao X, Ma M, Yuan H, Zhu J, Lan M. A simple temporal information matching mechanism for entity alignment between temporal knowledge graphs. In: *Proceedings of the 29th International Conference on Computational Linguistics*. 2022, 2075–2086
- [62] Li J, Song D, Wang H, Wu Z, Zhou C, Zhou Y. Entity alignment for temporal knowledge graphs via adaptive graph networks. *Knowledge-Based Systems*, 2023, 274: 110631
- [63] Cao S, Lu W, Xu Q. GraRep: learning graph representations with global structural information. In: *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. 2015, 891–900
- [64] Ou M, Cui P, Pei J, Zhang Z, Zhu W. Asymmetric transitivity preserving graph embedding. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016, 1105–1114
- [65] Tang J, Qu M, Wang M, Zhang M, Yan J, Mei Q. LINE: large-scale information network embedding. In: *Proceedings of the 24th International Conference on World Wide Web*. 2015, 1067–1077
- [66] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks. In: *Proceedings of the 5th International Conference on Learning Representations*. 2017
- [67] Hamilton W L, Ying Z, Leskovec J. Inductive representation learning on large graphs. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2017, 1025–1035
- [68] Veličković P, Cucurull G, Casanova A, Romero A, Liò P, Bengio Y. Graph attention networks. In: *Proceedings of the 6th International Conference on Learning Representations*. 2018
- [69] Wang H, Wang J, Wang J, Zhao M, Zhang W, Zhang F, Xie X, Guo M. GraphGAN: graph representation learning with generative adversarial nets. In: *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*. 2018
- [70] Li J, Wang H. Graph diffusive self-supervised learning for social recommendation. In: *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2024, 2442–2446
- [71] Xia L, Huang C, Huang C, Lin K, Yu T, Kao B. Automated self-supervised learning for recommendation. In: *Proceedings of the ACM Web Conference 2023*. 2023, 992–1002
- [72] Wang Q, Mao Z, Wang B, Guo L. Knowledge graph embedding: a survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 2017, 29(12): 2724–2743
- [73] Ji S, Pan S, Cambria E, Marttinen P, Philip S Y. A survey on knowledge graphs: representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*, 2022, 33(2): 494–514
- [74] Pan S, Luo L, Wang Y, Chen C, Wang J, Wu X. Unifying large language models and knowledge graphs: a roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 2024, 36(7): 3580–3599
- [75] Zhu Y, Wang X, Chen J, Qiao S, Ou Y, Yao Y, Deng S, Chen H, Zhang N. LLMs for knowledge graph construction and reasoning: recent capabilities and future opportunities. *World Wide Web*, 2024, 27(5): 58
- [76] Bordes A, Usunier N, Garcia-Duran A, Weston J, Yakhnenko O. Translating embeddings for modeling multi-relational data. In: *Proceedings of the 27th International Conference on Neural Information Processing Systems*. 2013, 2787–2795
- [77] Jiang J, Ferrara E. Social-LLM: modeling user behavior at scale using language models and social network data. 2023, arXiv preprint arXiv: 2401.00893
- [78] Zhang Y, Sharma K, Du L, Liu Y. Toward mitigating misinformation and social media manipulation in LLM era. In: *Proceedings of the ACM Web Conference 2024*. 2024, 1302–1305
- [79] Li C, Wong C, Zhang S, Usuyama N, Liu H, Yang J, Naumann T, Poon H, Gao J. LLaVA-med: training a large language-and-vision assistant for biomedicine in one day. In: *Proceedings of the 37th International Conference on Neural Information Processing Systems*. 2023, 1240
- [80] Skitsas K, Orłowski K, Hermanns J, Mottin D, Karras P. Comprehensive evaluation of algorithms for unrestricted graph alignment. In: *Proceedings 26th International Conference on Extending Database Technology*. 2023, 260–272
- [81] Sun Z, Zhang Q, Hu W, Wang C, Chen M, Akrami F, Li C. A benchmarking study of embedding-based entity alignment for knowledge graphs. *Proceedings of the VLDB Endowment*, 2020, 13(11): 2326–2340
- [82] Zhao X, Zeng W, Tang J, Wang W, Suchanek F M. An experimental study of state-of-the-art entity alignment approaches. *IEEE Transactions on Knowledge and Data Engineering*, 2022, 34(6): 2610–2625
- [83] Zhang R, Trisedya B D, Li M, Jiang Y, Qi J. A benchmark and comprehensive survey on knowledge graph entity alignment via representation learning. *The VLDB Journal*, 2022, 31(5): 1143–1168
- [84] Zhang S, Tong H, Xu J, Hu Y, Maciejewski R. ORIGIN: non-rigid network alignment. In: *Proceedings of 2019 IEEE International Conference on Big Data (Big Data)*. 2019, 998–1007
- [85] Liu L, Cheung W K, Li X, Liao L. Aligning users across social networks using network embedding. In: *Proceedings of the 25th International Joint Conference on Artificial Intelligence*. 2016, 1774–1780
- [86] Liu L, Li X, Cheung W K, Liao L. Structural representation learning for user alignment across social networks. *IEEE Transactions on Knowledge and Data Engineering*, 2020, 32(9): 1824–1837
- [87] Zhou F, Liu L, Zhang K, Trajcevski G, Wu J, Zhong T. DeepLink: a deep learning approach for user identity linkage. In: *Proceedings of the*

- IEEE Conference on Computer Communications. 2018, 1313–1321
- [88] Li G, Sun L, Zhang Z, Ji P, Su S, Yu P S. MC²: unsupervised multiple social network alignment. In: Proceedings of 2019 IEEE International Conference on Big Data (Big Data). 2019, 1151–1156
- [89] Fu S, Wang G, Xia S, Liu L. Deep multi-granularity graph embedding for user identity linkage across social networks. Knowledge-Based Systems, 2020, 193: 105301
- [90] Su S, Sun L, Zhang Z, Li G, Qu J. Master: across multiple social networks, integrate attribute and structure embedding for reconciliation. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence. 2018, 3863–3869
- [91] Chen H, Yin H, Sun X, Chen T, Gabrys B, Musial K. Multi-level graph convolutional networks for cross-platform anchor link prediction. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020, 1503–1511
- [92] Li C, Wang S, Yu P S, Zheng L, Zhang X, Li Z, Liang Y. Distribution distance minimization for unsupervised user identity linkage. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 2018, 447–456
- [93] Li C, Wang S, Wang Y, Yu P, Liang Y, Liu Y, Li Z. Adversarial learning for weakly-supervised social network alignment. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence. 2019, 996–1003
- [94] Zhou Y, Ren J, Jin R, Zhang Z, Zheng J, Jiang Z, Yan D, Dou D. Unsupervised adversarial network alignment with reinforcement learning. ACM Transactions on Knowledge Discovery from Data (TKDD), 2022, 16(3): 50
- [95] García-Durán A, Dumančić S, Niepert M. Learning sequence encoders for temporal knowledge graph completion. In: Proceedings of 2018 Conference on Empirical Methods in Natural Language Processing. 2018, 4816–4821
- [96] Lacroix T, Obozinski G, Usunier N. Tensor decompositions for temporal knowledge base completion. 2020, arXiv preprint arXiv: 2004.04926
- [97] Watts D J, Strogatz S H. Collective dynamics of ‘small-world’ networks. Nature, 1998, 393(6684): 440–442
- [98] Bai J, Bai S, Chu Y, Cui Z, Dang K, Deng X, Fan Y, Ge W, Han Y, Huang F, Hui B, Ji L, Li M, Lin J, Lin R, Liu D, Liu G, Lu C, Lu K, Ma J, Men R, Ren X, Ren X, Tan C, Tan S, Tu J, Wang P, Wang S, Wang W, Wu S, Xu B, Xu J, Yang A, Yang H, Yang J, Yang S, Yao Y, Yu B, Yuan H, Yuan Z, Zhang J, Zhang X, Zhang Y, Zhang Z, Zhou C, Zhou J, Zhou X, Zhu T. Qwen technical report. 2023, arXiv preprint arXiv: 2309.16609
- [99] Touvron H, Martin L, Stone K, Albert P, Almahairi A, Babaei Y, Bashlykov N, Batra S, Bhargava P, Bhosale S, Bikel D, Blecher L, Ferrer C C, Chen M, Cucurull G, Esiobu D, Fernandes J, Fu J, Fu W, Fuller B, Gao C, Goswami V, Goyal N, Hartshorn A, Hosseini S, Hou R, Inan H, Kardas M, Kerkez V, Khabsa M, Kloumann I, Korenev A, Koura P S, Lachaux M A, Lavril T, Lee J, Liskovich D, Lu Y, Mao Y, Martinet X, Mihaylov T, Mishra P, Molybog I, Nie Y, Poulton A, Reizenstein J, Rungta R, Saladi K, Schelten A, Silva R, Smith E M, Subramanian R, Tan X E, Tang B, Taylor R, Williams A, Kuan J X, Xu P, Yan Z, Zarov I, Zhang Y, Fan A, Kambadur M, Narang S, Rodriguez A, Stojnic R, Edunov S, Scialom T. Llama 2: open foundation and fine-tuned chat models. 2023, arXiv preprint arXiv: 2307.09288
- [100] Baidu Intelligent Cloud Documentation. ERNIE-Speed-8K - Qianfan Large Model Platform. See cloud.baidu.com/doc/WENXIN-WORKSHOP/s/klqx7b1xf website, 2025 (in Chinese)
- [101] Baidu Intelligent Cloud Documentation. ERNIE-Speed-128K - Qianfan Large Model Platform. See cloud.baidu.com/doc/WENXIN-WORKSHOP/s/6ltgkzya5 website, 2025 (in Chinese)
- [102] Guo D, Yang D, Zhang H, Song J, Zhang R, Xu R, Zhu Q, Ma S, Wang P, Bi X, Zhang X, Yu X, Wu Y, Wu Z F, Gou Z, Shao Z, Li Z, Gao Z, Liu A, Xue B, Wang B, Wu B, Feng B, Lu C, Zhao C, Deng C, Zhang C, Ruan C, Dai D, Chen D, Ji D, Li E, Lin F, Dai F, Luo F, Hao G, Chen G, Li G, Zhang H, Bao H, Xu H, Wang H, Ding H, Xin H, Gao H, Qu H, Li H, Guo J, Li J, Wang J, Chen J, Yuan J, Qiu J, Li J, Cai J L, Ni J, Liang J, Chen J, Dong K, Hu K, Gao K, Guan K, Huang K, Yu K, Wang L, Zhang L, Zhao L, Wang L, Zhang L, Xu L, Xia L, Zhang M, Zhang M, Tang M, Li M, Wang M, Li M, Tian N, Huang P, Zhang P, Wang Q, Chen Q, Du Q, Ge R, Zhang R, Pan R, Wang R, Chen R J, Jin R L, Chen R, Lu S, Zhou S, Chen S, Ye S, Wang S, Yu S, Zhou S, Pan S, Li S S, Zhou S, Wu S, Ye S, Yun T, Pei T, Sun T, Wang T, Zeng W, Zhao W, Liu W, Liang W, Gao W, Yu W, Zhang W, Xiao W L, An W, Liu X, Wang X, Chen X, Nie X, Cheng X, Liu X, Xie X, Liu X, Yang X, Li X, Su X, Lin X, Li X Q, Jin X, Shen X, Chen X, Sun X, Wang X, Song X, Zhou X, Wang X, Shan X, Li Y K, Wang Y Q, Wei Y X, Zhang Y, Xu Y, Li Y, Zhao Y, Sun Y, Wang Y, Yu Y, Zhang Y, Shi Y, Xiong Y, He Y, Piao Y, Wang Y, Tan Y, Ma Y, Liu Y, Guo Y, Ou Y, Wang Y, Gong Y, Zou Y, He Y, Xiong Y, Luo Y, You Y, Liu Y, Zhou Y, Zhu Y X, Xu Y, Huang Y, Li Y, Zheng Y, Zhu Y, Ma Y, Tang Y, Zha Y, Yan Y, Ren Z Z, Ren Z, Sha Z, Fu Z, Xu Z, Xie Z, Zhang Z, Hao Z, Ma Z, Yan Z, Wu Z, Gu Z, Zhu Z, Liu Z, Li Z, Xie Z, Song Z, Pan Z, Huang Z, Xu Z, Zhang Z, Zhang Z. DeepSeek-R1: incentivizing reasoning capability in LLMs via reinforcement learning. 2025, arXiv preprint arXiv: 2501.12948
- [103] Liao H, He S, Hao Y, Li X, Zhang Y, Zhao J, Liu K. SKIntern: internalizing symbolic knowledge for distilling better CoT capabilities into small language models. In: Proceedings of the 31st International Conference on Computational Linguistics. 2025, 3203–3221
- [104] Chen X, Sun Z, Guo W, Zhang M, Chen Y, Sun Y, Su H, Pan Y, Klakow D, Li W, Shen X. Unveiling the key factors for distilling chain-of-thought reasoning. 2025, arXiv preprint arXiv: 2502.18001
- [105] Douillard A, Feng Q, Rusu A A, Chhaparia R, Donchev Y, Kuncoro A, Ranzato M, Szlam A, Shen J. DiLoCo: distributed low-communication training of language models. 2023, arXiv preprint arXiv: 2311.08105
- [106] Jaghouar S, Ong J M, Hagemann J. OpenDiLoCo: an open-source framework for globally distributed low-communication training. 2024, arXiv preprint arXiv: 2407.07852
- [107] Li S, Tang H. Multimodal alignment and fusion: a survey. 2024, arXiv preprint arXiv: 2411.17040
- [108] Koushik A, Manoj M, Nezamuddin N. Shapley additive explanations for explaining artificial neural network based mode choice models. Transportation in Developing Economies, 2024, 10(1): 12
- [109] Gebreyesus Y, Dalton D, De Chiara D, Chinnici M, Chinnici A. AI for automating data center operations: model explainability in the data Centre context using Shapley additive explanations (SHAP). Electronics, 2024, 13(9): 1628
- [110] Davoudi S O. On the interpretability and explainability of

prototype-based methods and reinforcement learning. Carleton University, Dissertation, 2024

[111] Zhou L, Ma X J, Pan D H, Fan D M, Zhang H F, Zhong K. HE-SNA: an efficient cross-platform network alignment scheme from privacy-aware perspective. *Complex & Intelligent Systems*, 2023, 9(5): 6009–6022

[112] Chen J, Hu C, Sheng W, Li R, Zhao R, Yu J. A trust-based personalized differential privacy guarantees for online social networks. *IEEE Transactions on Network and Service Management*, 2025

[113] Yang X, Liu Y, He S. Overview of privacy set intersection protocol based on heterogeneous network and social network user alignment. *IEEE Transactions on Network Science and Engineering*, 2024, 11(6): 6692–6703

[114] Zhang Y, Li P, Hong J, Li J, Zhang Y, Zheng W, Chen P Y, Lee J D, Yin W, Hong M, Hong M, Wang Z, Liu S, Chen T. Revisiting zeroth-order optimization for memory-efficient LLM fine-tuning: a benchmark. In: *Proceedings of the 41st International Conference on Machine Learning*. 2024, 2444

[115] Gao B, He Z, Sharma P, Kang Q, Jevdjic D, Deng J, Yang X, Yu Z, Zuo P. Cost-efficient large language model serving for multi-turn conversations with CachedAttention. In: *Proceedings of 2024 USENIX Conference on USENIX Annual Technical Conference*. 2024, 7



Yutong WU is currently a PhD student at the Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology, China. Her research interests focus on storage and processing systems for big data. She publishes multiple papers in major international conferences in these areas, including USENIX ATC, SC, and APWeb-WAIM.



Feiyang LI is currently studying for a PhD degree at Huazhong University of Science and Technology, China. His research interests include large language models and knowledge graphs.



Zhan SHI is currently an associate professor in Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology, China. His present research interests include distributed storage, cloud storage, and graph storage. He publishes multiple papers in major journals and international conferences in these areas, including DAC, ATC, EuroSys, ICPP, ICDCS, CloudCom, CCGrid, SC, ICC, ACM TOS, IEEE TSC, etc.



Zhipeng TIAN is currently pursuing a master's degree at the School of Computer Science and Technology, Huazhong University of Science and Technology, China under the supervision of Associate Professor Shi Zhan. He graduated with a bachelor's degree from the School of Information Science and Engineering at Hunan University, China in 2022. His research interests include streaming graph processing.



Wang ZHANG is currently pursuing the PhD degree with the Institute of Artificial Intelligence, Huazhong University of Science and Technology, China. Before that, he was a research assistant at South-Central Minzu University, China. His research interests include AI for storage system performance optimization and AI for advanced data management. He publishes multiple papers in major international conferences and journals in these areas, including SC, DASFAA, IEEE TII, IEEE TITS, KBS.



Peng FANG received his PhD degree in computer architecture from Huazhong University of Science and Technology, China in 2023. He is currently a postdoctoral researcher with Huazhong University of Science and Technology. His research interests include computer architecture, graph learning, and big data analysis. He has several publications in major international conferences and journals, including VLDB, ICDE, ATC, ICPP, IWQoS, SBP-BRiMS, IEEE TKDE, IEEE TBD, FCS, and Pyhisica A.



Renzhi XIAO received PhD degree majoring in computer system architecture from Huazhong University of Science and Technology, China in 2024. He is currently a lecturer of computer science and technology at Jiangxi University of Science and Technology, China. His research focuses on Non-Volatile Memory (NVM) or Persistent Memory (PM), In-Memory Key-Value Stores, and PM-based Data Structures. He has published over ten papers in international conferences and leading journals, including ICCD, Euro-Par, ISPA, TACO, JSA, JPDC, and TODAES.



Fang WANG received her PhD degree in computer architecture in 2001 from Huazhong University of Science and Technology (HUST), China. She is a professor of computer science and engineering at HUST, China. Her interests include distribute file systems, parallel I/O storage systems, and graph processing systems. She has more than 80 publications in major journals and conferences, including IEEE-TC, IEEE-TPDS, IEEE-NSM, ACM TACO, SC, MSST, DATE, HiPC, ICDCS, HPDC, ICCD, ICDE, and ICPP.



Dan FENG received the BE, ME, and PhD degrees in Computer Science and Technology from Huazhong University of Science and Technology (HUST), China in 1991, 1994, and 1997, respectively. She is a professor and dean of the School of Computer Science and Technology, HUST. Her research interests include computer architecture, massive storage systems, and parallel file systems. She has more than 100 publications in major journals and international conferences, including IEEE-TC, IEEE-TPDS, ACM-TOS, JCST, FAST, USENIX ATC, ICDCS, HPDC, SC, ICS, IPDPS, and ICPP. She is a member of the Association for Computing Machinery and the Chair of the Information Storage Technology Committee, Chinese Computer Academy.