Temporal Network Prediction and Interpretation

Zou, Li; Zhan, Xiu xiu; Sun, Jie; Hanjalic, Alan; Wang, Huijuan

DOI
10.1109/TNSE.2021.3138643

Publication date
2022

Document Version
Accepted author manuscript

Published in
IEEE Transactions on Network Science and Engineering

Citation (APA)
Zou, L., Zhan, X. X., Sun, J., Hanjalic, A., & Wang, H. (2022). Temporal Network Prediction and Interpretation. *IEEE Transactions on Network Science and Engineering*, *9*(3), 1215-1224. https://doi.org/10.1109/TNSE.2021.3138643

Temporal Network Prediction and Interpretation

Li Zou, Xiu-Xiu Zhan, Jie Sun, Alan Hanjalic, *Fellow, IEEE* and Huijuan Wang

Abstract—Temporal networks refer to networks like physical contact networks whose topology changes over time. Predicting future temporal network is crucial e.g., to forecast the epidemics. Existing prediction methods are either relatively accurate but black-box, or white-box but less accurate. The lack of interpretable and accurate prediction methods motivates us to explore what intrinsic properties/mechanisms facilitate the prediction of temporal networks. We use interpretable learning algorithms, Lasso Regression and Random Forest, to predict, based on the current activities (i.e., connected or not) of all links, the activity of each link at the next time step. From the coefficients learned from each algorithm, we construct the prediction backbone network that presents the influence of all links in determining each link's future activity. Analysis of the backbone, its relation to the link activity time series and to the time aggregated network reflects which properties of temporal networks are captured by the learning algorithms. Via six real-world contact networks, we find that the next step activity of a particular link is mainly influenced by (a) its current activity and (b) links strongly correlated in the time series to that particular link and close in distance (in hops) in the aggregated network.

Index Terms—temporal network, link prediction, prediction backbone network

◆

1 INTRODUCTION

Real-world systems can be represented as complex networks, where nodes denote the components and links denote relations or interaction between these components. In many cases, however, the interactions are not continuously active. For example, individuals connect via email, text message, phone call or physical contact at specific time stamps instead of constantly. Temporal networks [1], [2], [3] could represent these systems more realistically with time-varying network topology. A temporal network can be regarded as a static network where each link is further associated with a time series specifying whether an interaction (contact) occurs or not at each time step. Temporal networks display non-trivial properties, which may have profound effect on the dynamic processes deployed on them. For example, the inter-event (contact, activation) time between a node pair has been found to follow a heavy-tail or power-law distribution in many temporal networks [4], [5], [6]. It has been shown that temporal network properties such as community structure, the degree distribution in the aggregated network and inter-event time influence the diffusion processes on the temporal network [7], [8], [9], [10], [11], [12], [13], [14], [15], [16].

Temporal network prediction is a task of predicting temporal interactions/contacts at a future time step based on the temporal network topology observed in the past. Predicting the temporal network such as a physical contact network in the future is essential to forecast performance of a process upon the network like the prevalence of epidemic spreading. The temporal network prediction problem is also equivalent to problems in recommender systems, e.g., predicting which user will purchase which product, which individuals will become acquaintance [17], [18], [19].

Existing prediction methods are either relatively accurate but black-box, or white-box but less accurate, although progress has also been made recently in evaluating the predictability of a temporal network [20]. Markovian Methods and machine learning algorithms have been developed to predict temporal network in short term, i.e., at the next time step based on the network observed so far within a given time window. Markovian models [21] can be developed by considering the time series or activity of each link and predict a link's future activity based on its previous activities. Markovian models have also been built by regarding the temporal network or the link activated at each time step as the state [22], [23]. Deep learning methods have been further developed to improve the temporal link prediction. Examples include temporal network embedding [24], [25], [26], restricted Boltzmann machine (RBM) based methods [27], [28] and Graph neural networks [29], [30], [31], [32]. These methods, however, do not allow for insightful interpretation regarding which inherent property or mechanism of the temporal networks could these methods capture when predicting temporal networks.

In this work, we address the problem of temporal network prediction, and its interpretation with respect to what underlying properties of temporal networks a prediction algorithm possibly captures or utilizes. We confine ourselves to the problem of predicting the activity of each link at a given time step based on the activities of all the links at the previous step. A statistical learning algorithm, i.e., Lasso Regression and a basic machine learning algorithm, i.e., Random Forest have been used for network prediction because of their interpretability. We further construct the prediction backbone network using the coefficients learned from the algorithms. The weighted backbone network suggests the influence of every link in determining a given's activity. Characterizing the backbone network in relation to the time series of all the links and the aggregated network unveils other patterns underlying the temporal networks

---

- *L. Zou, A. Hanjalic and H. Wang are with the Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, 2628 CD, Delft, The Netherlands.*
- *X.-X. Zhan is with the Alibaba Research Center for Complexity Sciences, Hangzhou Normal University, Hangzhou, 311121, China.*
- *J. Sun is with Theory Lab, Central Research Institute, 2012 Labs, Huawei Technology Co., Ltd.*
- *Corresponding author: X.-X. Zhan (zhanxiuxiu@hznu.edu.cn) and H. Wang (H.Wang@tudelft.nl)*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2021.3138643, IEEE Transactions on Network Science and Engineering

that possibly facilitate the prediction. We find that a link's current state is largely determined by its own activity but also influenced by the activities of other links, at the previous time step. Links tend to influence each other more if they have a shorter and/or more shortest paths in the aggregated network and are more strongly correlated in their time series.

These findings, when combined with modern deep learning techniques can potentially lead to interpretable yet accurate prediction models. They may also inspire the development of temporal network models and strategies to mitigate epidemic spreading on physical contact networks.

2 TEMPORAL NETWORK REPRESENTATION

A temporal network can be represented as a sequence of network snapshots $G = \{G_1, G_2, ..., G_T\}$, where T is duration of the observation window, $G_t = (V; E_t)$ is the snapshot at time step t with V and E_t being the set of nodes and contacts, respectively. If node j and k have a contact at time step t, $(j, k) \in E_t$. Here, we assume all snapshots share the same set of nodes, i.e., V. The links in the aggregated network G_w are defined as $E = \cup_{t=1}^{T} E_t$. That is, a pair of nodes is connected with a link in the aggregated network if at least one contact occurs between them in the temporal network. Hence, the link set E in the aggregated network contains all the node pairs that have contact(s) in the temporal network and the total number of links is $M = |E|$. We give each link in the aggregated network an index i, where $i \in [1, M]$. The temporal connection or activity of link i over time could then be represented by a T-dimension vector $\mathbf{x_i}$ whose element is $x_i(t)$, where $t \in [1, T]$, $x_i(t) = 1$ when node pair i has a contact at time t and $x_i(t) = 0$ if no contact occurs at t. The activity of all links can be captured by a $M \times T$ dimensional matrix $\mathbf{X}$ with its element $X(i, t) = x_i(t)$ where $t \in$ [1,T] and $i \in$[1,M].

3 EMPIRICAL DATA SETS

Most real-world temporal network data sets available are contact networks. Without losing the generality, we choose six empirical networks that range from physical and virtual human contact networks to animal contact networks: Hypertext 2009 [33], [34], Highschool [35], Call [36], Sms [36], Baboons [37], and Ant [38]. Basic description is given in section 1 in Supplementary and properties of these data sets are given in Table 1. Note that the time steps at which there is no contact in the whole network have been deleted. Basic description of how each temporal network is measured and constructed explains to some extent the difference of these networks in, for example, the average number of contacts per link.

We report the distribution of inter-event(contact) time in Figure 1, i.e., the interval between two consecutive contacts between a node pair. As it is often the case for human dynamics, the distributions of inter-contact time are heterogeneous. All our six systems show a heavy-tail distribution. It means the networks we consider exhibit burstiness which corresponds to frequent activities over a short period of time followed by a long period of inactivity [39], [40], [41].

TABLE 1: The number of nodes ($N = |V|$), the number of node pairs that have contact(s) (M), the length of the observation time window (T), time resolution (δ sec), the average number of contacts within the observation time window per link (η) and the type of contacts in each empirical network.

Network	N	M	T	δ	η	Type
Hypertext 2009	113	2196	5246	20	9.5	Human contact (conference)
Highschool	312	2242	899	20	12.8	Human contact (high school)
Call	75	270	8597	1	34	Human contact (phone call)
Sms	110	210	60932	1	291	Human contact (message)
Baboons	26	303	10072	5	1401	Animal contact (Baboons)
Ant	89	649	993	0.5	2.8	Animal contact (ants)

4 TEMPORAL NETWORK PREDICTION METHODS

In this section, we propose our methodology which allows not only temporal network prediction but also the deduction of the relationship between links in the aggregated network G_w in influencing each other's activity, i.e., the dynamic of link activities. Specifically, we aim to understand to what extend a link's activity (active/having contact or not) at a given time step is determined by the other links' and its own activity at the previous time step.

Firstly, we introduce a statistical learning algorithm, i.e., Lasso Regression and a basic machine learning algorithm, i.e., Random Forest, to predict temporal networks. In view of the heavy-tail distribution of inter-event time, thus the possibility that the activity of a link remains the same within a short period, we introduce two baseline models that assume the activity of a link is determined only by its own activity at the previous time step. These four models predict the activity of a link at a given time based on its and/or other links' activities at the previous time step.

Afterwards, we illustrate how to deduce the influence between links in activities via applying these proposed models to real-world temporal network data. This requires the calibration of the coefficients of the models and entails the the setup of training and test data sets.

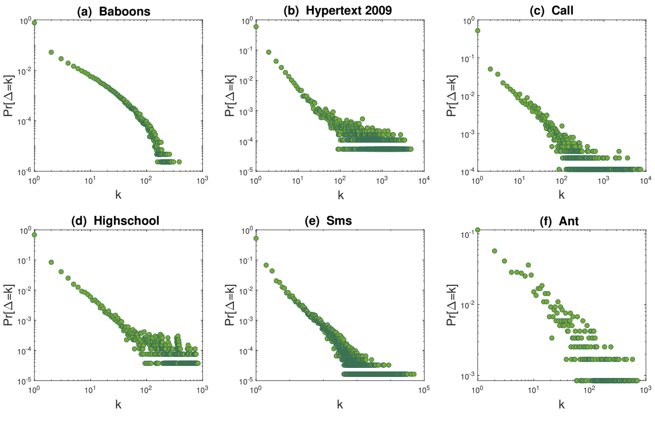

Fig. 1: The probability distribution $Pr[\Delta = k]$ of the inter-event time Δ in number of time steps in log-log scale for (a) Baboons, (b) Hypertext 2009, (c) Call, (d) Highschool, (e) Sms and (f) Ant.

4.1 Lasso Regression and Random Forest Model

Our method applies to a generic temporal network with N nodes and M links (node pairs that have at least one contact) whose activities are recorded within a time window $[1, T]$. The activities of the M links are recorded by a $M * T$ matrix X. The state or activity of link i at time $t+1$ is $x_i(t+1)$ ($t \in [p, p+L-1]$), which equals 1 when link i is active, and equals 0 otherwise. We assume that the activity of link i at time $t+1$ is a function of the activities of all the links at time t, i.e.,

$$x_i(t+1) = f_i(x_1(t), x_2(t), ..., x_M(t)). \tag{1}$$

The mapping function f_i is unknown and link-specific. It can be learned from the activities of all links, i.e., $[x_i(p), x_i(p+1), \cdot\cdot\cdot, x_i(p+L)]$ where $i \in [1, M]$ within a time window $[p, p+L]$, and denoted as $f_i^{p,L}$. We construct in total L training data samples for each link i based on the temporal network observed within $[p, p+L]$: we use link $i's$ state at each time step $t+1 \in [p+1, p+L]$ as target and the corresponding features are the states of all links at time step t. The training data samples for node pair i is expressed as a set $\mathcal{D}_i(p, L)$:

$$\mathcal{D}_i(p,L) = \{x_i(t+1); x_1(t), x_2(t), \cdots, x_M(t)\}_{t=p}^{p+L-1}. \tag{2}$$

A learning algorithm assumes a given function $f_i^{p,L}$, whose coefficients can be learned from a training set $\mathcal{D}_i(p, L)$. The learned function $f_i^{p,L}$ tells us to what extent $x_i(t+1)$ can be estimated by the activity of each link at t respectively.

We explore a statistical learning (Lasso Regression) and a machine learning algorithm (Random Forest) to learn $f_i^{p,L}$.

Lasso Regression assumes f_i to be a linear function [42], [43]

$$x_i(t+1) = \sum_{j=1}^{M} x_j(t)\beta_{ij} + c_i. \tag{3}$$

The objective is

$$\min_{\beta_i}\{\sum_{t=p}^{p+L-1} (x_i(t+1) - \sum_{j=1}^{M} x_j(t)\beta_{ij} - c_i)^2 + \alpha \sum_{j=1}^{M} |\beta_{ij}|\}. \tag{4}$$

where L is the number of training samples, M is the number of features as well as the number of links, c_i is the constant coefficient and $\beta_i = \{\beta_{i1}, \beta_{i2}, \cdots, \beta_{iM}\}$ are the regression coefficients of all the features for link i. A large coefficient β_{ij} indicates that feature $x_j(t)$ influences or determines significantly the target $x_i(t+1)$.

We use $L1$ regularization, which adds a penalty to the sum of the magnitude of coefficients $\sum_{j=1}^{M} |\beta_{ij}|$. The parameter α controls the penalty strength. The regularization forces some of the coefficients to be zero and thus lead to models with few non-zero coefficients (relevant features). If α is zero, Lasso Regression reduces to the classical linear regression algorithm. Given a training data set $\mathcal{D}_i(p, L) = \{x_i(t+1); x_1(t), x_2(t), \cdots, x_M(t)\}_{t=p}^{p+L-1}$, the coefficients $\beta_i(p, L)$ of the Lasso Regression model for each node i can be learned. The optimal α that achieves the best prediction is chosen from 50 logarithmically spaced points within $[10^{-4}, 10]$.

Random Forest is a non-linear ensemble learning algorithm for tasks such as classification [44], [45]. A large number of decision trees can be constructed from a training set. A decision tree is a flowchart-like structure in which each internal node represents a “test” on a feature, each branch represents the outcome of the test, and each leaf node represents a class label. The paths from root to leaf represent classification rules. Each tree is grown based on each training set $\mathcal{D}_i(p, L) = \{x_i(t+1); x_1(t), x_2(t), \cdots, x_M(t)\}_{t=p}^{p+L-1}$ as follows: 1) choose randomly a set of m ($m << M$) features out of the M features as the nodes in the tree 2) collect from each training sample the m features and the corresponding target 3) construct the decision tree based on the data collected from 2). The optimal m that leads to the highest prediction precision is chosen.

Random Forest could rank the importance of the features in estimating the target in a nonlinear way. Considering training set $\mathcal{D}_i(p, L) = \{x_i(t+1); x_1(t), x_2(t), \cdots, x_M(t)\}_{t=p}^{p+L-1}$, the value of the jth feature in the first sample is $x_j(p)$, the value in the second sample is $x_j(p+1)$. The values for the jth feature are ordered as $\{x_j(p), x_j(p+1), \cdots, x_j(p+L-1)\}$ from the first sample to the $(p+L)th$ sample. To measure the importance of the jth feature, its values $\{x_j(p), x_j(p+1), \cdots, x_j(p+L-1)\}$ are randomized/permuted. Random Forest model is then trained by the original training set and the permuted training set respectively. The importance of a feature is reflected by the difference between the prediction errors of the model learned from the original and permuted training set respectively. The coefficient β_{ij} is obtained as the normalized difference in prediction error. A larger difference in prediction error means a larger contribution of the feature to the target prediction. We use TreeBagger implementation in Matlab with 1000 trees and use default values for other parameters.

4.2 Training and test data

The temporal network observed in each sub-window $[p, p+L]$ where $p \in [1, T-L-1]$ is considered as a training set and the learned model function will be tested in predicting the temporal network observed at $p+L+1$, using the temporal network observed at $p+L$. For each learning algorithm, the coefficients $\{\beta_i(p, L)\}$, $i = 1, 2, \cdots, M$, learned from each training set $D_i(p, L)$ will be used to predict the activity of the links in the test set $Q_i(p, L) = \{x_i(p+L+1); x_1(p+L), x_2(p+L), \cdots, x_M(p+L)\}$. In total, $T-L-1$ training sets, together with their corresponding test sets, will be considered for each temporal network.

4.3 Baseline models

We introduce two baseline models that predict a link’s future activity based on its current activity. The probability that a link has the same state at two consecutive time steps is high, above 0.93 in each network. Hence, the baseline model 1 predicts the activity of a link at the next time step equal to the link’s own activity at the current step, i.e., $x_i(t+1) = x_i(t)$. If link i is active (inactive) at time step $t-1$, then its state at t is predicted to active (inactive) in baseline model 1.

Baseline model 2 is the corresponding Lasso Regression, $x_i(t+1) = \beta_{i,i} x_i(t) + c_i$ where a link's current activity is a linear function its own previous activity. The same training and test sets have been used as introduced in section 4.1.

5 MODEL EVALUATION

For a given length L of the training sets, we evaluate each model via its average quality in predicting links' activities in a test set $Q_i(p, L)$ using the coefficients $\{\beta_i(p, L)\}$ learned from the corresponding training set $D_i(p, L)$, where $i = 1, 2, ...M$. The average is over all test sets, i.e., $p \in [1, T-L-1]$.

The prediction quality in a test set is measured via the area under the ROC curve (AUC) [46], [47]. AUC provides an aggregate measure of performance across all possible classification thresholds. It ranges in value from 0 to 1. A high AUC implies high prediction quality.

Different lengths $L \in [1, T-50]$ of the training set are considered when evaluating the performance of each model. The maximum $L_{max} = T - 50$ ensures a minimum of 50 training/test sets for each temporal network.

Figure 2 shows that the training set length L indeed affects the prediction the quality AUC in all the networks. A relatively good performance tends to be obtained by a medium training length L, e.g., $L \sim 100$. A small length, e.g., $L \sim 10$ is insufficient for a model to learn the coefficients that to have reasonable prediction quality. A model with a large length may not capture the change of network dynamics over time, if there is. When the length is extremely large, e.g., $L \to T-50$, the number of training set $T-(L+1)$ is small and the corresponding test sets lie mainly at the end of the observation window $[0, T]$. Such boundary effect leads to low robustness of the model against abrupt change in data at the end of the observation window. For example, Supplementary Figure 2 shows an abrupt change in the total number of contacts per time step at the end of the time window in network Call. Correspondingly, the prediction quality AUC changes sharply when L is around $T - 50$.

Almost all the AUC values are larger than 0.5, which corresponds to the performance of random guessing. This suggests that all the models including the simple baseline models perform better than random guessing.

Lasso Regression performs the best in Hypertext 2009, Call, Sms and Ant networks. And for Baboons and Highschool networks, Lasso regression and Random Forest perform comparably, better than the baseline models. Random Forest does not perform evidently better than the baseline model2 in Call, Sms and Ant, which have a lower number of contacts per step on average than the other networks (see Supplementary Figure 1). In general, the linear relationship of Lasso Regression models the link activity dynamic in temporal networks the best. In contrast, the baseline models that predict a link activity based on the link's own activity in the previous time step, gain a smaller AUC. Hence, the activities of other links contribute to the prediction of a given link's activity. Both Lasso Regression and Random Forest could achieve a reasonably good prediction quality via the choice of the training length L.

The area under the precision recall curve $AUPR$ [48] is also considered to measure the prediction quality. It is considered as a more suitable measure for imbalanced classification problems. A larger $AUPR$ suggests a better prediction quality. Similar results are obtained when $AUPR$ is used to measure the link prediction quality for model evaluation (see Supplementary Figure 1). The prediction quality is the lowest in network Ant, which is possibly due to its lowest average number of contacts per link observed within the observation time window.

6 MODEL INTERPRETATION

The relatively good performance of the two models motivates us to further explore which links' activities influence a given link's activity more via the coefficients learned from the two models. We firstly introduce how to construct the prediction backbone network using the coefficients learned from a model. The backbone is a directed weighted network where nodes are the links in the aggregated network and weight B_{ij}, $i, j \in [1, M]$ represents the influence of link j in the aggregated network on link i in predicting link i's activity. Furthermore, we unravel which links' activities influence a given link's activity more via analyzing properties of the backbone as well as its relation to the aggregated network and the time series of the links.

6.1 Construction of the prediction backbone network using influence coefficients

The coefficients of each algorithm can be derived as follows. From a training set $D_i(p, L)$, where $i = 1, 2, ..., M$, we can obtain the coefficients or coefficient matrix $\{\beta_{ij}(p, L)\}_{i,j=1}^{M}$ for each learning algorithm, either Lasso Regression or Random Forest. Each element $\beta_{ij}(p, L)$ indicates the contribution or influence of the activity of link j at a time $t-1$ in determining the activity of link i at t, where $t \in [p+1, L+p]$.

For each network, we consider from now on the training set length L at which the Lasso Regression obtains the maximal AUC value. Furthermore, we randomly choose 50 out of $T - (L+1)$ training sets. We consider the coefficient matrices obtained from these 50 training sets via Lasso Regression and Random Forest, respectively, as samples to understand the influence between links.

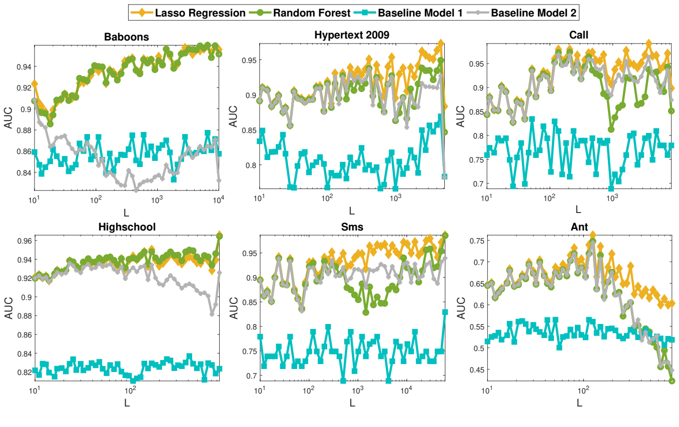

Fig. 2: The prediction quality AUC for Lasso Regression, Random Forest and Baseline model 1 and 2 respectively in six temporal networks at different training set lengths L.

We find a positive correlation between the coefficients $\beta_{ij}(p, L)$ obtained from the two algorithms respectively. Their Pearson correlation coefficients is higher than 0.5 in all networks and is higher than 0.8 in network Baboons, Hypertext2009, Call and Sms. It indicates that the coefficients, i.e., the influence between links, obtained by these two learning algorithms are consistent with each other. Hence, we will focus on the coefficients and the corresponding prediction backbone of Lasso Regression since now on, which performs the best in link prediction.

The prediction backbone network can be constructed as follows. The nodes of the backbone correspond to the M links in the aggregated network. The backbone is a directed and weighted complete network with self-loops. The weight $B_{ij} = E[\beta_{ij}(p, L)]$ where $i, j \in [1, M]$ is the average of the coefficient over the 50 samples, representing the influence of link j in the aggregated network on link i in determining link i's activity. The coefficient $\beta_{ij}(p, L)$ where $i, j \in [1, M]$ derived from a sample, possibly positive or negative, represents to what extent the contact of link j leads to the contact of link i at the next step. Among the 50 samples, the coefficients of any two samples are positively correlated on average. The average Pearson correlation of the coefficients from two random samples is 0.77, 0.72, 0.45, 0.74, 0.88 and 0.07 for Baboons, Hypertext 2009, Call, Highschool, Sms, and Ant respectively. Hence, the weight $B_{ij} = E[\beta_{ij}(p, L)]$ in the backbone suggests the average influence of link j on link i in activity.

We evaluate to what extent a link's activity is influenced by the activity of its own and of the other links. The probability density function[1] $f_{B_{ij}}(x)$ where $i = j$ of the influence of a link on its own activity and $f_{B_{ij}}(x)$ where $i \neq j$ of the influence of a different link are given in Figure 3. The influence of the link itself $B_{i=j}$ tends to be larger than the influence of another link $B_{i \neq j}$ on link i's activity in most networks except for Hypertext 2009 and Ant, where the self-influence $B_{i=j}$ can be negative. This suggests that Lasso's out-performance than the baseline model is because Lasso considers other links' influence on a given link's activity and Lasso considers a link's its own influence differently from the baseline model.

To have a better understanding of our backbone networks which are weighted directed complete graphs with self-loops, we visualize the sub-network of a backbone network. The sub-network includes only the none self-loop links in the backbone network that have the highest weights/influence and the corresponding end nodes of these links, such that the average degree of the sub-network is 2. We take the Sms backbone as an example, since it has the smallest number of nodes among all data sets, and visualize the sub-network of its backbone in Figure 4. Since the backbone is a directed network, a node pair in the sub-network may have none, one or two unidirectional links. Figure 4 shows that few node pair is connected by two unidirectional links or a bidirectional link, which is represented by two green links, whereas most node pairs are connected by an unidirectional link, which is colored in red. This suggests that a high weight B_{ij} of link from i to j in the backbone does not imply a high weight of B_{ji}. This observation is in line with the weak correction $\rho(B_{ij}, B_{ji})$ between the weight of the two reciprocal links of a node pair in the original un-sampled backbone network, as shown in in Table 2. Furthermore, node size and node color in Figure 4 are proportional to the node's in-strength and out-strength in the sub-graph respectively. A dark blue (white) color of a node represents a large (small) out-strength. We find that a node with a large in-strength in the sub-network does not necessarily have a large out-strength. This finding via visualization is consistent with the weak Pearson correlation $\rho(S_{in}, S_{out})$ between the in-strength and out-strength of a node in the original un-sampled backbone network, as given in Table 2.

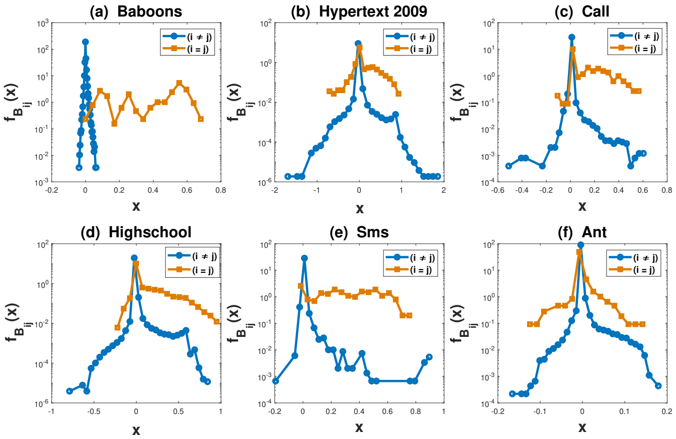

Fig. 3: The probability density function $f_{B_{ij}}(x)$ of the weight B_{ij} in the backbone network, when $i = j$ and $i \neq j$ respectively.

6.2 Backbone network in relation to time series

To understand which kind of links influence a given link i's activity more, we explore the relation between the weight B_{ij} in the backbone and the correlation of the time series corresponding to link i and j.

We first explore whether the relatively high coefficients B_{ii} can be explained by the auto-correlation of a link i's time series $\{x_i(t)\}_{t=1,2,\cdots,T}$. Auto-correlation describes the degree of similarity between a given time series and its lagged version. It measures the correlation between current value of a time series and its past value. Our models use links' activities at the previous time step to predict a link's activity at current time step. Hence, we compute, for

TABLE 2: The Pearson correlation coefficient $\rho(S_{in}, S_{out})$ between in-strength and out-strength of node, and $\rho(B_{ij}, B_{ji})$ between the weight $B(i, j)$ and $B(j, i)$ of the two reciprocal links of a node pair in the backbone network.

Network	$\rho(S_{in}, S_{out})$	$\rho(B_{ij}, B_{ji})$
Baboons	0.30	0.27
Hypertext 2009	-0.02	-0.01
Call	0.01	0.01
Highschool	-0.22	0.00
Sms	-0.40	0.08
Ant	0.19	-0.01

---

[1] The probability density function $f_{B_{ij}}(x)$ of a continuous variable B_{ij} is defined as $f_{B_{ij}}(x) = lim_{\Delta x \to 0} \frac{Pr[x < B_{ij} \leq x + \Delta x]}{\Delta x}$, the probability that the variable is within each range or bin $(x, x + \Delta x]$ normalized by the size of the bin Δx.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2021.3138643, IEEE Transactions on Network Science and Engineering

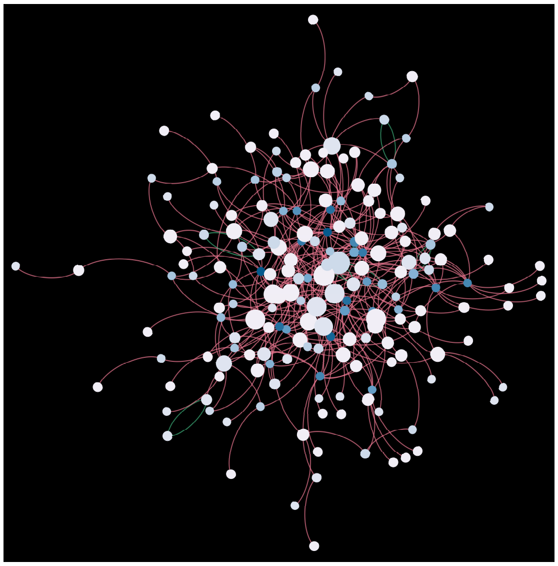

Fig. 4: Visualization of a sub-network of the Sms backbone network. The sub-network is composed of only the none self-loop links in the backbone network that have the highest weights and their corresponding end nodes, such that average degree of the sub-networks is 2. When a node pair is connected by one (two) unidirectional links, the connection is represented by a link in red (two links in green). Node size and node color are proportional to the node's in-strength and out-strength in the sub-graph respectively. A dark blue (white) color of a node represents a large (small) out-strength.

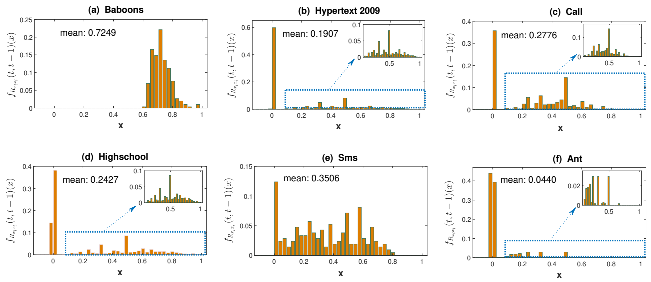

Fig. 5: The probability density function $f_{R_{x_i x_j}(t,t-1)}(x)$ of the auto-correlation coefficient $R_{x_i x_i(t,t-1)}$ in each of the six networks.

each link i, the Pearson correlation coefficient $R_{x_i x_i}(t,t-1)$ between $\{x_i(t)\}_{t=1,2,\ldots,T-1}$ and $\{x_i(t)\}_{t=2,3,\ldots,T}$ as its auto-correlation coefficient. The distribution of the auto-correlation coefficient of a link in each empirical temporal network is shown in Figure 5. In networks, such as Baboons (Hypertext 2009 and Ant) where the average auto-correlation coefficient is high (low), the self-influence $B_{i=j}$ tends (not) to be evidently larger than the influence of another link $B_{i\neq j}$ on the given link i's activity. Moreover, we find that the ranking of these networks in the average number of contacts within the observation time window per link (see Table 1) is the same as their ranking in the average auto-correlation (see Figure 5). Hence, a network with a large average number of contacts per link tends to have a high auto-correlation. Correspondingly, its self-influence $B_{i=j}$ in the backbone tends to be evidently larger than the influence of another link $B_{i\neq j}$ (see Figure 3).

Similarly, we study further the relation between the coefficient B_{ij} and the Pearson correlation coefficient $R_{x_i x_j}(t,t-1)$ between $\{x_i(t)\}_{t=2,3,\ldots,T}$ and $\{x_j(t)\}_{t=1,2,\ldots,T-1}$, where $i \in [1,M]$ and $i \neq j$. This aims to understand whether the influence B_{ij} of another link j on i can be explained by the cross correlation $R_{x_i x_j}(t,t-1)$ between the two links' activity series. The cross correlation is non-reciprocal $R_{x_i x_j}(t,t-1) \neq R_{x_j x_i}(t,t-1)$. The Pearson correlation coefficients between $R_{x_i x_j}$ and $R_{x_j x_i}$ are 0.99, 0.29, 0.25, 0.42, 0.68 and 0.05 for Baboons, Hypertext 2009, Call, Highschool, Sms and Ant respectively. Specifically, we compute the Pearson correlation coefficient at each node i between the influence B_{ij} and cross correlation $R_{x_i x_j}(t,t-1)$ where $i \in [1,M]$ and $i \neq j$. The probability density function of this Pearson correlation coefficient at a random node in Figure 6 shows that the influence B_{ij} tends to be positively correlated with thus can be partly explained by the cross correlation coefficient $R_{x_i x_j}(t,t-1)$. The correlation $R_{x_i x_j}(t,t-1)$ between the activities of link i and j allows our models to predict the activity of link i at t using the activity of link j at $t-1$. The Pearson coefficient BC between B_{ij} and cross-correlation $R_{x_i x_j}$ is the strongest in Sms. The skewed probability density function $f_{R_{x_i x_j}(x)}$ of the cross-correlation in Sms (see Supplementary Figure 3) explains the skewed distribution $f_{B_{ij}}(x)$ where $i \neq j$ of the influence of another link in the backbone (see Figure 3).

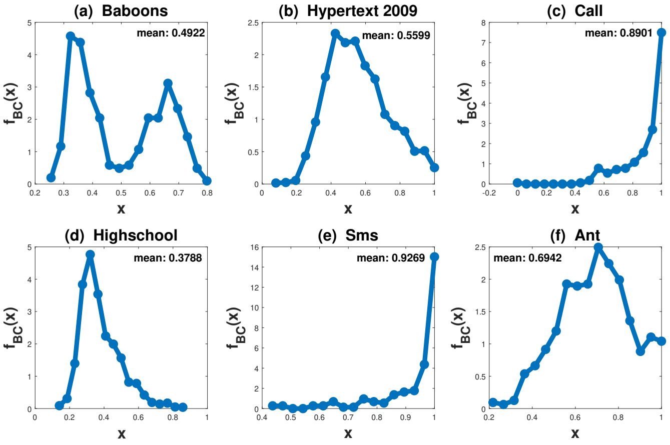

Fig. 6: The probability density distribution of Pearson coefficient BC between B_{ij} and cross-correlation coefficient $R_{x_i x_j}$.

6.3 The backbone network in relation to the aggregated network

The activities of other links have been shown to contribute to a better prediction of a link's activity. We would like to explore which kind of other links have more influence B_{ij} on a link. Would links that are close in the aggregated network tend to have a high influence on each other in activity prediction? Firstly, we define the topological distance or hopcount between two links in the aggregated network. The topological distance, also called hopcount, between two nodes on a static network is the number of links along the shortest path between the two nodes. The distance H_{ij} between two links i and j is defined on the aggregated network. The distance H_{ii} between the same link i is 0. The distance H_{ij} between two different links i and j is defined as the minimal hopcount between one end node of link i and

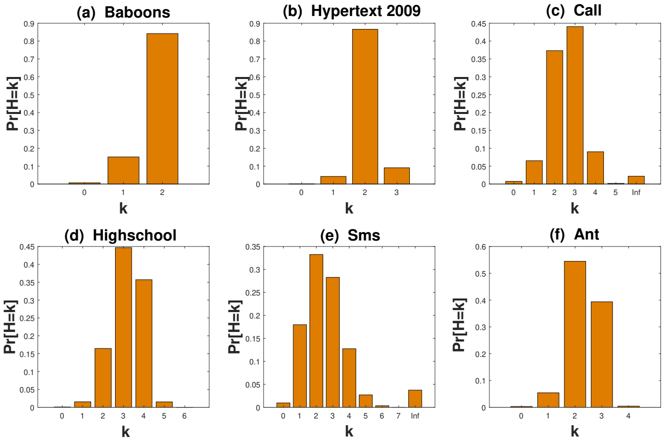

Fig. 7: The probability distribution $Pr[H = k]$ of the topological distance H of two random links in the aggregated network. All six real-world networks are considered.

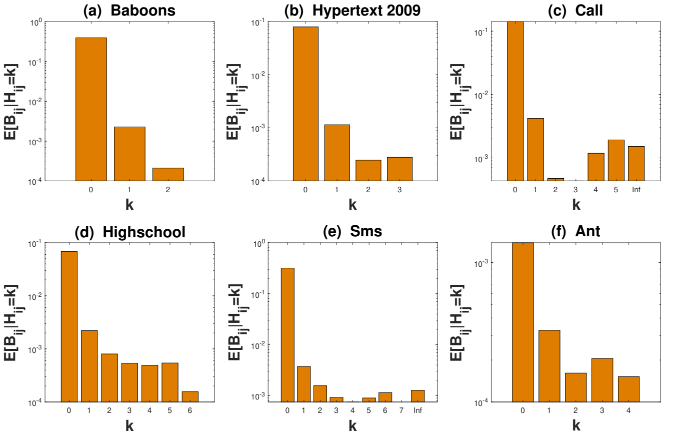

Fig. 8: The average weight in Lasso Regression $E[B_{ij}|H_{ij} = k]$ given the distance of the two links in each of the six networks.

one end node of link j plus one. Hence, two links that share one end node in common have a hopcount 1. The line graph, e.g, G_w^* of an aggregated network G_w can be constructed by considering each link in G_w as a node, and two nodes are connected in G_w^* if the two corresponding links in G_w share a common end node. The distance between two links in G_w equals the hopcount between their corresponding nodes in the line graph G_w^*.

Figure 7 shows the distribution $Pr[H = k]$ of link distance for all the six networks. The distance is in general small, except that few nodes are isolated from the largest connected component, leading to an infinite distance to the other nodes. Networks that are measured within a small (large) spatial space like Ant, Baboons, Hypertext (Call, Sms, Highschool) tend to have a small (large) average hopcount and a large (small) link density in the aggregated network, which is the number of links M normalized by $N(N-1)/2$. Of course, the average hopcount of a network could be also influenced by the number N of nodes.

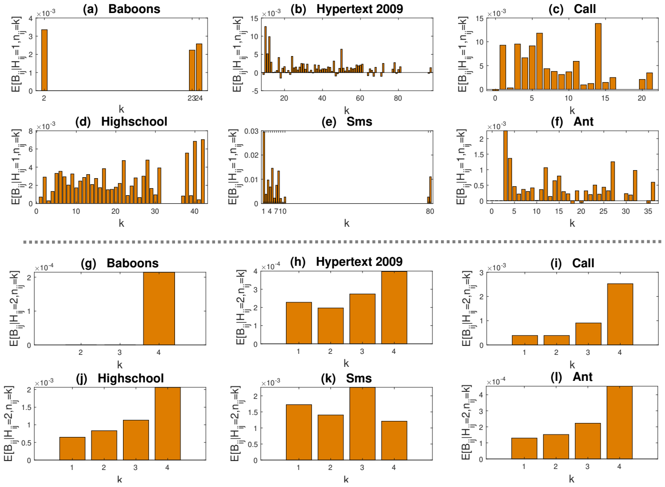

Fig. 9: The average weight $E[B_{ij}|H_{ij} = 1, n_{ij} = k]$ and $E[B_{ij}|H_{ij} = 2, n_{ij} = k]$ in Lasso Regression coefficient given the distance H_{ij} and number of common neighbors n_{ij} between link i and j.

Ref. [49] has shown that nodes that are closer to a target node have more influence on the target node's state. So here we further explore whether links that are closer in topological distance tend to have more influence on each other's activity. Fig. 8 illustrates influence between two links $E[B_{ij}|H_{ij} = k]$ given their topological distance. Still, we focus on Lasso Regression. The influence $E[B_{ij}|H_{ij} = k]$ in general decreases with k. This is also supported by the negative Pearson correlation between the B_{ij} and the distance H_{ij} of two links for Baboons (-0.32), Hypertext 2009 (-0.13), Call (-0.014), Highschool (-0.068), Sms (-0.035) and Ant (-0.027). In network call and Ant, the probability that two links have a distance 4 or larger is small. The average influence $E[B_{ij}|H_{ij} = k]$ when $k \leq 4$ is thus derived from few link pairs and can be noisy. Since B_{ij} can be negative, $E[B_{ij}|H_{ij} = k]$ represents to what extent on average the contact of link j could lead to the contact of link i given that i and j have a distance k. The average strength of influence $E[|B_{ij}||H_{ij} = k]$ also tends to decrease with k, as shown in Supplementary Figure 4.

It is interesting to notice that the average weight $E[B_{ij}|H_{ij} = inf]$ is none zero when the distance is infinity in network Call and Sms. Normally, it is assumed that links can influence each other at least when they are connected. The none zero coefficient $E[B_{ij}|H_{ij} = inf]$ may suggest that the two links are possibly connected in the aggregated network, but not observable in the current aggregated network or the temporal network measured within the short time window $[1, T]$.

Figure 8 shows that $E[B_{ij}|H_{ij} = k]$ is relatively large when $k = 1, 2$. We consider further the link pairs that have a distance $H_{ij} = 1$ and $H_{ij} = 2$ respectively. Among those link pairs, we explore whether link pairs that are well connected tend to have a high influence. How well link i and j are connected can be measured by the number of common neighbors n_{ij} of their corresponding nodes in the line graph G_w^*. Figure 9 demonstrates the influence between two links $E[B_{ij}|H_{ij} = 1, n_{ij} = k]$ and $E[B_{ij}|H_{ij} = 2, n_{ij} = k]$ respectively as a function k of the number of common neighbors. We find no clear relation between $E[B_{ij}|H_{ij} = 1, n_{ij} = k]$ and k. The distance $H_{ij} = 1$ means that node i an j are connected by a link in the line graph G_w^*. In this case, the number of common neighbors, which equals the number of two-hop paths, paths that are longer than the shortest path is not correlated with the influence. However, $E[B_{ij}|H_{ij} = 2, n_{ij} = k]$ does increase with k. Node pairs that have more shortest paths, i.e., two hop paths tend to have a higher influence. Similarly the average influence strength $E[|B_{ij}||H_{ij} = 2, n_{ij} = k]$ also tends to increase with k, as shown in Supplementary Figure 5. Links that have a short distance and are connected by many shortest paths tend to influence each other strongly.

7 LONG-TERM EFFECT

The Lasso regression assumes that a link's current state is a linear function of the activities of all the links at the previous step. In this section, we generalize this assumption as:

$$\begin{aligned} x_i(t) \;\; = \;\; & \sum_{j=1}^{M} \{ x_j(t-1)\beta_{ij}^{1} + x_j(t-2)\beta_{ij}^{2} + ... \\ & + x_j(t-p)\beta_{ij}^{p} + c_i \}. \end{aligned} \tag{5}$$

where a link's current state is a linear function of the activities of all the links in the previous p steps. The Lasso regression (see Eq. 3) that we have investigated in the previous sections corresponds to the case when $p = 1$. The prediction quality AUC is shown in table 2 for different choices of p. In general, considering the activity of a longer influence period, i.e., $p > 1$ can hardly improve the link prediction quality. The same is observed when the prediction quality is evaluated via $AUPR$ (see Supplementary Table 1). The current activity of a link is mainly influenced by the activities of links at the previous time step.

The small memory length p that we have considered is insufficient to capture periodic or pseudo-periodic behavior, in view the time resolution or duration of a time step, which is in the order of seconds. This choice is limited by two factors: the observation window of most real-world temporal networks is short and the computational complexity of Lasso Regression [50] is high: $O((Mp)^3 + (Mp)^2L)$, where M and L are number of nodes in the backbone and the length of training set, respectively. We deem it as an important future work to explore how underlying periodic behavior in the temporal network is captured by a learning model and influences the link prediction. Our finding with regard to which kind of links in the backbone tend to have a high weight may shed light on the selection of model features to reduce the computational complexity.

TABLE 3: The prediction quality AUC for Lasso Regression for different influence period p.

Network	$p = 1$	$p = 2$	$p = 3$	$p = 4$	$p = 5$
Hypertext 2009	0.97	0.97	0.97	0.93	0.93
Highschool	0.93	0.94	0.93	0.93	0.93
Call	0.97	0.97	0.96	0.96	0.96
Sms	0.96	0.96	0.96	0.96	0.96
Baboons	0.96	0.96	0.96	0.96	0.96
Ant	0.76	0.77	0.76	0.76	0.76

8 CONCLUSION

In this work, we illustrate our method that enables interpretable temporal network prediction. Interpretable learning algorithm Lasso Regression and Random Forest are employed to predict the activity (connected or not) of each link at the next time step based on the current activities of all links. The coefficients learned from each algorithm are further used to construct the prediction backbone network, presenting the influence or contribution of all links in determining each link's activity. Via exploring the properties of the backbone network and its relation to the activity time series of links and its relation to the aggregated network, we find the following in six real-world physical and virtual contact networks. A link's next step activity is mainly influenced by the current activity of the link itself and of other links that are better connected with the link. Two links are better connected if they have shorter and/or more shortest paths in the aggregated network. The influence between two links tend to be large if their corresponding activity time series are strongly correlated.Hence, the learning algorithm also captures the underlying network properties and correlation in activity time series, which are usually utilized by network property based prediction methods. Finally, both algorithms' performance can be hardly improved by considering the activities of more than one time steps in the past. The physical contact networks considered differ in the average number of contacts per link observed due the nature of these networks and the methods they are measured or defined. A low average number of contacts per link, e.g., in Ant, may suggest a low prediction quality, a low auto-correlation of the time series in the network and a less evident self-influence in the backbone.

These findings, when combined with modern deep learning techniques, can potentially lead to interpretable and more accurate prediction. Our findings may also shed lights on the modeling of the long-term temporal network evolution, in contrast to short-term network prediction. Such models are crucial to forecast the long-term performance of e.g. epidemic/information spreading on the network. The linear regression assumed by Lasso could be one elementary mechanism to model temporal networks. The influence patterns that we have discovered can be further used to adapt other dynamic processes to model temporal networks. Our findings of the backbone network and its association with other network properties may also inspire the solution of network classification and optimization problems. For example, the spread of epidemic/information can be mitigated by blocking the temporal interactions of selected links [51], [52]. The influence of a link on and by the other links could possibly help with the selection of the links to block.

High-order models have been explored recently to account for various types of high-order dependencies in data on complex systems [53]. It has been found, for example, high-order models of paths in temporal networks, could improve node ranking [54], [55] and community detection [56]. Our Lasso Regression and Random Forest Model are high-order in broad sense: the state of a node pair depends on the states of many node pairs in the past over a period. These models together with the interpretations may shed light on the possible mechanism by which the temporal network may emerge and patterns in paths [53] may emerge.

ACKNOWLEDGMENTS

The authors would like to thank the support of the Netherlands Organisation for Scientific Research NWO (TOP Grant no. 612.001.802), Zhejiang Provincial Natural Science Foundation of China (Grant Nos.LQ22F030008) and Scientific Research Foundation for Scholars of HZNU (2021QDL030).

REFERENCES

[1] P. Holme and J. Saramäki, "Temporal networks," *Phys. Rep.*, vol. 519, no. 3, pp. 97–125, 2012.

[2] N. Masuda and R. Lambiotte, *A Guide to Temporal Networks*, ser. complexity science. World scientific (Europe), 2016, vol. 4.

[3] P. Holme and J. Saramäki, *Temporal Network Theory*, 1st ed., ser. Computational Social Sciences. Springer International Publishing, 2019.

[4] B. Min, K.-I. Goh, and A. Vazquez, "Spreading dynamics following bursty human activity patterns," *Phys. Rev. E*, vol. 83, p. 036102, 2011.

[5] G. Grinstein and R. Linsker, "Power-law and exponential tails in a stochastic priority-based model queuens," *Phys. Rev. E*, vol. 77, p. 012101, 2008.

[6] H.-H. Jo, M. Karsai, J. Kertész, and K. Kaski, "Circadian pattern and burstiness in mobile phone communication," *New J. Phys.*, vol. 14, p. 013055, 2012.

[7] M. Morris and M. Kretzschmar, "Concurrent partnerships and transmission dynamics in networks," *Soc. Networks*, vol. 17, pp. 299–318, 1995.

[8] S. Bansal, J. Read, B. Pourbohloul, and L. A. Meyers, "The dynamic nature of contact networks in infectious disease epidemiology," *J. Biol. Dyn.*, vol. 4, no. 5, pp. 478–489, 2010.

[9] N. Masuda, K. Klemm, and V. M. Eguíluz, "Temporal networks: Slowing down diffusion by long lasting interactions," *Phys. Rev. Lett.*, vol. 111, no. 18, p. 188701, 2013.

[10] B. Ribeiro, N. Perra, and A. Baronchelli, "Quantifying the effect of temporal resolution on time-varying networks," *Sci. Rep.*, vol. 3, no. 1, p. 3006, 2013.

[11] J. L. Iribarren and E. Moro, "Impact of human activity patterns on the dynamics of information diffusion," *Phys. Rev. Lett.*, vol. 103, no. 3, p. 038702, 2009.

[12] M. Karsai, M. Kivelä, R. K. Pan, K. Kaski, J. Kertész, A.-L. Barabási, and J. Saramäki, "Small but slow world: How network topology and burstiness slow down spreading," *Phys. Rev. E*, vol. 83, no. 2, pp. 1550–2376, 2011.

[13] T. Takaguchi, N. Masuda, and P. Holme, "Bursty communication patterns facilitate spreading in a threshold-based epidemic dynamics," *Plos ONE*, vol. 8, p. e68629, 2013.

[14] D. G. Rand, S. Arbesman, and N. A. Christakis, "Dynamic social networks promote cooperation in experiments with humans," *Proc. Natl. Acad. Sci.*, vol. 108, no. 48, pp. 19 193–19 198, 2011.

[15] X.-X. Zhan, A. Hanjalic, and H. Wang, "Information diffusion backbones in temporal networks," *Sci. Rep.*, vol. 9, p. 6798, 2019.

[16] X.-X. Zhan, Z. Li, N. Masuda, P. Holme, and H. Wang, "Susceptible-infected-spreading-based network embedding in static and temporal networks," *EPJ Data Science*, vol. 9, p. 30, 2020.

[17] L. Lü, M. Medo, C. H. Yeung, Y.-C. Zhang, Z.-K. Zhang, and T. Zhou, "Recommender systems," *Phys. Rep.*, vol. 519, no. 1, pp. 1–49, 2012, recommender Systems.

[18] A. Aleta, M. Tuninetti, D. Paolotti, Y. Moreno, and M. Starnini, "Link prediction in multiplex networks via triadic closure," *Phys. Rev. Res.*, vol. 2, p. 042029(R), 2020.

[19] C. Ma, T. Zhou, and H.-F. Zhang, "Playing the role of weak clique property in link prediction: A friend recommendation model," *Sci. Rep.*, vol. 6, p. 30098, 2016.

[20] D. Tang, W. Du, L. Shekhtman, Y. Wang, S. Havlin, X. Cao, and G. Yan, "Predictability of real temporal networks," *Natl. Sci. Rev.*, vol. 7, no. 5, p. 929–937, 2020.

[21] J. G. Kemeny and J. L. Snell, *Markov Chains*. New York: Springer-Verlag, 1976.

[22] S. Das and S. K. Das, "A probabilistic link prediction model in time-varying social networks," in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–6.

[23] R. R. Sarukkai, "Link prediction and path analysis using markov chains," *Computer Networks*, vol. 33, pp. 377–386, 2000.

[24] L. Zhou, Y. Yang, X. Ren, F. Wu, and Y. Zhuang, "Dynamic network embedding by modeling triadic closure process," *Thirty-Second AAAI Conference on Artifcial Intelligence*, vol. 32, no. 1, Apr. 2018.

[25] G. H. Nguyen, J. B. Lee, R. A. Rossi, N. K. Ahmed, E. Koh, and S. Kim, "Continuous-time dynamic network embeddings," in *Companion Proceedings of the The Web Conference 2018*, ser. WWW '18. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, 2018, p. 969–976.

[26] M. Rahman, T. K. Saha, M. A. Hasan, K. S. Xu, and C. K. Reddy, "Dylink2vec: Effective feature representation for link prediction in dynamic networks," *CoRR*, vol. abs/1804.05755, 2018.

[27] X. Yu and T. Chu, "Dynamic link prediction using restricted boltzmann machine," in *2017 Chinese Automation Congress (CAC)*, 2017, pp. 4089–4092.

[28] X. Li, N. Du, H. Li, K. Li, J. Gao, and A. Zhang, "A deep learning approach to link prediction in dynamic networks," in *2014 SIAM International Conference on Data Mining*, M. Zaki, Z. Obradovic, P. N. Tan, A. Banerjee, C. Kamath, and S. Parthasarathy, Eds., 2014, p. 289–297.

[29] J. Chen, X. Xu, Y. Wu, and H. Zheng, "Gc-lstm: Graph convolution embedded lstm for dynamic link prediction," *CoRR*, 2018.

[30] A. Pareja, G. Domeniconi, J. Chen, T. Ma, T. Suzumura, H. Kanezashi, T. Kaler, T. B. Schardl, and C. E. Leiserson, "Evolvegcn: Evolving graph convolutional networks for dynamic graphs," *CoRR*, 2019.

[31] J. Li, J. Peng, S. Liu, and C. L. L. Weng, "Tsam: Temporal link prediction in directed networks based on self-attention mechanism," *CoRR*, 2021.

[32] L. Qu, H. Zhu, Q. Duan, and Y. Shi, "Continuous-time link prediction via temporal dependent graph neural network," *Proceedings of The Web Conference 2020*, 2020.

[33] "Hypertext 2009 network dataset–konect," http://konect.uni-koblenz.de/networks/sociopatterns-hypertext.

[34] L. Isella, J. Stehlé, A. Barrat, C. Cattuto, J.-F. Pinton, and W. V. den Broeck, "What's in a crowd? analysis of face-to-face behavioral networks," *J. Theor. Biol.*, vol. 271, pp. 166–180, 2011.

[35] "Sociopatterns," http://www.sociopatterns.org/datasets/.

[36] P. Sapiezynski, A. Stopczynski, L. D. D., and L. S., "Interaction data from the copenhagen networks study," *Sci. Data*, vol. 6, p. 315, 2019.

[37] V. Gelardi, J. Fagot, A. Barrat, and N. Claidiére, "Detecting social (in)stability in primates from their temporal co-presence network," *Animal Behaviour*, vol. 157, pp. 239–254, 2019.

[38] D. P. Mersch, A. Crespi, and L. Keller, "Tracking individuals shows spatial fidelity is a key regulator of ant social organization," *Science*, vol. 340, pp. 1090–1093, 2013.

[39] A.-L. Barabási, "The origin of bursts and heavy tails in human dynamics," *Nature*, vol. 435, pp. 207–211, 2005.

[40] A. Moinet, M. Starnini, and R. Pastor-Satorras, "Burstiness and aging in social temporal networks," *Physical review letters*, vol. 114 10, p. 108701, 2015.

[41] B. Min and K. I. Goh, *Burstiness: Measures, models, and dynamic consequences*, ser. Understanding Complex Systems. Springer Verlag, 2013, pp. 41–64.

[42] F. Santosa and W. W. Symes, "Linear inversion of band-limited reflection seismograms," *SIAM J. Sci. and Stat. Comput.*, vol. 7, no. 4, p. 1307–1330, 1986.

[43] R. Tibshirni, "Regression shrinkage and selection via the lasso," *J. R. Stat. Soc. Series B*, vol. 58, no. 1, pp. 267–88, 1996.

[44] T. K. Ho, "The random subspace method for constructing decision forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832–844, 1998.

[45] E. M. Kleinberg, "Stochastic discrimination," *Ann. Math. Artif. Intell.*, vol. 1, pp. 207–239, 1990.

[46] F. Tom, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, pp. 861–874, 2006.

[47] L. Zou, C. Wang, A. Zeng, Y. Fan, and Z. Di, "Link prediction in growing networks with aging," *Social Networks*, vol. 65, pp. 1–7, 2021.

[48] M. Kitsak, I. Voitalov, and D. Krioukov, "Link prediction with hyperbolic geometry," *Phys. Rev. Res.*, vol. 2, p. 043113, Oct 2020.

[49] J. Sun, F. Quevedo, and E. M. Bollt, "Data fusion reconstruction of spatially embedded complex networks," *preprint at arXiv:1707.00731v1*, 2017.

[50] B. Efron, T. Hastie, and I. Johnstone, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[51] H. Cherifi, S. Gaito, J. F. Mendes, E. Moro, and L. M. Rocha, Eds., *Suppressing information diffusion via link blocking in temporal networks*, ser. Studies in Computational Intelligence, vol. 881. Lisbon, Portugal: Springer, Cham, 2020.

[52] A. Arenas, W. Cota, J. G.-G. nes, S. Gómez, C. Granell, J. T. Matamalas, D. S.-P. nos, and B. Steinegger, "Modeling the spatiotemporal epidemic spreading of covid-19 and the impact of mobility and social distancing interventions," *Phys. Rev. X*, vol. 10, p. 041055, 2020.

[53] R. Lambiotte, M. Rosvall, and I. Scholtes, "From networks to optimal higher-order models of complex systems," *Nature Physics*, vol. 15, pp. 313–320, 2019.

[54] I. Scholtes, N. Wider, and A. Garas, "Higher-order aggregate networks in the analysis of temporal networks: path structures and centralities," *Eur. Phys. J. B*, vol. 89, p. 61, 2016.

[55] M. Rosvall, A. V. Esquivel, A. Lancichinetti, and J. D. West, "Memory in network fows and its efects on spreading dynamics and community detection," *Nat. Commun.*, vol. 5, p. 5630, 2014.

[56] A. R. Benson, D. F. Gleich, and J. Leskovec, "Higher-order organization of complex networks," *Science*, vol. 353, pp. 163–166, 2016.

Li Zou is currently a PhD candidate in the Multimedia Computing Group, Department of Intelligent Systems at Delft University of Technology. She received the MSc degree from Beijing Normal University, Beijing, China, in 2019. Her research interests are prediction and modeling of temporal networks.

Xiu-Xiu Zhan received the Doctoral degree in computer science from the Delft University of Technology, Delft, The Netherlands in 2020. She is currently an associate Professor with Alibaba Business School, Hangzhou Normal University, Hangzhou, China. Her research interests include complex networks, information diffusion, and machine learning. She has authored or coauthored more than 20 international journal papers and conference papers in physics and computer science, including Physics Reports, Applied Mathematics and Computation, and New Journal of Physics.

Jie Sun is a chief researcher at Huawei Hong Kong Research Center. During the period of completing this project, he was an associate professor in the Department of Mathematics at Clarkson University and the Clarkson Center for Complex Systems Sciences, where he worked between 2012 and 2019, initially as a tenure-track assistant professor and promoted to associate professor with tenure in 2017. He received the BS degree in Physics from Shanghai Jiao Tong University, China, in 2006 and PhD degree in Applied Mathematics from Clarkson University, USA, in 2009 and subsequently received postdoctoral training at Northwestern University, USA, and Princeton University, respectively. With over 50 journal publications, his research spans a broad range of topics centered around complex systems, nonlinear dynamics and control aspects of network science; with a recent focus on causal inference, machine learning and optimization with dynamical systems. He is a member of SIAM and APS, and an editor of the journal Chaos: an interdisciplinary journal of nonlinear science.

Alan Hanjalic is a professor of computer science, head of the Multimedia Computing Group and head of the Intelligent Systems Department at the Delft University of Technology (TU Delft), The Netherlands. His research interests are in the fields of multimedia information retrieval and recommender systems, in which he (co-)authored more than 250 publications. He is co-recipient of the Best Paper Award at the ACM Conference on Recommender Systems (ACM RecSys) 2012, the ACM International Conference on Multimedia (ACM Multimedia) 2017 and the IEEE International Conference on Multimedia Big Data (IEEE BigMM) 2019. He served as the Chair of the Steering Committee of the IEEE Transactions on Multimedia, the Associate Editor-in-Chief of the IEEE MultiMedia Magazine, and an Associate Editor of many scientific journals, including the IEEE Transactions in Multimedia, IEEE Transactions on Affective Computing and ACM Transactions on Multimedia Computing, Communications and Applications. He also served as the General and Program (Co-)Chair in the organizing committees of all major conferences in the multimedia domain, including ACM Multimedia, ACM CIVR/ICMR, and IEEE ICME.

Huijuan Wang is currently an Associate Professor in the Multimedia Computing Group, Department of Intelligent Systems at Delft University of Technology. She received the MSc degree and PhD degree in Electrical Engineering from Delft University of Technology in 2005 and 2009, respectively. She has been a visiting scientist in the Department of Physics at Boston University since 2011 and in the Department of Electrical Engineering at Stanford University in 2015. Her main research areas are prediction, modeling and control of dynamic processes (such as information, disease, failure propagation and social or financial contagion) on interdependent and time-evolving networks combining data and network science approaches.