Social Network Analysis and Mining (2020) 10:18
https://doi.org/10.1007/s13278-020-0630-6

ORIGINAL PAPER

Link and interaction polarity predictions in signed networks

Tyler Derr[1] · Zhiwei Wang[1] · Jamell Dacon[1] · Jiliang Tang[1]

Received: 6 January 2019 / Revised: 25 February 2020 / Accepted: 27 February 2020

Abstract

In today's world, users typically take to online social media to express their opinions, which can inherently be both positive and negative. In fact, online social networks can be best modeled as signed networks, where opinions in the form of positive and negative links can exist between users, such as our friends and foes (e.g., "unfriended" users), respectively. Furthermore, users can also express their opinions to content generated by others through online social interactions, such as commenting or rating. Intuitively, these two types of opinions in the form of links and interactions should be related. For example, users' interactions are likely to be positive (or negative) to those they have positively (or negatively) established links with. Similarly, we tend to establish positive (or negative) links with those whose generated content we frequently positively (or negatively) interact with online. Hence, in this paper, we first verify these assumptions by understanding the correlation between these two types of opinions from both a local and global perspective. Then, we propose a framework that jointly solves the link and interaction polarity prediction problem based on our newly found understanding of how these two problems are correlated. We ultimately perform experiments on a real-world signed network to demonstrate the effectiveness of our proposed approach to help mitigate both the data sparsity and cold-start problems found in the two tasks of link and interaction polarity prediction.

Keywords Signed networks · Social media · Link prediction · Interaction polarity prediction

1 Introduction

Traditionally, network analysis has focused on unsigned networks (or networks with only positive links). Many social networks in social media can have positive and negative links (or signed networks, Cartwright and Harary 1956; Heider 1946). Such examples include the Epinions network with trust and distrust and the Slashdot network with friend and foe links. Furthermore, many popular social media networks can incorporate "unfriending," "unfollowing," or blocking as negative links, while the currently established links can represent positive links (e.g., in Facebook, Twitter, etc.) It has been proven that negative links can advance various network analysis tasks such as link prediction (Guha et al. 2004; Hsieh et al. 2012; Leskovec et al. 2010a), node classification (Tang et al. 2016a), community detection (Chiang et al. 2014; Kunegis et al. 2010; Sharma 2012; Zheng and Skillicorn 2015), and recommendations (Forsati et al. 2014; Ma et al. 2009; Victor et al. 2009, 2013). Meanwhile, a recent study has shown that invisible negative links in social media are predictable (Tang et al. 2015) and that they can help convert many social media unsigned networks such as Facebook friendship and Twitter following into signed networks. Therefore, signed networks are ubiquitous and have attracted increasing attention in recent years (Tang et al. 2016b).

Meanwhile, social media has been increasingly used by online users to share and exchange opinions. In signed networks, users can directly express positive (or negative) opinions to others by establishing positive (or negative) links. They can also specify positive (or negative) opinions to content created by others via various interactions such as commenting and rating. These two types of opinions should be related inherently. For example, a user receiving more positive (or negative) links is likely to receive more positive (or negative) opinions for his/her content, while users are likely to give positive (or negative) opinions to content generated by those with positive (or negative) links. In reality, users may also explicitly only give opinions to a small

✉ Tyler Derr
derrtyle@msu.edu

[1] Data Science and Engineering Lab, Michigan State University, East Lansing, MI, USA

Published online: 10 March 2020

number of users or content. For example, both positive and negative links follow a power-law-like distribution—a small number of users specify many positive (or negative) links, while a large proportion of users specify a few positive (or negative) links (Tang et al. 2014). Hence, link prediction (Leskovec et al. 2010a) and interaction polarity prediction (Tang et al. 2013) are proposed to infer implicit opinions of these two types, respectively.

Recent years have witnessed a large number of algorithms for signed link prediction (Chiang et al. 2011; Hsieh et al. 2012; Leskovec et al. 2010a; Symeonidis and Mantas 2013) and interaction polarity prediction (Moghaddam et al. 2012, 2011; Tang et al. 2013; Wang et al. 2015). However, the majority of them have tackled these two tasks independently. As aforementioned, the corresponding opinions in these two tasks could be correlated and we can utilize one to power the other. Thus, we could boost the performance by joining these two tasks. Meanwhile, due to the sparsity nature of social media data, both tasks have been shown to severely suffer from the data sparsity and cold-start problems (Chiang et al. 2014; Wang et al. 2015). By capturing the correlation between these two types of opinions, one can enrich the other and therefore have more information to use for their corresponding tasks. Hence, a joint framework has the potential to mitigate the data sparsity and cold-start problems for both tasks.

In this paper, we study the problem of joint link and interaction polarity predictions in signed networks. In particular, we investigate—(a) whether opinions in the two tasks are related? and (b) how to utilize their correlations for joint link and interaction polarity predictions? Providing answers to these two questions, we propose a novel framework LIP that can infer links and polarities of interactions jointly. Our main contributions in this work have been summarized as follows:

- We validate the correlations between link signs and interaction polarities from both global and local perspectives;
- We propose a joint link and interaction prediction framework (LIP) that explicitly incorporates the correlations to predict links and interaction polarities simultaneously;
- We conduct experiments in a real-world signed network to demonstrate (a) the effectiveness of LIP and (b) the robustness of LIP to the data sparsity and cold-start problems.

When compared with Derr et al. (2018d), in this work, we have also: (1) performed a thorough data analysis from both a local and global perspective that motivates a joint learning model, (2) provided a detailed gradient-based optimization scheme for learning our proposed joint framework, which greatly improves the reproducibility of this work (along with making the code publicly available), (3) conducted further experiments investigating the performance when varying the amount of induced cold-start users, which we use to further show the robustness/effectiveness of LIP, and more specifically, how opinion information is able to flow between the two problems to show that opinions indeed power opinions and finally (4) we have performed a parameter analysis to observe the tradeoffs between the two main hyperparameters of our proposed framework.

The rest of this paper is organized as follows. In Sect. 2, we formally define the joint prediction problem. In Sect. 3, we describe the dataset used in our work, along with our analysis of the correlations. We discuss our proposed novel joint framework in Sect. 4. In Sect. 5, the experimental results and findings are presented. We briefly review related work in Sect. 6. Conclusions and future work are given in Sect. 7.

2 Problem

Let $\mathcal{U} = \{u_1, u_2, \dots, u_n\}$ denote the set of n users. We represent signed links between users in an adjacency matrix, $\mathbf{T} \in \mathbb{R}^{n\times n}$, where $\mathbf{T}_{ij} = 1$ if u_i creates a positive link to u_j, -1 if u_i creates a negative link to u_j, and 0 otherwise (i.e., when u_i has shown no link to u_j). Let $\mathcal{R} = \{r_1, r_2, \dots, r_m\}$ be the set of m content items generated by $\mathcal{U}$. We use $\mathbf{A} \in \mathbb{R}^{n\times m}$ to denote the authorship matrix where $\mathbf{A}_{ij} = 1$ if u_i creates r_j and $\mathbf{A}_{ij} = 0$ otherwise. Social media provides multiple ways for users to express their opinions to content items generate by other users. For example, Facebook and Twitter allow their users to comment on content; YouTube provides thumbs-up and thumbs-down buttons; and Epinions enables its users to rate the helpfulness of the content with scores from 1 to 6. We use $\mathbf{H} \in \mathbb{R}^{n\times m}$ to denote opinions expressed by $\mathcal{U}$ to $\mathcal{R}$, where $\mathbf{H}_{ik} = 1(or - 1)$ if u_i gives a positive (or negative) opinion to r_k and we use $\mathbf{H}_{ik} = 0$ to indicate no explicit opinion is expressed from u_i to r_k. Note that in this paper, we define positive (or negative) interactions between u_i and u_j as u_i giving positive (or negative) opinions to content items generated by u_j. In other words, an interaction between users is defined as a triplet (u_i, r_k, u_j) where u_i expresses opinions to r_k that was generated by u_j.

With the above notations and definitions, our problem is stated as follows: *given the signed relations* $\mathbf{T}$, *the authorship matrix* $\mathbf{A}$ *and the user-item opinion matrix* $\mathbf{H}$, *we aim to learn a predictor that can infer signed links and interaction polarities simultaneously by leveraging* $\mathbf{T}$, $\mathbf{A}$ *and* $\mathbf{H}$.

Note that when the content of the item is available, we also can utilize the content of $\mathcal{R}$. However, in this paper, we focus on leveraging $\mathbf{T}$, $\mathbf{A}$ and $\mathbf{H}$ and would like to leave the problem of exploiting content as one future work.

Table 1 Epinions dataset statistics

# of users	233,429
# of positive links	717,667
# of negative links	123,705
Density of **T**	7.75×10^{-5}
# of reviews	755,722
# of positive interactions	12,581,553
# of negative interactions	1,086,551
Density of **H**	1.54×10^{-5}

3 Data analysis

In this section, we conduct preliminary analysis on the correlation between signed links and interaction polarities. We begin by introducing the dataset in our study that can be found in our signed network dataset repository (Derr 2020).[1]

3.1 Dataset

We collected a dataset from Epinions for this investigation. Epinions users can give positive and negative links to each other, which we use to construct the **T** matrix. They also can write reviews, and we use this data to construct the authorship matrix **A**. For each review, others can use scores from 1 to 6 to indicate the helpfulness of the given reviews and that we use these to construct the matrix **H**. We define positive and negative helpfulness ratings to be $\{4, 5, 6\}$ and $\{1, 2, 3\}$, respectively. Some statistics of the dataset are shown in Table 1. From the table, we can observe that (1) there are more positive links (or interactions) than negative ones and (2) both links and interactions are very sparse. The task of creating (or receiving) a signed link to others can be thought of as an explicit form of expressing one's opinion of (or from) others. In contrast, when a user interacts with the content authored by others, they are implicitly marking their opinion toward others in these interactions. Therefore, it is reasonable to assume that the implicit and explicit opinions among users are correlated. Next, we investigate these correlations from both global and local perspectives.

3.2 A global perspective

From a global perspective, we want to examine the correlations between these explicit and implicit opinions from one user. In particular, we aim to answer the following questions—(1) is a user, giving more positive (or negative) links, likely to give more positively (or negatively) on content from others? and (2) is a user, receiving more positive (or negative) links, likely to receive more positive (or negative) opinions on his/her content? In this work, we refer to giving links or opinions on content as giving behaviors, while receiving links or opinions on content as receiving behaviors.

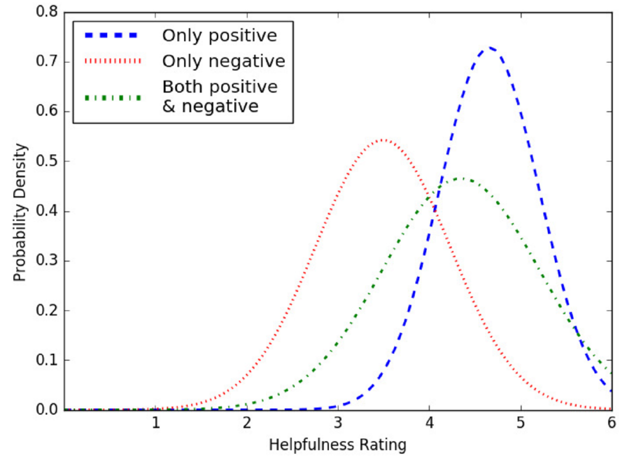


Fig. 1 Giving behaviors from the global perspective on opinion correlations

To answer the first question, we group users into three classes based upon their outgoing links as follows: (1) users who only have positive outgoing links (76,819 users); (2) users having only negative outgoing links (7138 users); and (3) users who have both positive and negative outgoing links (11,361 users). Then, we calculate the opinions (or helpfulness ratings) they gave to content from others for each group and we plot kernel smoothing density estimation for each group in Fig. 1. We note that on average, users who only create positive links also tend to interact more positively with the content generated by other users as compared to users who only create negative links. Furthermore, users who create both positive and negative links show a higher variance than the only positive and only negative classes and thus are more likely to express both positive and negative behaviors in their interactions.

To answer the second question, we divide users into three groups based upon their incoming links as follows: (1) users who only have positive incoming links (52,810 users); (2) users having only negative incoming links (14,701 users); and (3) users who have both positive and negative incoming links (17,090 users). Following the similar procedure, we plot kernel smoothing density estimation of receiving behaviors for each group in Fig. 2. From Figs. 1 and 2, we can make very similar observations for receiving behaviors as giving behaviors, which lead to a positive answer to the second question—users, receiving more positive (or negative) links, are likely to obtain more positive (or negative) opinions on their content.

[1] https://github.com/TylersNetwork/awesome-signed-network-datasets.

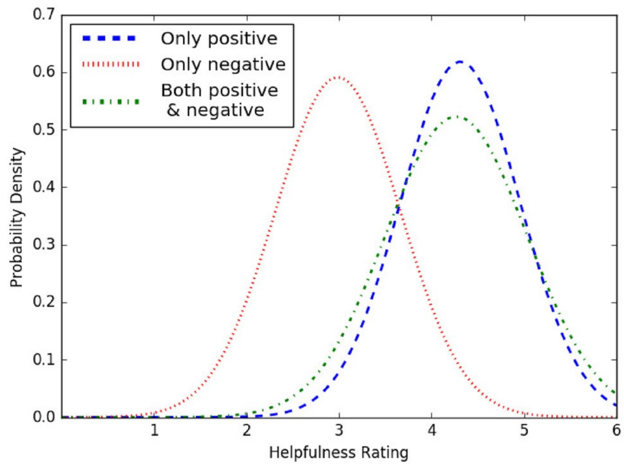


Fig. 2 Receiving behaviors from the global perspective on opinion correlations

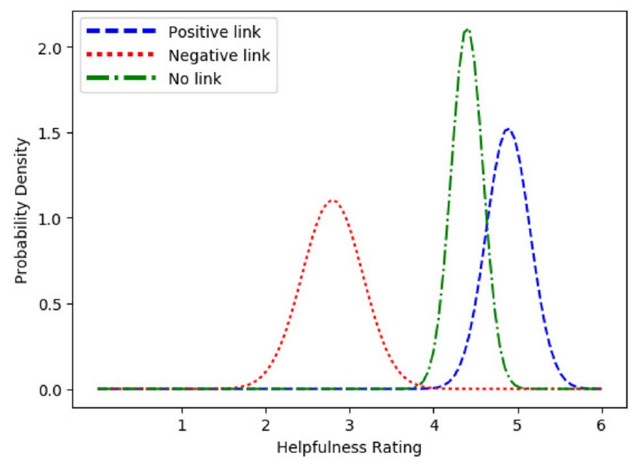


Fig. 3 Giving behaviors from the local perspective on opinion correlations

3.3 A local perspective

The global perspective in Sect. 3.2 focuses on correlations between one user and the remaining network. In this subsection, we focus on a pair of users and we want to investigate whether the existence of a positive (or negative) link for a pair of users makes a difference on how they give (or receive) opinions on each other's content. In particular, for a pair of users u_i to u_j, we aim to answer—(1) if u_i gives a positive (or negative) link to u_j, is u_i likely to give positive (or negative) opinions to content from u_j? and (2) if u_j receives a positive (or negative) link from u_i, is u_j likely to give positive (or negative) opinions to the content from u_i? Note that in this work, we use $u_i + u_j$, $u_i - u_j$ and $u_i ? u_j$ to denote a positive, negative and no link from u_i to u_j.

To answer the first question, we divide all pairs of users into three groups—(a) positive pairs $u_i + u_j$; (b) negative pairs $u_i - u_j$; and no-link pairs $u_i ? u_j$. For each pair in each group, we calculate the average opinion (or helpfulness ratings) from u_i to the content of u_j. We apply kernel smoothing density estimation for each group, and the distributions are shown in Fig. 3. From this figure, we note that on average, positive pairs have higher helpfulness scores than no-link pairs, which have higher scores than negative pairs. Hence, it is quite evident from the figure that if u_i gives a positive (or negative) link to u_j, u_i is likely to give positive (or negative) opinions to the content from u_j.

Intuitively, if u_j receives a positive link from u_i, u_j is likely to be friendly to u_i, and as a consequence, u_j is likely to give positive opinions to the content of u_i. On the other hand, if u_j receives a negative link from u_i, u_j could do revenge back and give negative opinions to the content of u_i. We follow a similar procedure of answering the first question for the second question. The results are demonstrated in Fig. 4. From this figure, we observe that (1) on average, u_j mostly gives positive opinions to the content from those who give positive links to u_j, while u_j mostly gives negative opinions to the content from those who give negative links to u_j. These observations support that if u_j receives a positive (or negative) link from u_i, then u_j is likely to give opinions being more positive (or negative) to the content from u_i.

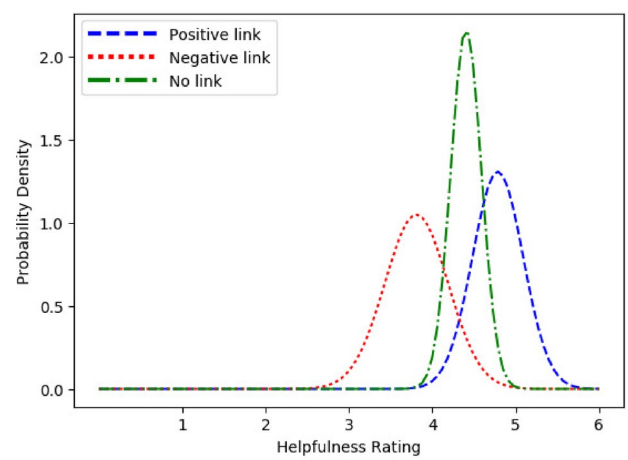


Fig. 4 Receiving behaviors from the local perspective on opinion correlations

4 A framework for joint link and interaction polarity predictions

In Sect. 3, we validated that there exist correlations between a user's opinion of other users in regard to the links they form in signed social networks and the polarities of the interactions between them. Thus, these findings naturally lead us to the question of whether this knowledge can benefit the two prediction tasks that are found in the two domains: link and interaction polarity prediction. In this section, we first briefly discuss a basic framework to solve the two tasks of link and interaction polarity predictions individually. We then discuss how to model the opinion correlations that enable us to have the opinions in one task power the other. Finally, we present our proposed framework LIP, which directly incorporates these correlations into a joint optimization algorithm that can infer links and polarities of interactions jointly.

4.1 Basic prediction models

The low-rank matrix factorization approach has gained popularity recently and is now being used across various applications such as link prediction (Agrawal et al. 2013; Hsieh et al. 2012) and recommender systems (Forsati et al. 2015; Wang et al. 2015). In this work, we choose to build the basic prediction models based on the low-rank matrix factorization approach.

4.1.1 Link prediction

Let $\mathcal{T} = \{(u_i, u_j) \mid \mathbf{T}_{ij} \neq 0\}$ be the set of pairs with links. In terms of the link prediction task, we would like to find two latent matrices $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n] \in \mathbb{R}^{K_L \times n}$ and $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n] \in \mathbb{R}^{K_L \times n}$,with K_L being the number of latent dimensions, by solving the following optimization problem:

$$\min_{\mathbf{U},\mathbf{V}} \frac{1}{2} \sum_{(u_i,u_j)\in\mathcal{T}} (\mathbf{T}_{ij} - \mathbf{u}_i^\top \mathbf{v}_j)^2 + \frac{\beta_1}{2}\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right) \tag{1}$$

where $\mathbf{u}_i$ and $\mathbf{v}_i$ are the user latent vectors representing giving and receiving link behaviors of u_i, respectively. Thus, $\mathbf{u}_i^\top \mathbf{v}_j$ models the sign of a link from u_i to u_j, and therefore, after optimizing the above formulation, we can use such inner products as a prediction for unknown user-user signed links in the network. Note that $\|\mathbf{U}\|_F^2$ denotes the Frobenius norm of $\mathbf{U}$ and is used as a regularization term to prevent overfitting, similarly for $\mathbf{V}$, and both are controlled by the parameter β_1.

4.1.2 Interaction polarity prediction

Let $\mathcal{H} = \{(u_i, r_k, u_j) \mid \mathbf{H}_{ik} \neq 0, \mathbf{A}_{jk} \neq 0\}$ be the set of interaction triplets and $\mathbf{H}_{ik}$ denotes the opinion from u_i to the content r_k authored by u_j. The main difference between the basic models for this task from traditional matrix factorization-based recommender systems is that we now have a third piece of information, the author. Thus, rather than taking the typical user-item formulation, we instead want to formulate the model so that we can include information about the author of the content.

In this problem, we wish to find three latent matrices $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n] \in \mathbb{R}^{K_I \times n}$, $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n] \in \mathbb{R}^{K_I \times n}$ and $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_m] \in \mathbb{R}^{K_I \times m}$, where $\mathbf{p}_i$ and $\mathbf{q}_i$, respectively, denote the giving and receiving interaction behaviors of u_i, and $\mathbf{s}_k$ is the latent vector for content r_k. One way to represent this would be to ignore the author and want $\mathbf{p}_i^\top \mathbf{s}_k$ to model the interaction between user u_i on content r_k that was authored by u_j. Similarly, we could ignore the content and only use the author, i.e., $\mathbf{p}_i^\top \mathbf{q}_j$, but each of these is lacking information. Hence, we propose to use $\mathbf{p}_i^\top (\mathbf{q}_j + \mathbf{s}_k)$, which includes both the context of the author and the content itself. These three matrices can be obtained via solving the following optimization problem:

$$\min_{\mathbf{P},\mathbf{Q},\mathbf{S}} \frac{1}{2} \sum_{(u_i,r_k,u_j)\in\mathcal{H}} (\mathbf{H}_{ik} - \mathbf{p}_i^\top (\mathbf{q}_j + \mathbf{s}_k))^2 + \frac{\beta_2}{2}\left(\|\mathbf{P}\|_F^2 + \|\mathbf{Q}\|_F^2 + \|\mathbf{S}\|_F^2\right) \tag{2}$$

where the term $\left(\|\mathbf{P}\|_F^2 + \|\mathbf{Q}\|_F^2 + \|\mathbf{S}\|_F^2\right)$ is introduced to avoid overfitting, which is controlled by β_2. Note that another way of modeling could be to linearly combine the author and content representation. In that way, we could define $\mathbf{M} \in \mathbb{R}^{K_I \times 2K_I}$ with $\mathbf{p}_i(\mathbf{M}(\mathbf{q}_j || \mathbf{s}_k))$, where $||$ is used to denote concatenation. However, this would add extra complexity by needing to learn $\mathbf{M}$, so we use $\mathbf{p}_i^\top (\mathbf{q}_j + \mathbf{s}_k)$, and leave other formulations as future work. Next, we will discuss how to capture correlations based on the two aforementioned basic models.

4.2 Modeling opinion correlations

In Sect. 3, we found that the giving (or receiving) behaviors in terms of links and interactions are correlated. In the basic models from Sect. 4.1.2, we use $\mathbf{u}_i$ and $\mathbf{v}_i$ to denote users' behaviors when giving and receiving links, respectively. While we use $\mathbf{p}_i$ and $\mathbf{q}_i$ to, respectively, indicate users' behaviors when giving and receiving interactions, separately. Therefore, we can capture the opinion correlations by bridging the two giving behaviors via $\mathbf{u}_i$ and $\mathbf{p}_i$, and the two receiving behaviors via $\mathbf{v}_i$ and $\mathbf{q}_i$.

Since the two giving behaviors are correlated, we can find a linear mapping matrix $\mathbf{W}_O \in \mathbb{R}^{K_I \times K_L}$ that can map u_i's latent vector $\mathbf{u}_i$, which denotes his/her underlying behavior on how to create links, to the latent vector $\mathbf{p}_i$, which captures their behavior toward how they give opinions to the content authored by other users in the network. Given a set of latent vectors for all users $u_i \in \mathcal{U}$, it can then be easily seen that the linear mapping between them would be a solution to the following optimization problem:

$$\min_{\mathbf{W_O}} \sum_{u_i \in \mathcal{U}} \|\mathbf{W}_O \mathbf{u}_i - \mathbf{p}_i\|_2^2 \tag{3}$$

Similarly, we seek to find a matrix $\mathbf{W}_I \in \mathbb{R}^{K_I \times K_L}$ to represent the mapping between the user u_j's latent vectors $\mathbf{v}_j$ and $\mathbf{q}_j$, which denote their receiving behaviors of receiving links and interactions, respectively. The mapping $\mathbf{W}_I$ can be learned as follows:

$$\min_{\mathbf{W_I}} \sum_{u_j \in \mathcal{U}} \left\|\mathbf{W}_I \mathbf{v}_j - \mathbf{p}_j\right\|_2^2 \tag{4}$$

Equations (3) and (4) can capture opinion correlations for links and interactions. They also allow us to bridge the two basic models for link and interaction polarity predictions together. Next, we will introduce the proposed joint framework.

4.3 The proposed joint framework

Now, we have formulated a model on how to optimize a linear mapping between both the giving and receiving behaviors in the two tasks. Next, we show how these mappings can be used as two additional terms in our joint matrix factorization framework, LIP, for the purpose of joint link and interaction polarity prediction. LIP solves the following optimization problem:

$$\begin{aligned}
\min_{\substack{\mathbf{U},\mathbf{V},\mathbf{P},\mathbf{Q},\\ \mathbf{S},\mathbf{W}_I,\mathbf{W}_O}} \quad & \mathcal{L}(\mathbf{U},\mathbf{V},\mathbf{P},\mathbf{Q},\mathbf{S},\mathbf{W}_I,\mathbf{W}_O) \\
= & \frac{1}{2} \sum_{(u_i,u_j)\in\mathcal{T}} (\mathbf{T}_{ij} - \mathbf{u}_i^\top \mathbf{v}_j)^2 \\
& + \frac{\eta}{2} \sum_{(u_i,r_k,u_j)\in\mathcal{H}} (\mathbf{H}_{ik} - \mathbf{p}_i^\top(\mathbf{q}_j + \mathbf{s}_k))^2 \\
& + \frac{\gamma}{2}\Big(\sum_{u_i\in\mathcal{U}} \|\mathbf{W}_O\mathbf{u}_i - \mathbf{p}_i\|_2^2 + \sum_{u_j\in\mathcal{U}} \left\|\mathbf{W}_I\mathbf{v}_j - \mathbf{q}_j\right\|_2^2 \Big) \\
& + \frac{\beta_1}{2}\Big(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\Big) + \frac{\beta_2}{2}\Big(\|\mathbf{P}\|_F^2 + \|\mathbf{Q}\|_F^2 + \|\mathbf{S}\|_F^2\Big) \\
& + \frac{\beta_3}{2}\Big(\|\mathbf{W}_I\|_F^2 + \|\mathbf{W}_O\|_F^2\Big)
\end{aligned} \tag{5}$$

where the first term is a standard user-user matrix factorization model (as discussed in Sect. 4.1) for the link prediction problem. The second term is a modification to the user-review matrix factorization model that also incorporates the additional vector $\mathbf{q}_j$ $\forall u_j \in \mathcal{U}$ to represent the influence of the author u_j in the prediction of u_i's opinion on r_k, when r_k was written by u_j. The third and fourth terms capture the correlations of giving and receiving behaviors, respectively, and their contributions are controlled by a parameter γ. Other terms in Eq. (5) are added to avoid overfitting.

We note that the balance between optimizing for the two tasks (sign link prediction and user interactions polarities) is balanced by the parameter η, where a small increase in this value will result in an increase to the importance of the user interaction polarity prediction task, and similarly toward the link prediction task when decreasing its value. Also, this transfer of information between problems is done by the linear mapping used in LIP. [More specifically, the terms controlled by γ in Eq. (5).] If a user u_i has no link information, they are deemed a cold-start user in the link prediction task. Thus, there is no way to learn $\mathbf{u}_i$ and $\mathbf{v}_i$ in the basic model and we fail to do link prediction for u_i. However, if u_i has had some interactions with other users in the network, we can learn $\mathbf{p}_i$ and $\mathbf{q}_i$ from his/her interaction data. Thus, the proposed framework LIP can also learn $\mathbf{u}_i$ and $\mathbf{v}_i$ via the model components of capturing giving and receiving correlations via the third and fourth terms in Eq. (5). Similarly, LIP can also help when u_i has no interaction data but has link information. Via the above analysis, we note that LIP has the potential to mitigate the data sparsity and cold-start problems in either link prediction or interaction polarity prediction.

4.4 An optimization method for LIP

Given the optimization objective shown above, we now present how to solve this problem. We have chosen to use stochastic gradient descent (SGD) due to the non-convexity of the joint optimization formulation. First, we compute the partial derivatives with respect to each of the parameters (i.e., $\mathbf{u}_i, \mathbf{v}_j, \mathbf{p}_i, \mathbf{q}_j, \mathbf{s}_k, \mathbf{W}_O$ and $\mathbf{W}_I$) and then iteratively update them using SGD until convergence. We use the combined training data $\mathcal{X} = \{\mathcal{T} \cup \mathcal{H}\}$, where $\mathcal{T}$ and $\mathcal{H}$ are the link and interaction training data, respectively.

For simplicity in the below, let $e_{\mathbf{T}_{ij}} = (\mathbf{T}_{ij} - \mathbf{u}_i^\top \mathbf{v}_j)$ be the error of estimating the link (which in some social networks, such as Epinions, can represent trust–distrust) from user u_i to user u_j, $e_{\mathbf{H}_{ikj}} = (\mathbf{H}_{ik} - \mathbf{p}_i^\top(\mathbf{q}_j + \mathbf{s}_k))$ be the error of estimating the interaction value user u_i gave to content r_k that had been authored by user u_j, $e_O = (\mathbf{W}_O\mathbf{u}_i - \mathbf{p}_i)$ be the error for our linear mapping from user u_i's latent vector $\mathbf{u}_i$ (representing the way they give links) to their latent vector $\mathbf{p}_i$ (representing how they interact with content created by others), and finally, we denote $e_I = (\mathbf{W}_I\mathbf{v}_j - \mathbf{q}_j)$ be the error for our linear mapping from user u_j's latent vector $\mathbf{v}_j$ (representing the way they receive links) to their latent vector $\mathbf{q}_j$ (representing how the content they had authored receives interactions).

Gradients of $\mathcal{L}$ with respect to $\mathbf{U}$ *and* $\mathbf{V}$ The gradients of Eq. (5) w.r.t. $\mathbf{u}_i$ and $\mathbf{v}_j$ are as follows, respectively:

$$\frac{\partial \mathcal{L}(\mathbf{U})}{\partial \mathbf{u}_i} = \sum_{\{j \mid (u_i,u_j)\in \mathcal{T}\}} \Big(- e_{\mathbf{T}_{ij}} \mathbf{v}_j \Big) + \gamma \mathbf{W}_O^\top e_O + \beta_1 \mathbf{u}_i$$

$$\frac{\partial \mathcal{L}(\mathbf{V})}{\partial \mathbf{v}_j} = \sum_{\{i \mid (u_i,u_j)\in \mathcal{T}\}} \Big(- e_{\mathbf{T}_{ij}} \mathbf{u}_i \Big) + \gamma \mathbf{W}_I^\top e_I + \beta_1 \mathbf{v}_j$$

Gradients of $\mathcal{L}$ with respect to $\mathbf{P}$, $\mathbf{Q}$ *and* $\mathbf{S}$ The gradients of Eq. (5) w.r.t. $\mathbf{p}_i$, $\mathbf{q}_j$ and $\mathbf{s}_k$ are the following, respectively:

$$\frac{\partial \mathcal{L}(\mathbf{P})}{\partial \mathbf{p}_i} = \sum_{\{(k,j) \mid (u_i,r_k,u_j)\in \mathcal{H}\}} \Big(- \eta e_{\mathbf{H}_{ikj}} (\mathbf{q}_j + \mathbf{s}_k) \Big) - \gamma e_O + \beta_2 \mathbf{p}_i$$

$$\frac{\partial \mathcal{L}(\mathbf{Q})}{\partial \mathbf{q}_j} = \sum_{\{i \mid (u_i,r_k,u_j)\in \mathcal{H}\}} \Big(- \eta e_{\mathbf{H}_{ikj}} \mathbf{p}_i \Big) - \gamma e_I + \beta_2 \mathbf{q}_j$$

$$\frac{\partial \mathcal{L}(\mathbf{S})}{\partial \mathbf{s}_k} = \sum_{\{i \mid (u_i,r_k,u_j)\in \mathcal{H}\}} \Big(- \eta e_{\mathbf{H}_{ikj}} \mathbf{p}_i \Big) + \beta_2 \mathbf{s}_k$$

Gradients of $\mathcal{L}$ with respect to $\mathbf{W}_O$ *and* $\mathbf{W}_I$ Finally, we present the gradients of Eq. (5) w.r.t. $\mathbf{W}_O$ and $\mathbf{W}_I$, which are shown below, in respective order.

$$\frac{\partial \mathcal{L}(\mathbf{W_O})}{\partial \mathbf{W}_O} = \sum_{u_i \in \mathcal{U}} \Big(\gamma e_O \mathbf{u}_i^\top \Big) + \beta_3 \mathbf{W}_O$$

$$\frac{\partial \mathcal{L}(\mathbf{W}_I)}{\partial \mathbf{W}_I} = \sum_{u_j \in \mathcal{U}} \Big(\gamma e_I \mathbf{v}_j^\top \Big) + \beta_3 \mathbf{W}_I$$

With update rules to optimize Eq. (5), we use SGD to optimize the framework using the combined training data $\mathcal{X} = \{\mathcal{T} \cup \mathcal{H}\}$, where $\mathcal{T}$ and $\mathcal{H}$ are the link and interaction training data, respectively. Note that although there are additional methods for optimizing matrix factorization-based methods, SGD has been shown to be both efficient and easy to tune, e.g., adaptive learning rates.

With gradients calculated above to optimize Eq. (5), the detailed optimization algorithm is presented in Algorithm 1. Next, we briefly introduce Algorithm 1. In line 1, we randomly initialize model parameters. In line 2, the learning data include links and interactions. From line 3 to line 14, we use stochastic gradient descent to optimize the framework. In particular, for each iteration, we first shuffle the data in line 4 and then update model parameters using gradient descent methods from line 5 to line 12. When having a signed user-user link training example, the algorithm utilizes lines 6 through 8 to calculate the gradients, as compared to when having an interaction training example, lines 9 through 11 are used. Then, on line 12, the model parameters for the respective part of the problem (based on whether we are updating on a signed link or interaction) can be updated using a gradient-based method.

Algorithm 1: The optimization method for the proposed framework LIP.

Input: $\mathcal{T} = \{(u_i, u_j) | \mathbf{T}_{ij} \neq 0\}$ be the set of pairs with links and $\mathcal{H} = \{(u_i, r_k, u_j) | \mathbf{H}_{ik} \neq 0, \mathbf{A}_{jk} \neq 0\}$ be the set of interaction triplets

Output: $\mathbf{U}$ and $\mathbf{V}$ for link predictions; and $\mathbf{P}, \mathbf{Q}$, and $\mathbf{S}$ for interaction polarity predictions

1 Randomly initialize $\mathbf{U}, \mathbf{V}, \mathbf{P}, \mathbf{Q}, \mathbf{S}, \mathbf{W}_O, \mathbf{W}_I$
2 Construct the learning data set $\mathcal{X} = \{\mathcal{T} \cup \mathcal{H}\}$
3 **while** Not convergent **do**
4   Shuffle($\mathcal{X}$)
5   **foreach** $x \in \mathcal{X}$ **do**
6   **if** $x \in \mathcal{T}$ **then**
7   Calculate gradients of $\mathcal{L}(\mathbf{U}, \mathbf{V}, \mathbf{P}, \mathbf{Q}, \mathbf{S}, \mathbf{W}_I, \mathbf{W}_O)$ w.r.t. $\mathbf{u}_i, \mathbf{v}_j, \mathbf{W}_I$, and $\mathbf{W}_O$
8   **end**
9   **if** $x \in \mathcal{H}$ **then**
10   Calculate gradients of $\mathcal{L}(\mathbf{U}, \mathbf{V}, \mathbf{P}, \mathbf{Q}, \mathbf{S}, \mathbf{W}_I, \mathbf{W}_O)$ w.r.t. $\mathbf{p}_i, \mathbf{q}_j, \mathbf{s}_k, \mathbf{W}_I$, and $\mathbf{W}_O$
11   **end**
12   Update the respective parameters using gradient descent methods
13   **end**
14 **end**

5 Experiments

In this section, we conduct experiments to answer the following two questions: (1) Can our joint model help alleviate the sparsity problem in these two prediction tasks? (2) Do the terms based upon correlated user opinions/behaviors in LIP provide a transfer of information between the two problems? To address the first question, we perform experiments in which we increase the sparsity of the training data and compare the performance with representative baselines. We address the second question by examining whether our algorithm is robust to handle some cold-start users. In the next subsection, we will further introduce our dataset and how it was used, the metric used in evaluating the two prediction tasks, and then, we introduce the experimental settings for the two types of experiments we have performed.

5.1 Experimental settings

As mentioned in Sect. 3, we have collected a dataset from Epinions for these experiments. Note that for the purpose of this study, we have filtered our collected Epinions dataset to form more dense user-user and user-content matrices. The first step is to preprocess the data such that we have the appropriate training, validation and testing sets from our dataset.

The filtering we perform only keeps users that have both given and received a link and also requires the users to have given at least one helpfulness rating and have also authored at least one review that has received at least one helpfulness rating. For all selected users to be filtered out, we remove all their user links, reviews they had written and helpfulness ratings associated with that user. The reason for this filtering is that it will allow us to later remove portions of the data to artificially create training sets that have a varying percentage of cold-start users and also different levels of sparsity and therefore seemingly becoming more similar to the raw dataset.

The original dataset had contained 233,429 users, 841,373 user-user links and 13,668,105 helpfulness ratings. After the above-mentioned filtering process, we were left with 29,901 users, 600,976 user-user links and 11,555,599 helpfulness ratings. The dataset has been randomly split into 70% for training, 10% for validation and 20% for testing. Note that we then balanced our testing dataset to be 50% positive and 50% negative similar to that done in Leskovec et al. (2010a).

To evaluate and compare the performance of LIP,[2] we present the F1 measure for the interaction polarity and the link prediction tasks. Note that the higher the value, the better the performance. F1 measure (the harmonic mean of precision and recall) is formally defined as follows:

$$F1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

For all the models that required parameters to be tuned, we used the validation set to obtain the best parameters for each respective model.

Also, the parameter settings for each experiment was fixed (e.g., all LIP results for the five varying cold-start experiments were selected based on a commonly "good" set of parameters for all five percentages, and not separately for each of the five). However, between the two experiments, we allowed for different parameters as the dynamics of cold-start users and varying the amount of induced sparsity required a different set of parameters for our model and similarly for the baselines.

5.2 Sparsity experiments

To answer the first question, we compare the proposed framework, LIP, with existing interaction polarity and link prediction methods. We first present the baselines for the interaction polarity prediction task followed by those for the link prediction task.

We choose the following representative interaction polarity prediction baselines for comparison:

- *uCF* User-based collaborative filtering approach where we used the five most similar users (in terms of cosine similarity) based on their helpfulness rating history for making the predictions. For details on collaborative filtering, please see Su and Khoshgoftaar (2009). We use the user-based collaborative filtering approach as our first baseline for predicting the user interaction polarities. Here, we present the results where we used the five most similar users (in terms of cosine similarity) based on their helpfulness rating history for making the predictions.
- *MF* Our low-rank matrix factorization method as shown in Eq. (2). Here, a comparison is made with the low-rank matrix factorization method that attempts to find a lower-dimensional representation of the user-review matrix. Note this follows the same formulation as that in Eq.(2) where we use the matrices **P**, **Q**, **S** and **H** equivalently as they are in LIP for the predictions.

For link prediction, the representative baselines are presented below and details of the methods can be found in their respective cited work.

- *SSA* A spectral-based method using the signed Laplacian matrix (Kunegis et al. 2010) and regularized Laplacian kernel (Ito et al. 2005) is used. Due to the fact that this method was presented for undirected networks, we convert the directed link information by making **T** symmetric, thus resulting in an undirected network, we use the undirected version of the dataset by removing the directions of the links, but keep the testing set the same.
- *HOC-3* It is an approach that was based on the social balance and status theories (Leskovec et al. 2010b). Features for a supervised approach are extracted from triads and also node features (e.g., number of incoming positive edges). A total of 23 features are created based on 16 possible directed triad configurations and seven node features. The details of this method can be found in Leskovec et al. (2010a).
- *MF* Low-rank matrix factorization method as shown in Eq. (1), which was first introduced in Hsieh et al. (2012). The final comparison is with the low-rank matrix factorization method, which was first introduced for this problem in Hsieh et al. (2012). This is the natural baseline predictor for our model since LIP is built upon this MF technique. This method optimizes the squared error, has the regularization parameter β and uses SGD. We note that it is formulated just as shown in Eq.(1) and the

[2] https://github.com/TylersNetwork/link-interaction-polarity.

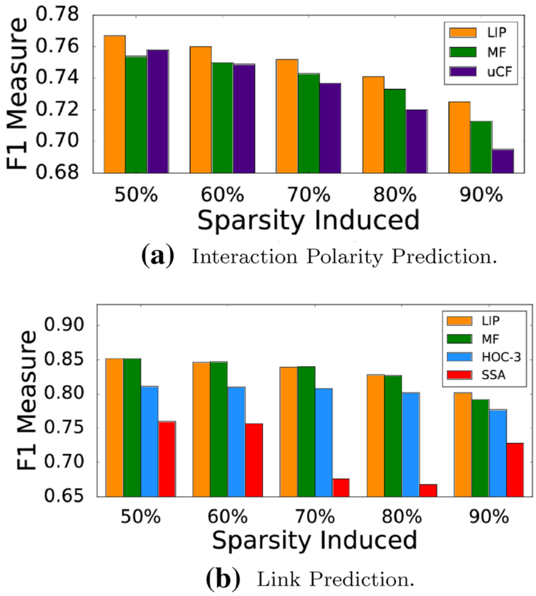


Fig. 5 Experimental results with varied sparsity settings

matrices **U**, **V** and **T** are used equivalently to those found in LIP.

In the first experiment, we are able to simulate a ranging sparsity across each user, since we have already limited our attention to a subset of the data that is denser than the original dataset. We remove x% of the links and interactions for each user and vary x in {50, 60, 70, 80, 90}. we are able to simulate a ranging sparsity across each user. We vary the sparsity of the dataset by removing 50–90% of the data, in increments of 10%.

5.2.1 Experimental results

The interaction polarity prediction results can be found in Fig. 5a. Most of the times, we see that the baseline MF method outperforms the user-based collaborative filtering method. Similarly, we have LIP finding significant gains over MF across the levels of sparsity induced. *Another thing to mention is that since we had first increased the density of the user-review matrix, it is not until the 80% sparsity that the density of the network drops below that of the original matrix* **H**. Therefore, in fact at 80% sparsity, the density of this induced sparse network is quite similar to that of the original network. We report the results of the sparsity experiments for the link prediction in Fig. 5b. LIP and MF obtain much better performance than SSA and HOC-3. We are able to observe that LIP performs comparable to the MF method for the lower sparsity settings, but upon reaching the higher sparsity level, LIP achieves better performance than MF.

From the results in the sparsity experiment, we have seen LIP's ability to help alleviate the sparsity problem found in the interaction polarity and link prediction tasks, thus providing evidence that our joint framework is able to partially alleviate the sparsity problem inherent in signed networks. More specifically, we see a significant improvement in the interaction polarity predictions and increasing improvement for the link prediction with the increase in the sparsity.

5.3 Cold-start experiments

Note that one of the main contributions of this work is the ability of the framework to handle not just the data sparsity problem, but also to help alleviate issues that are commonly faced with cold-start users in signed networks, which are quite common characteristics in these datasets. Therefore, to answer the second question, we compare LIP with existing algorithms that are able to handle cold-start users in both of the two prediction tasks.

For this experiment, we want to empirically evaluate the robustness of LIP when faced with networks having cold-start users. Note that this is a very difficult problem to overcome due to the fact if there is no knowledge about a user in a certain domain, then it becomes difficult, if not impossible, to make reasonable predictions involving them. However, since LIP is jointly predicting the signed links and user interaction polarities, the opinions formulated in one task can power those in the other task and simultaneously they should be able to gain information for users that previously had none in one of the tasks.

Under the cold-start setting, we choose the following user interaction polarity prediction baselines:

- *RG* The random guessing method for user interactions first calculates the class distributions and then selects randomly based on that distribution to make predictions for unknown values.
- *AvgG* The average guessing method (AvgG) first calculates the average interaction value found in the entire training set, next it predicts that value for all missing values and then it predicts that same value for all other edges in the network that have yet to be assigned.
- *MFwRG* We note that the typical matrix factorization method would not be applicable in this experiment, since if we have no training information for a given user, then the latent vectors of such users would never be updated. Thus, this would leave the predicted value to be assigned based on the dot product of two randomly initialized vectors. So instead we modify *MF* by adding the condition that if either of the two users' vectors have not been updated (i.e., they had no data in the training set and thus are a cold-start user), then instead of using the dot

Table 2 Interaction polarity prediction cold-start results

Method	Induced percent cold-start users				
	5%	10%	15%	20%	25%
RG	0.655	0.655	0.655	0.655	0.655
AvgG	0.667	0.667	0.667	0.667	0.667
MFwRG	0.769	0.764	0.754	0.746	0.739
LIP	0.773	0.771	0.769	0.766	0.763

Table 3 Link prediction cold-start results

Method	Induced percent cold-start users				
	5%	10%	15%	20%	25%
RG	0.641	0.641	0.641	0.641	0.640
MFwRG	0.848	0.837	0.825	0.813	0.797
LIP	0.860	0.858	0.853	0.848	0.839

product as we normally would with *MF* for predicting links, we instead use the *RG* method for the given link.

We note that the typical matrix factorization method would not be applicable in this experiment, since if we have no interaction information for a given user, then the latent vectors of such users would never be updated. This would leave the predicted value to be assigned based on the inner product of two randomly initialized vectors. Thus, we modified *MF* by adding the condition that if either of the two users' vectors have not been updated (i.e., they had no training interaction data and are therefore a cold-start user), then instead of using the inner product as we normally would with *MF* for predicting links, we instead use the *RG* method for that given link.

We compare the proposed framework LIP with the following link prediction baselines:

- *RG* Randomly guess missing links to be positive or negative based on training data class distribution.
- *MFwRG* This method has the identical extension for the cold-start users as described in *MFwRG* for the interaction polarity prediction task.

For these experiments, we vary the percentage of users that become cold-start users in a given task, but do not modify the testing set. We randomly select x% of the users and remove all their links and then randomly select x% of the users (who we have not already selected) and remove their interaction information while varying x in {5, 10, 15, 20, 25}, i.e., the number of cold-start users from 5% of the training dataset users to 25%, in intervals of 5%, thus making five data subsets.

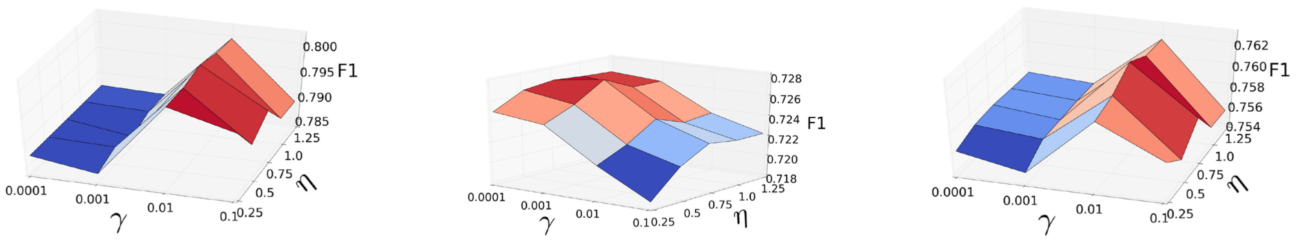
5.3.1 Experimental results

Table 2 holds the results of the cold-start experiments for the interaction polarity prediction task when varying the number of cold-start users. The very naive baseline RG is just shown to provide a reference for the F1 measure, but the *MFwRG* is expected to perform quite well. In this table, we are able to observe LIP's superiority over the baseline methods when observing cold-start users. We also see that LIP's performance as compared to the baselines drastically increases as the number of cold-start users increases, which is extremely intuitive based upon the use of the correlation terms. This is because even if a user has no current helpfulness rating information, LIP is able to transfer information (i.e., their opinions) through the linear mapping matrices $\mathbf{W}_O$ and $\mathbf{W}_I$ and use information that the user had from their link information.

In Table 3, we present the link prediction results when varying the amount of cold-start users in the training set. Upon seeing these results, the advantages of LIP over the other baseline methods become even more obvious. We note that whenever *MFwRG* has the ability to learn a low-dimensional representation for a user, it can then perform the prediction using its learned low-dimensional latent vectors. But when there is no link information for a given user, the user must resort to randomly guessing. Similarly to the interaction polarity prediction task, as the percentage of cold-start users increases, the performance gap in terms of F1 becomes larger in favor of LIP having the best prediction.

5.4 Discussions

This leads us back to our second question, where we set out to determine whether the linking terms based upon the correlated user opinions in LIP are able to provide a transfer of information between the two tasks that ultimately have a user's opinion in one task power the other. Based upon the results presented in this section, for both the sparsity and the cold-start experiments, we have shown that indeed, LIP is able to utilize the inherent correlations behind the opinions expressed in the two tasks to boost the performance in both the prediction tasks simultaneously. Next, we present our analysis on the parameters of LIP. We seek to not only to gain a better understanding of the relation between these two prediction tasks (i.e., η), but perhaps even more important in this study, and is the focus on γ, since it controlled the amount of opinion information to be transferred from one prediction task to the other, specifically the ones that control the correlation terms and the balance between optimizing the interaction polarity prediction task along with the link prediction task.



(a) Link prediction.

(b) Interaction polarity prediction.

(c) Trade-off between the two tasks (i.e., mean of 6(a) and 6(b)).

Fig. 6 Performance variations in LIP on the 90% data sparsity experiment w.r.t. η and γ

Based on the above experimental results, we have successfully verified our claim that our joint matrix factorization model uses additional terms for modeling the fact that users in social networks express their opinions in correlated ways across tasks when faced with sparse datasets. However, the most obvious claim we are now able to express is that LIP does indeed help alleviate the cold-start problem over the baseline MF method and the other baselines. In the next subsection, we perform a parameter analysis to gain a better understanding to not only the relation between these two prediction tasks (i.e., η), but perhaps more important in this study and is the focus on γ, since it controlled the amount of opinion information to be transferred from one prediction task to the other.

5.5 Parameter analysis for LIP

First, we will discuss the hyperparameters used in LIP. Thereafter, we discuss an analysis on some of the important hyperparameters in our model.

In this work, β_1, β_2 and β_3 are used as the typical regularization hyperparameters and we noticed they behave normally. In fact, they could be collapsed into a single regularization hyperparameter β without much change to the performance (as compared to splitting them into three separate hyperparameters). The other hyperparameters are quite necessary and typical for joint modeling (and similarly for cross-domain recommendation problems). For η, this is used to balance between the two tasks, which is assumed to result in large changes in performance when varying this hyperparameter greatly. This is because it controls to what extend the optimization is favoring higher performance (perhaps at the cost of the other) for one of the two problems over the other. As for γ, we have introduced this as a Lagrange multiplier used to solve this challenging optimization problem. In other words, based on our analysis, it appears there should be a transformation between the two domains of links and interactions, and to solve this problem, we have relaxed this constraint of finding such a mapping to instead find a mapping with minimal error (since we also assume the data is noisy). Hence, we introduce the hyperparameter γ to solve the optimization problem. Finally, we have K_L and K_I that denote the length of the representations in the link and interaction domains, respectively. These are the typical hyperparameters for embedding-based methods, and we have observed similar results as other methods that vary the embedding, i.e., the performance starts to increase, but then drops once the embedding becomes too large. Next, we will discuss an analysis on η and γ as these are the most interesting hyperparameters of LIP.

The parameters η and γ control the balance between optimizing the link prediction and user interaction polarity tasks, and how strongly to keep the two tasks low-dimensional representations correlated, respectively. In this subsection, we perform an analysis on how changing these two parameters affects the performance of LIP. We first fix all other parameters (i.e., the regularization parameters β_1, β_2 and β_3 and dimension sizes K_L and K_I) based upon the best parameters found against our validation set when performing a grid search over the parameter space. We evaluate the performance on all paired (η, γ) values, while we vary the value of η as 0.25, 0.5 0.75, 1.0, 1.25 and γ as 0.0001, 0.001, 0.01, 0.1, providing us with 20 possible combinations for running the grid search. Although the best parameter settings varied between the two above-mentioned experiments, we only display one representative from the sparsity user experiment, since we have similar observations in every other experimental setting. We present the analysis on the 90% sparsity experiment since it had the most variation in performance across the different settings.

In Fig. 6, we have shown the 3D surfaces for the mentioned combination of parameters. In Fig. 6a, we can see that $\gamma = 0.01$ is shown to clearly be a good region for this parameter, as both to the left and right, the performance in terms of F1 drops for the link prediction. However, there is little to no significant difference between the link predictions when varying η in the range provided. It can also be noticed that for the interaction polarity prediction task (as shown in Fig. 6b), the larger η leads to much better performance, which intuitively makes sense because a larger η relates to increasing the weight of how much we were to optimize the interaction polarity prediction as compared to the link prediction task. Unlike what we observed in the link prediction task, the interaction polarity prediction performs better with a smaller γ, meaning the two tasks have a different preferred weight to be associated with the correlation between the user latent vectors.

Finally, Fig. 6c shows that there is a drastic trade-off between the two tasks. Where if one of the tasks has a large increase in F1, then the other task becomes slightly worse. Thus, to obtain better performance in both tasks, we would want to choose a parameter setting such that the trade-off between the two tasks is balanced. Based on our analysis, such a point would have $\gamma = 0.01$, but as for the value of η, there is not a decisive value to choose. Thus, we have shown that the balance between optimizing the two tasks is not very sensitive, although from the figure, it appears choosing $\eta = 0.75$ has a slight advantage in both of the two tasks.

6 Related work

In this section, we present and discuss related work on signed networks, link prediction and interaction polarity predictions.

While some problems such as community detection (Chiang et al. 2014; Kunegis et al. 2010; Sharma 2012; Zheng and Skillicorn 2015) and centrality (Kunegis et al. 2009; Wu et al. 2016) have been extensively investigated, other directions such as network modeling (Li et al. 2018; Derr et al. 2018b), node relevance (Derr et al. 2018c; Wan et al. 2019) and network embedding (Wang et al. 2017; Derr and Ma 2018) are still in early development (Tang et al. 2016b) and many of their drawbacks have been that they are optimized each task one at a time.

Previous work on link prediction in signed networks can be split into two primary categories: supervised and unsupervised methods. The first of which formulates link prediction into a classification problem, having the existing positive and negative links as the labels and constructing features for each node and/or link. The unsupervised methods, however, can be further categorized into methods based on similarity (Javari and Jalili 2014; Symeonidis and Tiakas 2014) , propagation (Guha et al. 2004; Ziegler and Lausen 2005) and low-rank approximation (Agrawal et al. 2013; Cen et al. 2013; Hsieh et al. 2012).

It was in Leskovec et al. (2010a) that the supervised method, HOC-3, was first introduced. They had used the social balance theory to derive 16 features based upon the possible triad configurations and also included seven additional node features. Later in Chiang et al. (2011), HOC-3 was extended to higher-order cycles, and although it obtained slight improvements, it came at great time complexity costs when the network size becomes large, as compared to just local triads. Although they were able to have slight improvements in their predictions, it came at great time complexity cost for collecting the features as the network size becomes large. Thus, it is not as practical for current large real-world networks. The first low-rank approximation method for signed networks was presented in Hsieh et al. (2012), where multiple methods for matrix completion and matrix factorization were discussed, in which the later could be solved using either ALS or SGD and discussed multiple loss functions that could be interchanged (e.g., square hinge).

A more recent trend in signed network analysis is the use of advanced signed network embedding techniques. The specific goal of signed network embedding is to learn a set of vector representations that can be used in many tasks such as link prediction (Bhowmick et al. 2019; Lu et al. 2019; Islam et al. 2018), node classification (Bhowmick et al. 2019) and even visualization (Bhowmick et al. 2019). Most previous works have focused on representation learning (i.e., network embedding) for unsigned networks (Zhang et al. 2018) with a recent trend in using deep learning on graphs—graph neural networks (GNNs) (Wu et al. 2019). More recently graph convolutional networks (GCNs) (Hamilton et al. 2017; Kipf and Welling 2016) are a type of GNN that has been extended to signed network embedding (Huang et al. 2019; Derr and Ma 2018). We note that none of these methods are capable of handing the joint learning in their current state, but our framework could easily be modified from matrix factorization to another form of learning the embeddings, such as a signed network GNN.

The literature on the interaction polarity prediction is quite limited in comparison with the number of methods proposed for the classical link prediction task. It was in Tang et al. (2013) that the authors had the objective of specifically attempting to predict the rating a user would give the content generated by another user. Unlike our work, they included information about the content of the reviews, whereas we have only focused on predictions based upon the network information, although (as mentioned before) we have left this as a future work to include the content information as a means to gain even better prediction results. In Wang et al. (2015), they used the interactions for increased performance in recommendations to the users. This achieved better performance over the classical recommender system approaches primarily because they had included the role of users rating reviews as compared to only focusing on the information present in the reviews made by the user themselves. In Moghaddam et al. (2012), the authors focused on personalized predictions for review helpfulness where they presented a tensor factorization model. Note that we did not compare with tensor-based factorization methods due to the fact they require a higher time and space complexity, and instead, we had chosen to use matrix factorization as the base method to extend showing that the correlation between the two tasks is able to provide performance improvements. Another interaction-based sign prediction by Derr and Tang (2018); Derr et al. (2019) had been on the interactions of congress members and bills voted on in the US Congress where balance theory was studied in signed bipartite networks.

7 Conclusion and future work

In signed networks, users can express their opinions via two activities, i.e., creating signed links and expressing opinions on the content from others. Intuitively, the opinions and behaviors that the users have when performing these two activities online should be related. We first performed an analysis to validate the correlations between the signed links and user interaction polarities from both global and local perspectives. Our results show that indeed, there is a strong relation between the way users behave in expressing their opinions when performing these two aforementioned activities. We next proposed a joint optimization framework, LIP, for the prediction of signed links and interaction polarities that was built upon having the opinions in one task power the other.

This novel framework was able to boost the performance in both prediction tasks when jointly solving the two problems as compared to separately solving them individually. The significance becomes even more important in settings where the social network data are sparse or involve cold-start users. This is due to the fact that LIP is able to partially avoid and mitigate these problems since it can transfer information about users opinions from one problem to another by capturing the correlations between them. Our experiments on a real-world signed network have demonstrated both the effectiveness of LIP and also its robustness to the data sparsity and cold-start problems.

Future work in this domain will be to seek other problems that users might have correlated opinions or behaviors that can be harnessed to increase the performance in multiple tasks simultaneously. We also would like to investigate the underlying dynamics in signed networks that are causing these correlations, or other phenomenon, such as high reciprocity in some networks and not in others. More specifically how reciprocity relates to ways in which users express their opinions and perhaps sometimes even seek revenge. Furthermore, we plan to use this direction to perform the negative link prediction task (Tang et al. 2015; Abbasi et al. 2018; Shen et al. 2019) by harnessing interaction data for learning explicit negative links.

Acknowledgements The authors wish to thank the anonymous reviewers for their helpful comments. Tyler Derr, Zhiwei Wang, Jamell Dacon and Jiliang Tang are supported by the National Science Foundation (NSF) under Grant Numbers IIS-1714741, IIS-1715940, IIS-1845081 and CNS-1815636, and a Grant from Criteo Faculty Research Award.

References

Abbasi F, Muzammal M, Qu Q (2018) A decentralized approach for negative link prediction in large graphs. In: 2018 IEEE international conference on data mining workshops (ICDMW). IEEE, pp 144–150

Agrawal P, Garg VK, Narayanam R (2013) Link label prediction in signed social networks. In: Proceedings of the twenty-third international joint conference on artificial intelligence, IJCAI '13. AAAI Press, pp 2591–2597

Bhowmick AK, Meneni K, Mitra B (2019) On the network embedding in sparse signed networks. In: Pacific-Asia conference on knowledge discovery and data mining. Springer, pp 94–106

Cartwright D, Harary F (1956) Structural balance: a generalization of Heider's theory. Psychol Rev 63(5):277

Cen Y, Gu R, Ji Y (2013) Sign inference for dynamic signed networks via dictionary learning. J Appl Math 2013(3):708581

Chiang KY, Natarajan N, Tewari A, Dhillon IS (2011) Exploiting longer cycles for link prediction in signed networks. In: Proceedings of the 20th ACM international conference on Information and knowledge management. ACM, pp 1157–1162

Chiang KY, Hsieh CJ, Natarajan N, Dhillon IS, Tewari A (2014) Prediction and clustering in signed networks: a local to global perspective. J Mach Learn Res 15(1):1177–1213

Derr T (2020) Network analysis with negative links. In: Proceedings of the 13th international conference on web search and data mining. ACM

Derr T, Tang J (2018) Congressional vote analysis using signed networks. In: 2018 IEEE international conference on data mining workshops (ICDMW). IEEE, pp 1501–1502

Derr T, Ma Yao, J Tang (2018a) Signed graph convolutional networks. In: 2018 IEEE international conference on data mining (ICDM). IEEE

Derr T, Aggarwal C, Tang J (2018b) Signed network modeling based on structural balance theory. In: Proceedings of the 2018 ACM on conference on information and knowledge management. ACM

Derr T, Wang C, Wang S, Tang J (2018c) Relevance measurements in online signed social networks. In: Proceedings of the 14th international workshop on mining and learning with graphs (MLG)

Derr T, Wang Z, Tang J (2018d) Opinions power opinions: joint link and interaction polarity predictions in signed networks. In: Proceedings of the 2018 IEEE/ACM international conference on advances in social networks analysis and mining, ASONAM '18. ACM

Derr T, Johnson C, Chang Y, Tang J (2019) Balance in signed bipartite networks. In: Proceedings of the 28th ACM international conference on information and knowledge management. ACM, pp 1221–1230

Forsati R, Mahdavi M, Shamsfard M, Sarwat M (2014) Matrix factorization with explicit trust and distrust side information for improved social recommendation. ACM Trans Inf Syst (TOIS) 32(4):17

Forsati R, Barjasteh I, Masrour F, Esfahanian AH, Radha H (2015) Pushtrust: an efficient recommendation algorithm by leveraging trust and distrust relations. In: Proceedings of the 9th ACM conference on recommender systems. ACM, pp 51–58

Guha R, Kumar R, Raghavan P, Tomkins A (2004) Propagation of trust and distrust. In: Proceedings of the 13th international conference on World Wide Web. ACM, pp 403–412

Hamilton WL, Ying R, Leskovec J (2017) Representation learning on graphs: methods and applications. arXiv preprint arXiv:1709.05584

Heider F (1946) Attitudes and cognitive organization. J Psychol 21(1):107–112

Hsieh CJ, Chiang KY, Dhillon IS (2012) Low rank modeling of signed networks. In: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp 507–515

Huang J, Shen H, Hou L, Cheng X (2019) Signed graph attention networks. arXiv preprint arXiv:190610958

Islam MR, Prakash BA, Ramakrishnan N (2018) Signet: scalable embeddings for signed networks. In: Pacific-Asia conference on knowledge discovery and data mining. Springer, pp 157–169

Ito T, Shimbo M, Kudo T, Matsumoto Y (2005) Application of kernels to link analysis. In: Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. ACM, pp 586–592

Javari A, Jalili M (2014) Cluster-based collaborative filtering for sign prediction in social networks with positive and negative links. ACM Trans Intell Syst Technol (TIST) 5(2):24

Kipf TN, Welling M (2016) Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907

Kunegis J, Lommatzsch A, Bauckhage C (2009) The slashdot zoo: mining a social network with negative edges. In: Proceedings of the 18th international conference on World wide web. ACM, pp 741–750

Kunegis J, Schmidt S, Lommatzsch A, Lerner J, De Luca EW, Albayrak S (2010) Spectral analysis of signed graphs for clustering, prediction and visualization. In: Proceedings of the 2010 SIAM international conference on data mining. SIAM, pp 559–570

Leskovec J, Huttenlocher D, Kleinberg J (2010a) Predicting positive and negative links in online social networks. In: Proceedings of the 19th international conference on World wide web. ACM, pp 641–650

Leskovec J, Huttenlocher D, Kleinberg J (2010b) Signed networks in social media. In: Proceedings of the SIGCHI conference on human factors in computing systems. ACM, pp 1361–1370

Li L, Gu K, Zeng A, Fan Y, Di Z (2018) Modeling online social signed networks. Physica A 495:345–352

Lu C, Jiao P, Liu H, Wang Y, Xu H, Wang W (2019) SSNE: status signed network embedding. In: Pacific-Asia conference on knowledge discovery and data mining. Springer, pp 81–93

Ma H, Lyu MR, King I (2009) Learning to recommend with trust and distrust relationships. In: Proceedings of the third ACM conference on Recommender systems. ACM, pp 189–196

Moghaddam S, Jamali M, Ester M (2011) Review recommendation: personalized prediction of the quality of online reviews. In: Proceedings of the 20th ACM international conference on Information and knowledge management. ACM, pp 2249–2252

Moghaddam S, Jamali M, Ester M (2012) Etf: extended tensor factorization model for personalizing prediction of review helpfulness. In: Proceedings of the fifth ACM international conference on Web search and data mining. ACM, pp 163–172

Sharma T (2012) Finding communities in weighted signed social networks. In: 2012 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM). IEEE, pp 978–982

Shen P, Liu S, Wang Y, Han L (2019) Unsupervised negative link prediction in signed social networks. Math Probl Eng 2019:7348301

Su X, Khoshgoftaar TM (2009) A survey of collaborative filtering techniques. Adv Artif Intell 2009:4

Symeonidis P, Mantas N (2013) Spectral clustering for link prediction in social networks with positive and negative links. Soc Netw Anal Min 3(4):1433–1447

Symeonidis P, Tiakas E (2014) Transitive node similarity: predicting and recommending links in signed social networks. World Wide Web 17(4):743–776

Tang J, Gao H, Hu X, Liu H (2013) Context-aware review helpfulness rating prediction. In: Proceedings of the 7th ACM conference on Recommender systems. ACM, pp 1–8

Tang J, Hu X, Liu H (2014) Is distrust the negation of trust? The value of distrust in social media. In: Proceedings of the 25th ACM conference on Hypertext and social media. ACM, pp 148–157

Tang J, Chang S, Aggarwal C, Liu H (2015) Negative link prediction in social media. In: Proceedings of the eighth ACM international conference on web search and data mining. ACM, pp 87–96

Tang J, Aggarwal C, Liu H (2016a) Node classification in signed social networks. In: Proceedings of the 2016 SIAM international conference on data mining. SIAM, pp 54–62

Tang J, Chang Y, Aggarwal C, Liu H (2016b) A survey of signed network mining in social media. ACM Comput Surv (CSUR) 49(3):42

Victor P, Cornelis C, De Cock M, Teredesai A (2009) Trust-and distrust-based recommendations for controversial reviews. In: Web Science Conference (WebSci'09: Society On-Line), 161

Victor P, Verbiest N, Cornelis C, Cock MD (2013) Enhancing the trust-based recommendation process with explicit distrust. ACM Trans Web (TWEB) 7(2):6

Wan C, Fang Y, Wang C, Lv Y, Tian Z, Wang Y (2019) SignRank: a novel random walking based ranking algorithm in signed networks. Wirel Commun Mobile Comput 2019:4813717

Wang S, Tang J, Liu H (2015) Toward dual roles of users in recommender systems. In: Proceedings of the 24th ACM international on conference on information and knowledge management. ACM, pp 1651–1660

Wang S, Tang J, Aggarwal C, Chang Y, Liu H (2017) Signed network embedding in social media. In: Proceedings of the 2017 SIAM international conference on data mining. SIAM, pp 327–335

Wu Z, Aggarwal CC, Sun J (2016) The troll-trust model for ranking in signed networks. In: Proceedings of the Ninth ACM international conference on Web Search and Data Mining. ACM, pp 447–456

Wu Z, Pan S, Chen F, Long G, Zhang C, Yu PS (2019) A comprehensive survey on graph neural networks. arXiv preprint arXiv:190100596

Zhang D, Yin J, Zhu X, Zhang C (2018) Network representation learning: a survey. IEEE Trans Big Data 6(1):3–28

Zheng Q, Skillicorn DB (2015) Spectral embedding of signed networks. In: Proceedings of the 2015 SIAM international conference on data mining. SIAM, pp 55–63

Ziegler CN, Lausen G (2005) Propagation models for trust and distrust in social networks. Inf Syst Front 7(4–5):337–358