APPENDIX A
MORE RELATED WORK

The problem of interlayer link prediction in the multiplex network is typically solved by leveraging feature or structure information accessed from the multiple SMNs [1]. Early studies focused on feature information; they analyzed user profiles, location trajectories, and user-generated content to link nodes across different SMNs belonging to the same user. Profile features included username, image, position, birthday, job, and experience, among others [2]. The authors of Refs. [3]–[7] explored ways of using usernames for prediction. References [8]–[13] considered various profile attributes to improve prediction performance. With the rapid development of SMNs, many users began using different usernames in different SMNs for security reasons. Meanwhile, the accessible profile information among SMNs became increasingly fragmented, unavailable, and disruptive [14]. These SMN characteristics marginalized the traditional profile-based resolutions. The trajectory-based method has been popular since the emergence of the mobile-phone-based Internet. SMN users who wish to announce their location to their friends on some SMN applications can tap a "check-in" button to see a list of nearby places and choose the place that matches their location. References [15]–[17] focused on these check-in data and used them to link identities. Such trajectory-based methods, however, often face data sparsity problems, and users usually share different locations on different SMNs. User-generated content can reveal some unique characteristics of an SMN user, such as his or her writing style [18], [19] or footprint [20]. These methods rely heavily on the availability of excellent natural language processing (NLP) techniques and text preprocessing algorithms because user-generated content often includes spoken words, emotion icons, and abbreviations.

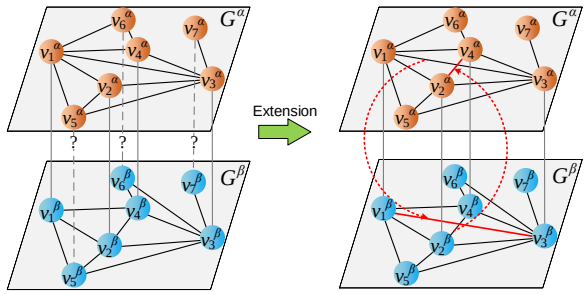
Different from feature information, network's structural information is highly accessible and difficult to counterfeit. In addition, a user's friend circle is highly personalized; i.e., few people share the same friend circle [21]. Therefore, network-based methods are an ideal solution for the interlayer link prediction problem and have attracted the interest of an increasing number of researchers in recent years. Network-based methods can be divided into non-embedding-based methods and embedding-based methods according to whether network embedding techniques are used. We will introduce non-embedding-based methods as follows.

Given the completeness and connectivity of a network structure, two kinds of structural information can be used to solve the interlayer link prediction problem. The first is local network information, which focuses on the one-hop neighborhood (e.g., follower/followee/friend relationships) of the unmatched nodes [1]. Narayanan and Shmatikov proposed a re-identification algorithm, which was the first method to use a graph-theoretic model based on the node neighborhood to solve this problem [22]. Later, Korula et al. [23] computed a similarity score for an unmatched node pair by counting the number of CMNs and then keeping all the links above a specific threshold. To avoid the possible problem of mismatching low-degree nodes in the early phases, only nodes whose degree is higher than a specified threshold are allowed to be matched. Zhou et al. [21] proposed a friend-relationship-based user identification (FRUI) algorithm that counts the number of shared friends to calculate the degree of match for all candidate-matched node pairs and chooses pairs that have the maximum value as the final set of matched pairs. Tang et al. [24] further investigated the importance of the scale-free property of real-world SMNs for accomplishing interlayer link prediction and proposed a degree penalty principle to calculate the degree of match of all unmatched node pairs. Ren et al. [25] defined a set of meta-diagrams for feature extraction and used greedy link selection for the interlayer link prediction. In Ref. [26], an algorithm is proposed to resolve the one-to-one constraint in the situation of prediction cross-multiple layers. This algorithm matches different layers of the multiplex network by minimizing the friendship inconsistency and selects the candidate node pairs which can lead to the maximum confidence scores across multiple layers.

The second type of structural information is global network information. Zhu et al. [27] transformed the interlayer link prediction problem into a maximum common subgraph problem and maximized the number of intralayer links to obtain a cross-layer mapping. Zafarani and Liu [28] also explored a solution utilizing global network information. They calculated the Laplacian matrices for each layer and used a matrix optimization method to perform the prediction. The authors

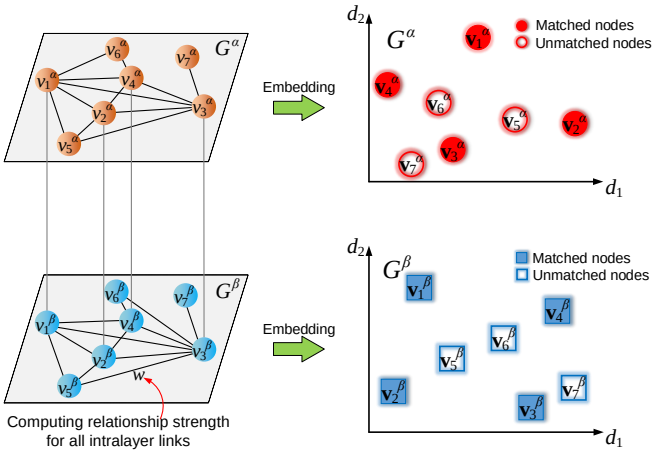
TABLE I: Symbols and notations

Symbol	Description
$\mathcal{M}$	The multiplex network.
G	A SMN which is one layer of $\mathcal{M}$.
u, v	Nodes in $\mathcal{M}$.
$\mathbf{u}, \mathbf{v}$	Embedding vectors of nodes u and v respectively.
α, β	Layer indices of $\mathcal{M}$.
e^{α}, e^{β}	Intralayer links in G^{α} and G^{β} respectively.
$\mathbf{e}$,$\mathbf{E}$	Intralayer links vector and intralayer link matrix respectively.
$e^{\alpha\beta}$	Interlayer link.
i, j, a, b	Node indices.
n^{α}, n^{β}	Number of nodes in G^{α} and G^{β}.
n, m	Number of a priori interlayer links and unobserved interlayer links respectively.
$\Gamma(v_i)$	Set of neighbors of node v_i.
k_v	Degree of node v
$p, \mathbf{P}$	Degree of vector consistency and Degree of vector consistency matrix respectively.
$q, \mathbf{Q}$	Degree of distance consistency and Degree of distance consistency matrix respectively.
$r, \mathbf{R}$	Degree of match and Degree of match matrix respectively.
d	Dimensionality of the latent representation space.
ϕ	Mapping function.
w	weight of the intralayer link.
δ	control parameter.
Φ	Set of a priori interlayer links.
Ψ	Set of unobserved interlayer links.



(a) Original multiplex network (b) Extended multiplex network

Fig. 1: Example of cross-layer extension. (a) Original multiplex network. Given a multiplex network with two layers and each layer having seven nodes, seven interlayer links exist between the nodes across the layers; four of these are a priori interlayer links, whereas the remaining three are unobserved interlayer links. (b) Extended multiplex network. Some of the missing intralayer links between the matched nodes in one layer can be assumed to be present with the assistance of their counterpart nodes in the other layer. For example, there exists an intralayer link e_{24}^{β} between the matched nodes v_2^{β} and v_4^{β}; this enables extension of the layer α with the intralayer link e_{24}^{α} (red line in layer α).



(a) Weighted multiplex network (b) Latent representation spaces

Fig. 2: Example of network embedding. (a) Weighted multiplex network: The relationship strengths between the nodes having intralayer links are computed to model each layer as a weighted graph. (b) Latent representation spaces: The nodes in different layers are represented as low-dimensional vectors in separate latent spaces. For example, node v_1^{α} is represented as a two-dimensional vector $\mathbf{v}_1^{\alpha}$ in the latent space G^{α}.

of Ref. [29] considered both local and global consistency to match nodes across more than two layers: local consistency for matching nodes across just two layers, and global consistency for dealing with the cases involving more than two layers. In Refs. [30], [31], the authors studied ways of predicting interlayer links in the absence of a priori interlayer links using the global network information.

Appendix B
Table of Symbols and Notations

Table I displays the main symbols and notations used in this paper. We follow the common symbolic conventions, wherein bold uppercase letters denote matrices, bold lowercase letters denote column vectors, and lowercase letters denote scalars.

Appendix C
Example of Different Steps of MulCEV

The example of different steps of MulCEV are as shown in Figs. 1, 2, 3 and 4.

Appendix D
Details of Optimization

To reduce the time complexity, we optimized the calculation of the degrees of vector consistency and distance consistency.

A. Optimization of Vector Consistency Calculation

The degree of vector consistency for each unmatched node pair can be calculated using Eq. (10) of the paper. However, many calculations are repeated, such as that of $||\mathbf{u}_b^{\beta}||$. We propose an approach based on a matrix operation to reduce the computational time complexity.

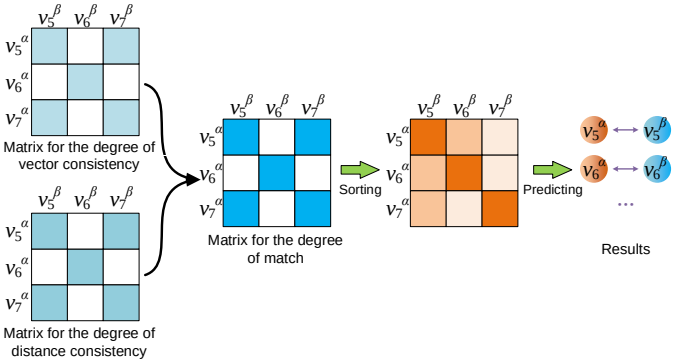


Fig. 4: Example of making prediction. By associating the two types of consistency, the matrix for the degree of match can be obtained. Then, we sort the matrix by row. In the third subfigure, the brighter the color, the greater the degree of match. Finally, the prediction results can be provided by the sorted list.

For all unmatched nodes, denoting $\boldsymbol{B}^{\alpha} = [\mathbf{u}_1^{\alpha}, \mathbf{u}_2^{\alpha}, \ldots, \mathbf{u}_{n^{\alpha}-n}^{\alpha}]$, $\boldsymbol{B}^{\beta} = [\mathbf{u}_1^{\beta}, \mathbf{u}_2^{\beta}, \ldots, \mathbf{u}_{n^{\beta}-n}^{\beta}]$, $\phi(\boldsymbol{B}^{\alpha}) = [\phi(\mathbf{u}_1^{\alpha}), \phi(\mathbf{u}_2^{\alpha}), \ldots, \phi(\mathbf{u}_{n^{\alpha}-n}^{\alpha})]$, $\boldsymbol{b}^{\alpha} = [||\mathbf{u}_1^{\alpha}||, ||\mathbf{u}_2^{\alpha}||, \ldots, ||\mathbf{u}_{n^{\alpha}-n}^{\alpha}||]^{\mathrm{T}}$, $\boldsymbol{b}^{\beta} = [||\mathbf{u}_1^{\beta}||, ||\mathbf{u}_2^{\beta}||, \ldots, ||\mathbf{u}_{n^{\beta}-n}^{\beta}||]^{\mathrm{T}}$, and $\boldsymbol{\phi}(b^{\alpha}) = [||\phi(\mathbf{u}_1^{\alpha})||, ||\phi(\mathbf{u}_2^{\alpha})||, \ldots, ||\phi(\mathbf{u}_{n^{\alpha}-n}^{\alpha})||]^{\mathrm{T}}$, the degree of vector consistency for all unmatched node pairs can be expressed as

$$\mathbf{P} = \frac{\phi(\mathbf{B}^{\alpha})^{\mathrm{T}} \cdot \mathbf{B}^{\beta}}{\phi(\mathbf{b}^{\alpha})^{\mathrm{T}} \cdot \mathbf{b}^{\beta}}. \quad (1)$$

B. Optimization of Distance Consistency Calculation

If h_{ai-bj} is denoted by $\exp(-(s_{ai}^{\alpha} \cdot |s_{ai}^{\alpha} - s_{bj}^{\beta}| \cdot s_{bj}^{\beta}))$, it is clear that if interlayer node pair $(v_i^{\alpha}, v_j^{\beta})$ is the CMN of

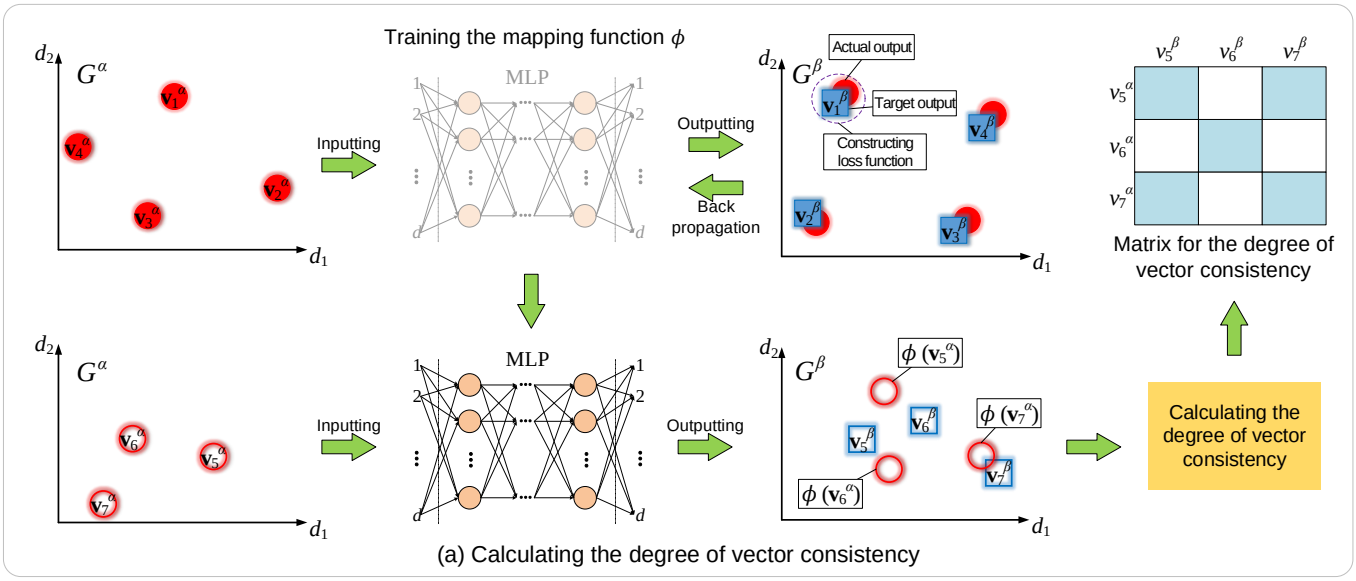


Fig. 3: Example of calculating degree of match. (a) Calculating the degree of vector consistency: The embedding vectors of the matched nodes in layer α, i.e., $\mathbf{v}_1^\alpha$, $\mathbf{v}_2^\alpha$, $\mathbf{v}_3^\alpha$, and $\mathbf{v}_4^\alpha$, are specified as inputs while the embedding vectors of the matched nodes in layer β, i.e., $\mathbf{v}_1^\beta$, $\mathbf{v}_2^\beta$, $\mathbf{v}_3^\beta$, and $\mathbf{v}_4^\beta$, are specified as the target outputs for the MLP to train the mapping function. Then, the embedding vectors of the unmatched nodes in layer α, i.e., $\mathbf{v}_5^\alpha$, $\mathbf{v}_6^\alpha$, and $\mathbf{v}_7^\alpha$, are applied as inputs to the trained MLP to obtain their mapped vectors in layer β, i.e., $\phi(\mathbf{v}_5^\alpha)$, $\phi(\mathbf{v}_6^\alpha)$, and $\phi(\mathbf{v}_7^\alpha)$. Thereafter, the cosine similarities between the mapped vectors of the unmatched nodes in layer α and the embedding vectors of the unmatched nodes in layer β are calculated to obtain the matrix for the degree of vector consistency. (b) Calculating the degree of distance consistency: The Euclidean distances between the unmatched nodes, i.e., v_5^α, v_6^α, or v_7^α, in layer α and their matched neighbors are computed; similar calculations are performed for the unmatched nodes in layer β. Then, the differences between pairs of Euclidean distances formed by the unmatched nodes across different layers and their CMNs are calculated. Finally, these differences and the numbers of CMNs between the unmatched nodes across the different layers are combined to obtain the matrix for the degree of distance consistency.

unmatched node pair (u_a^α, u_b^β), e_{ai}^α and e_{bj}^β will be equal to 1; thus, $e_{ai}^\alpha \cdot h_{ai-bj} \cdot e_{bj}^\beta = h_{ai-bj}$. In contrast, if interlayer node pair (v_i^α, v_j^β) is not the CMN of unmatched node pair (u_a^α, u_b^β), e_{ai}^α or e_{bj}^β will be equal to 0; thus, $e_{ai}^\alpha \cdot h_{ai-bj} \cdot e_{bj}^\beta = 0$. Therefore, Eq. (11) can be rewritten as

$$q(u_a^\alpha, u_b^\beta) = \sum_{\forall (v_i^\alpha, v_j^\beta) \in \Phi} e_{ai}^\alpha \cdot h_{ai-bj} \cdot e_{bj}^\beta. \tag{2}$$

If node v_i^α is an a priori interlayer node in layer α, a counterpart node must exist in layer β, and vice versa. Based on this, we can make the a priori interlayer nodes uniform, as follows: $(v_1^\alpha, v_1^\beta), \ldots, (v_i^\alpha, v_i^\beta), \ldots, (v_n^\alpha, v_n^\beta)$. Therefore, Eq. (2) can be replaced with

$$q(u_a^\alpha, u_b^\beta) = \sum_{i=1}^{n} e_{ai}^\alpha \cdot h_{ai-bj} \cdot e_{bi}^\beta. \tag{3}$$

Using the vector form, Eq. (3) can be replaced by

$$q(u_a^\alpha, u_b^\beta) = [e_{a1}^\alpha, \ldots, e_{ai}^\alpha, \ldots, e_{an}^\alpha] \cdot \begin{bmatrix} h_{a1-b1} \cdot e_{b1}^\beta \\ \vdots \\ h_{ai-bi} \cdot e_{bi}^\beta \\ \vdots \\ h_{an-bn} \cdot e_{bn}^\beta \end{bmatrix}. \tag{4}$$

We can use the Hadamard product to rewrite $[h_{a1-b1} \cdot e_{b1}^\beta, \ldots, h_{ai-bi} \cdot e_{bi}^\beta, \ldots, h_{an-bn} \cdot e_{bn}^\beta]^{\mathrm{T}}$ as

TABLE II: Statistics of Real-world Datasets. $|V|$ and $|E|$ are the number of nodes and intralayer links respectively. k_{max} is the maximum degree, $\langle k\rangle$ is the average degree, r is the degree-degree correlation, c is the clustering coefficient, H is the degree heterogeneity, as $H = \langle k^2\rangle//\langle k\rangle^2$, and $|E^{\alpha\beta}|$ is the number of interlayer links.

Network	$\|V\|$	$\|E\|$	k_{max}	$\langle k\rangle$	r	c	H	$\|E^{\alpha\beta}\|$
Foursquare	5,313	76,972	552	20.42	-0.193	0.23	3.446	3,148
Twitter	5,120	164,920	1725	51.01	-0.214	0.30	4.489	
DBLP_DataMining	11,526	47,326	117	36.68	0.110	0.85	2.176	1,295
DBLP_MachineLearning	12,311	43,948	552	20.42	-0.193	0.23	3.446	
Higgs_FS	4,288	122,826	1365	57.29	-0.140	0.27	2.943	3,760
Higgs_MT	3,777	13,413	1072	7.10	-0.095	0.22	9.619	
Higgs_FS	4,184	101,618	1086	48.57	-0.106	0.27	2.840	3,219
Higgs_RT	3,238	13,571	626	8.38	-0.090	0.09	5.653	

$[h_{a1-b1},\ldots,h_{ai-bi},\ldots,h_{an-bn}]^{\mathrm{T}} \circ [e^{\beta}_{b1},\ldots,e^{\beta}_{bi},\ldots,e^{\beta}_{bn}]^{\mathrm{T}}$. By denoting $\mathbf{h}_{ab} = [h_{a1-b1},\ldots,h_{ai-bi},\ldots,h_{an-bn}]^{\mathrm{T}}$, $\mathbf{e}^{\alpha}_{a} = [e^{\alpha}_{a1},\ldots,e^{\alpha}_{ai},\ldots,e^{\alpha}_{an}]^{\mathrm{T}}$, $\mathbf{e}^{\beta}_{b} = [e^{\beta}_{b1},\ldots,e^{\beta}_{bi},\ldots,e^{\beta}_{bn}]^{\mathrm{T}}$, Eq. (4) can be rewritten as

$$q(u^{\alpha}_{a},u^{\beta}_{b}) = (\mathbf{e}^{\alpha}_{a})^{\mathrm{T}}\cdot(\mathbf{h}_{ab}\circ\mathbf{e}^{\beta}_{b}). \tag{5}$$

By using Eq. (5), the degree of distance consistency for unmatched node pair $(u^{\alpha}_{a},u^{\beta}_{b})$ can be represented in vector operation form. Then, if we want to obtain the degree of distance consistency between node u^{α}_{a} and all the unmatched nodes in layer β, we can express Eq. (5) in matrix operation form. In a similar manner, we denote $\mathbf{H} = [\mathbf{h}_{a1},\ldots,\mathbf{h}_{ab},\ldots,\mathbf{h}_{a(n^{\beta}-n)}]$, $\mathbf{E}^{\beta} = [\mathbf{e}^{\beta}_{1},\ldots,\mathbf{e}^{\beta}_{b},\ldots,\mathbf{e}^{\beta}_{n^{\beta}-n}]$. The degree of distance consistency between node u^{α}_{a} and all the unmatched nodes in layer β can be calculated as follows:

$$\mathbf{q}^{\alpha}_{a} = ((\mathbf{e}^{\alpha}_{a})^{\mathrm{T}}\cdot(\mathbf{H}\circ\mathbf{E}^{\beta}))^{\mathrm{T}}. \tag{6}$$

By denoting $\mathbf{s}^{\alpha}_{a} = [s^{\alpha}_{a1},\ldots,s^{\alpha}_{ai},\ldots,s^{\alpha}_{an}]$, $\mathbf{s}^{\beta}_{b} = [s^{\beta}_{b1},\ldots,s^{\beta}_{bi},\ldots,s^{\beta}_{bn}]^{\mathrm{T}}$, $\mathbf{S}^{\beta} = [\mathbf{s}^{\beta}_{1},\ldots,\mathbf{s}^{\beta}_{b},\ldots,\mathbf{s}^{\beta}_{n^{\beta}-n}]^{\mathrm{T}}$, matrix $\mathbf{H}$ can be calculated as

$$\mathbf{H} = \exp\{(\mathbf{i}\cdot\mathbf{s}^{\alpha}_{a})\circ|\mathbf{i}\cdot\mathbf{s}^{\alpha}_{a} - \mathbf{S}^{\beta}|\circ\mathbf{S}^{\beta}\}, \tag{7}$$

where $\mathbf{i}$ is a column vector with $n^{\beta}-n$ elements. The value of each element in vector $\mathbf{i}$ is 1.

By joining the vector for the degree of distance consistency for all unmatched nodes in layer α, we can obtain the matrix for the degree of distance consistency for all unmatched node pairs, which can be represented as $\mathbf{Q} = [\mathbf{q}^{\alpha}_{1},\ldots,\mathbf{q}^{\alpha}_{a},\ldots,\mathbf{q}^{\alpha}_{n^{\alpha}-n}]^{\mathrm{T}}$.

Finally, the matrix for the degree of match for all unmatched node pairs can be obtained by

$$\mathbf{R} = \delta\cdot\mathbf{P} + (1-\delta)\cdot\mathbf{Q}. \tag{8}$$

The interlayer link prediction results are obtained by ranking each row or column of $\mathbf{R}$ in reverse order according to the degree of match.

APPENDIX E
TIME COMPLEXITY ANALYSIS

In the cross-layer extension stage, we search the intralayer links from the adjacency matrix comprising the matched nodes such that the time complexity of this stage is $O(n^2)$. In the network embedding stage, the time complexity for weight calculation is $O(|\Phi|\langle k\rangle^2)$, where $|\Phi|$ is the number of a priori intralayer links and $\langle k\rangle$ is the average degree of the nodes. Meanwhile, the time complexity for network embedding by LINE is $O(d|\Phi|\iota)$, where ι is the number of negative samples [32]. Therefore, the total time complexity in this step is $O(|\Phi|(d\iota + \langle k\rangle^2))$. In the degree of match calculation stage, the time complexity for training the mapping function, MLP, is $O(kdn)$ [33]. The time complexity of for calculating the vector consistency for all unmatched node pairs is $O(d(n^{\alpha}-n)(n^{\alpha}-n)(n^{\beta}-n))$ and that for calculating the distance consistency for all unmatched node pairs is $O(dn(n^{\alpha}-n)(n^{\beta}-n))$. We use matrix multiplication to optimize the process of calculating the degree of match for all the unmatched node pairs. After optimization, the time complexity for calculating the two types ofconsistency are $O(d(n^{\alpha}-n)(n^{\alpha}-n)(n^{\beta}-n)/\varsigma)$ and $O(dn(n^{\alpha}-n)(n^{\beta}-n)/\varsigma)$, respectively, where ς is the number of computational nodes [34]. Suppose the number of unmatched nodes in each layer is n_u, i.e., $n_u = (n^{\alpha}-n) = (n^{\beta}-n)$; then, the total time complexity for calculating the degree of match is $O(kdn + n_u^3 d/\varsigma + nn_u^2 d/\varsigma)$. Lastly, the time complexity for predicting the interlayer links is $O(Nn_u^2)$, where N is the size of top-N list.

APPENDIX F
DETAILS OF EXPERIMENTAL CONFIGURATIONS

A. Datasets

To evaluate the performance of our proposed framework and baseline methods, we used three synthetic and four real-world multiplex network datasets in our experiments. The synthetic networks are Erdős-Rényi [35] (ER) random networks, Watts-Strogatz [36] (WS) small-world networks, and Barabási-Albert [37] (BA) networks. We leveraged method proposed in Ref. [24] to generate multiplex networks by these three synthetic networks. We set the network size to 2000 and the percentage to remaining nodes to 0.5. The real-world datasets are as follows (cf. Table II):

- **Foursquare–Twitter (FT)**: This dataset was collected from Foursquare and Twitter by Zhang et al. [38]. The ground truth for this dataset is provided in Foursquare's profiles, and the nodes of the two social networks are partially aligned.
- **DBLP_DataMining-DBLP_MachineLearning (DBLP)**: This dataset was collected from the Citation Network Dataset[3] [39] and processed by Liu et al. [40].

[3]https://www.aminer.cn/citation

It is a co-authored multiplex network, one layer of which consists of researchers who published articles in journals or conference proceedings related to data mining, and the other layer containing researchers who published articles in journals or conference proceedings related to machine learning. The ground truth was obtained by collecting the authors who published articles in both fields.

- **Higgs_Friendships-Higgs_Mention (Higgs-FSMT)**: The Higgs dataset is collected from Twitter by Domenico et al. [41] which focuses on the spreading processes of the messages on Twitter during and after the discovery of a new particle with the features of the Higgs boson. We choose the friendships (FS) and mention (MT) networks to construct the multiplex network. To facilitate processing, we only reserve nodes with degrees greater than five in each network.
- **Higgs_Friendships-Higgs_Retweet (Higgs-FSRT)**: We choose the FS and retweet (RT) networks of Higgs to construct the multiplex network.

B. Comparison Methods

We used several state-of-the-arts as baselines, which are as follows.

- **DeepLink**: DeepLink is a semi-supervised learning algorithm that leverages traditional random walks to generate social sequences for the network embedding and utilizes the duality of mapping to improve the prediction performance.
- **IONE**: Input–output network embedding (IONE) projects multiple social networks into a common embedded space and matches same-user accounts by calculating the cosine similarity between the vectors of two nodes. In IONE, it represents each account by three vectors: a node vector, an input context vector, and an output context vector.
- **ONE**: This method is a simplified version of IONE. In this method, an account is represented by two vectors: a node vector and an output context vector.
- **IONE-D**: This method is a refined version of IONE that explores the community structure of the SMNs and incorporates the structural diversity to characterize a set of interlayer links.
- **BootEA**: This is a bootstrapping approach that aligns the entities of different knowledge graphs based on network embedding. It iteratively labels potential entity pairs as training data to overcome the lack of a sufficiently large training set and leverages an editing method to reduce error accumulation during the iterations.
- **PALE**: This method projects each SMN into a unique low-dimensional space and represents nodes by low-dimensional vectors in a latent space. Then, it learns a cross-layer mapping function for predicting interlayer links.
- **MAH**: Manifold alignment on hypergraph (MAH) tries to map common users across SMNs based on the network embedding method. It adopts a hypergraph to model high-order relations of SMNs and represents nodes into a common latent space. It infers correspondence by comparing distances between the vectors of the unmatched nodes.
- **MAG**: Manifold alignment on traditional graphs (MAG) is a method that uses $w(u_i, u_j) = |R_{u_i} \cap R_{u_j}|/(|R_{u_i}| + R_{u_j})$ for the calculation of node-to-node pairwise weights to build a graph for each SMN. The method for obtaining the node ranking result is the same as that for MAH.
- **CRW**: Collective random walk (CRW) is a joint link fusion approach for predicting the intralayer links and interlayer links simultaneously; it transfers information relating to intralayer links from one layer to another.

C. The Other Experimental Configurations

We employed $Precision@N$ ($P@N$) [1], [42], F-measure ($F1$) [1], and MAP [1] as the metrics to evaluate the performance of all methods. $P@N$ is defined as

$$P@N = \sum_{i=1}^{m} \mathbb{1}_i\{success@N\}/m, \tag{9}$$

where $\mathbb{1}_i\{success@N\}$ indicates whether the correct interlayer link exists in the top-N list, and m represents the number of all unobserved interlayer links. It is noteworthy that $Precision@N$ is actually the same as $Recall@N$ and $F1@N$ in the field of interlayer link prediction because $Precision@N$ represents the true positive prediction rate.

MAP is used to evaluate the ranked performance of different methods and is defined as

$$MAP = (\sum_{i=1}^{n} \frac{1}{r_i})/m, \tag{10}$$

where r_i represents the rank of the ith unmatched interlayer link. The higher the values of $P@N$ and MAP, the better the performance of the method.

To test the performance, the set of all interlayer links was randomly divided into two parts for each experiment (i) a training set Φ, which was treated as the set of a priori interlayer links; and (ii) a test set Ψ, which was used for testing and can be considered a collection of the unmatched node pairs waiting for prediction. The ratio of the size of the training set to the size of the set of all interlayer links is called the training ratio, which we varied in some of the experiments. Our task was to uncover the interlayer links in the test set based on the information in the training set and each layer of the multiplex network. For the experiments with a training ratio of 90%, we adopted 10-fold cross-validation. For the other experiments, we conducted ten times with randomly divided training and testing sets and took the average values as the results.

APPENDIX G
DETAILS OF EFFECT OF CONTROL PARAMETER δ

In Eq. (12) of the paper, the parameter δ is leveraged to control the proportions of the vector consistency and distance consistency in the final degree of match. We studied the initialization strategy for δ and its effect on the predicted results through experiments. We set $30.0\%, 60.0\%$, and 90.0% of the interlayer links as the training set and the remaining

TABLE III: Performance of MulCEV on different δ.

Metric	Datasets	Training ratios	δ										
			0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
P@30	FT	0.3	0.4622	0.4822	0.4891	0.4974	0.5048	0.5153	0.5327	0.5509	0.5636	**0.5640**	0.5001
		0.6	0.5828	0.5884	0.5941	0.5982	0.6071	0.6169	0.6331	0.6534	0.6753	**0.6802**	0.6047
		0.9	0.6645	0.6738	0.6738	0.6801	0.6738	0.6957	0.7012	0.7031	0.7112	**0.7174**	0.7051
	DBLP	0.3	0.2407	0.2413	0.2416	0.2418	**0.2419**	0.2418	0.2366	0.2255	0.2143	0.2043	0.1726
		0.6	0.3782	0.3796	0.3796	0.3796	0.3796	**0.3809**	0.3649	0.3328	0.3087	0.2900	0.2272
		0.9	0.5152	0.5152	0.5152	0.5152	**0.5291**	0.5152	0.4943	0.4456	0.3759	0.2785	0.1601
	Higgs-FSMT	0.3	0.4310	0.4402	0.4431	0.4435	0.4442	0.4472	**0.4538**	0.4505	0.4417	0.4204	0.3621
		0.6	0.6351	0.6430	0.6476	0.6509	0.6555	**0.6691**	0.6686	0.6647	0.6364	0.6054	0.5092
		0.9	0.7909	0.7936	0.7989	0.8016	0.8043	**0.8170**	0.8097	0.7882	0.7614	0.7239	0.6434
	Higgs-FSRT	0.3	0.4996	0.5139	0.5175	0.5261	0.5306	**0.5496**	0.5459	0.5477	0.5387	0.5148	0.4573
		0.6	0.6924	0.7003	0.7019	0.7082	0.7098	**0.7261**	0.7224	0.7192	0.7177	0.6822	0.6065
		0.9	0.8317	0.8447	0.8479	0.8511	0.8511	0.8511	**0.8608**	**0.8608**	0.8544	0.8091	0.7314
	ER	0.3	**0.8994**	**0.8994**	**0.8994**	**0.8994**	**0.8994**	**0.8994**	**0.8994**	**0.8994**	0.8609	0.7633	0.6331
		0.6	**0.9797**	**0.9797**	**0.9797**	**0.9797**	**0.9797**	**0.9797**	**0.9797**	0.9746	0.9594	0.9086	0.7614
		0.9	**0.9943**	**0.9943**	**0.9943**	**0.9943**	**0.9943**	**0.9943**	**0.9943**	0.9924	0.9880	0.9627	0.8335
	WS	0.3	**0.9117**	**0.9117**	**0.9117**	**0.9117**	**0.9117**	**0.9117**	**0.9117**	0.9088	0.9031	0.886	0.8319
		0.6	**0.9853**	**0.9853**	**0.9853**	**0.9853**	**0.9853**	**0.9853**	**0.9853**	**0.9853**	**0.9853**	0.9706	0.9363
		0.9	**0.9980**	**0.9980**	**0.9980**	**0.9980**	**0.9980**	**0.9980**	**0.9980**	**0.9980**	**0.9980**	0.9899	0.9446
	BA	0.3	0.8049	0.8074	0.8071	0.8099	0.8113	**0.8125**	0.8116	0.8110	0.7948	0.7554	0.6720
		0.6	0.9269	0.9269	0.9259	0.9279	0.9279	**0.9294**	0.9284	0.9269	0.9141	0.8847	0.7958
		0.9	0.9681	0.9681	0.9681	0.9721	0.9723	**0.9781**	0.9681	0.9681	0.9620	0.9370	0.8567

TABLE IV: Supplementary experiments of different δ on ER and WS datasets.

Metric	Datasets	Training ratios	δ										
			0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
P@1	ER	0.3	0.4909	0.5188	0.5349	0.5577	0.5695	0.5725	**0.5761**	0.5746	0.5291	0.4110	0.1725
		0.6	0.8274	0.8376	0.8477	0.8528	0.8528	**0.8681**	0.8630	0.8630	0.8274	0.6954	0.2843
		0.9	0.9621	0.9688	0.9643	0.9680	0.9680	**0.9696**	0.9508	0.9231	0.9148	0.8582	0.3632
	WS	0.3	0.5499	0.5755	0.584	0.5954	0.6182	0.6282	**0.6296**	0.6239	0.6154	0.5299	0.2877
		0.6	0.8382	0.8382	0.8431	0.8676	0.8676	0.8873	**0.8971**	**0.8971**	0.8775	0.7892	0.4461
		0.9	0.9508	0.9512	0.9491	0.9484	0.9495	**0.9534**	0.9522	0.9529	0.9380	0.8891	0.5486

links as the test set. The value of δ was varied from 0 to 1 in steps of 0.1.

The $P@30$ of these experiments are displayed in Table III. From the table, we see that the values of $P@30$ initially exhibit an increasing trend and later decrease with additional increase in δ. This demonstrates that both the vector consistency and distance consistency positively affect interlayer link predictions. Maximum values in each line are obtained for different δ values; this may be attributed to the difference in network structural properties between these datasets. For some datasets, the mapping functions can be learned well; hence, a small δ achieves the best prediction results. For other datasets, the mapping functions are hard to learn; these maximum values are considered for larger δ. We observed that the maximum values were achieved for multiple columns in the ER and WS datasets. This renders it difficult to choose an appropriate value of δ. To overcome this problem, we further tested $P@1$ on the two datasets, as shown in Table IV. The results exhibit the same trends as the other datasets at $P@30$.

Considering that the prediction results of most datasets show best performance with $\delta = 0.5$, this value is recommended for new datasets. For more precise values, investigators may consider performing 10-fold cross validation using the a priori interlayer link set, which is divided into training and validation sets.

APPENDIX H
CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Rui Tang: Conceptualization, Methodology, Software, Validation, Investigation, Writing - original draft, Writing - review & editing. Xingshu Chen: Conceptualization, Methodology, Project administration, Resources, Writing - review & editing. Haizhou Wang: Conceptualization, Methodology, Project administration, Writing - review & editing. Zhenxiong Miao: Writing - review & editing, Validation, Investigation. Shuyu Jiang: Writing - review & editing, Supervision. Wei Wang: Supervision, Visualization, Investigation.

APPENDIX I
ACKNOWLEDGMENT

The authors want to thank Dr. Wenxian Wang and Mingyong Yin for their advice.

REFERENCES

[1] K. Shu, S. Wang, J. Tang, R. Zafarani, and H. Liu, "User identity linkage across online social networks: A review," ACM SIGKDD Explorations Newsletter, vol. 18, no. 2, pp. 5–17, Mar. 2017.

[2] D. Zhao, N. Zheng, M. Xu, X. Yang, and J. Xu, "An improved user identification method across social networks via tagging behaviors," in Proceedings of the 30th International Conference on Tools with Artificial Intelligence, Volos, Greece, Nov. 2018, pp. 616–622.

[3] R. Zafarani and H. Liu, "Connecting corresponding identities across communities," in Proceedings of the 3rd International Conference on Weblogs and Social Media, San Jose, California, USA, May 2009.

[4] D. Perito, C. Castelluccia, M. A. Kaafar, and P. Manils, "How unique and traceable are usernames?" in Proceedings of the 11st International Symposium on Privacy Enhancing Technologies Symposium, Waterloo, ON, Canada, July 2011, pp. 1–17.

[5] R. Zafarani and H. Liu, "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, Chicago, IL, USA, Aug. 2013, pp. 41–49.

[6] J. Liu, F. Zhang, X. Song, Y.-I. Song, C.-Y. Lin, and H.-W. Hon, "What's in a name? an unsupervised approach to link users across communities," in Proceedings of the 6th ACM International Conference on Web Search and Data Mining, Rome, Italy, Feb. 2013, pp. 495–504.

[7] Y. Li, Y. Peng, Z. Zhang, H. Yin, and Q. Xu, "Matching user accounts across social networks based on username and display name," World Wide Web, vol. 22, no. 3, pp. 1075–1097, Apr. 2019.

[8] F. Carmagnola and F. Cena, "User identification for cross-system personalisation," Information Sciences, vol. 179, no. 1-2, pp. 16–32, Jan. 2009.

[9] T. Iofciu, P. Fankhauser, F. Abel, and K. Bischoff, "Identifying users across social tagging systems," in Proceedings of the 5th International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 2011.

[10] F. Abel, E. Herder, G.-J. Houben, N. Henze, and D. Krause, "Cross-system user modeling and personalization on the social web," User Modeling and User-Adapted Interaction, vol. 23, no. 2-3, pp. 169–209, Nov. 2013.

[11] O. Goga, D. Perito, H. Lei, R. Teixeira, and R. Sommer, "Large-scale correlation of accounts across social networks," University of California at Berkeley, 2013.

[12] K. Cortis, S. Scerri, I. Rivera, and S. Handschuh, "An ontology-based technique for online profile resolution," in Proceedings of the 4th International Conference on Social Informatics, Kyoto, Japan, Nov. 2013, pp. 284–298.

[13] X. Mu, F. Zhu, E.-P. Lim, J. Xiao, J. Wang, and Z.-H. Zhou, "User identity linkage by latent user space modelling," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, Aug. 2016, pp. 1775–1784.

[14] S. Fu, G. Wang, S. Xia, and L. Liu, "Deep multi-granularity graph embedding for user identity linkage across social networks," Knowledge-Based Systems, vol. 193, p. 105301, Apr. 2020.

[15] C. Riederer, Y. Kim, A. Chaintreau, N. Korula, and S. Lattanzi, "Linking users across domains with location data: Theory and validation," in Proceedings of the 25th International Conference on World Wide Web, Montreal, Canada, Apr. 2016, pp. 707–719.

[16] W. Chen, H. Yin, W. Wang, L. Zhao, and X. Zhou, "Effective and efficient user account linkage across location based social networks," in Proceedings of the 34th IEEE International Conference on Data Engineering, Paris, France, Apr. 2018, pp. 1085–1096.

[17] J. Feng, M. Zhang, H. Wang, Z. Yang, C. Zhang, Y. Li, and D. Jin, "D-plink: User identity linkage via deep neural network from heterogeneous mobility data," in Proceedings of the 28th International Conference on World Wide Web, San Francisco, CA, USA, May 2019, pp. 459–469.

[18] R. Zheng, J. Li, H. Chen, and Z. Huang, "A framework for authorship identification of online messages: Writing-style features and classification techniques," Journal of the American Society for Information Science and Technology, vol. 57, no. 3, pp. 378–393, Feb. 2006.

[19] A. Narayanan, H. Paskov, N. Z. Gong, J. Bethencourt, E. Stefanov, E. C. R. Shin, and D. Song, "On the feasibility of internet-scale author identification," in Proceedings of the 33rd IEEE Symposium on Security and Privacy, San Francisco, California, USA, May 2012, pp. 300–314.

[20] O. Goga, H. Lei, S. H. K. Parthasarathi, G. Friedland, R. Sommer, and R. Teixeira, "Exploiting innocuous activity for correlating users across sites," in Proceedings of the 22nd international conference on World Wide Web, Rio de Janeiro, Brazil, May 2013, pp. 447–458.

[21] X. Zhou, X. Liang, H. Zhang, and Y. Ma, "Cross-platform identification of anonymous identical users in multiple social media networks," IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 2, pp. 411–424, Feb. 2016.

[22] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," in Proceedings of the 30th IEEE Symposium on Security and Privacy, Oakland, California, USA, May 2009, pp. 173–187.

[23] N. Korula and S. Lattanzi, "An efficient reconciliation algorithm for social networks," Proceedings of the VLDB Endowment, vol. 7, no. 5, pp. 377–388, Jan. 2014.

[24] R. Tang, S. Jiang, X. Chen, H. Wang, W. Wang, and W. Wang, "Interlayer link prediction in multiplex social networks: an iterative degree penalty algorithm," Knowledge-Based Systems, vol. 194, p. 105598, Apr. 2020.

[25] Y. Ren, C. C. Aggarwal, and J. Zhang, "Meta diagram based active social networks alignment," in Proceedings of the 35th IEEE International Conference on Data Engineering, Macau, China, Apr. 2019, pp. 1690–1693.

[26] J. Zhang and S. Y. Philip, "Multiple anonymized social networks alignment," in Proceddings of the 15th IEEE International Conference on Data Mining, Atlantic City, NJ, USA, Nov. 2015, pp. 599–608.

[27] Y. Zhu, L. Qin, J. X. Yu, Y. Ke, and X. Lin, "High efficiency and quality: large graphs matching," The International Journal on Very Large Data Bases, vol. 22, no. 3, pp. 345–368, Sept. 2012.

[28] R. Zafarani, L. Tang, and H. Liu, "User identification across social media," ACM Transactions on Knowledge Discovery from Data, vol. 10, no. 2, pp. 1–30, Oct. 2015.

[29] Y. Zhang, J. Tang, Z. Yang, J. Pei, and P. S. Yu, "Cosnet: Connecting heterogeneous social networks with local and global consistency," in Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, Aug. 2015, pp. 1485–1494.

[30] S. Zhang and H. Tong, "Final: Fast attributed network alignment," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, Aug. 2016, pp. 1345–1354.

[31] S. Zhang, H. Tong, R. Maciejewski, and T. Eliassi-Rad, "Multilevel network alignment," in Proceedings of the 28nd International Conference on World Wide Web, San Francisco, CA, USA, May 2019, pp. 2344–2354.

[32] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, May 2015, pp. 1067–1077.

[33] T. Man, H. Shen, S. Liu, X. Jin, and X. Cheng, "Predict anchor links across social networks via an embedding approach," in Proceedings of the 25th International Joint Conference on Artificial Intelligence, vol. 16, New York, USA, July 2016, pp. 1823–1829.

[34] J. S. Lee, S.-Y. Park, P. B. Berra, and S. Ranka, "I/o and memory-efficient matrix multiplication with user-controllable parallel i/o," in Proceedings 1997 International Conference on Parallel and Distributed Systems, Seoul, South Korea, Dec. 1997, pp. 59–66.

[35] P. Erdős and A. Rényi, "On the evolution of random graphs," Publications of the Mathematical Institute of the Hungarian Academy of Science, vol. 5, no. 1, pp. 17–60, 1960.

[36] D. J. Watts and S. H. Strogatz, "Collective dynamics of small-worldnetworks," nature, vol. 393, no. 6684, pp. 440–442, 1998.

[37] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," Science, vol. 286, no. 5439, pp. 509–512, Oct. 1999.

[38] J. Zhang and S. Y. Philip, "Integrated anchor and social link predictions across social networks," in Proceedings of the 24th International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, July 2015, pp. 2215–2132.

[39] J. Tang, J. Zhang, L. Yao, and J. Li, "Extraction and mining of an academic social network," in Proceedings of the 17th International Conference on World Wide Web, Beijing, China, Apr. 2008, pp. 1193–1194.

[40] L. Liu, X. Li, W. Cheung, and L. Liao, "Structural representation learning for user alignment across social networks," IEEE Transactions on Knowledge and Data Engineering, 2019.

[41] M. De Domenico, A. Lima, P. Mougel, and M. Musolesi, "The anatomy of a scientific rumor," Scientific reports, vol. 3, no. 1, pp. 1–9, 2013.

[42] L. Liu, W. K. Cheung, X. Li, and L. Liao, "Aligning users across social networks using network embedding." in Proceedings of the 25th International Joint Conference on Artificial Intelligence, New York, USA, July 2016, pp. 1774–1780.