Fig. S1. An example of the link prediction problem. The common process of link prediction is that a set of existing links is removed randomly from the original network, which is marked as the positive testing set \( L^P \) (the red dash lines in the training network). As the control group, a random set of node pairs that are not connected in the original network is selected as the negative testing set \( L^N \) (the blue dash lines in the training network). An index considers the topology based on the rest of the links \( L^T \) (the black solid lines in the training network) and assigns a value to each node pair in \( L^P \) and \( L^N \). In unsupervised prediction, the index values are directly used as the score of samples in \( L^P \) and \( L^N \). Assume that according to the index value we have \( S_{16} = 0.3, S_{34} = 0.58, S_{13} = 0.2 \), and \( S_{46} = 0.58 \). The prediction quality is measured by how samples in \( L^P \) are ranked ahead of those in \( L^N \). When using AUC to measure the prediction performance, we usually apply the random sampling approach. In each comparison, we randomly draw a node pair from \( L^P \) and a node pair from \( L^N \), and compare their scores. Suppose 3 random comparisons are made. Node pairs 1-6 and 1-3, node pairs 1-6 and 4-6, and node pairs 3-4 and 4-6 are selected in each comparison. We have one case where node pair from \( L^P \) outscores that from \( L^N \), and one case where node pairs from \( L^P \) and \( L^N \) have an equal score. According to Eq. (1) of the main text, the AUC can be estimated as \( \frac{0+0.5+0}{3} = 0.5 \). When using precision to measure the prediction performance, we rank node pairs according to their scores in descending order. In the example shown, the rank is 3-4, 4-6, 1-6, 1-3. If we select the hyper-parameter \( L_k = 2 \), the top-two node pairs (3-4 and 4-6) are considered. As node pair 3-4 is the true positive sample whereas node pair 4-6 is not, the precision is 0.5.
Fig. S2. The capability of the topological features measured by AUC in the unsupervised prediction. Eqs. (2) and (3) in the main text suggest that different indexes have different prediction performances, but all indexes associated with one topological feature share the same $AUC_{\text{upper}}$ and $AUC_{\text{lower}}$. This is confirmed by 21 indexes related to 4 topological features: common neighbor (a), path (b), path of length three (c), and heterogeneity (d). For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^I$ and $L^N$ sets. In the figure, we use the average value of 200 samples.
Fig. S3. The maximum capability of the topological features measured by AUC in the supervised prediction. Eq. (4) in the main text suggests that $AUC'_{\text{upper}}$ sets the upper bound of the supervised prediction. This is confirmed by 21 indexes related to 4 topological features: common neighbor (a), path (b), path of length three (c), and heterogeneity (d). For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^P$ and $L^N$ sets. In the figure, we choose the highest AUC from 200 samples as the performance of an index.
Almost all data points are above the line $y = x$, indicating that the supervised prediction in general gives rise to a higher AUC value compared with the unsupervised prediction in the same network based on the same index. In unsupervised prediction, the AUC value can be less than 0.5 (Fig. S2), suggesting that the feature is more prominent in negative samples. A simple fix in this circumstance is to consider the feature as the “negative feature”. When the AUC is below 0.5, the predictor will consider that a smaller index value corresponds to a higher probability that two nodes are truly connected. After this modification is applied, predictions with AUC < 0.5 will be lifted. The red solid line is $y = 1 - x$, corresponding to the performance after considering the “negative feature”. All data points are above $y = 1 - x$. Hence, the improvement by the supervised approach is not merely taking advantage of the “negative feature”. We use 21 indexes related to 4 topological features: common neighbor (a), path (b), path of length three (c), and heterogeneity (d). For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^P$ and $L^N$ sets. In the figure, we use the average value of 200 samples for unsupervised results. For supervised results, we choose the highest AUC from 200 samples as the performance of an index.
Fig. S5. The structural characteristics of the common neighbor feature. (a) In the analytical expression of $p'_2$ (Eq. (36)), the denominator is dominated by $N^2$. Hence, $p'_2$ is expected to decay fast with the network size $N$. For a sufficiently large network, $p'_2$ is expected to be zero. This is generally held in the empirical analyses of 550 networks. But exceptions are also found. Some large networks have a relatively large $p'_2$ value. Such networks have many “leaves” structures, as illustrated in (b). The leaves will make the number of open triangles proportional to $N^2$. Hence, $p'_2$ will not vanish in such networks. The network in (b) is “5886685ba411221d0e7c677e” in the data set.
Table S1. The $p_1 \times p_2$ value of a topological feature. The $p_1 \times p_2$ (mean ± standard deviation) of a topological feature on the testing set ($L^P$ and $L^N$) applied to 550 empirical networks. For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^P$ and $L^N$ sets. For the $p_1$ and $p_2$ values of each network, we use the average value of 200 samples.

| Features          | $p_1 \times p_2$ |
|-------------------|------------------|
| Common Neighbor   | 0.057 ± 0.152    |
| Path of Length Three | 0.155 ± 0.241    |
| Path              | 0.270 ± 0.315    |
| Heterogeneity     | 0.695 ± 0.126    |
S1. The 21 indexes used in this study and the classification of these indexes

In this study, we select 21 indexes associated with 4 topological features to validate our quantitative framework presented in the main text. Here, we describe in detail the 21 indexes and how they are classified into 4 families.

A. A family of indexes based on common neighbor. We consider 11 indexes that gauge the common neighbor feature.

1) Common Neighbor Index (CN)

The CN directly counts the number of common neighbors two nodes share (1). It is defined as

\[ S_{CN}^{ab} = |n(a) \cap n(b)|, \]  

where \( n(a) \) denotes the set over all neighbors of node \( a \).

2) Adamic-Adar Index (AA)

Adamic and Adar propose the AA index that computes the similarity between two web pages (2). The AA emphasizes less-connected common neighbors. It is defined as

\[ S_{AA}^{ab} = \sum_{c \in n(a) \cap n(b)} \frac{1}{\log k(c)}, \]  

where \( k(c) = |n(c)| \) is the degree of node \( c \).

3) Resource Allocation Index (RA)

Motivated by the physical process of resource allocation, Zhou et al. propose the RA index that puts penalties to large degree nodes (3). The RA is defined as

\[ S_{RA}^{ab} = \sum_{c \in n(a) \cap n(b)} \frac{1}{k(c)}. \]  

4) Salton Index (Salton)

The Salton index is also called cosine similarity (4), which is defined as

\[ S_{Salton}^{ab} = \frac{|n(a) \cap n(b)|}{\sqrt{k(a) \times k(b)}}. \]  

5) Sørensen Index (SI)

The SI is usually used in ecological science (5), which is defined as

\[ S_{SI}^{ab} = \frac{2 \times |n(a) \cap n(b)|}{k(a) + k(b)}. \]  

6) Hub Promoted Index (HPI)

The HPI aims to measure the degree of topological overlap between two nodes in metabolic networks (6), which is defined as

\[ S_{HPI}^{ab} = \frac{|n(a) \cap n(b)|}{\min(k(a), k(b))}. \]  

7) Hub Depressed Index (HDI)

The HDI is similar to the HPI. The difference is that HDI emphasizes the role of nodes with large degrees (3). It is defined as

\[ S_{HDI}^{ab} = \frac{|n(a) \cap n(b)|}{\max(k(a), k(b))}. \]  

8) Leicht-Holme-Newman Index (LHN-I)

Leicht et al. propose the LHN-I, which assigns a high value to the node pair with many common neighbors (7). It is defined as

\[ S_{LHN-I}^{ab} = \frac{|n(a) \cap n(b)|}{k(a) \times k(b)}. \]  

9) Jaccard Index (Jaccard)

The Jaccard directly normalizes the number of common neighbors (1). It is defined as

\[ S_{Jaccard}^{ab} = \frac{|n(a) \cap n(b)|}{|n(a) \cup n(b)|}. \]  

10) CH2-L2 Index (CH2-L2)

Muscoloni et al. adopt the Cannistraci-Hebb network automaton model to extend the local community paradigm to paths of length 2 (8–10). Here, we denote \( P_{ab} \) by the set of nodes on all paths of length 2 that connect nodes \( a \) and \( b \). If node \( u \) is the intermediate node on a path of length 2 linking nodes \( a \) and \( b \), the CH2-L2 is defined as

\[ S_{CH2-L2}^{ab} = \sum_u \frac{l_{uab}(1 + i(u))}{1 + e(u)}. \]
where \(i(u)\) is the number of internal links between node \(u\) and nodes except nodes \(a\) and \(b\) in \(P_{ab}\). \(e(u)\) is the number of external links between node \(u\) and nodes not in \(P_{ab}\). The \(l_{au} = 1\) if node \(a\) and node \(u\) have a link, \(l_{au} = 0\) otherwise.

(11) CH3-L2 Index (CH3-L2)

According to the Cannistraci-Hebb rule, Muscoloni et al. propose the CH3-L2 index implemented solely based on the minimization of \(e(u)\) (8–10). Similar to the CH2-L2, the CH3-L2 is defined as

\[
S_{ab}^{\text{CH3-L2}} = \sum_u \frac{l_{au}l_{ba}}{1 + e(u)}. \tag{11}
\]

**B. A family of indexes based on path.** We consider 5 indexes gauging the path feature. To reduce the computational complexity and to unify the path feature, we focus on path lengths less than or equal to 4.

1) Local Path Index (LP)

Motivated by the CN index, Lü et al. propose the LP index (11). A large number of studies have shown that the predictive ability of the LP index is stronger than that of the CN on many real networks (12–14). The definition of LP is

\[
S_{ab}^{\text{LP}} = A_{ab}^2 + \beta A_{ab}^3 + \beta^2 A_{ab}^4, \tag{12}
\]

where \(A^i_{ab}\) is the number of paths of length \(i\) that links node \(a\) and \(b\), and \(\beta\) controls the weight of paths with different lengths. In this study, we use \(\beta = 0.02\).

2) Katz Index (Katz)

The Katz can be regarded as an ensemble method that directly sums the number of all paths (15). It is defined as

\[
S_{ab}^{\text{Katz}} = \beta^2 A_{ab}^2 + \beta^3 A_{ab}^3 + \beta^4 A_{ab}^4, \tag{13}
\]

where \(\beta\) controls the weight of paths with different lengths. Katz is similar to LP. The difference between them is that Katz assigns exponentially decaying weights into long paths whereas the decaying weights in LP are slower. In this study, we use \(\beta = 0.02\).

3) FriendLink Index (FL)

The basic idea of the FL index is that two people who have more and shorter paths are more likely to become friends on social networks (16). It is defined as

\[
S_{ab}^{\text{FL}} = \sum_{i=2}^{l} \frac{1}{i-1} \frac{|\text{paths}^i_{ab}|}{\prod_{j=2}^{N-j}(N-j)}, \tag{14}
\]

where \(N\) is the number of nodes in a network, \(l\) is the length of the longest path between nodes \(a\) and \(b\). The \(|\text{paths}^i_{ab}|\) is the number of paths with length \(i\) between nodes \(a\) and \(b\). In this study, we control \(l = 4\).

4) Relation Strength Similarity Index (RSS)

The RSS proposed by Chen et al. is to measure the relative degree of similarity between two nodes (17). The relation strength is defined as

\[
R(ab) = \begin{cases} \sum_{\forall x \in n(a)} \alpha_{ax} & \text{if } a \text{ and } b \text{ are adjacent} \\ 0 & \text{otherwise}, \end{cases}
\]

where \(\alpha_{ax}\) is the weight between nodes \(a\) and \(x\), which can be any value. Here, the \(R(ab)\) is not symmetric, i.e. \(R(ab) \neq R(ba)\). To make arbitrary nodes available, Chen et al. also propose the generalized relation strength as

\[
R_{pi}^*(ac) = \prod_{k=1}^{K-1} R(b_kb_{k+1}),
\]

where the \(p_i\) is a set of paths between nodes \(a\) and \(c\). The \(p_i\) is formed by \(b_1, b_2, ..., b_K\) in which the \(b_1\) represents node \(a\) and the \(b_K\) represents node \(c\). To make the calculation tractable, Chen et al. control the path length less than \(r\), yielding

\[
R_{pi}^*(ac) = \begin{cases} \prod_{k=1}^{K} R(b_kb_{k+1}) & \text{if } K \leq r \\ 0 & \text{otherwise}. \end{cases}
\]

Taken together, if there are \(P_l\) paths with length shorter than \(r\) from node \(a\) to node \(b\), the RSS is defined as

\[
S_{ab}^{\text{RSS}} = \sum_{l=1}^{P_l} R_{pi}^*(ab). \tag{15}
\]

In this study, we control \(r = 4\).

5) Shortest Path Length Index (SPL)
Ran et al. show that the shorter the shortest path length between two unconnected nodes is, the more likely they are to form a link in the long-path networks (15, 18). The SPL is defined as

$$\bar{S}_{ab}^{SPL} = \frac{1}{d_{ab} - 1}, \quad [16]$$

where the $d_{ab}$ is the length of the shortest path between nodes $a$ and $b$. As the path length is limited to be not greater than 4, $d_{ab}$ in this study falls within the range [2, 4]. If there is no path within length 4 connecting node $a$ and $b$, $d_{ab} \to \infty$ which gives $\bar{S}_{ab}^{SPL} = 0$.

**C. A family of indexes based on path of length three.** Different from the path feature that includes connections with different lengths, the feature path of length three only considers connections with lengths equal to three. Therefore, the path of length three is different from the path feature. We consider 3 indexes that gauge the feature path of length three.

1. Paths of length three Index (L3)

Kovics et al. propose a degree-normalized index based on paths of length three (L3). L3 is found to have a remarkable advantage compared with the indexes based on common neighbors in predicting protein-protein interactions (19). The L3 index is defined as

$$S_{ab}^{L3} = \frac{\sum_{uv} l_{au}l_{uv}l_{vb}}{\sqrt{k(u)k(v)}}, \quad [17]$$

where nodes $u$ and $v$ are the intermediate nodes on a path of length 3 linking nodes $a$ and $b$. The $l_{au} = 1$ if node $a$ and node $u$ have a link, $l_{uv} = 0$ otherwise.

2. CH2-L3 Index (CH2-L3)

Motivated by the L3 (19), Muscoloni et al. adopt the Cannistraci-Hebb network automaton model to extend the local community paradigm to paths of length 3 (8–10). Denote $P_{ab}$ by the set of nodes on all paths of length 3 that connect nodes $a$ and $b$. The CH2-L3 is defined as

$$S_{ab}^{CH2-L3} = \frac{\sum_{uv} l_{au}l_{uv}l_{vb} \sqrt{(1 + \beta(u))(1 + \beta(v))}}{\sqrt{(1 + e(u))(1 + e(v))}}, \quad [18]$$

where $i(u)$ is the number of internal links between node $u$ and nodes except nodes $a$ and $b$ in $P_{ab}$, $e(u)$ is the number of external links between node $u$ and nodes not in $P_{ab}$.

3. CH3-L3 Index (CH3-L3)

According to the Cannistraci-Hebb rule, Muscoloni et al. also propose the CH3-L3 index that puts penalties to the $e(u)$ and $e(v)$ (8–10). Similar to the CH2-L3, the CH3-L3 is defined as

$$S_{ab}^{CH3-L3} = \frac{\sum_{uv} l_{au}l_{uv}l_{vb}}{\sqrt{(1 + e(u))(1 + e(v))}}. \quad [19]$$

**D. A family of indexes based on heterogeneity.** We consider 2 indexes gauging the heterogeneity feature.

1. Heterogeneity Index (HEI)

Shang et al. find that the indexes based on common neighbors fail to identify missing links in the tree-like networks (20). To solve this problem, they take advantage of network heterogeneity and propose the heterogeneity index (HEI). The HEI is defined as

$$S_{ab}^{HEI} = |k(a) - k(b)|^\beta, \quad [20]$$

where $\beta$ is a free heterogeneity exponent. In this work, we set $\beta = 0.02$.

2. Homogeneity Index (HOI)

Based on the assumption that network homogeneity plays a major role in homogeneous networks (20), Shang et al. define HOI as

$$S_{ab}^{HOI} = \frac{1}{|k(a) - k(b)|^\beta}. \quad [21]$$

In this work, we set $\beta = 0.02$.

**E. Index classification.** From the description of the 21 indexes, we can intuitively link the index with the associated topological feature. To further validate the index classification, we consider the following property.

*If two indexes are associated with the same topological feature, they will have the same $p_1$ and $p_2$ values for the given $L^P$ and $L^N$.*

$p_1$ is the percentage of samples in $L^P$ that hold the topological feature, and $p_2$ is the percentage of samples in $L^N$ that hold the topological feature. Technically, $p_1$ and $p_2$ can be measured by counting the number of samples whose index value is 0. From the definition of those indexes, it is obvious that for indexes in the family of the common neighbor feature, they all give $S_{ab} = 0$ for two nodes that do not share any common neighbor. For indexes in the family of the path feature, they all give $S_{ab} = 0$ for two nodes not connected by a path (within length 4). For indexes in the family of the path length three feature, they all give $S_{ab} = 0$ for two nodes not connected by a path of length 3. For indexes in the family of heterogeneity feature, they all give $S_{ab} = 0$ for two nodes with the same degree. We empirically calculate $p_1$ and $p_2$ values by each index and check if they are the same for indexes classified into the four categories. The numerical $p_1$ and $p_2$ values confirm the classification.
S2. The results associated with Preferential Attachment

The index Preferential Attachment (PA) is one of the early indexes proposed in the link prediction problem, which is related to the product of two node’s degree (15). Here we perform prediction results by PA in 550 different real networks. Fig. S6 shows that prediction results by PA in the unsupervised and supervised approaches still follow the capability predicted. For all networks, the prediction measured by precision should be below what Eq. (23) and Eq. (24) yield, respectively. This is also confirmed in Fig. S7. These show that our finding also works for this feature with one single index.

Fig. S6. The capability of the preferential attachment feature measured by AUC. Eqs. (2) and (3) in the main text suggest that different indexes have different prediction performances, but all indexes associated with one topological feature share the same $AUC_{upper}$ and $AUC_{lower}$. Here this is confirmed by the PA method related to the preferential attachment feature in (a) and (b). Eq. (4) in the main text suggests that $AUC'_{upper}$ sets the upper bound of the supervised prediction. This is also confirmed by the PA method related to the preferential attachment feature in (c).
Fig. S7. The maximum capability of the preferential attachment feature measured by precision. We choose the PA to measure the performance of the unsupervised and supervised prediction in all 550 networks. (a) For different choices of \( L_k \) \( (L_k = |L^p|/4, |L^p|/2, 3|L^p|/4, |L^p|) \), the measured precision is equal to or below Precision\(_{upper}\), supporting the claim that Precision\(_{upper}\) gives the maximum capability of a feature measured by precision in the unsupervised approach. (b) For different choices of \( L_k \) \( (L_k = |L^p|/4, |L^p|/2, 3|L^p|/4, |L^p|, 5|L^p|/4, 3|L^p|/2) \), the measured precision is equal to or below Precision\(_{upper}'\), supporting the theoretical results for the maximum capability of a feature measured by precision in the supervised approach.
S3. Two alternative experiment setups

To make the assessment more general, we here evaluate the other two alternative experiment setups. We call the first experiment setup sample1 in the following discussion. Sample1 uses balanced positive and negative samples. It differs from the experiment setup in the main text for the size of the training and testing set. In particular, sample1 randomly removes 20% of $L$ links as the missing links. 80% of the removed links (16% of $L$ links) are used as the positive set in the training and the rest of 20% of the removed links are used as the positive set in the testing. In both training and testing, randomly selected nonexistent links compose the negative set with the same size as the positive set. The second experiment setup is called sample2 in the following discussion. Sample2 uses imbalanced positive and negative samples. Specifically, sample2 randomly removes 20% of $L$ links as the missing links. In the training step, the positive set is composed of 80% of the removed links (16% of $L$ links), and the negative set is composed of 80% of all nonexistent links. In the testing step, the positive set is composed of the rest 20% of the removed links, and the negative set is composed of the rest 20% of all nonexistent links.

To make sure our results are not affected by different setups, we perform a robustness check by repeating the measurement in the main text using sample1 (Fig. S8 and Fig. S9) and sample2 (Fig. S10 and Fig. S11). The same capability applies to different setups, supporting the universality of the conclusion drawn.

![Graphs showing the AUC values for different indexes and topological features](image)

Fig. S8. Unsupervised prediction measured by AUC based on balanced positive and negative samples by sample1. Eqs. (2) and (3) in the main text suggest that different indexes have different prediction performances, but all indexes associated with one topological feature share the same AUC upper and AUC lower. This is confirmed by 21 indexes related to 4 topological features: common neighbor (a), path (b), path of length three (c), and heterogeneity (d). For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^P$ and $L^N$ sets. In the figure, we use the average value of 200 samples. The same quantitative analysis in the left panel of Fig. S2 is repeated.
Fig. S9. Supervised prediction measured by AUC based on balanced positive and negative samples by *sample*. Eq. (4) in the main text suggests that $\text{AUC}_{\text{upper}}'$ sets the upper bound of the supervised prediction. This is confirmed by 21 indexes related to 4 topological features: common neighbor (a), path (b), path of length three (c), and heterogeneity (d). For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^P$ and $L^N$ sets. In the figure, we choose the highest AUC from 200 samples as the performance of an index. The same quantitative analysis in the Fig. S3 is repeated.
Fig. S10. Unsupervised prediction measured by AUC based on imbalanced positive and negative samples by sample2. Eqs. (2) and (3)  in the main text suggest that different indexes have different prediction performances, but all indexes associated with one topological feature share the same AUC_{upper} and AUC_{lower}. This is confirmed by 21 indexes related to 4 topological features: common neighbor (a), path (b), path of length three (c), and heterogeneity (d). For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^p$ and $L^\infty$ sets. In the figure, we use the average value of 200 samples. The same quantitative analysis in the left panel of Fig. S2 is repeated.
Fig. S11. Supervised prediction measured by AUC based on imbalanced positive and negative samples. Eq. (4) in the main text suggests that $\text{AUC}'_{\text{upper}}$ sets the upper bound of the supervised prediction. This is confirmed by 21 indexes related to 4 topological features: common neighbor (a), path (b), path of length three (c), and heterogeneity (d). For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L_P$ and $L_N$ sets. In the figure, we choose the highest AUC from 200 samples as the performance of an index. The same quantitative analysis in the Fig. S3 is repeated.
S4. A topological feature’s maximum capability measured by precision

In link prediction, precision can also be used to evaluate the quality of prediction. The precision measures the percentage of the correct prediction (node pairs indeed in \(L^P\)) among the top-k predicted candidates (4, 5, 24–26). After ranking the node pairs in both \(L^P\) and \(L^N\) according to their scores in descending order, we select \(L_k\) node pairs with the highest score. The precision is given as

\[
\text{Precision} = \frac{L_r}{L_k} \quad [22]
\]

where \(L_r\) is the number of selected node pairs that are included in \(L^P\).

Let us first consider the best index value ranking in the unsupervised approach (Fig. 1c presented in the main text and Fig. S12), in which the lowest index value of \(L_1\) is greater than the highest index value of \(L_2\). Assume \(n_1 = |\overline{L}_1 \cup L_1| = |L^P|\) and \(n_2 = |\overline{L}_2 \cup L_2| = |L^N|\) for the number of node pairs in \(L^P\) and \(L^N\). When ranking node pairs in descending order of their index values, we have three segments in the ranking list. The first is \(n_1p_1\) entities of \(L_1\), followed by \(n_2p_2\) entities of \(L_2\). The \(n_1(1-p_1)+n_2(1-p_2)\) entities of \(\overline{L}_1 \cup \overline{L}_2\), which all have the same index value, are ranked at the end. Here, \(n_1 = |\overline{L}_1 \cup L_1| = |L^P|\) and \(n_2 = |\overline{L}_2 \cup L_2| = |L^N|\). Depending on the choice of \(L_k\), the Precision\textsubscript{upper} for the best index value ranking can be derived.

\[
\text{Precision}_{\text{upper}} = \begin{cases} 
1, & n_1p_1 \geq L_k, \\
\frac{n_1p_1}{L_k}, & n_1p_1 < L_k \leq n_1p_1 + n_2p_2, \\
\frac{n_1p_1 + n_1(1-p_1)(L_k-n_1p_1-n_2p_2)}{n_1(1-p_1)+n_2(1-p_2)}, & n_1p_1 + n_2p_2 < L_k \leq n_1 + n_2.
\end{cases} \quad [23]
\]

Note that node pairs in \(\overline{L}_1 \cup \overline{L}_2\) have the same index value. The relative position of one node pair among all \(n_1(1-p_1)+n_2(1-p_2)\) entities is random. In Eq. (23), \(\frac{n_1(1-p_1)(L_k-n_1p_1-n_2p_2)}{n_1(1-p_1)+n_2(1-p_2)}\) corresponds to the probability of finding a missing link in \(\overline{L}_1 \cup \overline{L}_2\). Therefore, \(n_1(1-p_1)(L_k-n_1p_1-n_2p_2)\) is the expected number of missing links for a given \(L_k\) value.

The above deduction suggests that under the best index value ranking, the measured precision should exactly follow Precision\textsubscript{upper} given by Eq. (23). To test it, we select one index from each family and identify networks in which the unsupervised prediction by this index has an AUC value greater than 95% of AUC\textsubscript{upper}. For such networks, we could approximate that the index value ranking is close to the best scenario. We then quantify the performance of the unsupervised prediction using precision. In these networks, the measured precision well follows what Eq. (23) depicts (Fig. S13). Moreover, when measuring the prediction performance of this index in all 550 networks, the measured precision is below Precision\textsubscript{upper} (Fig. S14), supporting the claim that Eq. (23) captures the maximum capability of a topological feature measured by precision in the unsupervised approach.
Fig. S13. Networks whose AUC is close to the upper bound in the unsupervised approach. We choose the index CN, LP, L3, and HEI from each of the four families. We select the networks in which the unsupervised prediction by one index is already close to the upper bound measured by AUC (measured AUC is more than 95% of AUC$_{upper}$). For such networks, it is expected that the performance of this index measured by precision should follow Eq. (23). Indeed, for different choices of $L_k$ ($L_k = |L^1|/4, |L^P|/2, 3|L^P|/4, |L^P|$), the precision measured is almost on the line $y = x$, supporting the theoretical prediction.
Fig. S14. The maximum capability of the topological features measured by precision in the unsupervised approach. We choose the index CN, LP, L3, and HEI from each of the four families and measure the performance of the unsupervised prediction by these indexes in all 550 networks. For different choices of $L_k$ ($L_k = |L^p|/4$, $|L^p|/2$, $3|L^p|/4$, $|L^p|$), the measured precision is equal to or below Precision$_{\text{upper}}$, supporting the claim that Precision$_{\text{upper}}$ gives the maximum capability of a topological feature measured by precision. For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^p$ and $L^N$ sets. In the figure, we use the average value of 200 samples.
In the supervised approach, the mapping function can further optimize the score ranking from the index value ranking. As discussed in the main text, the optimal score ranking is such that \( L_1 \) ranks ahead, followed by \( L_1 \cup L_2 \), and \( L_2 \) ranks at the end (Fig. S15). Similar to Eq. (23), we can derive the upper bound of supervised prediction measured by precision as

\[
\text{Precision}^\text{upper} = \begin{cases} 
1, & n_1 p_1 \geq L_k, \\
\frac{n_1 (1-p_1) + n_2 (1-p_2)}{n_1 (1-p_1) + n_2 (1-p_2)} \left( \frac{1}{L_k} \right), & n_1 p_1 < L_k \leq n_1 + n_2 - n_2 p_2, \\
\frac{n_1}{L_k}, & n_1 + n_2 - n_2 p_2 < L_k \leq n_1 + n_2,
\end{cases}
\]

[24]

where \( \frac{n_1 (1-p_1)}{n_1 (1-p_1) + n_2 (1-p_2)} \) corresponds to the probability of finding a missing link in \( L_1 \cup L_2 \). It is noteworthy that the Eq. (23) and Eq. (24) still hold when the \( L_k = |L^P| \) (where the precision is equivalent to the recall (21)).

We perform the same test for Eq. (24) as for Eq. (23). For networks in which an index gives a supervised prediction already close to the upper bound \( \text{AUC}^\text{upper} \), the prediction measured by precision is expected to follow Eq. (24). This is confirmed in Fig. (S16). For all networks, the prediction measured by precision should be below what Eq. (24) yields. This is also confirmed in Fig. (S17).
Fig. S16. Networks whose AUC is close to the upper bound in the supervised approach. We choose the index CN, LP, L3, and HEI from each of the four families. After we select the networks in which the supervised prediction by one index is already close to the upper bound measured by AUC (measured AUC is more than 95% of AUC′ upper). For such networks, it is expected that the performance of this index measured by precision should follow Eq. (24). Indeed, for different choices of \( L_k \) (\( L_k = |L^P|/4, |L^P|/2, 3|L^P|/4, |L^P|, 5|L^P|/4, 3|L^P|/2 \)), the precision measured is almost on the line \( y = x \), supporting the theoretical prediction.
Fig. S17. The maximum capability of the topological features measured by precision in the supervised approach. We choose the index CN, LP, L3, and HEI from each of the four families and measure the performance of the supervised prediction by these indexes in all 550 networks. For different choices of $L_k (L_k = |L^P|/4, |L^P|/2, 3|L^P|/4, |L^P|/5, 5|L^P|/4, 3|L^P|/2)$, the measured precision is equal to or below Precision$^{\prime}_{\text{upper}}$, supporting the theoretical results for the maximum capability of a topological feature measured by precision. For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^P$ and $L^N$ sets. In the figure, we choose the highest precision from 200 samples as the performance of an index.
Finally, some studies argue that the imbalanced samples extremely affect the results measured by precision (22, 27, 28). For this reason, we test Eq. (23) and Eq. (24) under the imbalanced sampling \( \text{sample}2 \). The imbalanced sample indeed reduces the performance by precision (Fig. S18 and Fig. S19). But Eq. (23) and Eq. (24) still capture the upper bound of the performance, which correctly gives the maximum capability of a topological feature. It is worth noting that the \( \text{Precision}_\text{upper} \) for supervised prediction will be lower than 0.5 when the positive testing set \( L^P \) and the negative testing set \( L^N \) are imbalanced (Fig. S19).

Fig. S18. Unsupervised prediction measured by precision based on imbalanced positive and negative samples by \text{sample}2\. We choose the index CN, LP, L3, and HEI from each of the four families and measure the performance of the unsupervised prediction by these indexes in all 550 networks. For different choices of \( L_k (L_k = |L^P|^4, |L^P|^2, 3|L^P|^4, |L^P|) \), the measured precision is equal to or below \( \text{Precision}_\text{upper} \), supporting the claim that \( \text{Precision}_\text{upper} \) gives the maximum capability of a topological feature measured by precision. The same quantitative analysis in the Fig. S14 is repeated.
Fig. S19. Supervised prediction measured by precision based on imbalanced positive and negative samples by sample2. We choose the index CN, LP, L3, and HEI from each of the four families and measure the performance of the supervised prediction by these indexes in all 550 networks. For different choices of $L_k$ ($L_k = \frac{|L^P|}{4}, \frac{|L^P|}{2}, 3\frac{|L^P|}{4}, |L^P|, 5\frac{|L^P|}{4}, 3\frac{|L^P|}{2}$), the measured precision is equal to or below $\text{Precision}^{\prime}_{\text{upper}}$, supporting the theoretical results for the maximum capability of a topological feature measured by precision. The same quantitative analysis in the Fig. S17 is repeated.
S5. A topological feature’s maximum capability measured by AUC-mROC

To deal with the imbalanced positive and negative samples, Muscoloni et al. propose the area under the magnified ROC (AUC-mROC) to measure the prediction performance of one index (29). The mROC can make the performance of a random predictor always equal to 0.5. Denote \( n_1 = |L^P| \) and \( n_2 = |L^N| \) by the size of the positive and negative testing set. Given the index value of a predictor, we here define the true positive (TP@\( r \)) and false positive (FP@\( r \)) as the number of samples in \( L^P \) and \( L^N \) at a varying ranking threshold \( r \in [1, n_1 + n_2] \). Hence, we can denote the non-normalized magnified TPR (nmTPR@\( r \)) and non-normalized magnified FPR (nmFPR@\( r \)) at a varying ranking threshold \( r \in [1, n_1 + n_2] \) as

\[
\text{nmTPR@}\!r = \frac{\ln(1 + \text{TP@}\!r)}{\ln(1 + n_1)},
\]

\[
\text{nmFPR@}\!r = \frac{\ln(1 + \text{FP@}\!r)}{\ln(1 + n_2)}.
\]

To make the AUC-mROC of the random predictor 0.5, the nmTPR@\( r \) is normalized to mTPR@\( r \). The mTPR@\( r \) is defined as

\[
\text{mTPR@}\!r = \text{nmFPR@}\!r + \frac{\text{nmTPR@}\!r - \ln(1 + n_1) \left( 1 + \text{FP@}\!r \cdot \frac{n_2}{n_2} \right)}{1 - \ln(1 + n_1) \left( 1 + \text{FP@}\!r \cdot \frac{n_1}{n_2} \right)} \cdot (1 - \text{nmFPR@}\!r).
\]

Therefore, the mROC curve is composed of the points at coordinates (nmFPR@\( r \), mTPR@\( r \)) for each \( r \in [1, n_1 + n_2] \). Finally, the AUC-mROC can be obtained by computing the area under the mROC curve using the trapezoidal rule.

---

Let us first consider the best index value ranking in the unsupervised approach (Fig. 1c presented in the main text and Fig. S20), in which the lowest index value of \( L_1 \) is greater than the highest index value of \( L_2 \). When ranking node pairs in descending order of their index values, we have three segments in the ranking list. The first is \( n_1 p_1 \) entities of \( L_1 \), followed by \( n_2 p_2 \) entities of \( L_2 \). The \( n_1 (1 - p_1) + n_2 (1 - p_2) \) entities of \( L_1 \cup L_2 \), which all have the same index value, are ranked at the end. Here, \( n_1 = |L_1 \cup L_2| = |L^P| \) and \( n_2 = |L_2| = |L^N| \). Depending on the different ranking threshold \( r \), the TP@\( r \) and FP@\( r \) for the best index value ranking can be derived.

---

**Fig. S20.** The best index value ranking for AUC-mROC in the unsupervised approach. In the best index value ranking illustrated in Fig. 1c of the main text, \( L_1 \) is ranked ahead of \( L_2 \). Therefore, when ranking node pairs in descending order of their index values, we have three segments in the rank list. The first is \( n_1 p_1 \) entities of \( L_1 \), followed by \( n_2 p_2 \) entities of \( L_2 \). The \( n_1 (1 - p_1) + n_2 (1 - p_2) \) entities of \( L_1 \cup L_2 \), which all have the same index value, are ranked at the end. Here, \( n_1 = |L_1 \cup L_2| = |L^P| \) and \( n_2 = |L_2| = |L^N| \). Depending on the different ranking threshold \( r \), the TP@\( r \) and FP@\( r \) for the best index value ranking can be derived.

---

\[
\text{TP@}\!r = \begin{cases} 
  r, & n_1 p_1 < r \\
  n_1 p_1 + \frac{n_1 (1 - p_1)(r - n_1 p_1 - n_2 p_2)}{n_1 (1 - p_1) + n_2 (1 - p_2)}, & n_1 p_1 \geq r,
\end{cases}
\]

\[
\text{FP@}\!r = \begin{cases} 
  0, & n_1 p_1 < r \\
  n_2 p_2 + \frac{n_2 (1 - p_2)(r - n_1 p_1 - n_2 p_2)}{n_1 (1 - p_1) + n_2 (1 - p_2)}, & n_1 p_1 \geq r,
\end{cases}
\]
Note that node pairs in $L_1 \cup L_2$ have the same index value. The relative position of one node pair among all $n_1(1-p_1) + n_2(1-p_2)$ entities is random. In Eqs. (28) and (29), $\frac{n_1(1-p_1)}{n_1(1-p_1) + n_2(1-p_2)}$ and $\frac{n_2(1-p_2)}{n_1(1-p_1) + n_2(1-p_2)}$ correspond to the probability of finding a missing link and nonexistent link in $L_1 \cup L_2$, respectively. Therefore, $\frac{n_1(1-p_1)(r-n_1p_1-n_2p_2)}{n_1(1-p_1) + n_2(1-p_2)}$ and $\frac{n_2(1-p_2)(r-n_1p_1-n_2p_2)}{n_1(1-p_1) + n_2(1-p_2)}$ are the expected number of missing links and nonexistent links for a given $r$ value, respectively.

By using Eqs. (25)-(29), we can derive a topological feature’s maximum capability ($AUC\text{-}m\text{ROC}_{\text{upper}}$) measured by AUC-mROC in the unsupervised approach. The above deduction suggests that under the best index value ranking, the measured $AUC\text{-}m\text{ROC}$ should exactly follow $AUC\text{-}m\text{ROC}_{\text{upper}}$. To test it, we measure the prediction performance of 21 indexes in all 550 networks. And we use the experiment setup sample2 to make the prediction. The sample2 randomly removes 20% of $L$ links as the missing links. In the training step, the positive set is composed of 80% of the removed links (16% of $L$ links), and the negative set is composed of 80% of all nonexistent links. In the testing step, the positive set is composed of the rest 20% of the removed links, and the negative set is composed of the rest 20% of all nonexistent links. The measured $AUC\text{-}m\text{ROC}$ is below $AUC\text{-}m\text{ROC}_{\text{upper}}$ (Fig. S21), supporting the claim that $AUC\text{-}m\text{ROC}_{\text{upper}}$ captures the maximum capability of a topological feature measured by AUC-mROC in the unsupervised approach.

![Fig. S21. The maximum capability of the topological features measured by AUC-mROC in the unsupervised approach. The imbalanced positive and negative samples are generated by sample2. We use 21 indexes from four families and measure the performance of the unsupervised prediction by these indexes in all 550 networks. The prediction measured by AUC-mROC is equal to or below AUC-mROC_{upper}, supporting the claim that AUC-mROC_{upper} gives the maximum capability of a topological feature measured by AUC-mROC. For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^P$ and $L^N$ sets. In the figure, we use the average value of 200 samples.](image-url)
The optimal score ranking for AUC-mROC in the supervised approach. The optimal score ranking after the mapping function is to have $L_1 \cup L_2$ ranked in between $L_1$ and $L_2$, as illustrated in Fig. 1d of the main text. When ranking node pairs in descending order of their scores, we have three segments in the rank list. The first is $n_1 p_1$ entities of $L_1$, followed by $n_1 (1 - p_1) + n_2 (1 - p_2)$ entities of $L_1 \cup L_2$, which all have the same score. $n_2 p_2$ entities of $L_2$ are ranked at the end. Here, $n_1 = |L_1 \cup L_1| = |L^p|$ and $n_2 = |L_2 \cup L_2| = |L^m|$. Depending on the different ranking threshold $r$, the TP@r and FP@r for the best index value ranking can be derived.

In the supervised approach, the mapping function can further optimize the score ranking from the index value ranking. As discussed in the main text, the optimal score ranking is such that $L_1$ ranks ahead, followed by $L_1 \cup L_2$, and $L_2$ ranks at the end (Fig. S22). Here we can derive the TP@r and FP@r for the best index value ranking in the supervised approach as

$$\text{TP}@r = \begin{cases} r, & n_1 p_1 \geq r, \\ n_1 p_1 + \frac{n_1 (1 - p_1)(r - n_1 p_1)}{n_1 (1 - p_1) + n_2 (1 - p_2)}, & n_1 p_1 < r \leq n_1 + n_2 - n_2 p_2, \\ \frac{n_1 (1 - p_1)(r - n_1 p_1)}{n_1 (1 - p_1) + n_2 (1 - p_2)}, & n_1 + n_2 - n_2 p_2 < r \leq n_1 + n_2, \end{cases}$$  

$$\text{FP}@r = \begin{cases} 0, & n_1 p_1 \geq r, \\ \frac{n_2 (1 - p_2)(r - n_1 p_1)}{n_1 (1 - p_1) + n_2 (1 - p_2)}, & n_1 p_1 < r \leq n_1 + n_2 - n_2 p_2, \\ \frac{n_2 (1 - p_2)(r - n_1 p_1)}{n_1 (1 - p_1) + n_2 (1 - p_2)}, & n_1 + n_2 - n_2 p_2 < r \leq n_1 + n_2, \end{cases}$$

where $\frac{n_1 (1 - p_1)}{n_1 (1 - p_1) + n_2 (1 - p_2)}$ and $\frac{n_2 (1 - p_2)}{n_1 (1 - p_1) + n_2 (1 - p_2)}$ correspond to the probability of finding a missing link and nonexistent link in $L_1 \cup L_2$, respectively.

By putting Eqs. (30) and (31) into Eqs. (25)-(27), we can derive a topological feature’s maximum capability (AUC-mROC'$_{upper}$) measured by AUC-mROC in the supervised approach. We perform the same test for AUC-mROC'$_{upper}$ as for AUC-mROC$_{upper}$. For all networks, the prediction measured by AUC-mROC should be below what AUC-mROC'$_{upper}$ yields. This is also confirmed in Fig. (S23).
Fig. S23. The maximum capability of the topological features measured by AUC-mROC in the supervised approach. The imbalanced positive and negative samples are generated by sample2. We use 21 indexes from four families and measure the performance of the supervised prediction by these indexes in all 550 networks. The prediction measured by AUC-mROC is equal to or below AUC-mROC_{upper}, supporting the claim that AUC-mROC_{upper} gives the maximum capability of a topological feature measured by AUC-mROC. For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of L\(^P\) and L\(^N\) sets. In the figure, we choose the highest AUC-mROC from 200 samples as the performance of an index.
S6. Extended discussion on the lowest index value

An index for a topological feature is designed to quantify the expression of this feature. Hence, the logic behind the index is that its value increases monotonically with the extent to which the feature is expressed. For entities that do not hold the feature at all, they should have the same and the lowest value. If an entity that does not hold the feature has a higher index value than that does hold the feature, the index must be incorrectly designed. Likewise, if two entities that do not hold the feature have different index values, the index is also incorrectly designed.

Currently, there is no unified rule on what value should be the lowest. In the main text, we consider the lowest value to be zero. Indeed, all 21 indexes in this study assign the value 0 to entities that do not hold the feature they are designed to quantify. There could be exceptions. For example, one can design an index equal to the CN value plus 0.5. This is a valid index. However, it is easy to see that our theoretical framework still holds when the lowest value is not zero. Indeed, the validity of based on the fact that entities that do not hold the feature are assigned the same and the lowest value, which is guaranteed by the design of the index.
S7. Extended discussion on the scaling of the unsupervised prediction performance

As illustrated in Eq. (2) and Eq. (3) of the main text, the gap between the upper and lower bound of the unsupervised prediction is $p_1 \times p_2$. Therefore, if the average $p_1 \times p_2$ is relatively small for a given feature, the performance of different indexes associated with it should roughly scale as $p_1 - p_2$. However, the final scaling depends on how even the measured AUC data points distribute within the gap. In the main text, we show the scaling behavior of the common neighbor feature and path feature. For the path of length three feature, the average $p_1 \times p_2$ is smaller than that of the path feature (Table S1). The scaling also holds (Fig. S24a). But because the AUC for the path of length three feature is closer to the upper bound compared with the path feature (Figs. S2b, c), the data points are systematically located in the upper corner. For the heterogeneity feature, the average $p_1 \times p_2$ is much larger (Table S1). Hence, the AUC does not follow $p_1 - p_2$.

Fig. S24. The scaling of the AUC values from the path of length three feature and the heterogeneity feature. Eq. (2) and Eq. (3) in the main text suggest that the actual prediction by an index fluctuates within $p_1 \times p_2$. (a) The path of length three feature has a relatively small average $p_1 \times p_2$ value. Hence the data points roughly scale as $p_1 - p_2$. (b) For the heterogeneity feature, as the average $p_1 \times p_2$ value is large, the data points do not collapse to the line $y = x$. For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^P$ and $L^N$ sets. In the figure, we use the average value of 200 samples.
S8. Test on other machine learning algorithms

In the main text, we use the Random Forest classifier to analyze the capability of a topological feature. Here, we also consider Gradient Boosting and AdaBoost classifiers to display similar tests. The sampling method is the same as that of the main text. The results from Figs. S25 and S26 show that our quantitative framework is not affected by the machine learning algorithm. Hence, this further indicates that the machine learning algorithm can find the optimal mapping function and improve the prediction compared with that of the unsupervised prediction.

Fig. S25. Supervised prediction measured by AUC based on the Gradient Boosting classifier. Eq. (4) in the main text suggests that $\text{AUC}_{\text{upper}}$ sets the upper bound of the supervised prediction. This is confirmed by 21 indexes related to 4 topological features: common neighbor (\(a\)), path (\(b\)), path of length three (\(c\)), and heterogeneity (\(d\)). For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^P$ and $L^N$ sets. In the figure, we choose the highest AUC from 200 samples as the performance of an index. The same quantitative analysis in the Fig. S3 is repeated.
Fig. S26. Supervised prediction measured by AUC based on the AdaBoost classifier. Eq. (4) in the main text suggests that $AUC_{\text{upper}}$ sets the upper bound of the supervised prediction. This is confirmed by 21 indexes related to 4 topological features: common neighbor (a), path (b), path of length three (c), and heterogeneity (d). For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^P$ and $L^N$ sets. In the figure, we choose the highest AUC from 200 samples as the performance of an index. The same quantitative analysis in the Fig. S3 is repeated.
The link prediction performance relies on the relative rank of the three sets: $L_1$, $L_2$, and $\bar{L}_1 \cup \bar{L}_2$. Totally there are 6 different rankings. In Figs. 1b, 1c and 1d of the main text, we show 3 scenarios based on different rankings (in descending order of the scores/values): $L_2 > L_1 > \bar{L}_1 \cup \bar{L}_2$, $L_1 > L_2 > \bar{L}_1 \cup \bar{L}_2$, and $L_1 > \bar{L}_1 \cup \bar{L}_2 > L_2$. Here we show the other 3 different permutations in Fig. S27 and derive the AUC value in these ranking scenarios.

For the ranking in Fig. S27a, we have $\bar{L}_1 \cup \bar{L}_2 > L_1 > L_2$. A positive sample outscores a negative sample only when one node pair is from $L_1 \cup \bar{L}_1$ (the whole set $L_1^P$), and the other is from $L_2$. Hence, this gives

$$\text{AUC}_a = \frac{n'}{n} + \frac{1}{2} \frac{n''}{n} = p_2 + \frac{1}{2} (1 - p_1)(1 - p_2).$$

[32]

For the ranking in Fig. S27b, we have $\bar{L}_1 \cup L_2 > L_2 > L_1$. A positive sample outscores a negative sample only when one node pair is from $\bar{L}_1$, and the other is from $L_2$ (Fig. S27b). Correspondingly, we have

$$\text{AUC}_b = \frac{n'}{n} + \frac{1}{2} \frac{n''}{n} = (1 - p_1)p_2 + \frac{1}{2} (1 - p_1)(1 - p_2).$$

[33]

For the ranking in Fig. S27c, we have $L_2 > \bar{L}_1 \cup \bar{L}_2 > L_1$. Node pairs in $L_2$ outscore all positive samples (the whole set $L_2^P$). Hence, this gives

$$\text{AUC}_c = \frac{n'}{n} + \frac{1}{2} \frac{n''}{n} = \frac{1}{2} (1 - p_1)(1 - p_2).$$

[34]

By comparing the 6 equations (Eq. (32), Eq. (33), Eq. (34), AUC$_{\text{lower}} = p_1(1 - p_2) + \frac{1}{2} (1 - p_1)(1 - p_2)$, AUC$_{\text{upper}} = p_1 + \frac{1}{2} (1 - p_1)(1 - p_2)$, and AUC$_{\text{upper}} = p_1 + (1 - p_1)p_2 + \frac{1}{2} (1 - p_1)(1 - p_2)$), we can find that the AUC$_{\text{upper}}$ gives the highest AUC value. Hence, Fig 1d of the main text gives the optimal rankings of the three sets. Indeed, this is in line with the theory of machine learning, as the score of the positive sample should be greater than that of the negative one if we assume the positive one is expected to be predicted during the learning process.

To further test if a machine learning algorithm practically re-arranges the ranking, we show an example below. The Salton index is used. The distribution of index values in different sets is shown in Fig. S28a. After applying the Random Forest
algorithm, the classifier finds the mapping function to transform the index value into the score. The corresponding distribution of the score is presented in Fig. S28b. The set $\mathcal{I}_1 \cup \mathcal{I}_2$ indeed moves to the middle, giving rise to a ranking similar to the optimal ranking derived theoretically.

Fig. S28. A real case that has achieved the mapping from the index value to the score. (a) The distribution of index values obtained. (b) The distribution of scores obtained by using the Random Forest classifier in the supervised approach. This network is “56e9e0d7a6d70217090cdff1a” in the data set.
S10. Another example of feature and index selection in link prediction

In the main text, taking the common neighbor feature as an example, we show the theoretical finding can help us to determine the feature and index selection. To further validate the preceding analysis, we here conduct a second analysis using the path feature (Table S2). When using the index SPL to make unsupervised predictions in two real networks, we obtain AUC = 0.567 in both networks. The prediction results are relatively low, so it is difficult for us to decide whether the path feature is suitable for these two real networks. Under this case, we have to try one by one for the other 4 indexes on the two networks when the $p_1$ and $p_2$ values are unknown. However, calculating the maximum capability of the path feature ($AUC_{upper}$ presented in the main text) shows that we should switch to a new feature for prediction in the network E and should try the other indexes based on the path feature in the network F (Table S2). This analysis further shows that the theoretical finding presented in the main text can be applied to optimize the feature and index selection.

Moreover, as the theoretical expression can confirm whether an index is superior to the other indexes, our theoretical finding can provide strong support for experimental validation such as the interaction between genes, and protein-protein interactions. More importantly, the theoretical expression can also help to estimate how close to the upper bound for observed performance in the same topological feature.

Table S2. Another example that utilizes the maximum capability for feature selection. The AUC performance of the unsupervised prediction using SPL is the same for both networks E and F. However, using the $p_1$ and $p_2$ values, the maximum capability of the path feature can be estimated. The path feature is not suitable for network E but has potential in network F. The network E is “56e98770a6d70217090cde08”, and the network F is “Cat_cerebral_hemisphere_cortex_only” in the dataset.

| SPL | $p_1$ | $p_2$ | $AUC_{upper}$ | LP | Katz | FL | RSS |
|-----|-------|-------|---------------|----|------|----|-----|
| Network E | 0.567 | 0.158 | 0.026 | 0.568 | 0.567 | 0.567 | 0.567 |
| Network F | 0.567 | 1.0 | 1.0 | 1.0 | 0.867 | 0.867 | 0.869 | 0.841 |
S11. The theoretical expression of $p_1$ and $p_2$

To give more insights into the structural characteristics that make a topological feature effective in link prediction, we deduct the theoretical expression of $p_1$ and $p_2$. Here, we take the common neighbor feature as an example. The idea behind the feature is that two unconnected nodes that share the same neighborhood nodes are likely to become a link. Hence, the indexes based on the common neighbor assign a value greater than 0 to a node pair only if this node pair belongs to a link of the closed triangle. Since $p_1$ is the fraction of samples in the positive set $L^P$ with an index value greater than 0, $p_1$ can be quantified as the probability that a randomly picked node pair from all existing links of a network is exactly from one link in a closed triangle. Therefore, the theoretical expression of $p_1$ is defined as

$$p_1' = \frac{3 \times N_\Delta - S_\Delta}{L},$$ \[35\]

where the $N_\Delta$ is the number of closed triangles in a network. The $S_\Delta$ is the number of times that a link is shared by multiple triangles (Fig. S29). Because a link can belong to multiple triangles, the $3 \times N_\Delta$ would over-count the number of links belonging to a triangle and has to subtract the number of times a link appears in other triangles.

Likewise, as $p_2$ is the fraction of samples in the negative set $L^N$ with an index value greater than 0, $p_2$ can be quantified as the probability that a randomly picked node pair from all nonexistent links of a network shares a common neighbor (this node pair and an open triangle constitute a closed triangle). Hence, the theoretical expression of $p_2$ is formulated as

$$p_2' = \frac{N_\wedge - S_\wedge}{\frac{N(N-1)}{2} - L},$$ \[36\]

where the $N_\wedge$ is the number of open triangles in a network (the triad that two nodes that are not directly connected but both connect to the third node). The $S_\wedge$ is the number of times that an unconnected node pair is shared by other open triangles, which can cause an overcount if it is not subtracted (Fig. S29). The $\frac{N(N-1)}{2} - L$ corresponds to the total number of unconnected node pairs.

![Fig. S29. An illustration of explaining parameters in Eqs. (35) and (36). Since the two closed triangles ($\Delta_{134}$ and $\Delta_{124}$) share the same link (1-4), $N_\Delta = 2$ and $S_\Delta = 1$. Similarly, the two open triangles ($\wedge_{123}$ and $\wedge_{243}$) share the unconnected node pair (2-3), hence $N_\wedge = 2$ and $S_\wedge = 1$.](image_url)

To validate Eqs. (35) and (36), we take 550 empirical networks to directly calculate the theoretical values $p_1'$ and $p_2'$. Shading multiple links will change the existing topology of the original network which makes the estimate of $N_\Delta$ or $N_\wedge$ incorrect. Still, we observe an overall nice agreement between the theoretical and empirical values even when 20% of links are temporally removed (Figs. S30a, b). Moreover, to give more reasonable evidence of Eqs. (35) and (36), we here also consider the situation that computing the $p_1'$ and $p_2'$ from the original network. The results from Fig. S30c and Fig. S30d show that the theoretical values ($p_1'$ and $p_2'$) are line with the empirical values ($p_1$ and $p_2$), demonstrating a perfect agreement. The theoretical expressions explain why $C$ itself is insufficient to characterize the capability of the common neighbor feature. More importantly, the Eqs. (35) and (36) help us understand network characteristics associated with the utilization of a topological feature in link prediction.
Fig. S30. Evidence of the theoretical values $p'_1$ and $p'_2$ applied to 550 real networks. (a, b) The theoretical values $p'_1$ and $p'_2$ are given by Eq. (35) and Eq. (36) from the rest 80% of links, respectively. (c, d) The theoretical values $p'_1$ and $p'_2$ are given by Eq. (35) and Eq. (36) from the original network, respectively. For $p_1$ and $p_2$, we generate 200 independent pairs of $L^P$ and $L^N$ sets based on the random sampling (removed 20% of links) in each network, and we use the average value of them.
S12. Extended discussion on the prediction performance measured by precision

The precision measure involves a hyper-parameter \( L_k \) for the cutoff of the top-k node pairs. Therefore, we might not well measure the performance of an index when only using precision. Here, we show the link prediction performance in two networks (Table S3). When using the CN index to make unsupervised predictions in network G, we obtain \( \text{AUC} = 0.533 \), suggesting that the CN index overall has limited potential (\( \text{AUC}_{\text{upper}} = 0.533 \)). But when measuring the performance by precision, the obtained measure can vary significantly with different choices of \( L_k \). When choosing \( L_k < p_1|L^P| \), the predicted upper bound is 1. In this case, we also obtain a high precision value \( \text{Precision} = 0.973 \). But because \( p_1 \) is very small for network G, a high precision value only suggests that the prediction is correct for the very top candidates. When \( L_k \) becomes larger, the precision drops drastically (\( \text{Precision} = 0.550 \) for \( L_k = 121 \)).

Likewise, in network H, the performance of unsupervised prediction by LP index yields \( \text{AUC} = 0.826 \). The LP index also has a high potential in terms of the AUC measure (\( \text{AUC}_{\text{upper}} = 0.977 \)). But if measured by precision, we obtain a low value (\( \text{Precision} = 0.381 \) for \( L_k < p_1|L^P| \)). Hence, the prediction accuracy is low for the top candidates. But if the number of candidates increases, the precision goes up again. When \( L_k > p_1|L^P| \), we obtain \( \text{Precision} = 0.711 \).

The two networks in Table S3 demonstrate a vivid example of the complexity of interpreting the precision measure. One has to take the \( p_1 \) into consideration in order to explain the number of populations the precision is measured from.

Table S3. The interpretation of the performance by precision measure. The precision gives different performance when choosing different \( L_k \). The network G is “Water_Distribution_Network_EXNET”, and the network H is “Freshwater_stream_webs_Stony” in the data set.

|       | AUC  | \( p_1 \) | | \( p_1 \) | | \( \text{Precision} \) | | \( \text{Precision} \) |
|-------|------|-------|---|---|---|---|---|
| Network G | 0.533 | 0.067 | 241 | 16 | 0.973 (\( L_k = 10 \)) | 0.550 (\( L_k = 121 \)) |
| Network H | 0.826 | 0.976 | 83  | 81 | 0.381 (\( L_k = 21 \)) | 0.711 (\( L_k = 83 \)) |
**S13. The analysis for 10% random removal links**

In the main text, we show the analysis for 20% random removal links. For robustness check, we here repeat the analysis for 10% random removal. The similar AUC results (the Fig. S32 is similar to Fig. S2, the Fig. S31 is similar to Fig. S3) are obtained. Analogously, the similar precision results (the Fig. S33 is similar to Fig. S14, the Fig. S34 is similar to Fig. S17) are obtained. These show that our quantitative framework is not affected by the random removal percentage.

**Fig. S31.** The supervised prediction measured by AUC based on 10% random removal. Eq. (4) in the main text suggests that $AUC'_{\text{upper}}$ sets the upper bound of the supervised prediction. This is confirmed by 21 indexes related to 4 topological features: common neighbor (a), path (b), path of length three (c), and heterogeneity (d). For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L_P$ and $L_N$ sets. In the figure, we choose the highest AUC from 200 samples as the performance of an index. The same quantitative analysis in the Fig. S3 is repeated.
Eqs. (2) and (3) in the main text suggest that different indexes have different prediction performances, but all indexes associated with one topological feature share the same AUC<sub>upper</sub> and AUC<sub>lower</sub>. This is confirmed by 21 indexes related to 4 topological features: common neighbor (a), path (b), path of length three (c), and heterogeneity (d). For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of L<sub>P</sub> and L<sub>N</sub> sets. In the figure, we use the average value of 200 samples. The same quantitative analysis in the left panel of Fig. S2 is repeated.
Fig. S33. The unsupervised prediction measured by precision based on 10% random removal. We choose the index CN, LP, L3, and HEI from each of the four families and measure the performance of the unsupervised prediction by these indexes in all 550 networks. For different choices of $L_k$ ($L_k = |L^p|/4, |L^p|/2, 3|L^p|/4, |L^p|$), the measured precision is equal to or below Precision$_{upper}$, supporting the claim that Precision$_{upper}$ gives the maximum capability of a topological feature measured by precision. For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^p$ and $L^N$ sets. In the figure, we use the average value of 200 samples.
Fig. S34. The supervised prediction measured by precision based on 10% random removal. We choose the index CN, LP, L3, and HEI from each of the four families and measure the performance of the supervised prediction by these indexes in all 550 networks. For different choices of $L_k$ ($L_k = |L^p|/4, |L^p|/2, 3|L^p|/4, |L^n|, 5|L^p|/4, 3|L^p|/2$), the measured precision is equal to or below Precision$^{\prime}_{\text{upper}}$, supporting the theoretical results for the maximum capability of a topological feature measured by precision. For each network, we randomly generate 200 realizations of networks with link removal, as well as 200 pairs of $L^p$ and $L^n$ sets. In the figure, we choose the highest precision from 200 samples as the performance of an index.
S14. The detailed information of empirical dataset

To verify the universality of the pattern uncovered about the maximum capability of the topological features, we use the dataset from “CommunityFitNet corpus” (23). The basic statistics of different networks in each domain are shown in Table S4.

Table S4. The basic statistics of 6 different-domain networks. We here show the average statistics of networks in 6 different domains. \( \langle N \rangle \) and \( \langle L \rangle \) are the average number of nodes and links, respectively. \( \langle C \rangle \) is the average number of clustering coefficients. \( \langle k \rangle \) is the average degree, \( \langle d \rangle \) is the average shortest path length.

| Network domains | \( \langle N \rangle \) | \( \langle L \rangle \) | \( \langle k \rangle \) | \( \langle C \rangle \) | \( \langle d \rangle \) |
|-----------------|----------------|----------------|----------------|----------------|----------------|
| biological      | 294.235        | 780.151        | 6.302          | 0.135          | 4.577          |
| social          | 558.581        | 1988.331       | 7.592          | 0.840          | 6.017          |
| economic        | 701.677        | 865.685        | 3.338          | 0.040          | 11.697         |
| technological   | 532.543        | 1061.029       | 4.034          | 0.118          | 6.076          |
| transportation  | 721.343        | 1274.143       | 3.493          | 0.101          | 12.555         |
| information     | 494.167        | 1266.222       | 5.315          | 0.223          | 3.419          |
References

1. Liben-Nowell D and Kleinberg J. The link prediction problem for social networks. J Am Soc Inf Sci Technol 2007; 58: 1019–1031.
2. Adamic LA and Adar E. Friends and neighbors on the web. Soc Networks 2003; 25: 211–230.
3. Zhou T, Li L and Zhang YC. Predicting missing links via local information. Eur Phys J B 2009; 71: 623–630.
4. Sun J, Feng L and Xie J et al. Revealing the predictability of intrinsic structure in complex networks. Nat Commun 2020; 11: 574.
5. Ghorbanzadeh H, Sheikhhahmadi A and Jalili M et al. A hybrid method of link prediction in directed graphs. Expert Syst Appl 2021; 165: 113896.
6. Ravasz E, Somera AL and Mongru DA et al. Hierarchical organization of modularity in metabolic networks. Science 2002; 297: 1551–1555.
7. Leicht EA, Holme P and Newman MEJ. Vertex similarity in networks. Phys Rev E 2006; 73: 026120.
8. Muscoloni A, Abdelhamid I and Cannistraci CV. Local-community network automata modelling based on length-three-paths for prediction of complex network structures in protein interactomes, food webs and more. BioRxiv 2018.
9. Muscoloni A, Michieli U and Zhang Y et al. Adaptive network automata modelling of complex networks. Preprints 2022.
10. Muscoloni A and Cannistraci CV. “Stealing fire or stacking knowledge” by machine intelligence to model link prediction in complex networks. Iscience 2023; 26: 105697.
11. Lü L, Jin CH and Zhou T. Similarity index based on local paths for link prediction of complex networks. Phys Rev E 2009; 80: 046122.
12. Cao RM, Liu SY and Xu XK. Network embedding for link prediction: the pitfall and improvement. Chaos 2019; 29: 103102.
13. Martínez V, Berzal F and Cubero JC. A survey of link prediction in complex networks. ACM Comput Surv 2016; 49: 69.
14. Kumar A, Singh SS and Singh K et al. Link prediction techniques, applications, and performance: a survey. Physica A 2020; 553: 124289.
15. Ran Y, Liu T and Jia T et al. A novel similarity measure for mining missing links in long-path networks. Chin Phys B 2022; 31: 068902.
16. Papadimitriou A, Symeonidis P and Manolopoulos Y. Friendlink: link prediction in social networks via bounded local path traversal. In: Proceedings of the International Conference on Computational Aspects of Social Networks. Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2011, 66–71.
17. Chen HH, Gou L and Zhang X et al. Discovering missing links in networks using vertex similarity measures. In: Proceedings of the 27th annual ACM Symposium on Applied Computing. New York, NY: Association for Computing Machinery, 2012, 138–143.
18. Ran Y, Liu SY and Yu X et al. Predicting future links with new nodes in temporal academic networks. J Phys Complex 2022; 3: 015006.
19. Kovács IA, Luck K and Spirohn K et al. Network-based prediction of protein interactions. Nat Commun 2019; 10: 1240.
20. Shang K, Li T and Small M et al. Link prediction for tree-like networks. Chaos 2019; 29: 061103.
21. Zhou T. Progresses and challenges in link prediction. Iscience 2021; 24: 103217.
22. Yang Y, Lichtenwalter RN and Chawla NV. Evaluating link prediction methods. Knowl Inf Syst 2015; 45: 751–782.
23. Ghasemian A, Hosseinnia H and Galstyan A et al. Stacking models for nearly optimal link prediction in complex networks. Proc Natl Acad Sci USA 2020; 117: 23393–23400.
24. Lü L, Pan L and Zhou T et al. Toward link predictability of complex networks. Proc Natl Acad Sci USA 2015; 112: 2325–2330.
25. García-Pérez G, Aliakbarisani R and Ghasemi A et al. Precision as a measure of predictability of missing links in real networks. Phys Rev E 2020; 101: 052318.
26. Nasiri E, Berahmand K and Li Y. A new link prediction in multiplex networks using topologically biased random walks. Chaos Solit Fractals 2021; 151: 1052318.
27. Lichtnwalter R and Chawla NV. Link prediction: fair and effective evaluation. In: Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2012, 376–383.
28. Fawcett T. An introduction to roc analysis. Pattern Recognit Lett 2006; 27: 861–874.
29. Muscoloni A and Cannistraci CV. Early retrieval problem and link prediction evaluation via the area under the magnified ROC. Preprints 2022.