The Co-Evolution Model for Social Network Evolving and Opinion Migration

Yupeng Gu
University of California, Los Angeles
Los Angeles, CA
ypgu@cs.ucla.edu

Yizhou Sun
University of California, Los Angeles
Los Angeles, CA
yzsun@cs.ucla.edu

Jianxi Gao
Northeastern University
Boston, MA
j.gao@neu.edu

ABSTRACT

Almost all real-world social networks are dynamic and evolving with time, where new links may form and old links may drop, largely determined by the homophily of social actors (i.e., nodes in the network). Meanwhile, (latent) properties of social actors, such as their opinions, are changing along the time, partially due to social influence received from the network, which will in turn affect the network structure. Social network evolution and node property migration are usually treated as two orthogonal problems, and have been studied separately. In this paper, we propose a co-evolution model that closes the loop by modeling the two phenomena together, which contains two major components: (1) a network generative model when the node property is known; and (2) a property migration model when the social network structure is known. Simulation shows that our model has several nice properties: (1) it can model a broad range of phenomena such as opinion convergence (i.e., herding) and community-based opinion divergence; and (2) it allows to control the evolution via a set of factors such as social influence scope, opinion leader, and noise level. Finally, the usefulness of our model is demonstrated by an application of co-sponsorship prediction for legislative bills in Congress, which outperforms several state-of-the-art baselines.

CCS CONCEPTS

•**Information systems** →**Data mining;**

KEYWORDS

Dynamic networks; network generation models; co-evolution

1 INTRODUCTION

Social network analysis has become prevalent as the variety and popularity of information networks increase. In the real world, networks are evolving constantly with links joining and dropping over time. Meantime, properties of social actors in these networks, such as their opinions, are constantly changing as well. One example is the political ideology migration for two parties in U.S. Figure 1 shows the 1-dimensional mean ideology for members in two political parties via ideal point estimation using their historical voting records [12]. A similar discovery can be seen in [2]. We can clearly observe the divergence of ideologies of the two communities (i.e. the Democrats and Republicans), especially the polarization trend since 1960s. A natural question raises, *why such divergence happens and is there any possible intervention we can have to alleviate such polarization?* In this paper, we attempt to interpret this phenomenon and thus propose a unified co-evolution model for link evolution as well as (latent) node property migration in social networks.

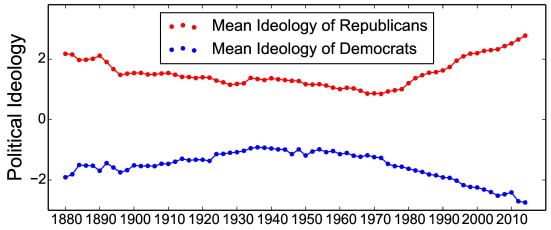


Figure 1: Ideology migration of the two parties in U.S.

On one hand, people in social networks exhibit great diversity and are associated with different properties (e.g., hidden properties such as political ideology). Interactions between individuals are more likely to happen within people that are alike, described as "homophily" in social network analysis [28]. With this principle, network generative models such as blockmodels [18, 44] and latent space models [17] have emerged, where each individual is assigned with a feature vector denoting her latent properties (i.e., a position in a latent space). Individuals that are close in the latent space are likely to have interactions in the network.

On the other hand, like flocks of collectively moving animals, people in social networks comprise a system of interacting, permanently moving units. In fact, the changing of location is ubiquitous among many kinds of creatures in real life: flocks of birds fly and migrate; colonies of ants and drones work and move to seek for foods. This phenomenon is also overwhelming in the realm of social network analysis, where people's latent position (e.g., ideology) are migrating with their crowds (e.g., parties). In other words, individuals are likely to be affected by their friends or who they interact with in the social network. This "social influence" [22, 41] assumption has been widely applied in literature. For example, in an information diffusion model, a person will be activated (i.e. the binary status is switched to "on") if she has enough activated neighbors [14].

Inspired by these observations, in this paper we propose a probabilistic co-evolution model that explains the evolution of networks

KDD'17, August 13–17, 2017, Halifax, NS, Canada.
 ISBN 978-1-4503-4887-4/17/08…$15.00
DOI: http://dx.doi.org/10.1145/3097983.3098002

as well as the migration of node properties, which contains two major components: (1) a network generative model when the node property is known; and (2) a property migration model when the social network structure is known. First, in terms of network evolution, similar to existing work, we assume the network is a reflection of node's latent properties. Our network generative model assumes (1) individuals have a higher chance to interact with people who are alike; and (2) opinion leaders attract more people and thus interact with more people. Second, in terms of property migration, we notice how creatures in biological systems and how particles in molecular systems propagate: they are influenced by their spatial neighbors to a large extent. We generalize the notion of "spatial neighbors" to "friends" in social network, and people's moving direction is influenced by their friends' moving directions.

Simulation shows that our model has several nice properties: (1) it can model a broad range of phenomena such as opinion convergence (i.e., herding) and community-based opinion divergence; and (2) it allows us to control the evolution via a set of factors such as social influence scope, opinion leader, and noise level. By learning system-level parameters via a series of historical snapshots of networks, predictions can be made about the evolution of the whole system in the future. We demonstrate the usefulness of our model by an application of co-sponsorship prediction for legislative bills in Congress, which outperforms several state-of-the-art baselines.

The contributions of our paper are summarized as follows:

- We propose a unified co-evolution model that captures the evolution of network structure as well as the migration of node properties.
- Under different system-level parameter settings, our model is able to exhibit different behaviors of network evolution and property migration.
- Our model is capable of inference via learning from real-world data. Empirical results reveal our advantage over state-of-the-art approaches in terms of a co-sponsorship prediction task.

2 PRELIMINARY OF COLLECTIVE MOTION

In the realm of biological systems, collective motion is one of the most common and spectacular manifestation of coordinated behavior [19, 43]. Flocks of birds fly and migrate uniformly as a group; ants are famous for their large and well-organized hierarchies, and individuals in each hierarchy exhibit highly coherent behaviors; a school of fish swim in a tightly organized way in terms of speed and direction. Collective motion is also observed in phase transition process as in many particle systems, and a well known line of work [42] describes their collective motion model as follows. Each particle moves at a constant rate v, while the direction of motion is determined by the average direction of all others within its neighborhood of radius r, plus some random perturbation. Denoting a particle n's position at time t by $\boldsymbol{x}_n(t)$, it is assumed to be updated according to

$$\frac{d}{dt}\boldsymbol{x}_n(t) = \boldsymbol{v}_n(t) \tag{1}$$

where $\boldsymbol{v}_n(t) = v \cdot (\cos\theta_n(t), \sin\theta_n(t))$ is its moving direction at t. The direction will be consistently adjusted by it spatial neighbors:

$$\theta_n(t+1) = \langle\theta_n(t)\rangle + \Delta\theta \tag{2}$$

where $\langle\theta_n(t)\rangle$ is the direction averaged by n's spatial neighbors within radius r, i.e. $\{m : ||\boldsymbol{x}_n(t) - \boldsymbol{x}_m(t)|| \leq r\}$. v is the absolute value of each particle's velocity and is assumed to remain the same for every particle during the transition process. Noise $\Delta\theta$ is randomly chosen uniformly from interval $[-\eta/2, \eta/2]$, where η controls the noise level.

Spatial neighbors play a crucial role in above systems. Notice that, however, in the setting of social networks, individuals are assumed to receive social influence only from their friends rather than anyone who are close to them. This inspires us to design the co-evolution model as introduced in next section.

3 THE CO-EVOLUTION MODEL

The position migration in biological and molecule systems mentioned in Section 2 are a good analogy to the opinion migration for individuals in social networks. Like flocks of collectively moving animals, people on social networks also comprise a system of interacting, permanently moving units in terms of latent opinions or stances. Different from biological systems, in social networks people form social ties where information propagate through. In other words, every individual is exposed to a group of "friends" and receives influence merely from them. This phenomenon is referred to as "social influence" or "social selection" [7, 22, 41] in literature. In turn, new/old links in social networks may form/drop as a result of individuals' opinion migration, due to "homophily" [28]. Since opinion is an important property of an entity, we use the terms *opinion*, *property* and *feature* interchangeably in this paper, to denote the intrinsic characteristics belonging to an individual on social network.

By putting (1) social influence-based opinion migration and (2) homophily-based network generation together, we then have our co-evolution model, which is introduced in the remaining of this section.

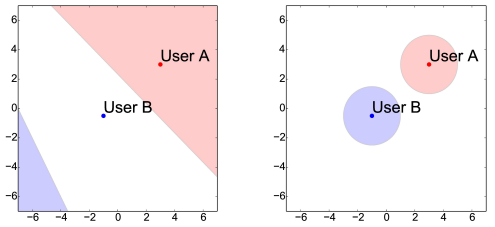
3.1 Social Network Generation

Latent space models [17] assume the snapshot of a static social network is generated based on the positions of individuals in an unobserved social space. This latent space consists of unobserved latent characteristics of people that represent potential tendencies in network relations. In these network generation models, the generation of each link is independent on each other, and is based purely on the positions of two users. We could design any score function $s : \mathcal{R}^K \times \mathcal{R}^K \to \mathcal{R}$ that assigns a score to a pair of node features $(\boldsymbol{x}_n, \boldsymbol{x}_m)$, which indicates the likelihood of observing the presence of the link in between. The score function is crucial to the network and its properties, and we discuss two possibilities below.

Dot Product-based Score Function. In tons of existing works, dot product of two features vectors is used to capture the similarity between them [3, 21, 30, 31, 40]. However, this generation model contradicts with the following observation.

Obviously, node degree is associated with the choice of score function. The higher chance of a node has to issue links to others, the larger degree it will be. Vector norm plays an important role in inner product; as a result, those actors with a large norm (i.e. $||\boldsymbol{x}_n||$) tend to attract interests from a large group of others, and thus become opinion leaders in the generation process. To demonstrate

this, we show the 2-dimensional position of two users A and B as well as their affected regions in Figure 2(a). The affected region of a user is defined as the set of people who can be influenced by her (i.e. their score function exceeds some threshold). User A has a position of $(3, 3)$ and B is located at $(-1, -0.5)$. It is obvious from the plot that user A are far more likely to befriend others (even those with less cosine similarity) than B, simply because A is further away from the origin than B is. In other words, people with extreme stances (i.e. large norms of latent feature vector) will become the opinion leader. However in most cases, the most popular people are either around the center of the entire population, or the center in their community. For example, it is found that radical politicians on the ideology spectrum are hardly party leaders [34]. In addition, each actor has limited resources and energy, which sets a constraint on one's spreadable radius. Preferably, the score function is invariant of the scale, and the affected region should have limited area (i.e. bounded).



(a) Inner product-based similarity (b) Distance-based similarity

Figure 2: Affected regions (colored area) for two users with different similarity functions. Nodes in the affected region are prone to interact with the corresponding user in the same color (i.e., red region for User A and blue region for User B).

Gravity-based Score Function. We recall that herds of animals have the notion of "spatial neighbors" when they migrate and collaborate. In molecule systems, nearby molecules also account for the majority of the interaction. Inspired by these observations, it is reasonable to set the score function between two users to be based on their Euclidean distance. We adapt the inverse squared gravity formula in our definition of score function. Using the new metric, we show the affected region of two users in Figure 2(b). Although the feature vectors of user A and B have different scales, the spaces of their friend candidates are comparable.

In the graph generation model, when we want to determine the link between two actors, the score function is mapped to a probability using Gaussian function:

$$p_{nm} = \exp(-\frac{1}{\epsilon^2} \cdot \frac{||\boldsymbol{x}_n - \boldsymbol{x}_m||^2}{b_n \cdot b_m}) \tag{3}$$

where ϵ is a model hyper-parameter, and $\{b_n\} \subset \mathbb{R}^+$ is another set of parameters which reflect the popularity of actors. The link will be generated if $p_{nm} > d$, where d is a system parameter which controls sparsity of the network, and a larger d means fewer neighbors an actor can interact with. For geometric interpretation, b_n is proportional to the radius of one's neighborhood, and opinion leaders will be the ones with largest values of b. In other words, opinion leaders are more likely (with higher probability) to interact with other actors. As the formula $\frac{b_n \cdot b_m}{||\boldsymbol{x}_n - \boldsymbol{x}_m||^2}$ resembles the law of gravity, we call this score function as gravity-based.

3.2 Opinion Migration

Similar to the migration of fish and flocks of birds, individuals in social networks also exhibit collective behaviors, which is modeled in this section.

Earlier work [6, 16, 36, 37, 45, 46] on modeling property change is quite straightforward: properties at adjacent timestamps (e.g. $\boldsymbol{x}^{\langle t \rangle}$, $\boldsymbol{x}^{\langle t+1 \rangle}$) are forced to be similar via various kinds of regularization/prior in order to avoid abrupt changes. For example, $\boldsymbol{x}^{\langle t+1 \rangle}$ is assumed to be generated from a Gaussian prior centered on its previous position $\boldsymbol{x}^{\langle t \rangle}$. However, this plausible strategy has two major flaws, which greatly reduce the power of the generation model.

First of all, let us investigate the activity of two actors in Figure 3. Here X-axis denotes the timestamp, and Y-axis denotes the 1-dimensional latent position. According to the migration prior defined above, the behavior of user X and Y are equally possible; however in real life, it is more likely to observe the trajectory of user Y (moving along the same direction) rather than X (oscillating). The same phenomenon is observed in flocks of animals as well: a school of fish tends to move towards some direction instead of wandering around some places.

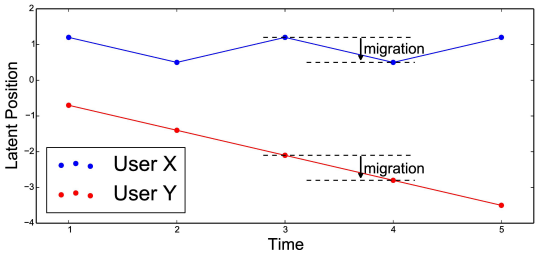


Figure 3: An example of two people's migration.

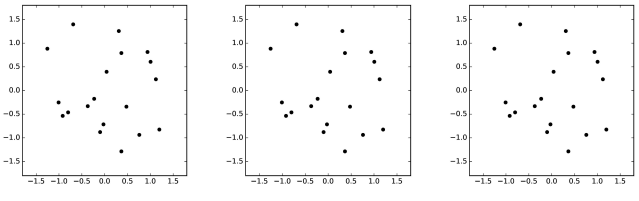
Secondly, social influence should be involved in the migration process, and the generation model should be able to express different properties of the random network under different system settings. For example, we may observe the polarization of opinions in some networks, i.e. multiple clusters of people heading towards different directions. However, if latent features evolve solely according to their previous positions, it is unlikely that individuals will automatically form several clusters.

In a recent work [16], social influence are included in the generation model. Simply generalizing their binary features into continuous features, we have

$$\boldsymbol{x}_n^{\langle t+1 \rangle} \sim \mathcal{N}\left((1-\lambda) \cdot \boldsymbol{x}_n^{\langle t \rangle} + \lambda \cdot \langle \boldsymbol{x}_n^{\langle t \rangle} \rangle, \sigma^2\right) \tag{4}$$

where $\langle \boldsymbol{x}_n^{\langle t \rangle} \rangle$ is the average position of user u_n's neighbors at time t, and $\mathcal{N}(\mu, \sigma^2)$ is the normal distribution with mean μ and variance

σ^2. A toy example of 2-dimensional feature migration under this framework is shown in Figure 4. We see that although two clusters emerge after several steps (nodes in the middle are going upwards and downwards), they are trapped in a local area and refuse to keep moving upwards or downwards since the clusters are formed. In other words, people's opinions will no longer change after communities are developed. The principal reason lies in that propagation model: the moving tendency of nodes is never captured; instead, entities update their positions arbitrarily, and they lack the motivation to move in a stable status.



(a) $t = 0$ (initial) (b) $t = 200$ (c) $t = 400$

Figure 4: Position migration of $N = 20$ nodes. 3 nearest neighbors of each node are chosen as friends in the network.

To overcome these problems, a natural approach is to track the historic features, such as setting a global regularization term in addition to features in adjacent steps. However, the lack of Markov property would make the generation process less intuitive and much more complex, and inference would be impossible due to high computational cost. Here we seek for a solution from the propagation in the nature. It is rare to observe a flock of animals turn around frequently; similarly, a person should gradually change her interest in some dimension (e.g. her enthusiasm of a topic may be dropping) instead of keeping switching between two viewpoints. Therefore, we keep track of velocity, i.e., the direction (which can be regarded as the first derivation of displacement), and punish its volatile changes.

Therefore, in terms of opinion migration, we introduce the direction that a user u_n is heading as an angle θ_n, and the latent position of each user will be updated according to the basic displacement formula:

$$\frac{d}{dt}\boldsymbol{x}_n(t) = v \cdot (\cos\theta_n(t),\ \sin\theta_n(t)) \tag{5}$$

where v is a constant factor indicating absolute speed, and the unit vector $(\cos\theta_n(t), \sin\theta_n(t))$ represents u_n's moving direction at time t. In reality, we observe discrete snapshots of social networks. Therefore, we write the above equation in its discrete form as

$$\boldsymbol{x}_n^{\langle t+1\rangle} = \boldsymbol{x}_n^{\langle t\rangle} + v \cdot (\cos\theta_n^{\langle t\rangle},\ \sin\theta_n^{\langle t\rangle}) \tag{6}$$

The remaining question is how $\theta_n^{\langle t\rangle}$ propagates. It is worth noticing how every member in a flock of birds picks its direction. When some flocks of birds head west and others head north, an observer bird is likely to pick either direction instead of south or east. During a migration, people are likely to take similar paths as their families and close friends. This strategy is believed to have advantages such as more efficient explorations for resources and improved decision making in larger groups [43]. In sum, it is very rare that a member chooses to behave oppositely to its friends. When it comes to social networks, people also adopt similar behaviors as their neighbors [22]. We probably have already observed the following facts in our real life. A scholar tends to raise interest in a research topic that is trending among her collaborators. A Democrat is likely to become more liberal, if she feels her acquaintances are going "left" (and vice versa). Social network provides exposure to one's neighbors, and this factor will be reflected in the formation of direction variables. Therefore in our model, a person's moving direction is assumed to be influenced by her neighbors' directions, and is subject to a noise of some magnitude:

$$\theta_n^{\langle t+1\rangle} \sim \mathcal{N}(\langle\theta_n^{\langle t\rangle}\rangle, \sigma^2) \tag{7}$$

where $\langle\theta_n^{\langle t\rangle}\rangle$ is the average direction of u_n's neighbors' (including herself) at time t. In the above case, when a bird observes 10 others heading west and 20 others heading north, the average direction of other birds is about 63° north of west. Therefore in most cases, the observer will fly in a similar direction (follows either the west or north group), as it would incur great penalty if it flies south or east instead. Intrinsically, the parameter σ controls how easily people are influenced by their neighbors (or how strictly a person should follow the trend of their neighbors): larger σ will relax the regularization.

In the discussion above, the dimension of node feature is set to 2 in order to make the propagation process more intuitive. Nevertheless, our method is not subject to this constraint and can be easily generalized to higher dimensional latent spaces using polar/hyperspherical coordinate systems [1]. For example, the direction $(\cos\theta_n(t),\ \sin\theta_n(t))$ in Equation 5 can be replaced by any dimensional unit-length vector with polar coordinates. The average direction determined by Equation 7 simply becomes the (normalized) vector summation. In the remaining of the paper, we will use 2-dimensional representations for visualization purposes.

Note that our regularization on the direction θ already implies the regularization of feature $\boldsymbol{x}$. This is trivial since the change of a variable is reflected in its first derivative. In particular, $||\boldsymbol{x}_n^{\langle t+1\rangle} - \boldsymbol{x}_n^{\langle t\rangle}||$ is fixed for every t, which means abnormal change in the feature space is impossible. Therefore, our model has further contributions while inheriting the advantages of existing propagation approaches.

3.3 Unified Model

Putting them together, the evolution of network and migration of entity opinions happen iteratively after each other in our co-evolution model. At each timestamp t, a network is generated given node latent features (homophily), and node directions are generated according to the network structure (social influence), thus determine the latent feature for the next timestamp $t + 1$ (migration). System-level parameters include sparsity parameter d which controls the sparsity of the graph (i.e. the average number of friends), and noise level σ which implies the deviation of one's direction from the expected value. The generative process of our co-evolution model is summarized in Algorithm 1.

input : number of users N; number of timestamps T; sparsity parameter d; noise level σ.
output: a series of graphs and users' latent positions.
initialization;
for $t = 1$ *to* T **do**
  // graph generation
  for $n, m = 1$ *to* N **do**
    calculate p_{nm};
    determine the link between n and m as $G_{nm}^{\langle t \rangle} = 1$ if $p_{nm} > d$;
  end
  // opinion migration
  if $t == 1$ **then**
    for $n = 1$ *to* N **do**
      sample $\theta_n^{\langle t \rangle} \sim \text{Uniform}[0, 2\pi)$;
      update $\boldsymbol{x}^{\langle t+1 \rangle} = \boldsymbol{x}^{\langle t \rangle} + v \cdot (\cos\theta_n^{\langle t \rangle}, \sin\theta_n^{\langle t \rangle})$;
    end
  else
    for $n = 1$ *to* N **do**
      sample $\theta_n^{\langle t \rangle} \sim \mathcal{N}(\langle \theta_n^{\langle t-1 \rangle} \rangle, \sigma^2)$;
      update $\boldsymbol{x}^{\langle t+1 \rangle} = \boldsymbol{x}^{\langle t \rangle} + v \cdot (\cos\theta_n^{\langle t \rangle}, \sin\theta_n^{\langle t \rangle})$;
    end
  end
end

Algorithm 1: Generation model for co-evolution

4 SIMULATION

To reveal the properties of our generation model, we run simulations and show the migration of individuals in the network for selected parameters. For initialization, every node is randomly assigned a 2-dimensional initial position in the lattice of $[-L/2, L/2] \times [-L/2, L/2]$ where $L = 5$, as well as a popularity $b \sim \text{Uniform}([1, 2])$. b will be fixed throughout the migration process. Initializations are identical across all parameter settings.

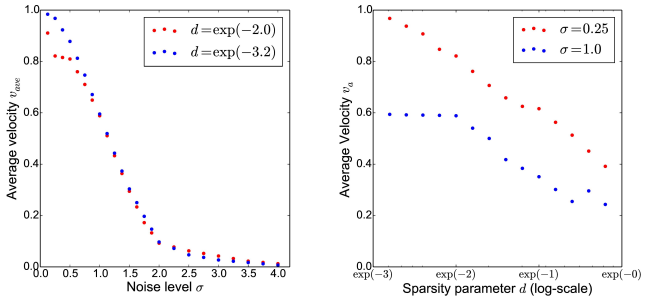
According to [42], we adopt the absolute value of average normalized velocity as a measure for the system status:

$$v_{ave} = \frac{1}{N} \left| \sum_{n=1}^{N} (\cos\theta_n, \sin\theta_n) \right| \tag{8}$$

$v_{ave} \in [0, 1]$ and in general, $v_{ave} = 1$ means completely coherent moving behavior, while $v_{ave} = 0$ means completely randomness, or two groups of equal number of people moving towards opposite directions. In Figure 5 we plot the metric v_{ave} under different parameter settings.

Noise level. Noise level σ controls how uniformly individuals proceed. Intuitively, a large σ will overwrite the direction determined by one's neighbors, thus leads to more random migration behaviors. In Figure 5(a) we can see $v_{ave} \approx 0$ for large σ. People tend to behave collectively in groups with small σ values.

Sparsity parameter. Sparsity parameter d plays a role in the emergence of clusters. A larger value of d leads to a sparser network, therefore people interact with only a few others. In this case, communities are allowed to maintain their own direction, and it is more likely to observe several clusters with different migration directions. On the other hand, when the threshold is small, an individual is easily linked to most others, therefore information is prone to spread through the entire network, making almost all the people to propagate coherently. In Figure 5(b) we can see v_{ave} is larger for smaller d values.



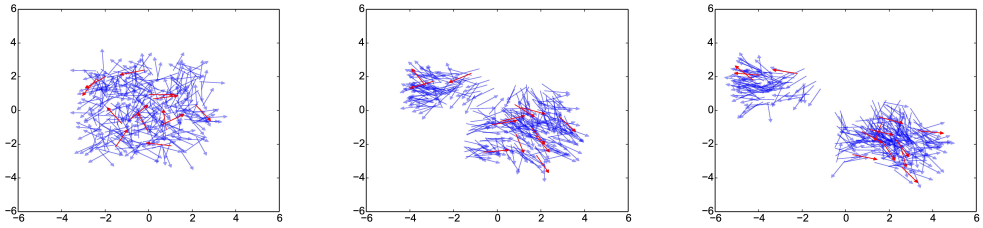
(a) Effect of noise level σ (b) Effect of sparsity parameter d

Figure 5: System-level parameter study

We show people's positions and their moving directions in Figure 6. Each row corresponds to a set of system-level parameters. Absolute value of velocity is set to $v = 0.03$ and moving directions are shown as unit-length arrows starting from one's position. Opinion leaders (top 5% people with largest b) are marked in red.

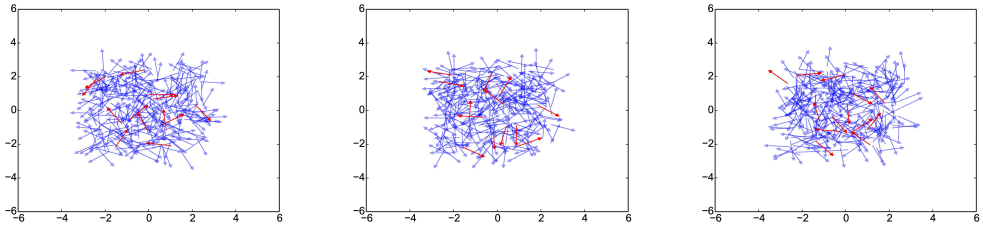
Observations. We can see in most cases, the opinion leaders are surrounded by others and appear in the center of a community, which agrees with our findings in Section 3.1. In addition, the effect of system-level parameters is also revealed in these examples: networks tend to be very random when noise level σ is large (comparing first and second row). Under a small noise level, sparsity parameter d comes into play: a small d makes the network denser, thus communities have more overlapping entities and are likely to act coherently; while a large d reduces the scope of individuals, and clusters may emerge and head towards different directions (comparing first and third row). In sum, initially, sparsity, small noise and different directions of opinion leaders are necessary in order for opinion convergence within each community, which eventually leads to emergence of clusters.

Intervention. Now back to the question raised in introduction section: how can we alleviate the divergence of communities? From the above observations, one solution is to reduce σ and enlarge d. Under this setting, people are exposed to many others, follow their directions without much perturbation and a uniform global trend is likely to occur. Another alternative is utilizing opinion leaders to advertise and propagate similar directions of migration. Thanks to their high popularity, they are likely to interact with more people in their neighborhood, and thus play a role in deciding others' directions. In Figure 6(a)-6(c), we already observe the emergence of two clusters with different directions. Following Figure 6(c), we flip and fix the directions of the three leaders in the left community as in Figure 7(a); as a result, people in the left cluster will gradually alter their directions following the leaders (Figure 7(a)-7(c)).



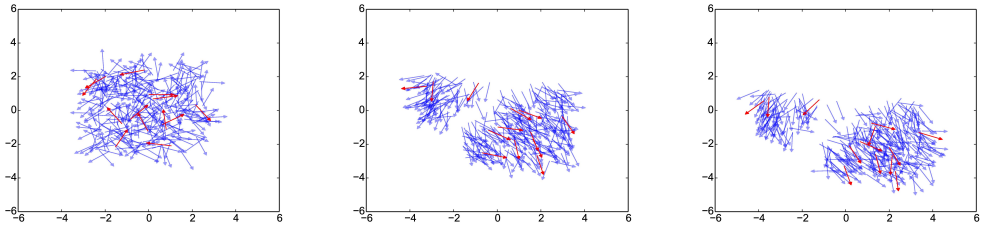
(a) $t = 0$ (init) (b) $t = 40$ (c) $t = 80$

Noise level $\sigma = 0.5$. Sparsity parameter $d = \exp(-0.4)$.



(d) $t = 0$ (init) (e) $t = 40$ (f) $t = 80$

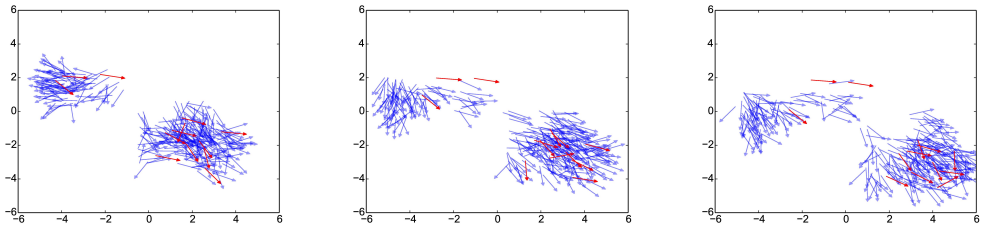
Noise level $\sigma = 2.0$. Sparsity parameter $d = \exp(-0.4)$.



(g) $t = 0$ (init) (h) $t = 40$ (i) $t = 80$

Noise level $\sigma = 0.5$. Sparsity parameter $d = \exp(-2.0)$.

Figure 6: Migration of entities in the network. Each row corresponds to one setting of system parameters.



(a) $t = 80$ (b) $t = 120$ (c) $t = 160$

Figure 7: Role of opinion leaders (under the same setting: $\sigma = 0.5$, $d = \exp(-0.4)$).

5 APPLICATION

Apart from the capability of modeling opinion migration and network evolution, a good generation model should be able to explain and predict the behavior of objects given observed data. In general, node properties could be regarded as vector representations or explanatory variables of a node, and are also referred to as node embeddings in some work (e.g. [40]). They usually convey meanings dependent on the network and context, and are flexible enough

to be inferred given a variety of real-world networks. In this section we show an application of our co-evolution model, where we predict the cosponsors of bills in the future. Here the node latent properties can be treated as multi-dimensional political ideology as in [5, 32].

5.1 Dataset

Co-sponsorship dataset. A *sponsor* of a bill is a legislator (usually a member from the congress) who introduces a bill or resolution for consideration. A *cosponsor* is another congress member who adds his or her name as a supporter to the sponsor's bill. Cosponsorship contains important information about the social support network between legislators: the closer the relationship between a sponsor and a cosponsor, the more likely it is that the sponsor has directly petitioned the cosponsor for support [11]. We crawled the legislative bills[1] from 1983 (98*th* congress meeting) till now (114*th* congress meeting), with a timeframe of 34 years. For bills with a sponsor, we extract all the cosponsors and build links between them. The minimal time unit is set to one month, and we use $H^{\langle t \rangle}$ to denote all the cosponsor links in month t. In order to make the evolution process smoother, a snapshot of network $G^{\langle t \rangle}$ consists of all the people and their cosponsor links within a 12-month period up to month t, and the time window is shifted forwards one month at a time. In other words, $G^{\langle t \rangle} = H^{\langle t-11 \rangle} \cup H^{\langle t-10 \rangle} \cup \cdots \cup H^{\langle t \rangle}$. Therefore, this series of graphs starts at $t_0 = 12$ and $G^{\langle t_0 \rangle}$ contains all the cosponsorship links from Jan. 1, 1983 to Jan. 1, 1984; $G^{\langle t_0+1 \rangle}$ contains all the cosponsorship links from Feb. 1, 1983 to Feb. 1, 1984, and so on. This series of evolving networks contain $T = 382$ time slices, $N = 2,180$ legislators, 130,692 bills and 2.1 million cosponsorship links in total.

5.2 Fitting the Data

A graphical model representing our model is shown in Figure 8.

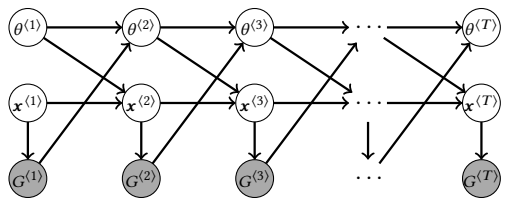


Figure 8: Graphical model representation of our model. Shadowed units represent observed variables.

Our model becomes a probabilistic model during the inference process, therefore each link is no longer deterministically established by a threshold d. The optimal parameters are inferred by maximizing the joint probability of G, $X = \{\boldsymbol{x}^{\langle t \rangle}\}_{t=1}^{T}$, $\Theta = \{\theta^{\langle t \rangle}\}_{t=1}^{T}$ and $\boldsymbol{b} = \{b_n\}_{n=1}^{N}$. From Figure 8 we have

$$X, \Theta, \boldsymbol{b} = \underset{X,\Theta,\boldsymbol{b}}{\operatorname{argmax}} \prod_{t=1}^{T} p(G^{\langle t \rangle} | \boldsymbol{x}^{\langle t \rangle}, \boldsymbol{b}) \cdot \prod_{t=2}^{T} p(\theta^{\langle t \rangle} | \theta^{\langle t-1 \rangle}, G^{\langle t-1 \rangle}) \tag{9}$$

[1]Data are collected at https://www.govtrack.us

s.t.

$$\boldsymbol{x}_n^{\langle t+1 \rangle} = \boldsymbol{x}_n^{\langle t \rangle} + v \cdot (\cos \theta_n^{\langle t \rangle}, \sin \theta_n^{\langle t \rangle}), \ \forall n, t \tag{10}$$

The constraint (Equation 10) makes it much harder to achieve a global estimation of parameters. Therefore, we adopt an approach similar to coordinate ascent algorithm, and update X, Θ and $\boldsymbol{b}$ given each other iteratively.

Update $\boldsymbol{b}$. $\boldsymbol{b}$ can be directly updated using traditional methods (e.g. stochastic gradient ascent) under a unconstrained optimization setting.

Update X and Θ. Initially ($t = t_0$), the optimal positions $\boldsymbol{x}^{\langle t_0 \rangle *}$ are estimated by maximizing the likelihood of the first observed graph:

$$\boldsymbol{x}^{\langle t_0 \rangle *} = \underset{\boldsymbol{x}^{\langle t_0 \rangle}}{\operatorname{argmax}} p(G^{\langle t_0 \rangle} | \boldsymbol{x}^{\langle t_0 \rangle}, \boldsymbol{b}) \tag{11}$$

and directions $\theta^{\langle t_0 \rangle}$ are initialized uniformly at random in $[0, 2\pi)$. Latent features at the next step $\boldsymbol{x}^{\langle t_0+1 \rangle}$ are updated deterministically by our propagation model (Equation 6).

After that, for each timestamp t ($t \geq t_0 + 1$), given the present position $\boldsymbol{x}^{\langle t \rangle}$, previous direction $\theta^{\langle t-1 \rangle}$ and the next graph $G^{\langle t+1 \rangle}$, we are able to estimate $\theta^{\langle t \rangle *}$ according to:

$$\begin{aligned} \theta^{\langle t \rangle *} &= \underset{\theta^{\langle t \rangle}}{\operatorname{argmax}} \ \log p(\theta^{\langle t \rangle} | G^{\langle t+1 \rangle}, \theta^{\langle t-1 \rangle}, \boldsymbol{x}^{\langle t \rangle}, \boldsymbol{b}) \\ &= \underset{\theta^{\langle t \rangle}}{\operatorname{argmax}} \ \Big(\log p(G^{\langle t+1 \rangle} | \theta^{\langle t \rangle}, \boldsymbol{x}^{\langle t \rangle}, \boldsymbol{b}) + \log p(\theta^{\langle t \rangle} | \theta^{\langle t-1 \rangle}, G^{\langle t-1 \rangle}) \Big) \\ &= \underset{\theta^{\langle t \rangle}}{\operatorname{argmax}} \ \Big(\log p(G^{\langle t+1 \rangle} | \boldsymbol{x}^{\langle t+1 \rangle}, \boldsymbol{b}) + \log p(\theta^{\langle t \rangle} | \theta^{\langle t-1 \rangle}, G^{\langle t-1 \rangle}) \Big) \end{aligned} \tag{12}$$

This concludes an outer-iteration of parameter update. We plot the objective versus the number of outer iterations in Figure 9. Empirically, only a few iterations are needed for convergence, and we let the number to be 3 in all following experiments.

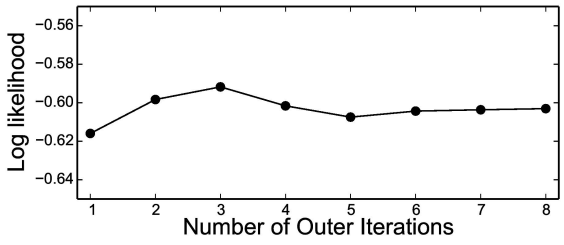


Figure 9: The log likelihood when parameters are updated for multiple rounds.

5.3 Baselines

We compare our co-evolution model of network structure and node opinions (CoNN) with the following baseline methods. For fair comparison, we compare with several models designed for dynamic networks, and dimension of latent features is set to $K = 2$ in all methods. The absolute value of velocity is fixed to be $v = 3 \times 10^{-3}$ in order for the process to be smoother. We let $\sigma = 1$ and $\epsilon = 0.8$ in our method for now. Parameter studies at the end of this section reveal that our method is not sensitive to these parameters.

- CoNN$_{dot}$: The first baseline is a variant of our model where the probability of a link involves a dot product: $p(G_{nm}=1) = 1/(1+e^{-(\boldsymbol{x}_n \cdot \boldsymbol{x}_m + b_n + b_m)})$. $\{b_n\}$ is a set of variables with meaning similar to our CoNN model.
- Latent feature propagation model (LFP) [16]: The second baseline is the binary latent feature propagation model. Local optimization is adopted in order for their method to scale with our data. The authors kindly share their code.
- Dynamic social network with latent space models (DSNL) [36]: We implement the dynamic social network analysis approach where no social neighbors are considered in propagation. Latent features evolve purely according to their previous positions.
- Phase transition model (PTM) [42]. This approach is proposed to model the behavior of molecules during a phase transition. Directions of molecules are treated as the parameter, and they propagate according to the average of their spatial neighbors. We use our estimated features at the beginning as their initialization, and run simulation for T steps.

We also compare with a state-of-the-art baseline method designed for static networks.

- Large-scale information network embedding (LINE) [40]. This approach embeds information network into low-dimensional vector spaces. We apply LINE on static snapshots of the social graph, and treat the embeddings as node features.

5.4 Co-sponsorship Prediction

In this task, we demonstrate the advantage of our co-evolution model by predicting cosponsors in the future. Specifically, given the observed cosponsor links up to time t_1, a bill in future time t_2 ($t_2 > t_1$) and its sponsor u_n, our goal is to predict the users who will cosponsor it.

Given $G^{\langle t_0:t_1\rangle}$, we are able to learn $\theta^{\langle t_0:t_1-1\rangle}$ and $\boldsymbol{x}^{\langle t_0:t_1\rangle}$. After that, the latent features propagate according to our evolution model, namely

$$\begin{aligned} \theta_n^{\langle s\rangle} &\sim \mathcal{N}(\langle\theta_n^{\langle s-1\rangle}\rangle, \sigma^2) \\ \boldsymbol{x}_n^{\langle s+1\rangle} &= \boldsymbol{x}_n^{\langle s\rangle} + v \cdot (\cos\theta_n^{\langle s\rangle},\ \sin\theta_n^{\langle s\rangle}) \end{aligned} \tag{13}$$

for $s = t_1, \cdots, t_2 - 1$ and every user n. Finally, we calculate the pairwise probability of a link from u_n to all other users, rank them and evaluate the AUC score in Figure 11. The X-axis denotes time gap between now and the prediction (in months) ($\Delta t = t_2 - t_1$), and Y-axis denotes the cosponsor prediction AUC for all bills at time t_2, averaged over all pairs of (t_1, t_2) which satisfy $t_2 - t_1 = \Delta t$.

For baseline methods which purely model the propagation of latent features, we have

$$\boldsymbol{x}_n^{\langle s+1\rangle} \sim p_m(\boldsymbol{x}_n^{\langle s\rangle}), \forall n \tag{14}$$

for $s = t_1, \cdots, t_2 - 1$, where $p_m(\boldsymbol{x}_n^{\langle s\rangle})$ is the prior (propagation probability) for the corresponding baseline method m. For baseline methods designed for static networks (i.e. no propagation in terms of latent features), we use the node representation at time t_1 to predict the cosponsors at t_2.

It would also be interesting to study the time delay that a legislator cosponsors a bill. After a bill is initialized, the sponsor may expend considerable efforts recruiting cosponsors with personal contacts so that others will add their names to support the bill later. Only those cosponsors who join within a year are considered. The distribution of the time delay between the initial sponsorship and cosponsor date is shown in Figure 10. When a legislator cosponsors a bill immediately after its initialization, it may indicate that the sponsor and cosponsor are close in some sense. Therefore, we assign a relevant score to cosponsors according to the date of cosponsorship: those who signed their names within the first quartile (most promptly) are assigned with the highest relevance score of 4; those between the first and the second quartile have a relevance score of 3, and so on. Thus, based on the ranking given by the likelihood of a cosponsor, we are able to calculate the normalized discounted cumulative gain (NDCG) of the cosponsorship prediction. The macro-average NDCG$_{10}$ score for each bill is reported in Figure 12. X-axis has the same meaning as the previous task, i.e., the time gap between now and the prediction.

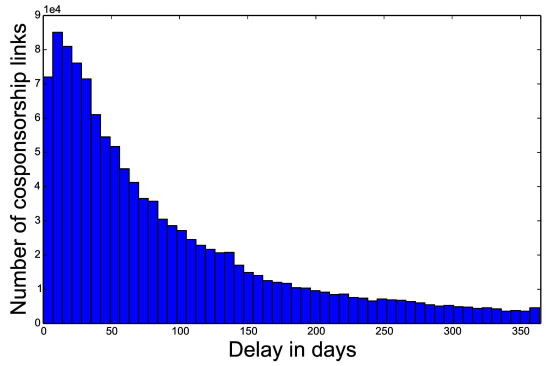


Figure 10: Distribution of time delay. Quartiles: Q_1 = 24, Q_2 = 58, Q_3 = 125.

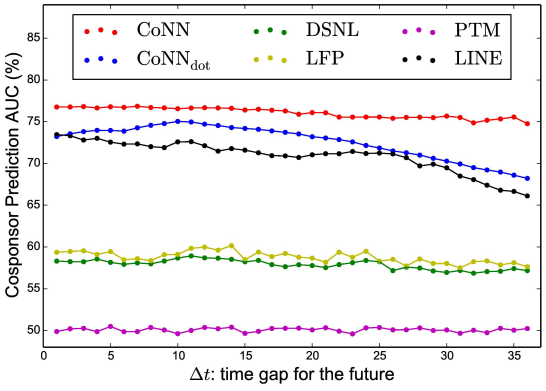


Figure 11: AUC score for cosponsor prediction.

In Table 1 we also show the top 10 people with largest b values in our timeframe (1983-present). They are popular in that many others legislators are likely to cosponsor the bills they drafted. We interpret them as opinion leaders, since cosponsorship implies endorsement and their ideas spread more widely among others. Among the

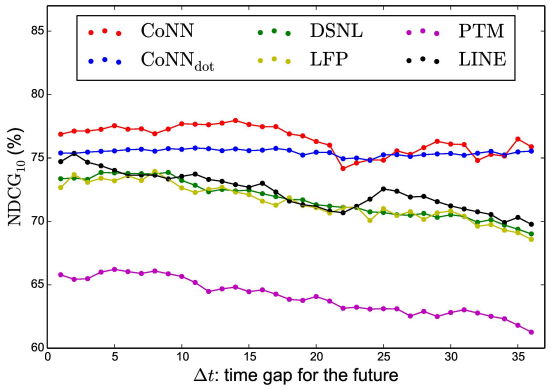


Figure 12: NDCG$_{10}$ for cosponsor prediction.

results, we identify John Kerry (68th U.S. Secretary of State), Albert Gore (45th U.S. Vice President) and Mitch McConnell (the majority leader of the Senate since 2015). Therefore, the opinion leaders and the actual leaders in the legislature have some overlap, and our approach can detect leaders from another perspective.

Rank	Name	Party-State	Time in Congress
1	Paul Simon	Democrat-IL	1975-1997
2	Jay Rockefeller	Republican-WV	1985-2015
3	John Kerry	Democrat-MA	1985-2013
4	Thomas Harkin	Democrat-IA	1975-2015
5	James Terry Sanford	Democrat-NC	1986-1993
6	Albert Gore	Democrat-TN	1983-1993
7	Kent Conrad	Democrat-ND	1987-2013
8	Edward Kennedy	Democrat-MA	1962-2009
9	Mitch McConnell	Republican-KY	1985-present
10	Frank Annunzio	Democrat-IL	1965-1993

Table 1: Popular legislators ranked by b in recent 34 years.

Parameter Study. We plot the performance curve under different choices of hyperparameters (σ, ϵ) in Figure 13. For intuitive comparison, we calculate the average evaluation measure over all possible lengths of time gap (i.e. from $\Delta t = 1$ to 36) as the value on Y-axis. In sum, our inference model is not sensitive to these parameters as long as they lie within a reasonable range.

6 RELATED WORK

Understanding the evolution of link structure and node property has been a promising research topic recently. Traditional interpretations of dynamic networks treat the two problems separately, i.e., the evolution of link structures [4, 9, 24–26, 39, 47] and the evolution of node attributes [13–15, 22].

Under a fixed network structure, various node property propagation models have been proposed, which are better known as the information diffusion model when the node features are binary. The binary feature of each node can be considered as a status,

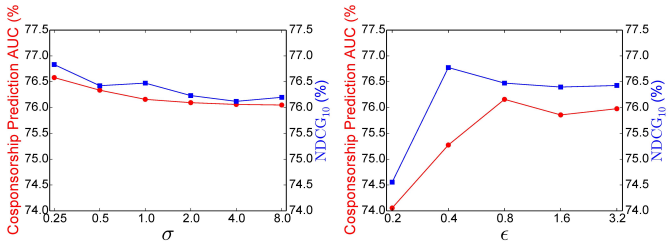


Figure 13: Parameter study on σ and ϵ for CoNN. In the left figure, ϵ is fixed to be 0.8. In the right figure, σ is fixed to be 1.

as whether the node is infested or activated, and it may change according to the network structure. Typically, information diffusion process occurs between nodes that are linked to each other. For example, linear threshold model [14] involves an aggregation of neighbors' weights, and a node is activated if the aggregated weight of its active neighbors exceeds some threshold. Independent cascade model [13] assumes an activating probability for each neighbor of a newly activated node. While binary features reflect the activation status of a node, probabilistic or real-valued features embed every node onto a continuous spectrum, which indicates the relative position between actors. In the DeGroot learning process, every time the opinions of agents are assumed to be updated according to the weighted average of their neighbors [8]. Adjustment in user features after interaction is studied in [7], where similarity of connected users are found to be increasing over time. These methods are limited to the case where network structure does not change over time, and more principled approaches are desired to model user behaviors in dynamic networks.

The evolution of networks is usually modeled as a result of the migration of individuals' features. To model the static snapshots of networks, a variety of methods assume vertices in the network are associated with a latent feature representation, and the observed links are a result of their interaction. Latent class models (blockmodels) assume the probability of a link depends on the communities that the corresponding users engage in [6, 16, 45, 46], and continuous latent feature models embed each node in the network as a position in a lower dimensional Euclidean space, where the features constitute a continuous spectrum that conveys more meaningful messages such as a user's stance (e.g. extreme/moderate) towards a specific topic. These approaches have broad applications in clustering, visualization and so on [17, 29, 33].

Migration of users' latent features is usually modeled as a hidden Markov model (HMM), with network structure being the observed sequence and node features being the latent variables [6, 16, 45, 46]. The distribution of the latent variables depends only on the their previous values, and the value of observed network depends only on the latent variables at the same timestamp. Optimization is usually done using standard forward-backward algorithm [16]. Feature dimension may also be learned automatically from the data, leading to nonparametric methods [10, 20, 23, 35]. The evolution of latent features is modeled as regression of a node's future features to accommodate dynamic networks [27, 36–38]. However, these methods fail to consider the feature migration as part of co-evolution

process. In other words, influence from network structure to node feature migration is totally ignored. In addition, as far as we are concerned, all of the existing approaches simply posit the propagation of node features can happen arbitrarily, without considering the direction or tendency when people change their opinions.

7 CONCLUSION

In this paper we present a novel approach for understanding the co-evolution of network structure and opinion migration. Our approach models both the migration of latent features by virtue of network structures, and the evolution of link structures as a result of the change of node features. We analogize the motion of entities in biological and molecular system to propose the latent feature migration model, and social influence is explicitly exhibited in terms of user's moving directions. Various properties of network can be charactered by adjusting the system-level parameters of our generation model, and applications on a real-world dataset reveal our advantage over the state-of-the-art co-evolution approaches.

ACKNOWLEDGEMENT

The authors would like to thank Tina Eliassi-Rad who kindly reviewed an earlier version of this manuscript and provided valuable feedback and suggestions. We would also like to thank the anonymous reviewers for their precious comments. This work is partially supported by NSF CAREER #1741634.

REFERENCES

[1] *N-sphere: Spherical coordinates.* https://en.wikipedia.org/wiki/N-sphere#Spherical_coordinates
[2] *The Polarization of the Congressional Parties.* http://www.voteview.com/political_polarization_2015.htm
[3] Amr Ahmed, Nino Shervashidze, Shravan Narayanamurthy, Vanja Josifovski, and Alexander J Smola. 2013. Distributed large-scale natural graph factorization. In *WWW'13*. 37–48.
[4] Albert-László Barabási and Réka Albert. 1999. Emergence of scaling in random networks. *Science* 286, 5439 (1999), 509–512.
[5] Joshua Clinton, Simon Jackman, and Douglas Rivers. 2004. The statistical analysis of roll call data. *American Political Science Review* 98, 02 (2004), 355–370.
[6] Marco Corneli, Pierre Latouche, and Fabrice Rossi. 2015. Modelling time evolving interactions in networks through a non stationary extension of stochastic block models. In *ASONAM'15*. 1590–1591.
[7] David Crandall, Dan Cosley, Daniel Huttenlocher, Jon Kleinberg, and Siddharth Suri. 2008. Feedback effects between similarity and social influence in online communities. In *KDD'08*. 160–168.
[8] Morris H DeGroot. 1974. Reaching a consensus. *J. Amer. Statist. Assoc.* 69, 345 (1974), 118–121.
[9] Mehrdad Farajtabar, Yichen Wang, Manuel Gomez-Rodriguez, Shuang Li, Hongyuan Zha, and Le Song. 2015. Coevolve: A joint point process model for information diffusion and network co-evolution. In *NIPS'15*. 1954–1962.
[10] James R Foulds, Christopher DuBois, Arthur U Asuncion, Carter T Butts, and Padhraic Smyth. 2011. A dynamic relational infinite feature model for longitudinal social networks. In *Int. Conf. on Artificial Intelligence and Statistics*. 287–295.
[11] James H Fowler. 2006. Connecting the Congress: A study of cosponsorship networks. *Political Analysis* (2006), 456–487.
[12] Sean Gerrish and David M Blei. 2011. Predicting legislative roll calls from text. In *ICML'11*. 489–496.
[13] Jacob Goldenberg, Barak Libai, and Eitan Muller. 2001. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing letters* 12, 3 (2001), 211–223.
[14] Mark Granovetter. 1978. Threshold models of collective behavior. *Amer. J. Sociology* (1978), 1420–1443.
[15] Daniel Gruhl, Ramanathan Guha, David Liben-Nowell, and Andrew Tomkins. 2004. Information diffusion through blogspace. In *WWW'13*. 491–501.
[16] Creighton Heaukulani and Zoubin Ghahramani. 2013. Dynamic probabilistic models for latent feature propagation in social networks. In *ICML'13*. 275–283.
[17] Peter D Hoff, Adrian E Raftery, and Mark S Handcock. 2002. Latent space approaches to social network analysis. *J. Amer. Statist. Assoc.* 97, 460 (2002), 1090–1098.
[18] Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. 1983. Stochastic blockmodels: First steps. *Social networks* 5, 2 (1983), 109–137.
[19] Petter Holme and Mark EJ Newman. 2006. Nonequilibrium phase transition in the coevolution of networks and opinions. *Physical Review E* 74, 5 (2006), 056108.
[20] Katsuhiko Ishiguro, Tomoharu Iwata, Naonori Ueda, and Joshua B Tenenbaum. 2010. Dynamic infinite relational model for time-varying relational data analysis. In *Advances in Neural Information Processing Systems*. 919–927.
[21] Mohsen Jamali and Martin Ester. 2010. A matrix factorization technique with trust propagation for recommendation in social networks. In *RecSys'10*. 135–142.
[22] David Kempe, Jon Kleinberg, and Éva Tardos. 2003. Maximizing the spread of influence through a social network. In *KDD'03*. 137–146.
[23] Myunghwan Kim and Jure Leskovec. 2013. Nonparametric multi-group membership model for dynamic networks. In *Advances in Neural Information Processing Systems*. 1385–1393.
[24] Paul L Krapivsky, Sidney Redner, and Francois Leyvraz. 2000. Connectivity of growing random networks. *Physical review letters* 85, 21 (2000), 4629.
[25] Jure Leskovec, Lars Backstrom, Ravi Kumar, and Andrew Tomkins. 2008. Microscopic evolution of social networks. In *KDD'08*. 462–470.
[26] Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, and Christos Faloutsos. 2005. Realistic, mathematically tractable graph generation and evolution, using kronecker multiplication. In *PKDD'05*. 133–145.
[27] Shawn Mankad and George Michailidis. 2013. Structural and functional discovery in dynamic networks with non-negative matrix factorization. *Physical Review E* 88, 4 (2013), 042812.
[28] Miller McPherson, Lynn Smith-Lovin, and James M Cook. 2001. Birds of a feather: Homophily in social networks. *Annual review of sociology* (2001), 415–444.
[29] Aditya Krishna Menon and Charles Elkan. 2011. Link prediction via matrix factorization. In *ECML/PKDD'11*. 437–452.
[30] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS'13*. 3111–3119.
[31] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation. In *EMNLP'14*, Vol. 14. 1532–1543.
[32] Keith T Poole and Howard Rosenthal. 1985. A spatial model for legislative roll call analysis. *American Journal of Political Science* (1985), 357–384.
[33] Guo-Jun Qi, Charu Aggarwal, Qi Tian, Heng Ji, and Thomas Huang. 2012. Exploring context and content links in social media: A latent space method. *PAMI'12* 34, 5 (2012), 850–862.
[34] Sebastián M Saiegh and others. 2015. Using joint scaling methods to study ideology and representation: Evidence from Latin America. *Political Analysis* 23, 3 (2015), 363–384.
[35] Purnamrita Sarkar, Deepayan Chakrabarti, and Michael Jordan. 2012. Nonparametric link prediction in dynamic networks. *arXiv preprint arXiv:1206.6394* (2012).
[36] Purnamrita Sarkar and Andrew W Moore. 2005. Dynamic social network analysis using latent space models. *ACM SIGKDD Explorations Newsletter* 7, 2 (2005), 31–40.
[37] Purnamrita Sarkar, Sajid M Siddiqi, and Geoffrey J Gordon. 2007. A latent space approach to dynamic embedding of co-occurrence data. In *AISTATS'07*. 420–427.
[38] Daniel K Sewell and Yuguo Chen. 2015. Latent space models for dynamic networks. *J. Amer. Statist. Assoc.* 110, 512 (2015), 1646–1657.
[39] Yizhou Sun, Jie Tang, Jiawei Han, Cheng Chen, and Manish Gupta. 2014. Co-evolution of multi-typed objects in dynamic star networks. *TKDE'14* 26, 12 (2014), 2942–2955.
[40] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *WWW'15*. 1067–1077.
[41] Jie Tang, Jimeng Sun, Chi Wang, and Zi Yang. 2009. Social influence analysis in large-scale networks. In *KDD'09*. 807–816.
[42] Tamás Vicsek, András Czirók, Eshel Ben-Jacob, Inon Cohen, and Ofer Shochet. 1995. Novel type of phase transition in a system of self-driven particles. *Physical review letters* 75, 6 (1995), 1226.
[43] Tamás Vicsek and Anna Zafeiris. 2012. Collective motion. *Physics Reports* 517, 3 (2012), 71–140.
[44] Yuchung J Wang and George Y Wong. 1987. Stochastic blockmodels for directed graphs. *J. Amer. Statist. Assoc.* 82, 397 (1987), 8–19.
[45] Eric P Xing, Wenjie Fu, Le Song, and others. 2010. A state-space mixed membership blockmodel for dynamic network tomography. *The Annals of Applied Statistics* 4, 2 (2010), 535–566.
[46] Kevin S Xu and Alfred O Hero III. 2013. Dynamic stochastic blockmodels: Statistical models for time-evolving networks. In *Social Computing, Behavioral-Cultural Modeling and Prediction*. 201–210.
[47] Elena Zheleva, Hossam Sharara, and Lise Getoor. 2009. Co-evolution of social and affiliation networks. In *KDD'09*. 1007–1016.