Artif Intell Rev (2019) 52:1961–1995
https://doi.org/10.1007/s10462-017-9590-2

A systemic analysis of link prediction in social network

Sogol Haghani[1] · **Mohammad Reza Keyvanpour**[2]

Published online: 17 November 2017

© Springer Science+Business Media B.V., part of Springer Nature 2017

Abstract Link prediction is an important task in data mining, which has widespread applications in social network research. Given a social network, the objective of this task is to predict future links which have not yet observed in the current state of the network. Owing to its importance, the link prediction task has received substantial attention from researchers in diverse disciplines; thus, a large number of methodologies for solving this problem have been proposed in recent decades. However, existing literatures lack a current and comprehensive analysis of existing link prediction methodologies. Couple of survey articles on link prediction are available, but they are out-dated as numerous link prediction methods have been proposed after these articles have been published. In this paper, we provide a systematic analysis of existing link prediction methodologies. Our analysis is comprehensive, it covers the earliest scoring-based methodologies and extends up to the most recent methodologies which are based on deep learning methods. We also categorize the link prediction methods based on their technical approach, and discuss the strength and weakness of various methods.

Keywords Link prediction · Social network · Approaches · Benefits · Challenges

1 Introduction

Relational data can be represented in the form of networks such as social network, biological network and knowledge graph (Lü and Zhou 2011; Nickel et al. 2016). Social networks have a vast diversity of relations such as friendship in Facebook and follower on Twitter (Davis et al.

✉ Sogol Haghani
s.haghani@student.alzahra.ac.ir

Mohammad Reza Keyvanpour
keyvanpour@alzahra.ac.ir

[1] Department of Computer Engineering and Data Mining Laboratory, Alzahra University, Vanak, Tehran, Iran

[2] Department of Computer Engineering, Alzahra University, Vanak, Tehran, Iran

2011; Chung et al. 2016). A network can be visualized as a graph in which nodes represent entities and edges correspond to links or interactions. The increasing availability of online social networks is useful not only in social network analysis, such as community detection or link prediction, but also in recommendation systems. Networks are powerful representations and are employed in different tasks such as machine learning and data mining (Al Hasan et al. 2006; Ngonmang et al. 2015). Developing models to analyze such relational data has attracted a considerable amount of attention, where link prediction is a fundamental task. Link prediction is a task of predicting unseen links with a given social graph. The prediction of missing links or links that will be formed in the future based on snapshots of the network, is formally defined as a link prediction problem (Liben-Nowell and Kleinberg 2007; Clauset et al. 2008; Lü and Zhou 2011; Dunlavy et al. 2011).

Link prediction is an important task in link mining (Getoor and Diehl 2005). In order to predict links in relational data, one needs to provide a model for different types of information in the graph (Kashima et al. 2009; Nguyen and Mamitsuka 2012). Node information is the first type of information used to predict links, and it is captured from entities such as node attributes(Rahman and Al Hasan 2016). The other one is the relationship between two nodes. While modeling entities are common practice, modeling links are usually more complex (Nickel et al. 2016). Due to the different types of information, various methods have been presented (Bliss et al. 2014). Therefore, designing a model that can cope with such information is a challenging task, which is contrary to nature, and extract latent features.

One of the remarkable features of the social network link prediction is the constant change in size of the network which increases and decreases links and entities over time (Dunlavy et al. 2011; da Silva Soares and Prudêncio 2012; Heaukulani and Ghahramani 2013; Rossetti et al. 2015). The dynamic of such a network makes the study of these graphs a challenging task. Moreover, a large dynamic network may be complicated by the multi relational data (Davis et al. 2011; Yang et al. 2012; Sarkar et al. 2014; Garcia-Duran et al. 2016). Mostly, social networks are heterogeneous and dealing with multiple links and nodes may be intricate. The sparsity of linked data introduces another challenge(Getoor and Diehl 2005; Nguyen and Mamitsuka 2012; Zhai and Zhang 2015). When a network is sparse, it is sensitive to noise and in a dynamic and heterogeneous network, noise rate changes abruptly over time. Over a period of time nonlinear transformations are commonly seen in dynamic networks with seasonal fluctuations (Sarkar et al. 2012; Li et al. 2014b). Catching such nonlinearity is expensive and time-consuming (Zhu et al. 2016b). The wide range of methods is presented to overcome some of the mentioned challenges. However, choosing an appropriate method could be challenging because there are problems in social network link prediction. With regard to this, a triplex analytical framework, which classifies link prediction approaches was introduced. Moreover, the proposed framework evaluates each category based on our presented functional measure. Our triplex framework led to empirical and technical comparison of link prediction methods and provides a reference point for future researches by recording different techniques and methods concerning the topic. They were also evaluated by specifying the key measures to reflect on the capabilities and characteristics of the proposed methods.

The rest of the paper is organized as follows: In the next section, related works are reviewed. Section 3 describes the social network. In Sect. 4, the formal definition of the link prediction problem is presented. Section 5 shows the general architecture of link prediction in the social network. In Sect. 6, the proposed framework is presented. Section 7 includes the conclusion and future works.

2 Previous related review study

In order to deal with the challenging task of link prediction, a considerable amount of literature have attempted to exploit the relational and temporal nature of data, so as to demonstrate improved performance by compounding related sources of information in their modeling framework. An earlier study on predicting links in social networks is that proposed by Liben-Nowell and Kleinberg (2007). Their model focuses on the unsupervised approach, most of which either generate scores based on node neighborhoods or path information. They examined various similarity indices, including common neighbors, preferential attachment, Adamic-Adar, Jacard, SimRank, hitting time, rooted PageRank and Katz. They found that these similarity indices work well as compared to a random predictor. Later, Al Hasan et al. (2006) expanded this work in two directions. They affirmed that using social network data as well as graph topology can significantly improve the prediction result. Further, they used various similarity indices as entries of feature vector in a supervised learning setup and the link prediction problem was mapped as a binary classification task. Since then, the supervised classification approach has been popular in various other works in link prediction (Bilgic et al. 2007; Wang et al. 2007; Doppa et al. 2009).

Lü and Zhou (2011) classified link prediction approaches into three main categories which include recent approaches such as: (1) similarity based approach, (2) Maximum Likelihood Methods, (3) Probabilistic Models. They reported that an obvious drawback of the maximum likelihood methods is that it is very time consuming; while the probabilistic model will optimize a built target function to establish a model which can best fit the observed data. It is noteworthy that they emphasized on studies conducted by statistical physicists.

Al Hasan and Zaki (2011) investigated feature-based link prediction, probabilistic models, graphical models and linear algebraic models as the most commonly used approaches in link prediction. However, with the exception of Taskar et al. (2003), most of these models were designed for homogeneous networks that consider the same type of nodes and the same type of links in the network. Another line of research, Wang et al. (2015) surveys current trends/methods in link prediction problem. The main topics covered by Wang et al. (2015) include latest link prediction techniques, link prediction applications and active research groups.

In conjunction with all the above studies, a comprehensive classification and evaluation of link prediction approaches based on the introduced evaluation criteria, is presented. This study covers more recent link prediction works, which earlier surveys (which are at least 5 years old) did not cover.

Notation

Before proceeding, let us define our mathematical notation. Scalars are denoted by lower case letters such as c; column vectors are denoted by bold lower case letters such as $\mathbf{a}$; matrices are denoted by bold upper case letters such as $\mathbf{A}$; and tensors are denoted by bold upper case letters with an underscore such as $\underline{\mathbf{A}}$ (mostly the order of tensor is 3, $N_1 \times N_2 \times N_3$) and the (i, j, k)'th element by a_{ijk} (which is a scalar).

3 Social networks

Formally, social networks include a set of entities with regard to certain criteria known as network entities. The other important elements for the network are relations and interactions or in general, Links (Brandes and Wagner 2004; Lü and Zhou 2011). A data structure in a graph depends on a number of links between two nodes. It means that, in single-relational data, there is just one type of link among network and in multi-relational data, the types of links are more than one (Yang et al. 2012; Litwin and Stoeckel 2016). The study of social network is both a knowledge representation and a how supply. Knowledge representation is on what the graph structure is. Although the overall network structure is a graph, due to the complexity and variety of social networks, it arises as a separate field. Some social networks involving one single type of nodes and links can be represented as homogeneous networks, while some other social networks have multiple links and node type (Keyvanpour and Azizani 2012). A graph of homogeneous network is defined as:

$$\begin{aligned} G_{homo} &= \langle V, E \rangle \\ V &= \{v \in V \mid \forall i, j \;\; v_i \;\; identical\ to\ v_j\} \\ E &= \{e \in E \mid \forall i, j \;\; e_i \;\; identical\ to\ e_j\} \end{aligned} \tag{1}$$

where V is the node set and E is the link set. A heterogeneous network is defined in a similar way, but it contains multiple kinds of nodes and links:

$$\begin{aligned} G_{hetro} &= \langle V, E \rangle \\ V &= \cup_i V_i \\ E &= \cup_i E_i \end{aligned} \tag{2}$$

where V is the node set that contains the union of different node type sets and E is the union of the heterogeneous link sets.

Together with studying the structure of social networks, supplying network is a field of interest in this domain. In almost every social network, all the entities are not available at first and by the time of adding to it (da Silva Soares and Prudêncio 2012; Nasim and Brandes 2014). A dynamic network is one in which the rates of changes in same time intervals are always changing (Kuhn and Oshman 2011; Aggarwal and Subbian 2014). In other words, nodes and edges shrink and grow quickly. Such a dynamic network can typically be modeled as a dynamic graph $G = \langle V, E \rangle$, where V is a set of nodes, and $E\colon N^+ \to 2^{V \times V}$ is a dynamic edge function assigned to each round $r \in N^+$ a set of edges $E(r)$ for that round. A round occurs between two times; round $r \in N^+$ occurs between time $r-1$ and r. $G(r) = \langle V, E(r) \rangle$ is instantaneous in round r. In the literature, such dynamic graphs have also been termed evolving graphs (Kuhn and Oshman 2011; Aggarwal and Subbian 2014). Naturally, what can be achieved in a dynamic network largely depends on how the dynamic set of edges is chosen. In Fig. 1, the overall view of social networks is presented.

4 Link prediction: problem definition

Link prediction is a the task of predicting the relations and interactions in a network. Machine learning techniques are proposed for the prediction of unknown links using the known links in a graph as training data. Independent of the procedure, predicting unknown links falls into two categories in accordance with the linked data: (i) Missing Link Prediction and (ii) Future Link prediction (Liben-Nowell and Kleinberg 2007; Lü and Zhou 2011; Dunlavy et al. 2011).

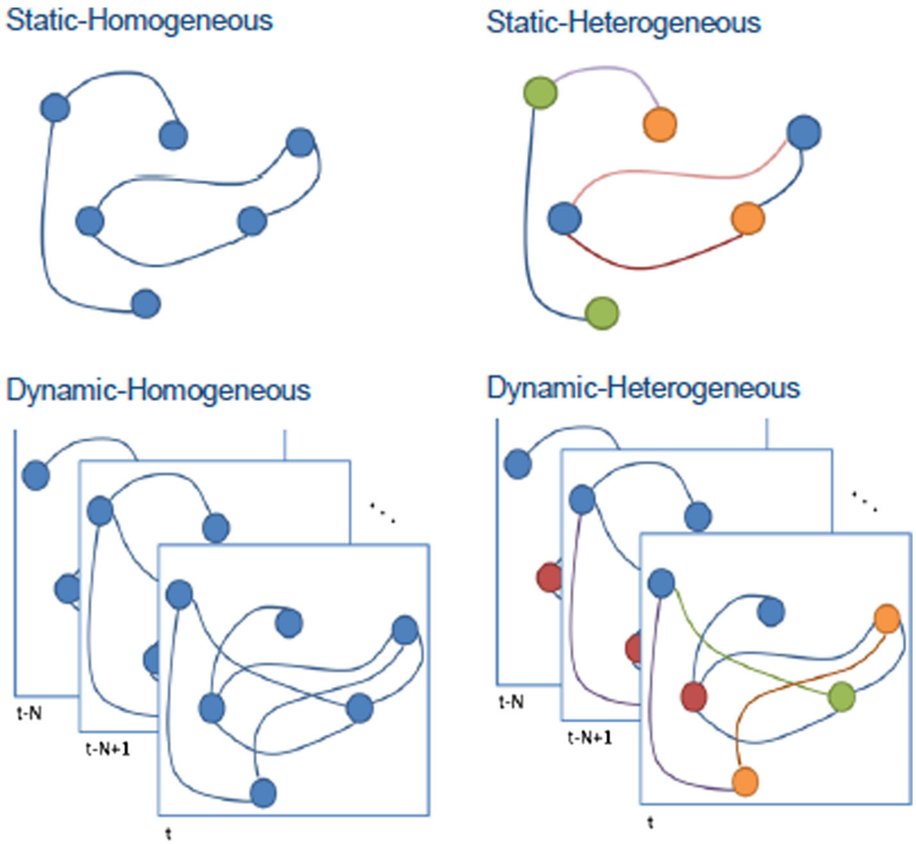

Fig. 1 Social networks types

4.1 Missing link prediction

Consider graph $G = \langle V, E \rangle$, where V is the set of nodes and E is the set of edges, $E \subseteq (V \times V)$ that each edge $e = (u, v) \in E$ represent a link between u and v. Let us denote the subgraph $G[k]$ consisting of all edges that are available (also called training graph) and $G[k']$ consisting of all missing edges (called test graph)(Lü and Zhou 2011;). In other words, the union of both subgraphs, $G[k]$ and $G[k']$ is equal to the original graph and the Intersection of these two subgraphs is empty (3).

$$\begin{aligned} E_k \cup E_{k'} &= E \\ E_k \cap E_{k'} &= \emptyset \end{aligned} \tag{3}$$

For an algorithm to access the subgraph $G[k]$, it must yield a list of edges not presented in $G[k]$ that are predicted to appear in the $G[k']$. Figure 2, describes a simple view of missing link prediction in a homogeneous social network.

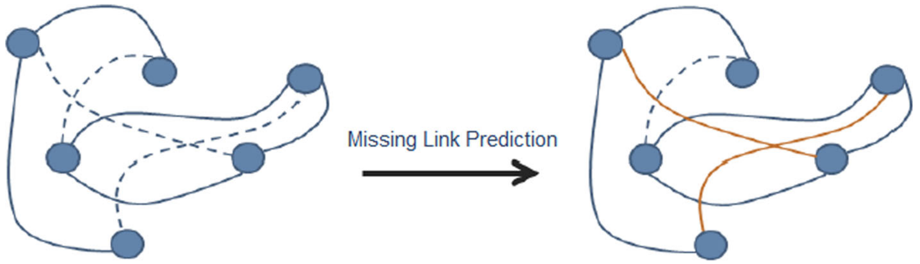

Fig. 2 An overview of the incoming and outgoing missing link predictor. Dashed edges in the left graph are missed and the model predictor cloud forecasts two of them as shown in the right

4.2 Future link prediction

Future link prediction can be grouped into the following categories: (i) periodic link prediction and (ii) non-periodic link prediction. The former type of link prediction considers the dynamic nature of the graph as a key feature due to the prediction. On the other hand, the latter type dose not discover changes over time but it focuses on the current state of the network. The look-out of future Link prediction is shown in Fig. 3.

- **Periodic link prediction**
  Given a series of snapshots $\{G_1, G_2, \cdots, G_t\}$ of an evolving graph $G_t = \langle V, E_t \rangle$ in which each $e = (u, v) \in E_t$ represent a link between u and v that took place at a particular time t (Brandes and Wagner 2004; Miller et al. 2009; Tylenda et al. 2009). We seek to predict the most likely link state in the next time step G_{t+1}. In almost every method analyzed in this paper, it is assumed that nodes V remain the same across all time steps but edges E_t changes for each time t. Also some new links were predicted, while some of the previous links were removed. In other words, the goal is to properly predict the next state graph snapshot. Also in dynamic heterogeneous network, the graph is defined as:

$$G = \langle V_1 \cup V_2 \cup \cdots \cup V_M, \{E_{1,t-n} \cup E_{2,t-n} \cup \cdots \cup E_{N,t-n}, \cdots, E_{1,t} \cup E_{2,t} \cup \cdots \cup E_{N,t}\}\rangle \quad (4)$$

  where $V_u,\ u \in M$ represents the set of nodes of the same u and $E_j,\ j \in N$ represent the set of links with type j (Yang et al. 2012).
- **Non-periodic link prediction**
  In the non-periodic type, instead of having a series of snapshot of an evolving graph, we have one snapshot of the current state of graph G_t. More formally, let $G = \langle V, E_t \rangle$, where V is the set of nodes and E denote its edges, $E \subseteq (V \times V)$. Considering two subgraphs corresponding to the current state, G_t and future G_{t+1} that $E_t \cup E_{t+1} = E$, $E_t \cap E_{t+1} = \emptyset$. Due to current state, we seek to predict the next time step of graph G_{t+1} (Al Hasan et al. 2006 ; Liben-Nowell and Kleinberg 2007). It is noteworthy that most times, this category refers to a heterogeneous network because of the complexity of multi-relational data. It is possible to expand these definitions in a heterogeneous network. This could be achieved just by considering the graph as:

$$G = \langle V_1 \cup V_2 \cup \cdots \cup V_M, E_1 \cup E_2 \cup \cdots \cup E_N\rangle \quad (5)$$

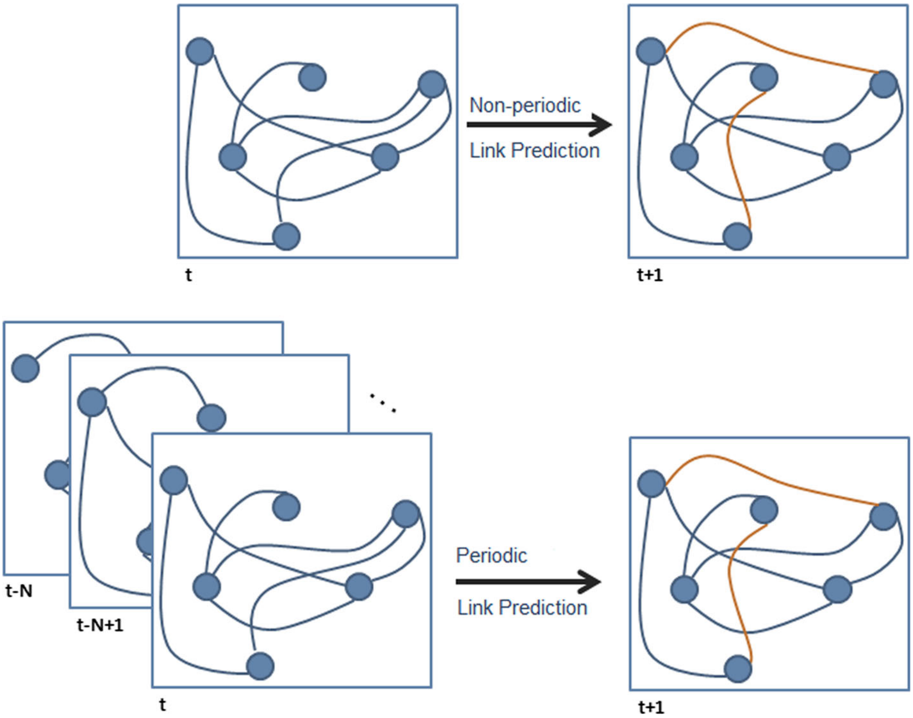

Fig. 3 Future Link prediction. The one at the bottom shows a periodic link prediction, where the inputs are the snapshot of the graph in different time intervals. The one at the top shows the non-periodic link prediction where the input is just a snapshot of the current

5 Link prediction in social network

Link prediction is one of the important tasks of link mining in a data mining community (Getoor and Diehl 2005). Therefore, many methods have been created to determine a better way of predicting links (missing links or future links). The most fundamental issue in social link prediction is features(Al Hasan et al. 2006; Nguyen and Mamitsuka 2012; Li et al. 2014a). These features are obtained from two information resources. One is the information from the entities of networks such as nodes. The other kind of information used to predict links is the structures (topologies) of the networks themselves, with or without node information (Lichtenwalter et al. 2010; Nguyen and Mamitsuka 2012). In other words, the structure of social linked-data comprises latent information that can be used in many link-mining tasks, especially in link prediction.

In social networks, link prediction consists of two principal phases: feature extraction and prediction (Al Hasan et al. 2006; Fire et al. 2011). It is noteworthy that in many methods, delimitation is not explicitly seen and often in the prediction phase, some features are extracted. As a result of these features and the previous ones, inferences are made. The general architecture for a link prediction system is shown in Fig. 4.

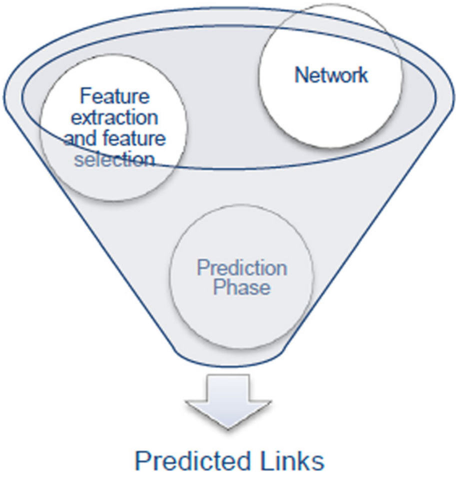

Fig. 4 General process of link prediction

6 Systemic analytical framework

In this section, a triplex analytical framework is presented and its usefulness in selecting an appropriate method is demonstrated. Since the structure of a social network is different, varieties of methods are presented to solve the link prediction problem. The aim of this analytical framework is to provide a platform which analyzes the most common and recent methods. It classifies these methods in a coherent way. Also, in this framework, with the introduction of evaluation criteria, the analysis and comparison of this classification was done. This analysis and comparison resulted in a better understanding of the approaches because the potential advantages of the methods when compared with each other is often unknown. This has a direct impact on performance in a particular status and lack of such framework poses challenges.

This paper explored a vast range of link prediction methods in social networks. Therefore, the structure of the framework is built based on three components. The aim of this study was to illustrate the different types of techniques and identify their fairness and effectiveness. The proposed framework is comprised of three components:

1. Classification of Link Prediction Approaches
2. Evaluation Criteria
3. Analytical evaluation

These three components have been described in details, as follows.

6.1 Classification of link prediction approaches

Classifying link prediction approaches, is the first component of the proposed framework. A review of the wide range of factors offered in this paper shows that link prediction methods can be classified as follows (Fig. 5). At the highest level, the methods are classified with learning based approaches and heuristic based approaches. First ,in the heuristic based approach, the prediction phase is done directly after determining the features (Liben-Nowell and Kleinberg 2007; Lichtenwalter et al. 2010; Sarkar et al. 2011). This group of algorithms compute a similarity score between a pair of nodes. The second one is the learning based models which extract patterns from the input data. This input data can be a preprocessed feature vector, an

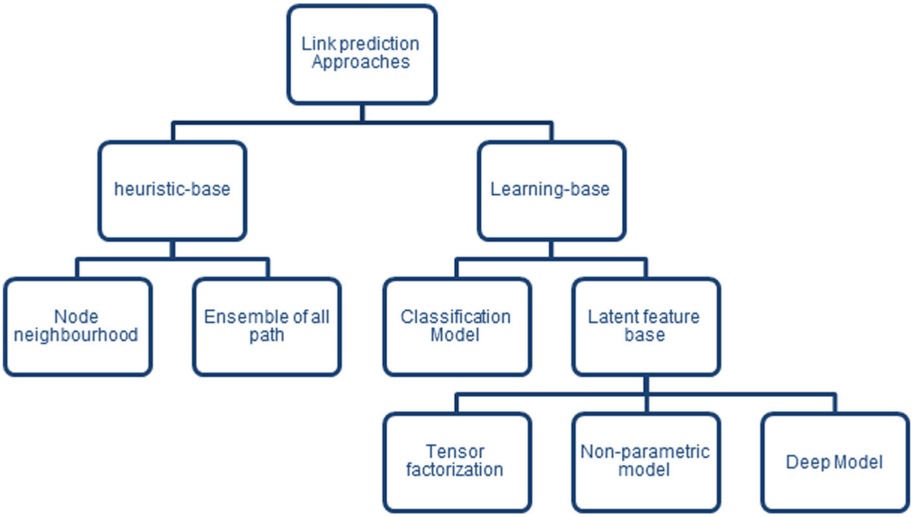

Fig. 5 Link prediction approaches

explicit graph structure or both of them(Wang et al. 2011; Yang et al. 2012; Tylenda et al. 2009; Ermiş et al. 2012). This decomposition is because learning algorithms itself extracts a pattern from data to predict unseen links for further heuristic based approaches prediction link by known similarities from the graph structure.

In the following, each approach is described in detail. We began with the heuristic based approach and subcategories and then reviewed the learning based model variations.

6.1.1 Heuristic based approaches

This approach includes methods that they attempt predict links via heuristic information. This information captures the shared characteristics or contexts of two nodes. Due to this captured information, heuristic based approach is categorized into classes: node neighborhood and ensemble of all paths (Liben-Nowell and Kleinberg 2007; Lü and Zhou 2011; Sarkar et al. 2011; Feng et al. 2012). For Dynamic networks, although most methods in this approach do not consider the dynamic nature of the network, the result is better than predicting by accident. Methods based on neighborhood, which consider local indicators and paths based methods, are known as global indicators (Lü and Zhou 2011; Bliss et al. 2014). In these techniques, a score is assigned to each unobserved link and the top K links with the highest score are predicted. The superiority of these algorithms is no domain knowledge necessary to compute similarity score (Lichtenwalter et al. 2010). Furthermore, it is applicable in discovering homophily pattern. Homophily pattern is the tendency of entities to be related to other entities with similar characteristics (Nickel et al. 2016). The heuristic based approach is also known as similarity-based approach.

- **Node neighborhoods**
  Node neighborhood encodes information about the relative overlap between node neighborhoods (Liben-Nowell and Kleinberg 2007). It is expected that the more "*similar*"nodes are more likely to be a predicted link. As a result of the simplicity and having fewer parameters, it is used in many studies on link prediction. The other priority is that being a

general solution, it means it does not need a specific knowledge of the domain. Together with these positive aspects, it does not have the capability to explore evolutionary and non-linear patterns (Dunlavy et al. 2011). Four popular node neighborhood indices are explained below.

(a) Common neighbors It is a building block of many other approaches (Liben-Nowell and Kleinberg 2007; Li et al. 2014b) with mathematical expression as:

$$score_{CN}(x, y) = |\Gamma(x) \cap \Gamma(y)| \tag{6}$$

where $\Gamma(x)$ and $\Gamma(y)$ denote the neighbor set of nodes x and y, respectively.
(b) Jaccard coeffcient This similarity metric is mostly used in information retrieval. Jaccard Coeffcient is a normalized neighbor (7).

$$score_{JC}(x, y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|} \tag{7}$$

In fact, it defines the probability that a common neighbor of a pair of nodes x and y would be selected if the selection is made randomly from the union of the neighbor-sets of x and y. However, from the experimental results, Liben-Nowell and Kleinberg (2007) showed that the performance of Jaccard coefficient is worse in comparison with the number of common neighbors.
(c) Adamic/Adar Adamic and Adar (Adamic and Adar 2003) proposed this score as a similarity index between two web pages. For link prediction, Liben-Nowell and Kleinberg (2007) customized these indices as below, where the common neighbors are considered as features.

$$score_{AA}(x, y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log |\Gamma(z)|} \tag{8}$$

Conceptually, Adamic/Adar gives more weight to the neighbors that are not shared with many others. From the reported results of the existing works on link prediction, Adamic/Adar works better than the previous two metrics.
(d) Preferential attachment The basic premise is that the probability that a new edge has node x as an endpoint is proportional to $\Gamma(x)$, the current number of neighbors of x. The probability that this new link will connect x and y is proportional to:

$$score_{PA}(x, y) = |\Gamma(x)|.|\Gamma(y)| \tag{9}$$

- **Ensemble of all paths**
Paths between two nodes are another heuristic that can be used for computing similarities between node pairs. Exploring the graph is both a strength and a weakness. Exploring through graph is a simple learning inside, but in a large scale network, it is time-consuming. Therefore, it is used as a basic idea of the other methods (Rossetti et al. 2015). A brief introduction of the four important global indices is given as follows.

(a) Katz Katz is a stated method among path-based approach that counts all paths between two nodes (Dunlavy et al. 2011). The paths are exponentially damped by length that can give mode weights to shorter paths. This measure is defined as follows, where $paths^{l}_{x,y}$ is the set of all paths from x to y which $l, \beta > 0$ (10) and the very small β will cause Katz metric much like CN metric because long length contributes very little to the final similarities (Liben-Nowell and Kleinberg 2007; Lü and Zhou 2011).

$$score_{katz}(x, y) = \sum_{l=1}^{\infty} \beta^l \cdot |path^l_{x,y}| = \beta A + \beta^2 A^2 + \beta^3 A^3 + \cdots \tag{10}$$

(b) Hitting time For two vertices, x and y in a graph, the hitting time, $H_{x,y}$ defines the expected number of steps required for a random walk starting at x to reach y. Shorter hitting time shows that the nodes are similar, so they can be unseen links (Al Hasan and Zaki 2011). For undirected graph it can be considered as:

$$score_{HT_{ugraph}}(x, y) = H_{x,y} + H_{y,x} \tag{11}$$

Hitting time metric is easy to compute by performing some trial random walks. On the downside, its value can have high variance; hence, prediction by this feature can be poor. Due to the scale free nature of a social network some of the vertices may have very high stationary probability (π) in a random walk; to safeguard against it, the hitting time can be normalized by multiplying it with the stationary probability of the respective node, as shown below:

$$score_{NHT_{ugraph}}(x, y) = H_{x,y}.\pi_y + H_{y,x}.\pi_x \tag{12}$$

(c) Rooted Pagerank Similarity score between two vertices x and y can be measured as the stationary probability of y in a random walk that returns to x with probability $1 - \beta$ in each step, moving to a random neighbor with probability β is pagerank for link prediction (Feng et al. 2012). Rooted Pagerank is a modification of Pagerank. Let D be a diagonal degree matrix with $D[i, i] = \sum_j A[i, j]$. Let, $N = D^{-1}A$ be the adjacency matrix with row sums normalized to 1. Then,

$$score_{RPR}(x, y) = (1 - \beta)(I - \beta N)^{-1} \tag{13}$$

(d) SimRank SimRank is defined in a self-consistent way, according to the assumption that two nodes are similar if they are connected to similar nodes.

$$score_{SR}(x, y) = \begin{cases} 1 & \text{if } x{=}y \\ \gamma . \frac{\sum_{a \in \gamma(x)} \sum_{b \in \gamma(y)} simRank(a,b)}{|\Gamma(x)|.|\Gamma(y)|} & \text{otherwise} \end{cases} \tag{14}$$

where $\gamma \in [0, 1]$ is the decay vector. The SimRank can also be interpreted by the random walk process, that is $score_{SR}(x, y)$ to measure how soon two random walkers, respectively starting from nodes x to y, are expected to meet at a certain node(Lü and Zhou 2011).

The brief description of these similarity indices are listed in Table 1. In the description column, the weaknesses and strengths of each index are demonstrated.

6.1.2 Learning-based approaches

At this level of abstraction, a model is presented which is learned with a given feature, extract patterns and eventually lead to link prediction (Nickel et al. (2016)). Conceptually, learning based models aim at abstracting the underlying structure of the input graph, and predicting the unseen links by using the learned model (Al Hasan et al. 2006). This approach is classified into two: *Classification model* and *Latent-feature-based models*. The key idea behind this grouping is the type of learning model. Classification of model extract patterns from input data and learning a model for the prediction of missing/future links. The input data are the pre-process feature vectors where each entry is known like similarity index, external node

Table 1 Popular similarity indices in the link prediction. The set $\Gamma(x)$ consist of the neighbors of x in G_{train}

Methods	Score function	Description
1. Common neighbors	$\lvert\Gamma(x) \cap \Gamma(y)\rvert$	Reasonable result on most data set
		Not normalized
		Scalable
		Cold start
2. Jaccard coeffcient	$\frac{\lvert\Gamma(x) \cap \Gamma(y)\rvert}{\lvert\Gamma(x) \cup \Gamma(y)\rvert}$	Normalized
		Poor Performance in comparison with common neighbor
3. Adamic/Adar	$\sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log\lvert\Gamma(z)\rvert}$	Use minor information
		Works better than common neighbor and Jaccard coeffcient
4. Preferential attachment	$\lvert\Gamma(x)\rvert.\lvert\Gamma(y)\rvert$	Works well in evolutionary networks
		It has the east computational complexity
5. Katz	$\sum_{l=1}^{\infty} \beta^{l} \cdot \lvert path_{x,y}^{l}\rvert = \beta A + \beta^2 A^2 + \beta^3 A^3 + \cdots$	Unscalable
		Works well in weighted networks
		High computational complexity
6. Hitting time	$H_{x,y} + H_{y,x}$	Works well in directed networks
	$H_{x,y}.\pi_y + H_{y,x}.\pi_x$	Sensitive dependence on parts of the graph far away from x and y
		High variance
7. Rooted pagerank	$(1-\beta)(I-\beta N)^{-1}$	Random walks on a graph
		Effective for capturing various relations among vertices of graphs
		Well defined for any given graph
8. SimRank	$\begin{cases} 1 & \text{if } x{=}y \\ \gamma . \frac{\sum_{a \in \gamma(x)} \sum_{b \in \gamma(y)} simRank(a,b)}{\lvert\Gamma(x)\rvert.\lvert\Gamma(y)\rvert} & \text{otherwise} \end{cases}$	The similarity score considers both the graph topological and attribute based similarity.
		Unscalable
		Computational feasibility

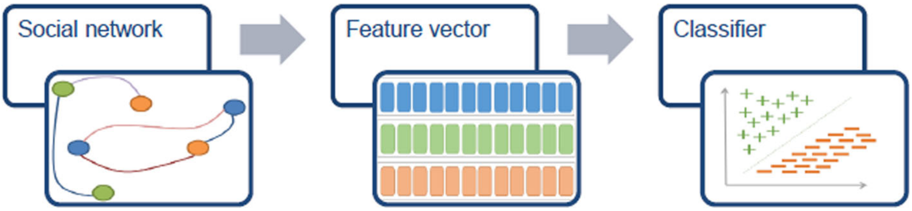

Fig. 6 The flowchart of classification model

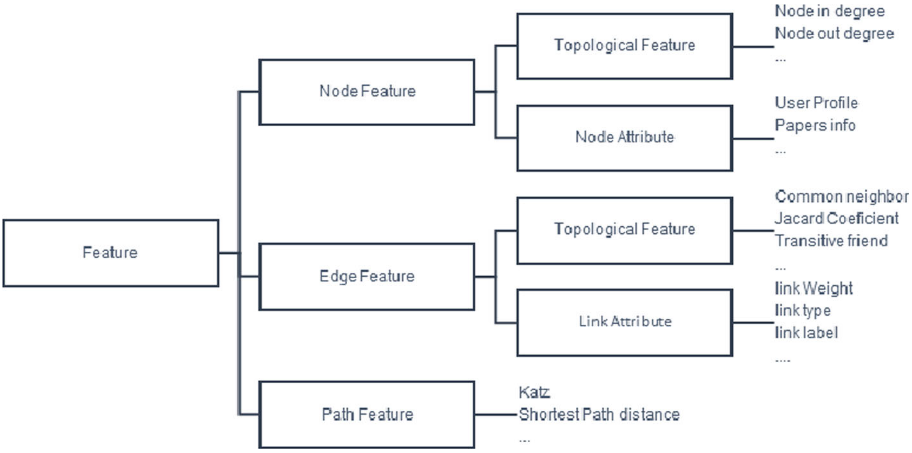

Fig. 7 Grouping most used features

or link information(Al Hasan et al. 2006; Liu et al. 2012; Fire et al. 2011). However, in the latent-feature-based model these pre-process feature vectors can be optional. Clearly, the latent-feature-based model, extracting latent features from input graph and learn a model. Meanwhile, this learning model can use external information analogous to social network data or pre-process feature vector. The following two items describe the classification model and latent-feature-based model.

- **Classification model**
  Almost every method that falls into this category is a supervised learning. After identifying the set of features which are key to supervised learning (Al Hasan et al. 2006; Backstrom and Leskovec 2011), the link prediction problem is mapped to a binary classification (Lee et al. 2013; Fig. 6). Although in the binary classification, it is important to predict both classes, in link prediction, the issue is to predict the missing link or future link. In a classification model for link prediction, researchers have used the supervised model, including the support vector machine (Bliss et al. 2014), decision trees (Wang et al. 2011), multi-layer perceptron (Al Hasan et al. 2006), supervised random walks (Backstrom and Leskovec 2011) and others. They found that the support vector machine performed most in the prediction of future links in the supervised model (Nguyen-Thi et al. 2015).
  Al Hasan et al. (2006) placed the feature under categories. Node Attribute and topological features are mostly obtained from heuristic-based methods. In other words, instead of directly predicting link, it is used as an entry of feature vector for learning a model. The

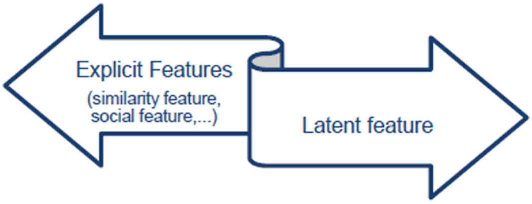

Fig. 8 Two different type of information can obtain from network

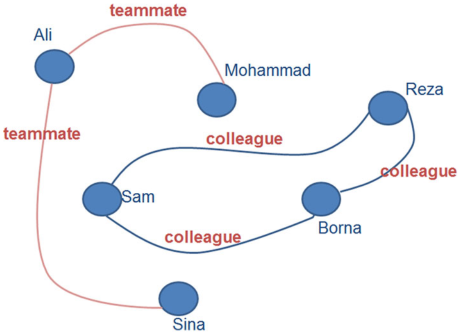

Fig. 9 Example of heterogeneous network

others refer to node attribute features like profile information in social networks. Figure 7 shows some of the properties in the feature vector. Although, just having a feature vector can predict link with any binary classification model, creating a valuable feature vector has an additional task.

- **Latent-feature-based model**

  An underlying assumption of latent-feature-based model is to build a model, which can discover latent features from the structure of the graph (Sarkar and Moore 2005; Miller et al. 2009; Sarkar et al. 2012; Sewell and Chen 2016). As mentioned earlier, there are two types of information that can be obtained from the network (Fig. 8). Nevertheless, the researchers in this domain believe that the structure of the graph and the combination of these two types of information have latent characteristic, which with the simple techniques, cannot be achieved, especially when capturing the dynamic characteristics or in heterogeneous network with multi-relational data (Miller et al. 2009; Bordes et al. 2014; Zhu et al. 2016b). The goal of latent feature based approach is to learn a model from observed links such as those that can predict the values of unobserved entries. The latent representation of each node corresponds to a point on the surface of a unit hypersphere. In the latent-feature-based model, each entity is associated with a vector $\mathbf{e}_i \in \mathbb{R}^{H_e}$, where $H_e \ll N_e$ (N_e is the number of entity in the graph). Each link is explained via latent features of entities. For instance, as show in Fig. 9 each node can be modeled via vectors

$$\mathbf{e}_{Ali} = \begin{bmatrix} 0.1 \\ 0.9 \end{bmatrix} \qquad \mathbf{e}_{Sina} = \begin{bmatrix} 0.15 \\ 0.85 \end{bmatrix} \qquad \mathbf{e}_{Mohammad} = \begin{bmatrix} 0.1 \\ 0.8 \end{bmatrix}$$

$$\mathbf{e}_{Sam} = \begin{bmatrix} 0.9 \\ 0.5 \end{bmatrix} \qquad \mathbf{e}_{Reza} = \begin{bmatrix} 0.92 \\ 0.35 \end{bmatrix} \qquad \mathbf{e}_{Borna} = \begin{bmatrix} 0.82 \\ 0.45 \end{bmatrix}$$

  where the component e_{i1} corresponds to the latent feature *proficient developer* and e_{i2} correspond to *being healthy*. Thus, Ali has a teammate connection, it can be inferred that he is healthier, or Sam has a colleague connection to Reza and Borna, so the latent feature *proficient developer* is higher for him. Note that, unlike this example, the latent features

that are inferred by the following models are typically difficult to interpret. The key intuition behind relational latent feature models is that the relationships between entities can be derived from interactions of their latent features(Sarkar and Moore 2005; Rastelli et al. 2016; Rahman and Al Hasan 2016; Sewell and Chen 2016). However, there are many possible ways to model these interactions, and many ways to derive the existence of a relationship from them. With the dimensionality constraint,the link prediction is efficient in both computational time and storage cost(Sarkar and Moore 2005; Nickel et al. 2016; Zhu et al. 2016b). In addition, varying the dimension of latent space offers an opportunity to fine-tune the compromise between computational cost and solution quality. The higher dimension leads to a more accurate latent space representation of each node, but also yields higher computational cost(Zhu et al. 2016b).
Several approaches have been presented to obtain these latent features. The following three items will introduce each approach in detail.

(a) Tensor factorization based
Significantly, tensor factorization is known as an approach for structured data in different learning contexts. The success of tensor factorization in link prediction problem is due to its high ability to model and analyze relational data (Dunlavy et al. 2011; Ermiş et al. 2012; Gao et al. 2011). Tensor based methods are usually in two (Matrix (Menon and Elkan 2011)) and three orders. In future link prediction, the third domain is considered as a different time snapshot. This approach has a reasonable ability to detect latent feature during the time. More formally, given a sequence of words:

$$z(i, j, t) = \begin{cases} 1 & \text{if node } i \text{ links to node } j \text{ at time } t. \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

which shows that the link from node i to j appeared at time t(Acar et al. 2009; Spiegel et al. 2011; Dunlavy et al. 2011; Yao et al. 2015; Zhu et al. 2016b; Han and Moutarde 2016). On the other hand, in heterogeneous networks with multi-relational data, the third dimension shows different types of links. It is most applicable in heterogeneous networks where links have a high dependency(Gao et al. 2011; Ermiş et al. 2012; Nickel and Tresp 2013b, a; London et al. 2013; Nickel et al. 2014; Krompaß et al. 2014). The third order tensor is used to define multi-relational data as follows:

$$z(i, j, k) = \begin{cases} 1 & \text{if } relation_k(node_i, node_j) \text{ is true.} \\ 0 & \text{otherwise.} \end{cases} \tag{16}$$

Factorization techniques like the CP (CanDecomp/Parafact) and Tucker model can be considered higher-order generalization of the matrix singular value decomposition (SVD)(Keyvanpour and Moradi 2014) and principal component analysis (PCA) (Kolda and Bader 2009 ; Spiegel et al. 2011). However, the CP model is more advantageous in terms of interoperability, uniqueness of solutions and determining the parameter (Spiegel et al. 2011). A three dimensional tensor $\underline{\mathbf{Z}}$ defined as $m * n * t$, its k-component CP factorization is defined as:

$$\sum_{k=1}^{k} \lambda_k a_k \circ b_k \circ c_k \tag{17}$$

Symbol $\circ$ stands for outer product, $\lambda_k \in \mathbb{R}^+$, $a_k \in \mathbb{R}^m$, $b_k \in \mathbb{R}^n$, $c_k \in \mathbb{R}^t$, where $k = 1, \cdots , K$. Each summand $(\lambda_k a_k \circ b_k \circ c_k)$ is called a component, each vector is

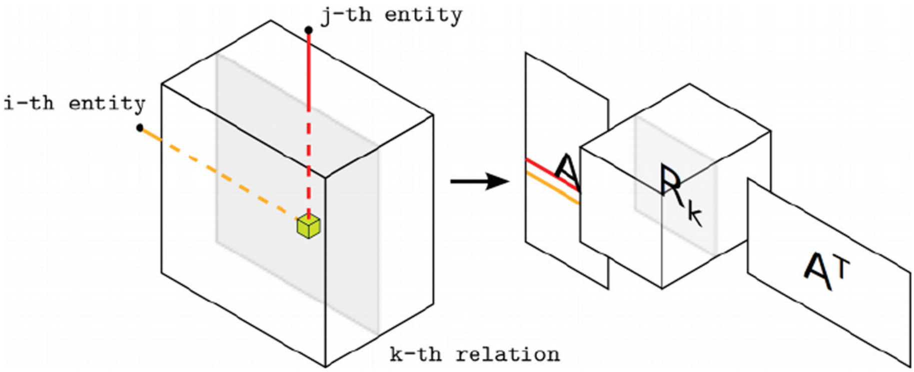

Fig. 10 Factorization of an adjacency tensor $\underline{\mathbf{X}}$ using RESCAL model. (Reproduced with permission from Nickel and Tresp 2013b)

called factor. RESCALL is one of the prominent latent factor models for relational learning(Nickel and Tresp 2013b), which factorizes an adjacency tensor $\underline{\mathbf{X}}$ into a core tensor $\underline{\mathbf{R}} \in \mathbb{R}^{r\times r\times m}$ and a factor matrix $\mathbf{A} \in \mathbb{R}^{n\times r}$ such that

$$\underline{\mathbf{X}} \approx \underline{\mathbf{R}} \times_1 \mathbf{A} \times_2 \mathbf{A} \tag{18}$$

Equation (18) can be equivalently specified as $x_{ijk} \approx \mathbf{a}_i^\top \mathbf{R}_k \mathbf{a}_j$ where the column vector $a_i \in R^r$ denotes the i-th row of $\mathbf{A}$ and the matrix $\mathbf{R}_k \in \mathbb{R}^{r\times r}$ denotes the k-th frontal slice of $\mathbf{R}$. Consequently, $\mathbf{a}_i$ corresponds to the latent representation of $entity_i$,while $\mathbf{R}_k$ models the interactions of the latent variables for $relation_k$ (Nickel et al. 2016; Fig. 10). Among the presented approach, Tensor factorization has a high capability of representation of multi-relational data (Ermiş et al. 2012; Spiegel et al. 2011), even the tensor can expand to higher order due to representing dynamic heterogeneous network. Nevertheless, as the network size is growing, the computational cost is grows speedily.

As shown by Narita et al. (2012), one significant obstacle in tensor factorization is that its prediction accuracy tends to be poor because observations made in real datasets are typically sparse. Several studies as such coupled tensor factorization (Yılmaz et al. 2011; Acar et al. 2011; Narita et al. 2012; Ermiş et al. 2012, 2015; Nakatsuji et al. 2016) have attempted to incorporate side information such as node attribute or social connection to improve prediction accuracy. To address couples tensor factorization, consider a third order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$ with some elements been observed and others remaining unobserved. Three non-negative matrices, $\mathbf{A}_1 \in \mathbb{R}^{+^{I\times I}}$, $\mathbf{A}_2 \in \mathbb{R}^{+^{J\times J}}$, $\mathbf{A}_3 \in \mathbb{R}^{+^{K\times K}}$, corresponds to one of three modes of $\underline{\mathbf{X}}$ and contain side information such as similarity or node attributes. Given these three matrices beside $\underline{\mathbf{X}}$ can fill unobserved parts of $\underline{\mathbf{X}}$ and overcome the sparsity problem.

Another approach to learning from graphs is based on matrix factorization, where, prior to the factorization, the adjacency tensor $\underline{\mathbf{Y}} \in \mathbb{R}^{N_e\times N_e\times N_t}$ is reshaped into matrix $\mathbf{Y} \in \mathbb{R}^{N_e^2\times N_t}$ by associating rows with node-node pairs (e_i, e_j) and columns with time t or relation r_k (Jiang et al. 2012; Riedel et al. 2013). Unfortunately, these formulations lose information as compared to tensor factorization. For instance, if each node-node pair is modeled via a different latent representation, the information that the relationships y_{ijk} and y_{pjq} share the same object is lost. It also leads to an increased memory complexity, since a separate latent representation is computed for each pair of entities.

(b) Nonparametric model

Using nonparametric trick is another type of latent feature model. In this model, methods mostly use Bayesian nonparametric methods to discover discriminative latent features and automatically infer the unknown social dimension (Wang et al. 2007; Miller et al. 2009; Yu et al. 2009; Cao et al. 2010; Kim et al. 2013; Zhu et al. 2016a). Basically, each entity is described by a set of binary features (Schmidt and Morup 2013). Nonparametric models allow simultaneous inferring of the number of latent features at the same time (Sarkar et al. 2012). On the other hand, some nonparametric models are based on kernel. These kernel based models include kernel regression, compute kernel similarities between query and all members of the training set(Collomb and Härdle 1986; Yu et al. 2006; Nguyen and Mamitsuka 2011; Sarkar et al. 2012).

Miller et al. (2009) introduced a basic Nonparametric model. They showed that each entity is described by a set of binary features and there is no priority for each of them. The probability of having a link from one entity to another is entirely determined by a combined effect of all pairwise feature interactions. If there are K features, then $\mathbf{Z}$ will be the $N \times K$ binary matrix where each row corresponds to an entity and each column corresponds to a feature such that $z_{ik} \equiv Z(i,k) = 1$ if the i^{th} entity has feature k and $z_{ik} = 0$ otherwise. The model has a real value weight matrix $(K \times K)$ where $w_{kk'} \equiv W(k,k')$ is the weight that affects the probability of occurrence of a link from entity i to j if entity i has feature k and entity j has feature k' .

It is assumed that links are independently conditioned on $\mathbf{Z}$ and $\mathbf{W}$, and that only the features of entities i and j influence the probability of a link between those entities. This defines the likelihood:

$$\Pr(\,Y \mid Z\,,\ W\,) \;=\; \prod_{i,j} \Pr\,(\,y_{ij} \mid Z_i Z_j\,,\ W\,) \tag{19}$$

where the product ranges over all pairs of entities. Given the feature matrix $\mathbf{Z}$ and weight matrix $\mathbf{W}$, the probability that there is a link from entity i to entity j is given as:

$$\Pr(\,y_{ij} = 1 \mid Z\,,\ W\,) = \sigma(\,Z_i W Z_j^{\top}\,) = \sigma\left(\sum_{k,k'} z_{ik}\, z_{jk'}\, w_{kk'}\right) \tag{20}$$

where $\sigma(.)$ is a sigmoid function that transforms values on $(-\infty, +\infty)$ to $(0, 1)$.

Nonparametric models have high ability to explore evolutionary patterns especially seasonal fluctuations. It is noteworthy that, this effectiveness is just compared to the heuristic based methods and not to the other latent feature based model (Sarkar et al. 2012, 2014; Zhu et al. 2016a). Among the learning based models, the nonparametric model is the fastest. Reason being that it has no or a few parameters. On the other hand, most of the methods use LSH implementation to make them faster. LSH or locality sensitive hashing is often used in database for table lookups or retrieving matching items from a large database (Sarkar et al. 2012). This model is also known as probabilistic model (Wang et al. 2007; Clauset et al. 2008; Schmidt and Morup 2013).

(c) Deep model

Due to the significant result of deep learning approach in computer vision, speech recognition, and natural language processing (Bengio et al. 2013; Li Deng 2014;

Goodfellow et al. 2016), the researchers were motivated to use a deep model in link prediction task (Socher et al. 2013; Liu et al. 2013; Perozzi et al. 2014; Li et al. 2014a, b; Zhai and Zhang 2015). In general, deep learning is a set of algorithms in machine learning that performs learning tasks in multiple levels, corresponding to different levels of abstraction. It typically uses artificial neural networks. The levels in these learned statistical models correspond to distinct levels of concepts, where higher-level concepts are defined from lower level ones, and the same lower-level concepts can help to define many higher-level concepts (Bengio et al. 2013; Goodfellow et al. 2016).

Deep model is new to link prediction. Multiple levels of representation help to detect nonlinear latent relationship (Li et al. 2014a, b; Liu et al. 2013). Although most of the methods using a deep neural network, Perozzi et al. (2014) and Grover and Leskovec (2016) are presented as deep random walk due to a notation of deep model. In fact, for the use of deep model, three different ideas have been used in this domain and they are:

- Using multiple levels of RBM as a building block of the deep model. The Restricted Boltzmann machine is a building block of many deep models. The reason for its success is that it supports distributed representation, has associative memory and the inference is easily done (Hinton et al. 2006). In this case, in order to use it for relational data, some modifications are applied (Liu et al. 2013; Li et al. 2014a, b). A RBM is a neural network that contains two layers. It has a single layer of hidden units that are not connected with each other. The hidden units have undirected, symmetrical connections to a layer of visible units. To each unit, including both hidden units and visible units, there is a bias in the network. The value of the visible and hidden units are often binary or stochastic units (assume 0 or 1 based on probability)(Hinton et al. 2006). When inputing a vector $\mathbf{v}(v_1, v_2, \cdots, v_n)$ to the visible layer, the binary state, h_j of each hidden unit is set to 1 with a probability given by:

$$p(h_j = 1|v) = \sigma(b_j + \sum_{i}^{n} v_i w_{ij}) \tag{21}$$

  where $\sigma(x) = 1/(1 + e^{-x})$, b_j is the bias of the hidden unit j.
  When input a vector $\mathbf{h}(h_1, h_2, \cdots, h_m)$ to the hidden layer, the binary state, v_i of each visible unit is set to 1 with probability by

$$p(v_i = 1|h) = \sigma(a_i + \sum_{j}^{n} h_j w_{ij}) \tag{22}$$

  where a_i is the bias of visible unit i.
  Liu et al. (2013) utilizes the Deep belief network(stack of Restricted Boltzmann Machines) to study the model used for unsupervised link prediction. They provide original link feature vectors as the first RBM's visible units' input and use the top RBM's hidden units' output as their represented features(Fig. 11a). After training the top RBM, each possible label is tried in turn with a test vector(Fig. 11b).
  Li et al. (2014b) introduces ctRBM, which inherits the advantages of the Restricted Boltzmann Machine family for predicting temporal link. It contains distributed hidden states which imply that it has an exponentially large state space to manage the complex nonlinear variations. It has conditional indepen-

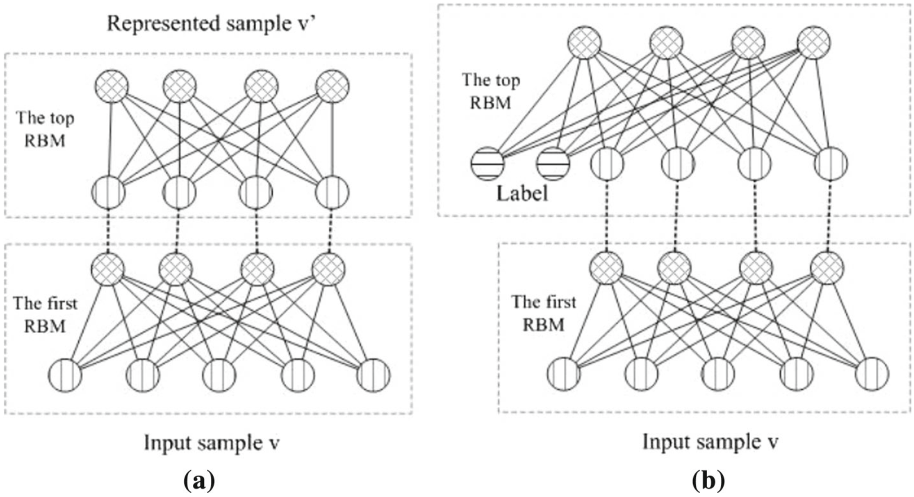

Fig. 11 **a** Left is DBN structure for unsupervised link prediction and feature representation. **b** Right is DBN structure for link prediction. (Reproduced with permission from Liu et al. 2013)

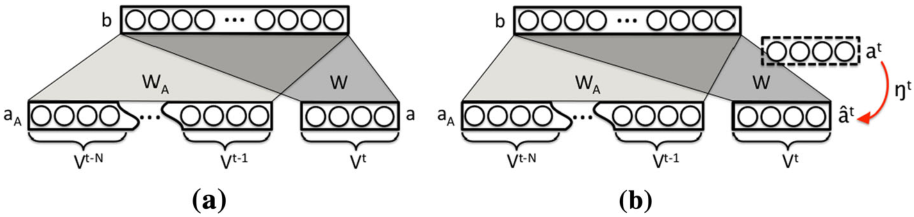

Fig. 12 **a** Left Restricted Boltzmann Machine with temporal information, where N is the window size. **b** Right The summarized neighbor influence η^t is integrated into the energy function as adaptive bias. (Reproduced with permission from Li et al. 2014b)

dent structure that makes it easy to plug in external features. Each node in that model has two types of directed connection to the hidden variables: temporal connection from nodes to historical observations and neighbor connections from the expectation of its local neighbors' prediction (Fig. 12).

- Since the deep model has an outstanding result in natural language processing and data in that field is relational, this prompted researchers to draw inspiration from the ideas in the field for link prediction (Perozzi et al. 2014). Link prediction algorithm such as deepwalk(Perozzi et al. 2014) and node2vec (Grover and Leskovec 2016)truncated random walks to learn latent representations by treating walks as the equivalent of sentences. They produce a vector representation for each entity as latent representation by word2vec models (Mikolov et al. (2013)). These vector representations of entities carry semantic meanings. The inputs of the model are sequences of nodes from the underlying network and turn a network into an ordered sequence of nodes (Perozzi et al. 2014; Tang et al. 2015; Grover and Leskovec 2016; Zhang et al. 2016). in this model, the algorithm iterates over all possible collections in the random walk. In fact, this approach generates representation of social networks that are low-dimensional, and exist in a continuous vector space. Its representation encodes the latent feature of nodes.

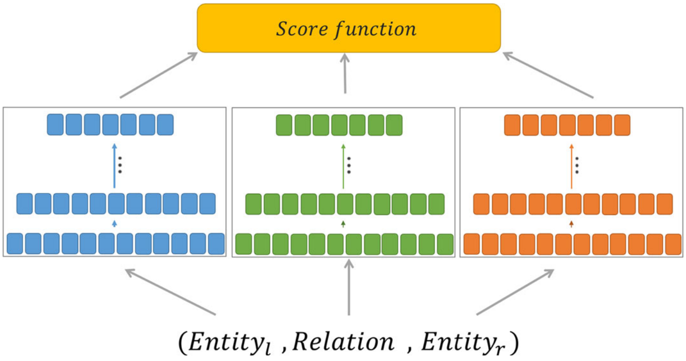

Fig. 13 Perspective of deep embedded model

- Creating a new embedded neural network that is compatible with the nature of relational data. In these embedding models each relation is defined as a triple:

$$(h, l, t)\ \ ,\ \ h, t \in E, l \in L \tag{23}$$

It is composed of two entities h,t and a relation l. Each model assigns a score to a triple using a score function which measures how likely it is that the triple is correct. Entities are represented by a low dimensional vector the embedding and the relation acts as an operator on them (Socher et al. 2013; Fig. 13). Both embedding and operators define a score function that is learned so that triples observed in knowledge bases have a higher score than the unobserved ones. In other words, link prediction is a complementary task in knowledge base completion (Bordes et al. 2014; Socher et al. 2013). Table 2 summarizes some popular scoring functions in the literature. In addition, Table 2 shows the transformation of each score function and the parameters.

For a more efficient representation, at the end of the Sect. 6.1 this classification together with related benefits and requirement is summarized in Table 3. In this table, the most important points and technical aspects are illustrated. It is noteworthy that the mentioned approaches in the table are in one-to-one correspondence with Fig. 5.

6.2 Evaluation criteria

The second component of the proposed framework is the introduction of evaluation criteria in the link prediction. Evaluation criteria plays a significant role in evaluating the effectiveness and weakness of an algorithm. In fact, there are evaluation criteria that determine the quality and validity of a method. As long as evaluation criteria are being discussed, quantitative criteria always come to mind. However, it is noteworthy that the evaluation criteria do not end only on quantitative criteria, and the qualitative criteria are also determinants of the validity of an algorithm. Qualitative criteria cannot measure and represent with a number or a curve. These metrics are derived or inferred from two or more parameters and is often expressed in comparative terms. In the following, each quantitative and qualitative criterion is introduced and each type is expressed. The purpose of this section is to provide an overview of all the evaluation criteria that can be used to examine the validity, quality and capability of a method.

Table 2 Comparisons among several multi-relational models in their scoring functions

Models	Transformation	Score function $s(h, l, t)$	Parameters	Pros&Cons
Bilinear (Jenatton et al. 2012)	Bilinear	$\mathbf{h}\mathbf{W}_l\mathbf{t}$	$\mathbf{W}_l \in \mathbb{R}^{d\times d}$	Incorporates the interaction of two entity vectors in a simple and efficient way Restricted in terms of expressive power and number of parameters by the entity vectors.
NTN (Socher et al. 2013)	Linear&Bilinear	$\mathbf{u}_l^\top f(\mathbf{h}^\top \underline{\mathbf{W}}_l^{[1:k]}\mathbf{t} + \mathbf{V}_l \begin{bmatrix}\mathbf{h}\\ \mathbf{l}\end{bmatrix} + \mathbf{b}_l)$	$f = \tanh$, $\underline{\mathbf{W}}_l^{[1:k]} \in \mathbb{R}^{d\times d\times k}$, $\mathbf{V}_l \in \mathbb{R}^{k\times 2d}$, $\mathbf{u}_l \in \mathbb{R}^k$, $\mathbf{b}_l \in \mathbb{R}^k$	Complex Has a more general parameterization Pre-trained vectors tend to capture syntactic and semantic information
SME (Bordes et al. 2014)	Linear	$(\mathbf{W}_{l1}\mathbf{h}^\top + \mathbf{W}_{l2}\mathbf{l}^\top + \mathbf{b}_l^\top)^\top(\mathbf{W}_{r1}\mathbf{t}^\top + \mathbf{W}_{r2}\mathbf{l}^\top + \mathbf{b}_r^\top)$	$\mathbf{W}_{l1}, \mathbf{W}_{l2}, \mathbf{W}_{r1}, \mathbf{W}_{r2} \in \mathbb{R}^{p\times d}$, $\mathbf{b}_l, \mathbf{b}_r \in \mathbb{R}^p$	Constrained version of RESCAL
	Bilinear	$((\underline{\mathbf{W}}_l \overline{\times}_3 \mathbf{l}^\top)\mathbf{h}^\top + \mathbf{b}_l^\top)^\top((\underline{\mathbf{W}}_r \overline{\times}_3 \mathbf{l}^\top)\mathbf{t}^\top + \mathbf{b}_r^\top)$	$\underline{\mathbf{W}}_l, \underline{\mathbf{W}}_r \in \mathbb{R}^{p\times d\times d}$,$\mathbf{b}_l, \mathbf{b}_r \in \mathbb{R}^p$, $\overline{\times}_3$ n-mode vector-tensor product	
TATEC (Garcia-Duran et al. 2016)	Bilinear	$\langle \mathbf{r}_1^l \vert \mathbf{h}_1\rangle + \langle \mathbf{r}_2^l \vert \mathbf{t}_1\rangle + \langle \mathbf{h}_1 \vert \mathbf{D} \vert \mathbf{t}_1\rangle + \langle \mathbf{h}_2 \vert \mathbf{R}^l \vert \mathbf{t}_2\rangle$	$\langle .\vert .\rangle$canonical dot product, $\mathbf{h}_1, \mathbf{t}_1, \mathbf{r}_1^l, \mathbf{r}_2^l \in \mathbb{R}^d$, $\mathbf{D} \in \mathbb{R}^{d\times d}$diagonal matrix, $\mathbf{h}_2, \mathbf{t}_2 \in \mathbb{R}^p$, $\mathbf{R}^l \in \mathbb{R}^{p\times p}$	Use of different embedding spaces Combines 2-way and 3-way interaction

Table 3 Benefits and challenges of Link prediction models

Approach	Benefit	Challenges	Type	Learning type	Description
1. Node neighborhoods	Simplicity	Failure to detect latent features	Missing link, future link	Unsupervised	Based on a similarity measure, give a score to a testing link. The higher the score the higher the chance predict.
	The basic idea for other methods	Failure to recognize the evolutionary patterns			
	It does not require specific domain knowledge	Failure in multi-relational data			
	Independent of the type and structure of the network				
2. Ensemble of all paths	The basic idea for other methods	Faced with a large network is powerless	Missing link, future link	Unsupervised	
	It does not require specific domain knowledge	Failure to detect latent features and multi-relational			
	In some dataset have a reasonable result	Failure to recognize the evolutionary patterns			
3. Classification model	Just by having feature vector it can predict with any binary classification model	Lack of attention to the multi-relational structure of graph	Missing link, future link	Supervised	Map to a binary classification problem. It can use proximity index as entry of feature vector.
		Need to have a feature vector			
		For future link prediction: Lack of attention to the formation of links over time			

Table 3 continued

Approach	Benefit	Challenges	Type	Learning type	Description
4. Tensor factorization based	Noise-resistant	High computational cost	Missing link, future link	Unsupervised	
	High capability in representation of multi-relational data	It is not able to detect non-linear structure			
	It can expand to higher order tensor easily due to capture dynamic information in multi-relational data				
5. Nonparametric model	Stability	Complexity	Missing link, future link	Supervised, Unsupervised, Semi-supervised	The proper kernel the effectiveness of detection nonlinear characteristic.
	Discover the latent features automatically	On large and dynamic network loses performance			
	Ability to explore nonlinear characteristic				
	The ability to detect seasonal fluctuations				
6. Deep model	Noise-resistant	In some deep neural network, it is less considering a graph structure.	Missing link, future link	Unsupervised	Restricted Boltzmann machine is a building block of deep neural network.
	The ability to detect seasonal fluctuations	Deep models are new born in link mining and capabilities are unknown.			
	Ability to explore nonlinear characteristic				
	The ability to detect seasonal fluctuations				

6.2.1 Quantitative evaluation metrics

Most of the quantitative evaluation metrics used in link prediction have been adopted from other applications such as information retrieval and binary classification (Junuthula et al. (2016)). Hence, their functionality has not been specifically investigated in the link prediction. Of course, Yang et al. (2015) and Junuthula et al. (2016) studies some of these evaluation criteria in temporal link prediction in dynamic networks. By reviewing a large volume of articles, it was revealed that some of the evaluation criteria are different in predicting temporal or lost link. The reason for this is explained as follows: In the temporal link prediction, all links that are not already in the network are in fact not present at all or in other words, they are negative links. In missing link prediction, this assumption is not true at all, and links that are not visible on the network are in the two subsets of lost links and negative links. These two distinct attitudes to the issue should also be taken into account in choosing the evaluation criterion.

Quantitative criteria can be considered in two broad categories: fixed-threshold metrics and threshold curves (Lichtnwalter and Chawla 2012; Yang et al. 2015; Davis and Goadrich 2006). Fixed-threshold metrics suffer from the limitation that some estimates of a reasonable threshold must be available in the score space (Lichtnwalter and Chawla 2012). Threshold curve criteria like ROC or PR are an alternative to these weaknesses.

(a) Fixed-threshold metrics

Fixed-threshold metrics rely on different types of thresholds: prediction score, percentage of instances, or number of instances Yang et al. (2015). The output of a link predictor is usually a set of real-valued scores, which are compared with a set of binary labels, where each label denotes the presence (1) or absence (0) of an edge. These scores should be transformed to the label with regard to the threshold and finally, the evaluation criteria should be used.

Accuracy In classification, the commonly used metrics is accuracy which is defined as:

$$accuracy = \frac{TP + TN}{P + N} \tag{24}$$

where TP is the true positive predicted and TN is the true negative predicted links and P, N are the total positive and negative links, respectively. Accuracy is often deceptive in the case of highly imbalanced data, where high accuracy can be obtained even by a random predictor. Typically, social networks are sparse and the existing link only constitutes less than 10% (this is approximate among data sets) of all possible links (vonWinckel 2014; Lichtnwalter and Chawla 2012). This means that accuracy is not a meaningful measure.

Precision In link prediction, precision or positive predictive value is defined as:

$$PPV = \frac{TP}{TP + FP} \tag{25}$$

where TP is the true positive predicted and FP is the False positive predicted links. This fraction shows the ratio of the true positive prediction among all positive predictions. Precision takes all positive predicted links into account, but it can also be evaluated at a given cut-of-rank, considering only the top-most results returns by the system. This measure is called precision at n or $P@n$. As might be expected, the accuracy of link prediction also varies according to the choice of precision measure.

Due to its robustness, $P@n$ is a frequently used measure in the domain of information retrieval and machine learning (Spiegel et al. (2011)).

Recall In link prediction, recall or true positive rate shows how many of the positive links are predicted. In formal literature it is defined as:

$$Recall(TPR) = \frac{TP}{TP + FN} = \frac{TP}{P} \tag{26}$$

where TP is the true positive predicted link and false negative predicted link.

F1-score Precision and recall are often combined into a single measure using their harmonic mean, known as the F1-score. Equation 27 describes how these two metrics combine into one.

$$F1 - score = 2.\frac{recall.precision}{recall + precision} = \frac{2TP}{2TP + FP + FN} \tag{27}$$

NDCG The normalized discounted cumulative gain (NDCG) over the top k link prediction scores is another information retrieval-based metric that has been used for evaluating link prediction accuracy Tylenda et al. (2009). It is a common evaluation metric used to measure the performance of ranking methods (Zhang et al. (2015). It is defined as:

$$NDCG_k = \frac{DCG_k}{IDCG} \tag{28}$$

where

$$\begin{aligned} DCG_k &= \sum_{i=1}^{k} \frac{2^{r_i} - 1}{\log_2(i + 1)} \\ IDCG_k &= \sum_{i=1}^{|r|} \frac{2^{r_i} - 1}{\log_2(i + 1)} \end{aligned} \tag{29}$$

In Eq. 29, $|r|$ represents a list of positive links in the network and $r_i \in \{0, 1\}$ is the rank of the link in binary mode. It is a fixed-threshold metric that suffers from the same drawbacks as other fixed-threshold metrics as discussed by Yang et al. (2015).

Mean Rank (MR) This evaluation criterion is solely used in missing link prediction. In order to evaluate with this measure, the dataset should be divided into two sets, the train and test. Both train and test sets include observed links. In fact, there are no negative links in these two sets. Considering these circumstances, the test procedure is as follows: For each test link, one node is removed and replaced by each of the entities of the dictionary in turn. Dissimilarities (or energies) of those corrupted links are first computed by the models and then sorted in ascending order; the rank of the correct entity is finally stored. Then the mean of those predicted ranks is reported as *mean rank*(Bordes et al. (2011)).

Hit@n The procedure of Hit@n is the same as mean rank. Hit@n is the proportion of correct entities ranked in the top n. Mostly it reports Hit@10 in articles (Bordes et al. 2013).

(b) Threshold curves

An alternative to fixed-threshold metrics is the use of threshold curve due to the rarity of cases when researchers are in possession of reasonable threshold (Davis and Goadrich 2006; Yang et al. 2015). Threshold curve works by shifting the threshold, computing metrics for each one and then drawing a curve with all computed metrics in all thresholds. They become popular when the class distribution is highly imbalanced. Moreover, in threshold curve, a single scalar measure known as area under curve is used, which serves as a single summary statistic of performance (Davis and Goadrich 2006). Receive Operation characteristic curve (ROC) and Precision-Recall curve (PR) are the two threshold curves mostly used in link prediction (Yang et al. 2015; Junuthula et al. 2016).

Receive Operation characteristic (ROC) ROC curve describes the fraction of true positive rate versus the fraction of false positive rate at various threshold setting (Davis and Goadrich 2006). The FPR measures the fraction of negative links that are misclassified as positive links. The TPR measures the fraction of positive links that are correctly predicted (Davis and Goadrich 2006; Lichtnwalter and Chawla 2012) . Although these metrics are widely used in link prediction, Yang et al. (2015) proves that ROC and AUC can be deceptive. They explain that, due to the severe class imbalance in link prediction, it is recommended to use PR curves and PRAUC in evaluating the link predictor rather than the ROC curve and AUC.

Precision-Recall (PR) The precision-recall curve shows precision with respect to recall at all thresholds Davis and Goadrich (2006) Lichtnwalter and Chawla (2012). The Precision-Recall curve considers only the prediction of positives link and ignores negative samples. Although in link prediction, it is desirable to predict a positive link, in temporal link prediction, removed edges are needed to predict which PR curve dose not give credit for correctly predicting removed or negative edges (Junuthula et al. 2016).

6.2.2 Qualitative evaluation metrics

This section describes qualitative criteria in link prediction. Although such qualitative criteria are not measurable, understanding their importance can aid in the selection of an appropriate method for link prediction. The objective is to identify fair and effective qualification measure for link prediction evaluation.

Cost The proposal of cost is on how time consuming the method. As size of the network goes up, computational process increases sharply (Dunlavy et al. 2011; Li et al. 2014b). It is obvious that the number of parameters has high impact on the computing time and memory (30).

$$\begin{aligned} N_p &\propto T \\ N_p &\propto M \end{aligned} \tag{30}$$

Scalability Social network is a large sparse graph. Actually, there are few observed links compared to the size of the graph. Especially in periodic link prediction, the total size of the network varies with time. Hence, the presented method should be able to cope with evolving network (Lü and Zhou 2011; Bordes et al. 2014).

Generalization It is appealed that the presented method has a reasonable result to most of the datasets. In other words, it utilizes less explicit features like profile information and explores the features just by the input data. If the method requires more explicit features then its evaluation is highly dependent on the selection of proper features (Dunlavy et al. 2011; Nguyen and Mamitsuka 2012; Litwin and Stoeckel 2016).

Exploring evolutionary patterns In a dynamic network, patterns will be formed over time. Discovering such, these patterns help to represent a better picture of network over time and increases accuracy. A key problem in the dynamic network is in understanding the seasonal fluctuations and detecting an ill-behaved node (Li et al. 2014b).

Knowledge Representation Multi-relational data in heterogeneous social network is also known as Knowledge base. The main issue with KBs is that they are far from being complete and knowledge can be represented as a triple(node head, edge relation, node tail). Extracting semantic relation is a task of demand (Brandes and Wagner 2004; Litwin and Stoeckel 2016).

6.3 Analytical evaluation

The third and perhaps the most important component of the proposed framework is the evaluation of link prediction categorization methods. This component evaluates link prediction techniques according to the criteria that have been introduced in the previous subsection. Our evaluation is summarized in Table 4. As observed in Table 4, the column headings is evaluation criteria and the row headings are the nodes of the classification tree (Fig. 5). In Table 4 the strengths and weaknesses of the approaches were compared to each other. An attempt was made to show the efficiency of link prediction models in a different view.

It is important to note that this is not a quantitative evaluation based on scientific experiments, but a qualitative assessment based on a detailed study of the results of previous papers. These results are based on a variety of datasets and with a variety of model validation. Since the number of datasets, the model validation and the purpose of the evaluation criteria are varied, so our mission in this survey article is to provide a qualitative assessment. This is a qualitative assessment to provide a comparative perspective on a macro level, even in a specific domain, with a high accuracy, different parameters such as datasets, test methods, and so on are affected. This assessment can be used in three ways: how to compare, how to choose a method and how to improve the method. This section presents a discussion and comparison of the approaches to each other with a variety of evaluation criteria, and then we will explain how this analytical component can be used in choosing the appropriate method. Ultimately, the solutions that can be used to improve the link prediction method are described.

6.3.1 How to compare

It shows that the learning based model has a higher accuracy than the heuristic base. This is because nearly almost every newly proposed method compares its accuracy with the heuristic based methods and it is obvious that the learning base model uses a different type of features for prediction especially the latent feature model which works well among others (Wang et al. 2007, 2011; Yang et al. 2012; Sarkar et al. 2012; da Silva Soares and Prudêncio 2012; Richard et al. 2014; Li et al. 2014a, b; Rahman and Al Hasan 2016). The accuracy of Tensor factorization models has an inverse proportion with network size. As The network size increases, the accuracy decreases. This is because social networks are sparse, and by increasing the network size, sparsity increases sharply(Nickel et al. 2016). Although some works are done in order to deal with the sparsity, like coupled tensor factorization; among the other latent feature based models which has a medium accuracy (Acar et al. 2011; Yılmaz et al. 2011; Nickel et al. 2016). Medium accuracy is assigned to the classification model; accuracy is highly dependent on selecting the feature vector (Al Hasan and Zaki 2011). It is important to note that although heuristic based approaches have low accuracy among others, they have reasonable performance in a homogeneous network for missing link prediction or non-periodic link prediction (Liben-Nowell and Kleinberg 2007; Lü and Zhou 2011; Feng et al. 2012).

As stated in the previous section, the number of parameters has high impact on the computing time and memory. In Table 4 this has been confirmed too. The reason is that we assigned

Table 4 Analytical framework for evaluation of ink prediction models

Link prediction approach	Accuracy	Cost	Scalability	Generalization	Noise-resident	Explore evolutionary patterns	Knowledge representation
Heuristic-based							
Node-neighborhood	Low	Low	Medium	Medium	Medium	Low	Low
Ensemble of all path	Low	High	Low	Low	Medium	Low	Low
Learning-base							
Classification	Medium	Medium	Medium	Low	Medium	Low	Medium
Latent feature base							
Tensor Factorization	Medium	High	Low	Medium	Low	Medium	Hight
Non-parametric model	High	Low	Medium	Medium	High	High	Medium
Deep model	High	High	Medium	High	High	Medium	High

a high cost to the Ensemble of all paths, which is the nature of the approach. Ensemble of all paths approach walk through a graph which has cost (Liben-Nowell and Kleinberg 2007; Lü and Zhou 2011).

Social networks are sparse. When a network becomes big, mostly it becomes sparser than before. In this case, it can be claimed that sparsity characteristic has a direct impact on scalability. There is no single clear winner among the approaches, and in each approach some of the techniques are more considerable about scalability challenges. In general, because ensemble of all paths behaves like an exhaustive search, it is not a proper approach to deal with a large dataset. On the other hand, when a dataset gets larger, tensor dimensions get larger too and for this reason tensor based models do not have a good performance compared to the smaller network (Nickel et al. 2016).

The classification model has less generalization ability, though a lot of researches have been done for scaling classifiers. As it can be seen in Table 4, deep models have more generalization power and this is because they are not mostly dependent on the feature vectors and they extract latent features in different level of abstraction (Li et al. 2014b; Perozzi et al. 2014; Grover and Leskovec 2016). However; deep models consume more time in comparison to the others.

It is worth noting that sparse networks are highly sensitive to noise (Nguyen and Mamitsuka 2012). Among these, heuristic based approaches have a good performance due to their simplicity. Particularly, global indicator works well and is not sensitive to the local changes caused by noise (Lü and Zhou 2011). However, due to the invariance characteristic of deep models, they are more noise resistant (Bengio et al. 2013). Deep architecture can lead to abstract representation because more abstract concepts can often be constructed in terms of less abstract ones. More abstract concepts are generally invariant to most local changes of the input. One considerable difficulty with tensor factorization is that its prediction accuracy tends to be poor because it is sensitive to noise. some studies tried to add side information to over-come these challenges (Yılmaz et al. 2011; Acar et al. 2011; Narita et al. 2012; Ermiş et al. 2012, 2015; Nakatsuji et al. 2016).

Exploring evolutionary pattern is a long-standing goal in periodic link prediction. As shown in Table 4 the latent feature based models have the ability to explore evolutionary patterns in dynamic networks. It indicates that evolutionary patterns are latent features and need a model to consider different time span as input (Sewell and Chen 2016). Among latent feature based models, non-parametric approaches have a high ability to model evolutionary pattern (Sarkar et al. 2014). Nevertheless, it is mostly dependent on feature vectors, so the result is being limited to a specific domain. Regarding deep models, nothing can be claimed yet, because much work has not been done, but it is obvious that the deep model itself has a lot of parameters, if it is applied to the dynamic network, then the process of learning gets too slow.

As it has been said, knowledge representation is a task of discovering non-linear relation in multi-relational data. Tensor-based and deep embedded models have a high ability of knowledge representation in multi-relational data (Dunlavy et al. 2011; Perozzi et al. 2014; Grover and Leskovec 2016; Nickel et al. 2016). This is because a model focuses on the structure of the graph especially relations. As it was stated earlier, the consideration of both the node attribute and structure of the network is the main key to building a successful model. Meanwhile, nonparametric models do not have access to such success. Actually, this approach works well, when the types of relation is not vast (Zhu et al. 2016a). Moreover, heuristic based approached due to their simplicity are unable to discover non-linear relations and it is not a good choice for knowledge representation (Zhai and Zhang 2015).

6.3.2 How to select

Unlike many classification and clustering methods, choosing the right approach for link prediction, accuracy, and cost are not the basic criteria for selecting a proper method. Link prediction is an off-line task and reaching a reasonable result in the shortest possible time has never been considered as a benchmark in link prediction. It should be noted that although link prediction is an off line task and does not always give the desired best time rate, it cannot be ignored in general. Most social networks have high dimensions and time-consuming methods are faced with the challenges of this huge amount of data. On the other hand, in the link prediction, attempt is always made to predict the maximum number of potential links. The question arises as to how the maximum number of potential links, under which conditions and with what definitions can be obtained?

As previously stated, link prediction falls into two categories: future link prediction and missing link prediction. This leads researchers in two separate lines to select an appropriate method for the link prediction problem. In periodic link prediction, evolutionary pattern recognition is always prioritized. As shown in Table 4, latent-feature-based models have better ability to detect evolutionary patterns. Latent-feature-based models are not only concerned with network structures, but also provide a model that considers the evolutionary structure of the graph. Tensor-factorization approach has high capabilities in modeling dynamic networks. It encodes evolutionary patterns in the model and efficiently represents it. Despite the high ability of this approach to code evolutionary patterns, it is not scalable and in high dimensions it is time-consuming. Hence, researchers have tried to combine it with scalable methods, such as similarity-based approaches. Similarity-based approaches are scalable and time saving.

Another criterion that is necessary for consideration in selecting a method is the type of relationship or link that can be predicted. Similarity algorithms have a good ability to predict links in homogeneous networks. It also has a high scalability due to its simplicity. On the other hand, this approach is incapable of dealing with heterogeneous networks because similarity index does not pay attention to the type of relationship and only predicts the existence of the link. In contrast, latent-feature-based approaches have high ability to model knowledge from multi-relationship data. In fact, tensor factorization and deep models are designed according to the multi-relational structure of the graph and have a high ability to discover knowledge and represent it. Tensor-based models and deep models, with the ability to disclose knowledge, have different accuracy on the same dataset. Tensor-based models work better on low-dimensional data sets, while a large amount of information is needed to explore the knowledge in deep models.

6.3.3 How to improve

The third aspect of the analytical assessment of this proposed framework is to provide an overview of open paths to improve methods. After comparing and recognizing the index criteria in choosing the basic approach, improving other methods is the main goal of this analytical assessment. The presentation of these solutions is based on three perspectives: the combination, the use of side information and the expression of ambiguous and dumb points in the link prediction.

Table 4 can be used explicitly for combining methods. For instance, Acar et al. (2009) have taken advantage of the strengths of similarity-based methods to combine tensor decomposition with the Katz method. The purpose of this combination is to use global indices throughout the graph, which, leads to an efficient method by combining the structural features of the tensor. Lee and colleagues also tried to find an evolutionary model by providing a deep model.

They also tried to use local indexes in an image of the graph using a neighboring joint with this deep model. Li et al. (2014b) have also tried to find an evolutionary pattern by providing a deep model. They have also attempted to use local indices in a snapshot of the graph using a common neighboring joint with this deep model. Due to the high computational cost, the combination of a deep model with time consuming approaches cause a challenging task and may lead to inefficient methods. While combining it with a low cost method prevents this discussion. Since in social networks, local features and homophily patterns are important, it is likely that such compounds have an effective effect, of which Li et al. (2014b) admits this assumption.

As long as the utilization of side information such as content of the social network is talked about, generalization drops. Generalization is at the minimum level in the classification and ensemble of all path approaches. The feature vector in this approach consists of structural information and social network content. This approach has a high potential for utilizing content information. On the other hand in latent feature based models, the feasibility of using side information has been seen is smaller. Similarity-based approaches, given the simplicity and low cost of computing, have a lot of potential for exploiting content information, and much work has been done in this direction.

by scrutinizing a large number of articles, vague and ambiguous points in relation to the evaluation criteria have been seen. In Yang et al. (2015) and Junuthula et al. (2016), some research has been conducted on the evaluation criteria in dynamic networks. It has been shown that, the ROC curve, a well known metrics in link prediction can be deceptive due the high distribution of imbalance. On the other hand PR curve that has been suggested to replace ROC, cannot distinguish removed links in the network. In order to improve the link prediction methods, it is necessary to determine the performance of the evaluation criteria carefully. For instance, there is no quantitative evaluation criterion in dynamic networks that can measure links that are created only over time. Although in this paper an attempt was made to present qualitative and comparative analyses of evolutionary patterns, it is necessary to make statistical calculations. Similar research is also needed to predict missing links.

In social networks, such a statistical pattern is known as homophily, that is, the tendency of entities to be related to other entities with similar characteristics. This has been widely observed in various social networks(Liben-Nowell and Kleinberg 2007). Now the question arises whether the patterns of social networking end with this pattern? What are the other latent patterns of social networks?

7 Conclusion and future work

This work was motivated by the need to understand potential advantage of link prediction methods compared to each other. This paper presented an analytical framework for link prediction in social networks and illustrated that there are different challenges and techniques. This analytical framework has a structural perspective, which has three components: classification of link prediction approach, evaluation criteria and analytical evaluation. This framework proposes a new classification for like prediction methods in a different view. An attempt was made to collect all the current and major works done and then evaluate this classification based on the presented evaluation criteria. This framework could be used to conduct future studies in social networks.

Acknowledgements This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

Acar E, Dunlavy DM, Kolda TG (2009) Link prediction on evolving data using matrix and tensor factorizations. In: 2009 IEEE international conference on data mining workshops, IEEE, pp 262–269

Acar E, Kolda TG, Dunlavy DM (2011) All-at-once optimization for coupled matrix and tensor factorizations. arXiv:1105.3422

Adamic LA, Adar E (2003) Friends and neighbors on the web. Soc Netw 25(3):211–230

Aggarwal C, Subbian K (2014) Evolutionary network analysis: a survey. ACM Comput Surv CSUR 47(1):10

Al Hasan M, Zaki MJ (2011) A survey of link prediction in social networks. In: Aggarwal C (eds) Social network data analytics. Springer, Boston, pp 243–275

Al Hasan M, Chaoji V, Salem S, Zaki M (2006) Link prediction using supervised learning. In: SDM06: workshop on link analysis, counter-terrorism and security

Backstrom L, Leskovec J (2011) Supervised random walks: predicting and recommending links in social networks. In: Proceedings of the fourth ACM international conference on Web search and data mining, ACM, pp 635–644

Bengio Y, Courville A, Vincent P (2013) Representation learning: a review and new perspectives. IEEE Trans Pattern Anal Mach Intell 35(8):1798–1828

Bilgic M, Namata GM, Getoor L (2007) Combining collective classification and link prediction. In: Seventh IEEE international conference on data mining workshops (ICDMW 2007), IEEE, pp 381–386

Bliss CA, Frank MR, Danforth CM, Dodds PS (2014) An evolutionary algorithm approach to link prediction in dynamic social networks. J Comput Sci 5(5):750–764

Bordes A, Weston J, Collobert R, Bengio Y (2011) Learning structured embeddings of knowledge bases. In: Conference on artificial intelligence, EPFL-CONF-192344

Bordes A, Usunier N, Garcia-Duran A, Weston J, Yakhnenko O (2013) Translating embeddings for modeling multi-relational data. In: Burges CJC (eds) Advances in neural information processing systems. Curran Associates Inc., pp 2787–2795

Bordes A, Glorot X, Weston J, Bengio Y (2014) A semantic matching energy function for learning with multi-relational data. Mach Learn 94(2):233–259

Brandes U, Wagner D (2004) Analysis and visualization of social networks. In: Jünger M, Mutzel P (eds) Graph drawing software. Mathematics and visualization. Springer, Berlin, pp 321–340

Cao B, Liu NN, Yang Q (2010) Transfer learning for collective link prediction in multiple heterogenous domains. In: Proceedings of the 27th international conference on machine learning (ICML-10), pp 159–166

Chung TS, Wedel M, Rust RT (2016) Adaptive personalization using social networks. J Acad Mark Sci 44(1):66–87

Clauset A, Moore C, Newman ME (2008) Hierarchical structure and the prediction of missing links in networks. Nature 453(7191):98–101

Collomb G, Härdle W (1986) Strong uniform convergence rates in robust nonparametric time series analysis and prediction: Kernel regression estimation from dependent observations. Stoch Process Their Appl 23(1):77–89

da Silva Soares PR, Prudêncio RBC (2012) Time series based link prediction. In: The 2012 international joint conference on neural networks (IJCNN), IEEE, pp 1–7

Davis D, Lichtenwalter R, Chawla NV (2011) Multi-relational link prediction in heterogeneous information networks. In: 2011 International conference on advances in social networks analysis and mining (ASONAM), IEEE, pp 281–288

Davis J, Goadrich M (2006) The relationship between precision-recall and roc curves. In: Proceedings of the 23rd international conference on Machine learning, ACM, pp 233–240

Doppa JR, Yu J, Tadepalli P, Getoor L (2009) Chance-constrained programs for link prediction. In: NIPS workshop on analyzing networks and learning with graphs

Dunlavy DM, Kolda TG, Acar E (2011) Temporal link prediction using matrix and tensor factorizations. ACM Trans Knowl Discov Data TKDD 5(2):10

Ermiş B, Acar E, Cemgil AT (2012) Link prediction via generalized coupled tensor factorisation. arXiv:1208.6231

Ermiş B, Acar E, Cemgil AT (2015) Link prediction in heterogeneous data via generalized coupled tensor factorization. Data Min Knowl Discov 29(1):203–236

Feng X, Zhao J, Xu K (2012) Link prediction in complex networks: a clustering perspective. Eur Phys J B 85(1):1–9

Fire M, Tenenboim L, Lesser O, Puzis R, Rokach L, Elovici Y (2011) Link prediction in social networks using computationally efficient topological features. In: 2011 IEEE third international conference on privacy, security, risk and trust (PASSAT) and 2011 IEEE third inernational conference on social computing (SocialCom), IEEE, pp 73–80

Gao S, Denoyer L, Gallinari P (2011) Link pattern prediction with tensor decomposition in multi-relational networks. In: 2011 IEEE symposium on computational intelligence and data mining (CIDM), IEEE, pp 333–340

Garcia-Duran A, Bordes A, Usunier N, Grandvalet Y (2016) Combining two and three-way embedding models for link prediction in knowledge bases. J Artif Intell Res 55:715–742

Getoor L, Diehl CP (2005) Link mining: a survey. ACM SIGKDD Explor Newsl 7(2):3–12

Goodfellow I, Bengio Y, Courville A (2016) Deep learning, http://www.deeplearningbook.org, book in preparation for MIT Press

Grover A, Leskovec J (2016) Node2Vec: Scalable feature learning for networks. In: Proceedings of the 22nd acm SIGKDD international conference on knowledge discovery and data mining. KDD'16. ACM, San Francisco, CA, USA, pp 855–864

Han Y, Moutarde F (2016) Analysis of large-scale traffic dynamics in an urban transportation network using non-negative tensor factorization. Int J Intell Transp Syst Res 14(1):36–49

Heaukulani C, Ghahramani Z (2013) Dynamic probabilistic models for latent feature propagation in social networks. In: Dasgupta S, McAllester D (eds) ICML (1). PMLR, pp 275–283

Hinton GE, Osindero S, Teh YW (2006) A fast learning algorithm for deep belief nets. Neural Comput 18(7):1527–1554

Jenatton R, Roux NL, Bordes A, Obozinski GR (2012) A latent factor model for highly multi-relational data. In: Pereira F, Burges CJC, Bottou L, Weinberger KQ (eds) Advances in neural information processing systems. Curran Associates Inc., pp 3167–3175

Jiang X, Tresp V, Huang Y, Nickel M (2012) Link prediction in multi-relational graphs using additive models. In: Proceedings of the 2012 international conference on semantic technologies meet recommender systems & big data-volume 919, CEUR-WS. org, pp 1–12

Junuthula RR, Xu KS, Devabhaktuni VK (2016) Evaluating link prediction accuracy in dynamic networks with added and removed edges. In: 2016 IEEE International conferences on big data and cloud computing (BDCloud), social computing and networking (SocialCom), sustainable computing and communications (SustainCom) (BDCloud-SocialCom-SustainCom), IEEE, pp 377–384

Kashima H, Kato T, Yamanishi Y, Sugiyama M, Tsuda K (2009) Link propagation: a fast semi-supervised learning algorithm for link prediction. In: Park H, Parthasarathy S, Liu H (eds) SDM, vol 9, SIAM, Philadelphia, pp 1099–1110

Keyvanpour MR, Azizani F (2012) Classification and analysis of frequent subgraphs mining algorithms. J Softw 7(1):220–227

Keyvanpour MR, Moradi SS (2014) A perturbation method based on singular value decomposition and feature selection for privacy preserving data mining. Int J Data Warehous Min 10(1):55–76

Kim DI, Gopalan PK, Blei D, Sudderth E (2013) Efficient online inference for bayesian nonparametric relational models. In: Burges CJC, Bottou L, Welling M, Ghahramani Z, Weinberger KQ (eds) Advances in neural information processing systems. Curran Associates, Inc., pp 962–970

Kolda TG, Bader BW (2009) Tensor decompositions and applications. SIAM Rev 51(3):455–500

Krompaß D, Nickel M, Tresp V (2014) Large-scale factorization of type-constrained multi-relational data. In: 2014 International conference on data science and advanced analytics (DSAA), IEEE, pp 18–24

Kuhn F, Oshman R (2011) Dynamic networks: models and algorithms. ACM SIGACT News 42(1):82–96

Lee C, Nick B, Brandes U, Cunningham P (2013) Link prediction with social vector clocks. In: Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp 784–792

Li K, Gao J, Guo S, Du N, Li X, Zhang A (2014a) Lrbm: a restricted boltzmann machine based approach for representation learning on linked data. In: 2014 IEEE international conference on data mining, IEEE, pp 300–309

Li X, Du N, Li H, Li K, Gao J, Zhang A (2014b) A deep learning approach to link prediction in dynamic networks. In: Proceedings of the 2014 SIAM international conference on data mining. SIAM, pp 289–297

Li Deng DY (2014) Deep learning: methods and applications. Tech. rep., https://www.microsoft.com/en-us/research/publication/deep-learning-methods-and-applications/

Liben-Nowell D, Kleinberg J (2007) The link-prediction problem for social networks. J Am Soc Inf Sci Technol 58(7):1019–1031

Lichtenwalter RN, Lussier JT, Chawla NV (2010) New perspectives and methods in link prediction. In: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp 243–252

Lichtnwalter R, Chawla NV (2012) Link prediction: fair and effective evaluation. In: Proceedings of the 2012 international conference on advances in social networks analysis and mining (ASONAM 2012), IEEE Computer Society, pp 376–383

Litwin H, Stoeckel KJ (2016) Social network, activity participation, and cognition a complex relationship. Res Aging 38(1):76–97

Liu F, Liu B, Wang X, Liu M, Wang B (2012) Features for link prediction in social networks: a comprehensive study. In: 2012 IEEE international conference on systems, man, and cybernetics (SMC), IEEE, pp 1706–1711

Liu F, Liu B, Sun C, Liu M, Wang X (2013) Deep learning approaches for link prediction in social network services. In: International conference on neural information processing, Springer, pp 425–432

London B, Rekatsinas T, Huang B, Getoor L (2013) Multi-relational learning using weighted tensor decomposition with modular loss. arXiv:1303.1733

Lü L, Zhou T (2011) Link prediction in complex networks: a survey. Phys A Stat Mech Appl 390(6):1150–1170

Menon AK, Elkan C (2011) Link prediction via matrix factorization. In: Joint European conference on machine learning and knowledge discovery in databases, Springer, pp 437–452

Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv:1301.3781

Miller K, Jordan MI, Griffiths TL (2009) Nonparametric latent feature models for link prediction. In: Bengio Y, Schuurmans D, Lafferty JD, Williams CKI, Culotta A (eds) Advances in neural information processing systems. Curran Associates, Inc., pp 1276–1284

Nakatsuji M, Toda H, Sawada H, Zheng JG, Hendler JA (2016) Semantic sensitive tensor factorization. Artif Intell 230:224–245

Narita A, Hayashi K, Tomioka R, Kashima H (2012) Tensor factorization using auxiliary information. Data Min Knowl Discov 25(2):298–324

Nasim M, Brandes U (2014) Predicting network structure using unlabeled interaction information. MMB & DFT 2014:57

Ngonmang B, Viennet E, Tchuente M, Kamga V (2015) Community analysis and link prediction in dynamic social networks. In: Gamatié A. (eds) Computing in research and development in Africa. Springer, Cham

Nguyen CH, Mamitsuka H (2011) Kernels for link prediction with latent feature models. In: Joint European conference on machine learning and knowledge discovery in databases, Springer, pp 517–532

Nguyen CH, Mamitsuka H (2012) Latent feature kernels for link prediction on sparse graphs. IEEE Trans Neural Netw Learn Syst 23(11):1793–1804

Nguyen-Thi AT, Nguyen PQ, Ngo TD, Nguyen-Hoang TA (2015) Transfer adaboost svm for link prediction in newly signed social networks using explicit and pnr features. Proc Comput Sci 60:332–341

Nickel M, Tresp V (2013a) Logistic tensor factorization for multi-relational data. arXiv:1306.2084

Nickel M, Tresp V (2013b) Tensor factorization for multi-relational learning. In: Joint European conference on machine learning and knowledge discovery in databases, Springer, pp 617–621

Nickel M, Jiang X, Tresp V (2014) Reducing the rank in relational factorization models by including observable patterns. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND, Weinberger KQ (eds) Advances in neural information processing systems. Curran Associates, Inc., pp 1179–1187

Nickel M, Murphy K, Tresp V, Gabrilovich E (2016) A review of relational machine learning for knowledge graphs. Proc IEEE 104(1):11–33

Perozzi B, Al-Rfou R, Skiena S (2014) Deepwalk: online learning of social representations. In: Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining, ACM, pp 701–710

Rahman M, Al Hasan M (2016) Link prediction in dynamic networks using graphlet. In: Joint European conference on machine learning and knowledge discovery in databases, Springer, pp 394–409

Rastelli R, Friel N, Raftery AE (2016) Properties of latent variable network models. Netw Sci 4(4):407–432

Richard E, Gaïffas S, Vayatis N (2014) Link prediction in graphs with autoregressive features. J Mach Learn Res 15(1):565–593

Riedel S, Yao L, McCallum A, Marlin BM (2013) Relation extraction with matrix factorization and universal schemas. In: HLT-NAACL. Curran Associates, Inc., pp 74–84

Rossetti G, Guidotti R, Pennacchioli D, Pedreschi D, Giannotti F (2015) Interaction prediction in dynamic networks exploiting community discovery. In: 2015 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM), IEEE, pp 553–558

Sarkar P, Moore AW (2005) Dynamic social network analysis using latent space models. ACM SIGKDD Explor Newsl 7(2):31–40

Sarkar P, Chakrabarti D, Moore AW (2011) Theoretical justification of popular link prediction heuristics. In: IJCAI proceedings-international joint conference on artificial intelligence, vol 22, p 2722

Sarkar P, Chakrabarti D, Jordan M (2012) Nonparametric link prediction in dynamic networks. arXiv:1206.6394

Sarkar P, Chakrabarti D, Jordan M et al (2014) Nonparametric link prediction in large scale dynamic networks. Electron J Stat 8(2):2022–2065

Schmidt MN, Morup M (2013) Nonparametric bayesian modeling of complex networks: an introduction. IEEE Signal Process Mag 30(3):110–128

Sewell DK, Chen Y (2016) Latent space models for dynamic networks with weighted edges. Soc Netw 44:105–116

Socher R, Chen D, Manning CD, Ng A (2013) Reasoning with neural tensor networks for knowledge base completion. In: Burges CJC, Bottou L, Welling M, Ghahramani Z, Weinberger KQ (eds) Advances in neural information processing systems. Curran Associates, Inc., pp 926–934

Spiegel S, Clausen J, Albayrak S, Kunegis J (2011) Link prediction on evolving data using tensor factorization. In: Pacific-Asia conference on knowledge discovery and data mining, Springer, pp 100–110

Tang J, Qu M, Wang M, Zhang M, Yan J, Mei Q (2015) Line: Large-scale information network embedding. In: Proceedings of the 24th international conference on world wide web, ACM, pp 1067–1077

Taskar B, Wong MF, Abbeel P, Koller D (2003) Link prediction in relational data. In: Thrun S, Saul LK, Schölkopf PB (eds) Advances in neural information processing systems. MIT Press

Tylenda T, Angelova R, Bedathur S (2009) Towards time-aware link prediction in evolving social networks. In: Proceedings of the 3rd workshop on social network mining and analysis, ACM, p 9

Wang C, Satuluri V, Parthasarathy S (2007) Local probabilistic models for link prediction. In: Seventh IEEE international conference on data mining (ICDM 2007), IEEE, pp 322–331

Wang D, Pedreschi D, Song C, Giannotti F, Barabasi AL (2011) Human mobility, social ties, and link prediction. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp 1100–1108

Wang P, Xu B, Wu Y, Zhou X (2015) Link prediction in social networks: the state-of-the-art. Sci China Inf Sci 58(1):1–38

Yang Y, Chawla N, Sun Y, Hani J (2012) Predicting links in multi-relational and heterogeneous networks. In: 2012 IEEE 12th International conference on data mining, IEEE, pp 755–764

Yang Y, Lichtenwalter RN, Chawla NV (2015) Evaluating link prediction methods. Knowl Inf Syst 45(3):751–782

Yao L, Sheng QZ, Qin Y, Wang X, Shemshadi A, He Q (2015) Context-aware point-of-interest recommendation using tensor factorization with social regularization. In: Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval, ACM, pp 1007–1010

Yılmaz KY, Cemgil AT, Simsekli U (2011) Generalised coupled tensor factorisation. In: Shawe-Taylor J, Zemel RS, Bartlett PL, Pereira F, Weinberger KQ (eds) Advances in neural information processing systems Curran Associates Inc., pp 2151–2159

Yu K, Chu W, Yu S, Tresp V, Xu Z (2006) Stochastic relational models for discriminative link prediction. In: Schölkopf PB, Platt JC, Hoffman T (eds) Advances in neural information processing systems, pp 1553–1560

Yu K, Lafferty J, Zhu S, Gong Y (2009) Large-scale collaborative prediction using a nonparametric random effects model. In: Proceedings of the 26th annual international conference on machine learning, ACM. MIT Press, pp 1185–1192

Zhai S, Zhang Z (2015) Dropout training of matrix factorization and autoencoder for link prediction in sparse graphs. In: Proceedings of the 2015 SIAM international conference on data mining. SIAM, pp 451–459

Zhang J, Lv Y, Yu P (2015) Enterprise social link recommendation. In: Proceedings of the 24th ACM international on conference on information and knowledge management, ACM, pp 841–850

Zhang X, Chen W, Yan H (2016) TLINE: Scalable transductive network embedding. In: Ma S et al (eds) Information retrieval technology. AIRS 2016. Lecture notes in computer science, vol 9994. Springer, Cham

Zhu J, Song J, Chen B (2016a) Max-margin nonparametric latent feature models for link prediction. arXiv:1602.07428

Zhu L, Guo D, Yin J, Ver Steeg G, Galstyan A (2016b) Scalable temporal latent space inference for link prediction in dynamic social networks. IEEE Trans Knowl Data Eng 28(10):2765–2777