Predicting the Evolution of Social Networks: Optimal Time Window Size for Increased Accuracy

Marcin Budka
Bournemouth University
Smart Technology Research Centre
School of Design,
Engineering and Computing
Talbot Campus, Fern Barrow
Poole, BH12 5BB, UK
mbudka@bournemouth.ac.uk
http://www.budka.co.uk/

Katarzyna Musial
King's College London
Department of Informatics
School of Mathematical
and Natural Sciences
Strand Campus
WC2R 2LS London, UK
katarzyna.musial@kcl.ac.uk
http://www.katemusial.com/

Krzysztof Juszczyszyn
Wroclaw University of Technology
Faculty of Computer Science
and Management
Wybrzeze Wyspianskiego 27
50-370 Wroclaw, PL
krzysztof.juszczyszyn@pwr.wroc.pl
http://www.ii.pwr.wroc.pl/~juszczyszyn/

***Abstract*—This study investigates the data preparation process for predictive modelling of the evolution of complex networked systems, using an e–mail based social network as an example. In particular, we focus on the selection of optimal time window size for building a time series of network snapshots, which forms the input of chosen predictive models. We formulate this issue as a constrained multi–objective optimization problem, where the constraints are specific to a particular application and predictive algorithm used. The optimization process is guided by the proposed Windows Incoherence Measures, defined as averaged Jensen-Shannon divergences between distributions of a range of network characteristics for the individual time windows and the network covering the whole considered period of time. The experiments demonstrate that the informed choice of window size according to the proposed approach allows to boost the prediction accuracy of all examined prediction algorithms, and can also be used for optimally defining the prediction problems if some flexibility in their definition is allowed.**

I. Introduction

In this paper we investigate the process of data preparation for subsequent modelling of the dynamics of complex networked systems. In our view this fundamental issue did not receive due attention in the complex networks literature and hence lacks wider recognition in the complex systems community. On the contrary though, various studies in the area of applied data mining [25], suggest that data preparation can take up to 80% of the modeling efforts and is absolutely crucial for development of top-performing and, even more importantly, meaningful models. Since in our recent research [16]–[18], [26] we have been focusing on the analysis of the dynamics of complex networks in the context of predictive tasks (e.g. which links are likely to appear or disappear in the future), we are effectively balancing at the boundary of the two disciplines. For the fusion of these sciences to be most beneficial, we find it necessary to transfer good practices between the two, with careful and concise data preparation being one of the key data mining practices that should be applied in other disciplines.

In particular, in this study we focus on the process of slicing the available data describing the evolution of an e-mail network over time, into a number of time windows, effectively forming a time series of network snapshots that can subsequently be used for training a predictive model. We propose and investigate a principled way of selecting an optimal time window size, which allows to achieve good prediction performance, at the same time respecting the application and predictive method specific constraints, like minimal length of the time series or required prediction horizon.

One of the main motivations of the presented approach is the intuition that some key characteristics, of an evolved network are not likely to change over time. As a result, for example a random network is not likely to evolve into a scale-free network. In most cases, the characteristics like degree and motif distributions, network diameter etc. remain relatively stable, despite the fact that the node activity fluctuates and networks undergo rewiring on the local topology level. Hence we postulate that it is crucial for the networks represented in the time windows to have similar properties to the global, evolved network for the prediction problem to be meaningful. If this condition is not met, due to the distorted and noisy nature of the input time series, it can be hardly possible to achieve prediction accuracy better than chance.

In this study we also argue that analysis of dynamics of a complex networked system should always be performed in a context of what the network actually represents. Given a graph which describes some unknown network, assuming that every edge corresponds to a relation might be an overstatement. In email network for example, the mere fact that a message has been sent from email account A to email account B does not necessarily imply a relation between account owners. If the message was sent by a computer virus or the network represents exchange of messages on a dating site and user B has never responded, we can hardly call this a relation.

This paper is structured as follows. In Section II we describe the classical network growth models, focusing on their applicability to prediction of network evolution in a given time frame on a local scale. Section III has been devoted to the definition of relation in a network, various time window types and units.

In Section IV we sketch a new approach to the problem of optimal choice of time window size by using the proposed Window Incoherence Measures in a course of constrained optimization. Section V presents the experimental results for an e-mail communication network, while the conclusions and future research directions can be found in Section VI.

II. Network types and growth models

There are many types of complex networked systems. One of the classifications distinguishes infrastructural (Internet, WWW, energy and transportation networks) and natural complex systems (biological networks, social systems and ecosystems) [6]. Although all of them consist of nodes and interactions between them, their structures and dynamics differ. Most of the existing network models were developed to some extent based on the observation of the real-world networks. Usage of these models results in creation of networks that follow specific node degree distributions. Additionally some of them feature communities and short paths between nodes. Most often used networks architectures are (1) regular networks, (2) random graphs (with short paths), (3) small-world networks (with short paths and high clustering), and (4) scale-free networks (with hubs – highly connected nodes) [5], [31]. The overview of the main characteristics of these models is presented in Table I.

Regular simple architectures such as chains, grids, lattices and fully-connected graphs do not model any specific characteristics of real-world networks but in turn they allow us to focus on the complexity caused by the nonlinear dynamics of the nodes, without being burdened by any additional complexity in the network structure itself [31].

Random graphs are characterized by node degree distribution $P(k)$, giving the probability that a node in the network is connected to k other nodes, that peaks at an average $< k >$ and decays exponentially for large k. The emerging structure features small-world effect, i.e. the degree of separation between two randomly picked nodes is small. At the same time random networks are not clustered, hence the communities are not present. Erdos and Renyi in [29] presented the formal theory of random networks. The detailed description of models and features of random graphs is presented in [9].

Small-world model is another approach to describe social networks [33]. In opposite to random graphs, small-world network is a structure with high clustering coefficient. The friend of a friend phenomenon where people tend to create dense groups is one of the main characteristics of these networks. At the same time short paths between nodes are also present. Both random and small-world models lead to a fairly homogeneous networks, in which each node has approximately the same number of links, $k \simeq < k >$ [3].

Another network model, which is widely used, is the scale-free model that is a useful approximation of large networks for which $P(k)$ follows a power-law, that is $P(k) \sim k^{-\gamma}$, free of a characteristic scale. This type of node degree distribution can be observed in the World-Wide Web [2], [14], the Internet [13] and other large networks [30]. Hubs – highly connected nodes are statistically significant in scale-free networks [3].

There has been extensive research devoted to analyzing the proposed models [1], [8]. Characteristics and processes such as weak links, synchronisation, spread of information/diseases, stability, robustness and resilience are well defined for existing models. The dependance on a small number of network properties (primarily the node degree distribution) is their main limitation. Even if a network can be described by a given node degree distribution, it was shown that the underlying local structure can change dramatically.

TABLE I
NETWORK MODELS CHARACTERISTICS; N–NUMBER OF NODES;

Feature	**Random Networks**	**Small-world Networks**	**Scale-free Networks**
Degree Distribution	Poisson dist.	Poisson dist.	Power–law dist.
Clustering Coefficient	Low $= \frac{k}{1/N}$	High $= \frac{3(k-1)}{2(2k-1)} \cdot (1-p)^3$ p – rewiring probability	Higher than in random network $\sim N^{-0.75}$
Average Path Length	Small $= \frac{\ln N}{\ln k}$	Small $= \frac{\ln(Nkp)}{k^2 p}$	Small $\sim \frac{\ln N}{\ln \ln N}$

As more and more data becomes available for network analysis, the need for more detailed modelling techniques is clearly visible. In addition, the dynamics of the systems is a very important element to model and as one can see the models themselves do not support such task. Thus, there were several methods proposed to describe the growth and dynamics of networked systems, such as preferential attachment or vertex coping models [28]. There are also some approaches that aim at developing specific models for online social networks and take into consideration some information characteristic to such networks [10], [21], [22].

One of the main drawbacks of all these growth models is that they are only able to account for global properties of the network. This can mean that for example in preferential attachment, although the number and degree of hubs might be predicted relatively well, this may not be true for the identity of the nodes which became hubs. Hence predicting the exact location of a new node/edge in the network is beyond the ability of this class of models. Also, all the network models discussed above are time independent, i.e. they assume that new node/edge will come to the network at some point that is not defined in the context of time. In the real-world networks time is the inseparable element of their evolution and it cannot be neglected during the analysis. Thus, it is crucial to investigate the system dynamics at the right level of granularity as depending on the size of time window one can obtain different results of the analysis. As it has been shown in [32] too small time window can result in a very noisy data to be analysed whereas too big can hide interesting patterns as the network may seem to be stable structure. Although the concept presented in [32] is worth mentioning, the authors did not perform extensive studies to investigate it further.

III. Relations and time windows

An important issue in social network analysis is the definition of a relation. As mentioned before, this is even more crucial in the case of an e-mail network, where assuming that the mere fact of sending a message forms a relation can be misleading [12]. The relation should be defined in the context of a particular application, taking into account application-specific phenomena like the bursty nature of communication [4] etc. In a company e–mail network it may for example make sense to define a relation as an exchange of e–mail messages, which takes place within a number of business days. The actual number of days can be determined from the data (e.g. how fast and how many messages must be exchanged in order for the communication to occur again in the future).

The data coming from the e–mail server log are sequential, i.e. every e-mail message has a timestamp, which imposes a natural ordering of the messages. This ordering paired with an appropriate definition of a relation allows us to reconstruct the history of the social network evolution. In order to use this data for predicting what the network will look like in the future, it first needs to be divided into sequential chunks (time windows) effectively forming a multivariate time series consisting of a number of network snapshots. As demonstrated in Section V the way in which this time series is constructed is critical for the usability of the resultant predictive model.

Before discussing the types of the time windows we first define what a fixed–size and variable–size window is, in the context of different units that can be used to measure window size. Intuitively, the general rule is that a fixed–size window covers a fixed number of window size units, while a variable-size window does not. This property however does not necessarily translate across the units – a fixed–size window in terms of unit Y can at the same time be a variable–size window in terms of unit Z. The window size measures we propose to use have been summarized in Table II.

TABLE II
Window size measures

measure	symbol	dependency		
		T	E_C	N_C
time	T	**FIX**	VAR	VAR
edge count	E_C	VAR	**FIX**	VAR
node count	N_C	VAR	VAR	**FIX**

As it can be seen, in our case fixing the window size S in terms of any of the measures, makes the remaining two variable across multiple windows (the 'dependency' column). For example, if we decide to keep the window fixed in terms of time and set its size to one week, the number of edges and nodes from one window to another will most likely vary.

One argument for using the edge count rather than time to measure window size is the need for thresholding of the scores, which are the outputs of many predictive algorithms, including the Triad Transition Matrix (TTM) predictor [16] used in our experiments. If we don't know how many new edges to expect in the next time window, determining the correct value of the cut–off threshold might be difficult. If however we use the edge count as a measure of window size and increase it by a fixed number every time, then this fixed step automatically becomes the number of new edges. In this case however we are not able to give a precise time horizon for their appearance, similarly to the classical network growth models discussed in Section II, although it can be estimated from past observations.

Depending on the *longevity* of the relation the following types of time windows can be distinguished:

- **Growing** time window, which accommodates new incoming data without discarding old data and thus reflecting the assumption that once formed, a relation never disappears. This kind of window is variable–size in terms of all three measures given in Table II and can be used to model some types of networks like for example a network of acquaintances, in which the state of knowing somebody lasts indefinitely.
- **Sliding** time window, which has a fixed size in terms of the selected measure from Table II and hence in order to accommodate new data, old data must be dropped/forgotten.

An important parameter of the time window approach is the step at which the window grows or slides. This parameter is crucial as it determines the prediction horizon. For example, if one is interested in predicting what new links will appear in the network during the next week, i.e. the prediction horizon H is set to 7 days, the historical data needs to be divided into time windows by growing or sliding the initial window by exactly 7 days before being used for predictive model training.

IV. Our approach

For the predictive problem to be meaningful, time windows should be constructed in such a way, that the properties of the networks contained within each window are as close as possible to the characteristic of the global network (i.e. the network within the whole period covered by the input data). In other words, while the definition of the relation should be application–driven, the parameterization of the time windows should be data–driven, whenever possible.

A list of network properties/characteristics we have used in this study is given in Table III. The list is by no means exhaustive but we have chosen a blend of both local and global properties, each having different computational complexity. All of these properties have also been thoroughly investigated in the literature, have well established interpretation in the context of network analysis and some of them have proven their usefulness in link prediction (e.g. local clustering coefficient in the Common Neighbours predictor).

A natural way of comparing the distributions $P(\Phi)$ of the properties listed in Table III is by using a divergence measure. Although the best known measure of this kind is probably the Kullback-Leibler divergence [20], according to the findings of our recent study [11] we have rather opted for

TABLE III
NETWORK PROPERTIES

id (Φ)	name	complexity[a]
ND	Node degree distribution (in/out)	$O(V^2)$
SP	Shortest path length distribution[b]	$O(VE\log(V))$
TC	Triad census distribution[c]	$O(E)$
CC	Clustering coefficient distribution	$O(Vd^2)$
KS	Katz score[d] distribution	$O(V^3)$
BC	Betweenness centrality[e] distribution	$O(VE)$

[a] V – vertex count, E – edge count, d – maximum vertex degree

[b] Using the Johnson's algorithm for sparse graphs [15]

[c] Using the algorithm from [7]

[d] Using matrix inversion to calculate exact scores [19]; approximate algorithm with complexity $O(V^2)$ can also be used

[e] For unweighted graphs; for weighted graphs the complexity is $O(VE + V(V+E)\log(V))$

the Jensen-Shannon divergence [24], which for two probability distributions $P(x)$ and $Q(x)$ has been defined as:

$$D_{JS}(P,Q) = \frac{1}{2}D_{KL}(P,M) + \frac{1}{2}D_{KL}(Q,M) \qquad (1)$$

where $M(\boldsymbol{x}) = \frac{1}{2}(P(\boldsymbol{x}) + Q(\boldsymbol{x}))$ and D_{KL} is the Kullback-Leibler divergence given by:

$$D_{KL}(P,Q) = \sum_{x} P(\boldsymbol{x}) \log \frac{P(\boldsymbol{x})}{Q(\boldsymbol{x})} \qquad (2)$$

The choice of D_{JS} has been dictated by the fact that it is symmetric and, unlike D_{KL}, defined also for $Q(x) = 0$.

The issue of determining the optimal values of time window size and prediction horizon can be seen as a constrained optimization problem, where the objective function is the divergence of the distributions of any network characteristic or even multiple divergences of a number of network characteristics (multi-objective optimization). There are two constraints in this problem: (1) the prediction horizon, largely determined by the particular application of the predictive model and (2) the minimum size of the resultant time series, determined by the predictive algorithm one wants to use and by the dynamics of the time series itself.

The time series length issue stems from the fact that this length depends on both time window size and prediction horizon which are also coupled. This functional dependence has been shown in Figure 1. As it can be seen, the time series length only marginally depends on the time window size (additive factor) but at the same time heavily depends on the prediction horizon (multiplicative factor).

Formally, the constrained multi-objective optimization problem can be states as follows. Denoting by N the input data size, by S the time window size, by H the prediction horizon, by L_{min} the minimal time series length, by L the actual time series length and by $\mu_i(S,H)$ the i^{th} objective function out of M, the optimization problem becomes:

$$\begin{aligned} \underset{S,H}{\operatorname{argmin}} \quad & [\mu_1(S,H), \mu_2(S,H), \ldots, \mu_M(S,H)]^T \\ \text{s. t.} \quad & L = \left\lfloor \frac{N-S}{H} + 1 \right\rfloor \geq L_{min} \end{aligned} \qquad (3)$$

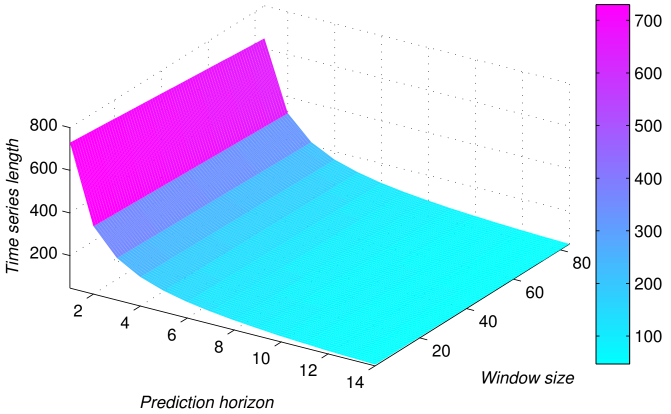

Fig. 1. The amount of training data (time series length) as a function of window size and prediction horizon (time step) for the input data of 730 days (two years) and a sliding time window

where in a special case considered in this paper $M = 1$.

The objective function we use in this study has been termed the Window Incoherence Measure (WIM) and for a network property Φ is defined as:

$$\mu^{(\Phi)} = \frac{1}{L}\sum_{i=1}^{L} D_{JS}\left(P(\Phi(w_i)), P(\Phi(W))\right) \qquad (4)$$

where w_i denotes the i^{th} time window in the time series and $W = \sum_L w_j$. Whenever we use a specific network property from Table III, for simplicity of notation we denote the WIM by the name of this property in *italics*, e.g. instead of writing $\mu^{(TC)}$ we simply write *TC*.

V. EXPERIMENTS AND RESULTS

A. Experimental protocol

1) Data: Due to the exhaustive character of our experiments we have used a dataset generated from the well–known ENRON e–mail database[1], where there are only 151 nodes in the social network. The database covers a period of over 3 years (1138 days). We have divided it into 1–day time windows forming an input dataset with $N = 1138$ instances (hence using time as a measure of window size). At this stage we have assumed that a relation between node A and B in a given time window exists, if A has sent B an e–mail within this time window. This choice has been driven by the fact, that if we defined the relation by a message–response pair over e.g. a 5 working days, the network would become very sparse and would not reflect the ongoing communication activity recorded in the mail server logs. This can be seen in Figure 2, which depicts the cumulative probability of receiving a response after a given number of working days. Therefore, in the network under consideration, we found the above approach justified.

The weight $W_{A\rightarrow B}$ of a relation between node A and node B has been calculated using the following formula:

$$W_{A\rightarrow B} = \frac{\sum_{i=1}^{N_{A\rightarrow B}} R_i^{-1}}{\sum_{X\in V\setminus\{A\}} N_{A\rightarrow X}} \qquad (5)$$

[1] http://www.cs.cmu.edu/~enron/

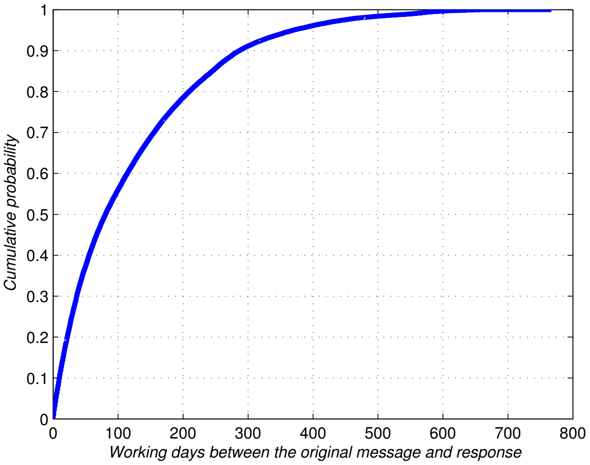

Fig. 2. Cumulative probability of receiving a response after a given number of working days

where $N_{A\rightarrow B}$ denotes the number of messages sent from A to B in a given time window, R_i is the number of recipients of the i^{th} message in this time window and V is the set of all nodes in the network.

Following a generally accepted approach for validation of models built on sequential data, the input dataset has been then divided into two parts:

- **Training dataset**, covering the period of the first 2 years ($N_{TR} = 730$ days) and used for all the operations required to build a predictive model like tuning various parameters (e.g. selecting the optimal time window size) and training the predictive model.
- **Test dataset**, covering the remaining period of over 1 year ($N_{TE} = 408$) and used exclusively for assessing the performance of the obtained predictive models.

In all the experiments we vary the prediction horizon between 1 day and 56 weeks, in order to cover a wide range of application scenarios.

2) Predictive models: In our experiments we have used the Triad Transition Matrix predictor (TTM) as a main method, which is based on a 1–st order probabilistic model of transitions between various triad types, resulting from an observation that there exist distinctive patterns which drive the evolution of connections between nodes. TTM has been first proposed in [16] and further evaluated in [17], demonstrating good performance and robustness, especially for sparse networks emerging in short time windows.

As mentioned before, the reported prediction performance has been obtained on the independent test set, which was used to train the predictors. In this study we did not follow the test-then-train protocol, which involves updating the predictive model after casting a prediction for a given time window, thus accommodating the latest available information. Although this would most likely further increase the accuracy of predictions, the test data size (and hence the number of updates) varies with the size of the prediction horizon, which could make the results incomparable across different horizons.

Two additional standard methods have also been used in order to illustrate the applicability of the proposed approach: (1) Preferential Attachment (PA), where the basic premise is that the probability that a new edge has node A as an endpoint is proportional to, the current number of neighbours of A, and (2) Common Neighbours (CN), which is based on the correlation between the number of common neighbors of A and B at any time instant and the probability that they will collaborate in the future. Both methods have been verified and discussed in [27].

3) Accuracy of predictions: We have followed the verification scheme proposed in [23], where it was assumed, that all the predictors assign a predicted connection weight $score(A,B)$ to unlinked pairs of nodes (A,B), based on the input graph, and then produce a ranked node pair list in decreasing order of $score(A,B)$. The scores are interpreted as the estimated probability of forming a new link between A and B. In this way each link predictor outputs a ranked list of node pairs which would eventually form predicted new links. The list is being sorted in decreasing values of scores and the set of first n entries is taken, then the size of its intersection with the set of new links (of the same size n) is computed. The percentage of links from the predicted set, which are also present in the set of new links, is the prediction accuracy. However, in [24] it was also postulated that it is reasonable to seek new links only between the nodes already connected, therefore only the links joining nodes adjacent to at least 3 other existing links are predicted. In our view, when one is dealing with sparse and dynamic networks (in addition to variable time windows) this assumption restricts link prediction only to the densely connected parts of the network and hence we did not impose this restriction in our experiments.

B. Minimization of the divergence

Figures 3 and 4[2] depict the values of the Window Incoherence Measure for all the network properties from Table III[3] for the training data and networks built using time windows/prediction horizons of varying size, for the unweighed and weighted cases respectively. The minimal window size has been limited to the size of the prediction horizon to ensure that no training data is skipped during the procedure (this results in half of each plot being empty). The black crosses denote minimal values for each column, marking optimal (with respect to the WIM used) time window sizes for each prediction horizon. Intuitively, one might be tempted to use as big time window as possible to cover as much data as possible. Our results however show, that this is not always the optimal choice. For example, in the case of *NDu* if we choose the prediction horizon to be 26 days long, the optimal window size is 32 days rather than the maximal possible 56 days (red

[2] The actual ‘min’ and ‘max’ values in these figures are different for every subplot, hence they are not reported in order to keep a single color scale.

[3] Please note that we have appended an ‘u’ or ‘w’ to the abbreviation of the property name to denote respectively ‘unweighted’ (binary) and ‘weighted’ network used as an input for the calculation of the property value.

circle in top-left Figure 3). This is even more visible in the case of other measures (*SPu*, *CCu* and *BCu*), while *KS* behaves in a way more consistent with the initial intuition.

The situation for the weighted network (Figure 4) confirms the above, with *SPw* consistently preferring small windows, which is also emphasized in the case of *BCw* (note that *TC* and *KS* are the same for both networks as the weights are ignored during calculation of these two).

The results presented in Figures 3 and 4 can also be used in another way. If the prediction horizon is flexible (e.g. we want it to be *around* 7 days) it may turn out that setting the horizon to 6 or 9 days instead might make the divergence below the levels possible to obtain with $H = 7$. Table IV contains the WIM values for a number of prediction horizons, minimized with respect to the window size. As it can be seen, according to different measures, rather than using a 7-, 14- or 21-day prediction horizon, a slight increase or decrease in horizon size can allow to reduce the WIM and potentially result in more accurate predictions, as will be discussed in Section V-C.

As has been shown, the proposed approach can be helpful in at least two situations: (1) when the problem is completely defined, i.e. a fixed H is given, an optimal time window size S for the given value H will be recommended; (2) when the problem is not completely defined, i.e. H is given as a range or set of possible values, optimal values for both H and S will be recommended, helping to completely define the problem at the same time.

C. Correlation with prediction accuracy and optimal window size

In Figure 5 we present the absolute values of the Spearman's rank correlation coefficient between WIM and prediction accuracy of the three predictive models that have been used in the experiments. As it can be seen, *NDu*, *TC*, *KS*, *NDw* and *CCw* are moderately (≥ 0.4) correlated with the performance of TTM, two of which (*TC* and *CCw*) feature a relatively low computational complexity (see Table III), which additionally increases their attractiveness. The correlation with the accuracy of PA and CN is weak to none.

The left parts of Figures 7, 8 and 9 depict the true optimal window size for each prediction horizon as determined by maximum accuracy of the respective predictive methods on the independent test set together with recommended window size on the basis of the proposed measures. Since according to the presented results none of the measures from Table III is individually highly correlated with the prediction accuracy, we have devised the following two approaches allowing to take advantage of the most appropriate subset of WIMs:

- **Globally-weighted combination (GWC)**, where the recommendation is a weighted combination of the recommendations of all incoherence measures, with the weights determined from the training dataset, irrespectively of the prediction horizon. The weights have been calculated as a fraction of the times each WIM's recommendation $S_{rec}^{(\Phi)}$ is closest to the optimal window size S_{opt}.

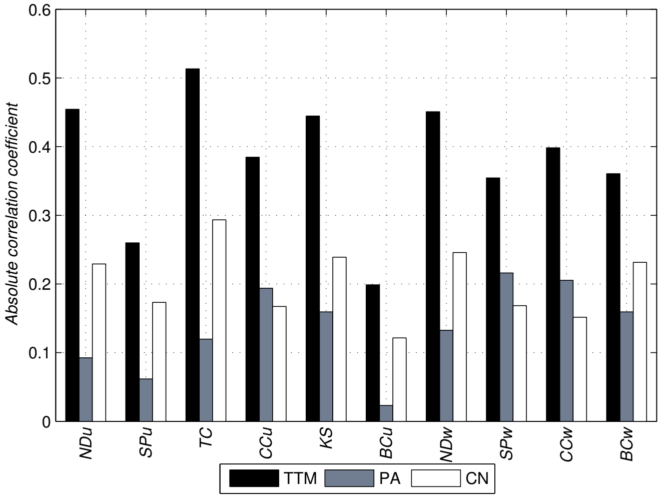

Fig. 5. Correlation between WIM and prediction accuracy

- **Locally-weighted combination (LWC)**, where the recommendation is a weighted combination as above but there is a separate set of weights for each prediction horizon. The weights decay exponentially with decreasing quality of the recommendation on the training set and have been calculated as: $\exp(-|S_{opt} - S_{rec}^{(\Phi)}|)$. Example weights of a local combination have been depicted in Figure 6. As it can be seen, in this case many weights are driven towards very low values, with 1-2 WIMs dominating for each prediction horizon.

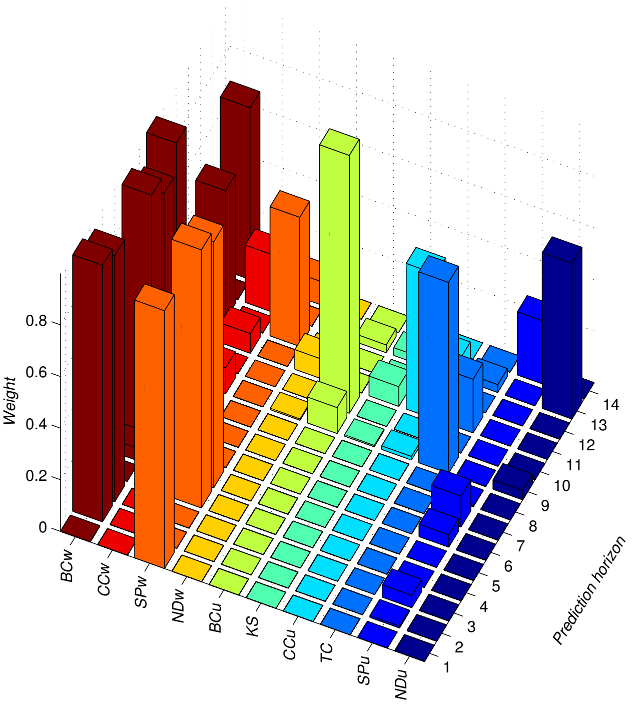

Fig. 6. Locally-weighted combination weights for TTM and $H \in (1, 14)$

From Figures 7, 8 and 9 (left) it is easy to notice that the window size recommendations obtained with LWC are much more accurate than the ones produced by GWC. This is most

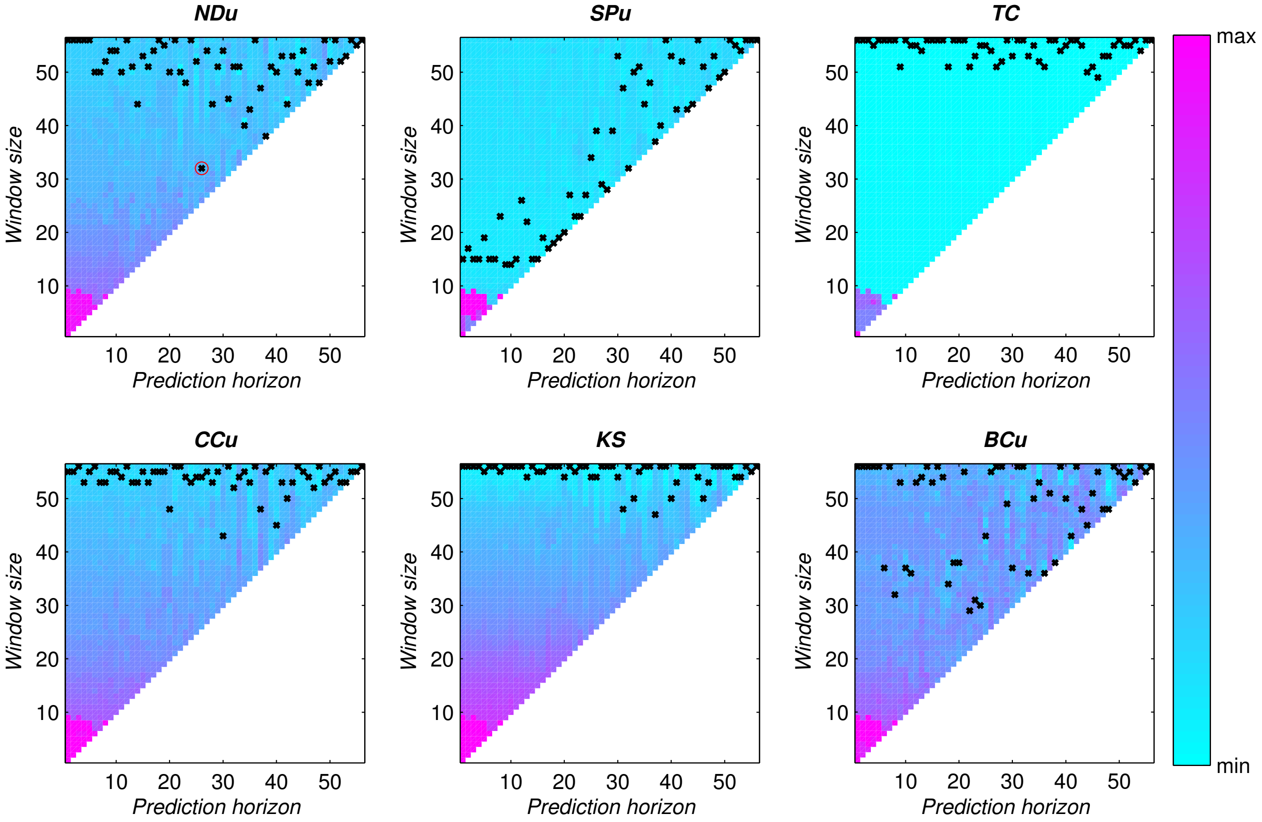

Fig. 3. JS-divergence (color-coded) v. window size [days] v. prediction horizon [days] for various unweighted network properties, averaged over all windows

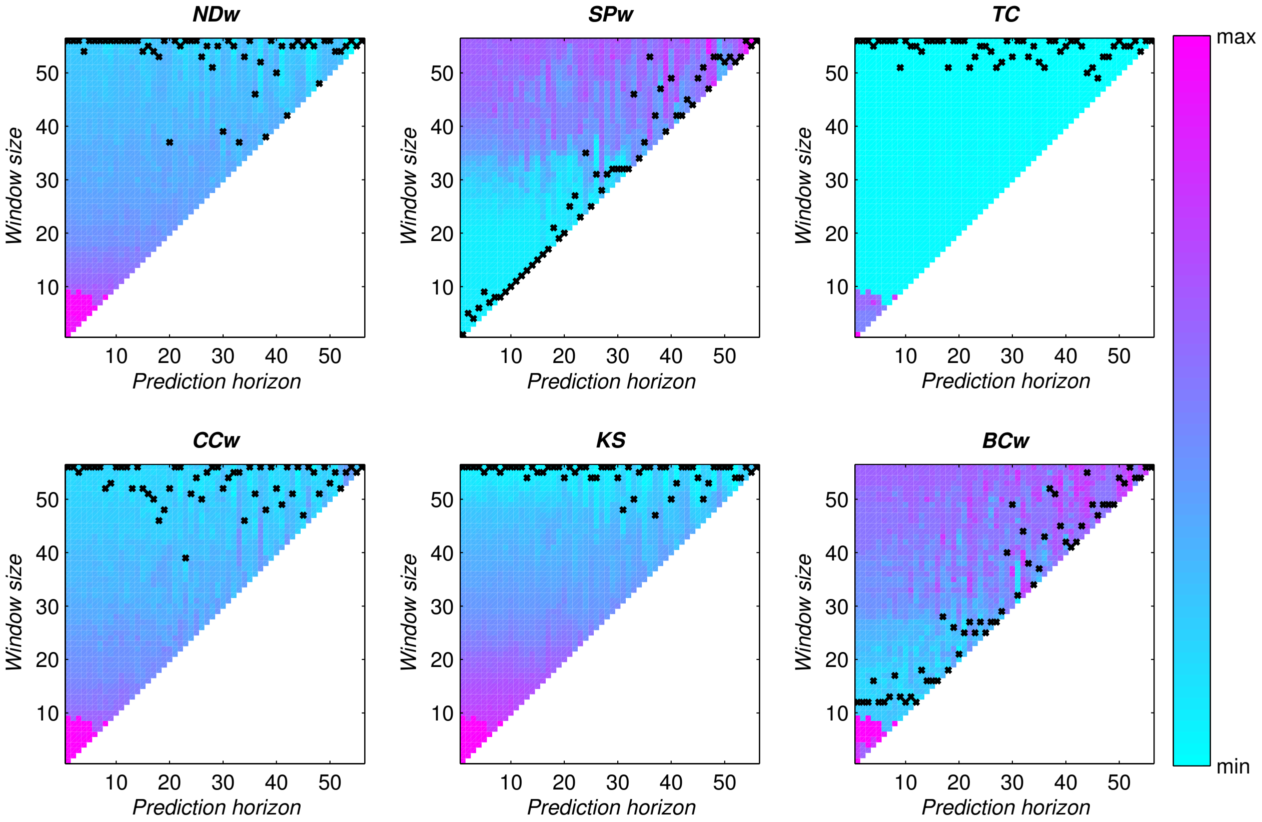

Fig. 4. JS-divergence (color-coded) v. window size [days] v. prediction horizon [days] for various weighted network properties, averaged over all windows

TABLE IV
OPTIMAL WINDOW SIZES FOR GIVEN PREDICTION HORIZONS +/- TOLERANCE

id	**prediction horizon +/- tolerance**														
	-1	**7**	**+1**	**-2**	**-1**	**14**	**+1**	**+2**	**-3**	**-2**	**-1**	**21**	**+1**	**+2**	**+3**
NDu	**.6708**	.6712	.6710	.6707	.6708	**.6705**	.6708	.6711	.6708	.6724	.6709	.6722	.6708	**.6700**	.6711
SPu	.1627	.1630	**.1618**	.1633	.1515	.1627	.1653	**.1369**	.1496	.1637	.1528	.1512	**.1307**	.1699	.1649
TC	**.4220**	.4270	.4240	.4180	.4210	.4240	.4180	**.4170**	.4210	.4140	.4210	.4240	.4150	**.4130**	.4270
CCu	**.6002**	.6025	.6021	.6028	.6011	.6019	**.5992**	.6021	.6035	.6031	.5997	**.5997**	.6019	.6011	.6077
KS	.4264	**.4262**	.4262	.4262	.4266	.4262	.4266	**.4260**	.4270	.4279	.4271	.4270	.4265	.4258	**.4257**
BCu	.5899	**.5885**	.5892	.5886	.5928	.5898	**.5878**	.5898	.5880	.5897	**.5862**	.5895	.5882	.5892	.5884
NDw	.6131	**.6105**	.6131	.6134	.6131	.6131	**.6106**	.6131	**.6094**	.6119	.6103	.6097	.6133	.6106	.6142
SPw	**.1990**	.2750	.2870	**.2870**	.2870	.2870	.2870	.2870	**.2870**	.2870	.3040	.3110	.2870	.2870	.4820
CCw	.5648	**.5598**	.5658	.5695	.5666	.5700	**.5655**	.5658	.5621	**.5595**	.5665	.5652	.5745	.5655	.5621
BCw	**.3246**	.3256	.3247	.3247	**.3245**	.3253	.3250	.3247	**.3222**	.3252	.3226	.3238	.3286	.3262	.3250

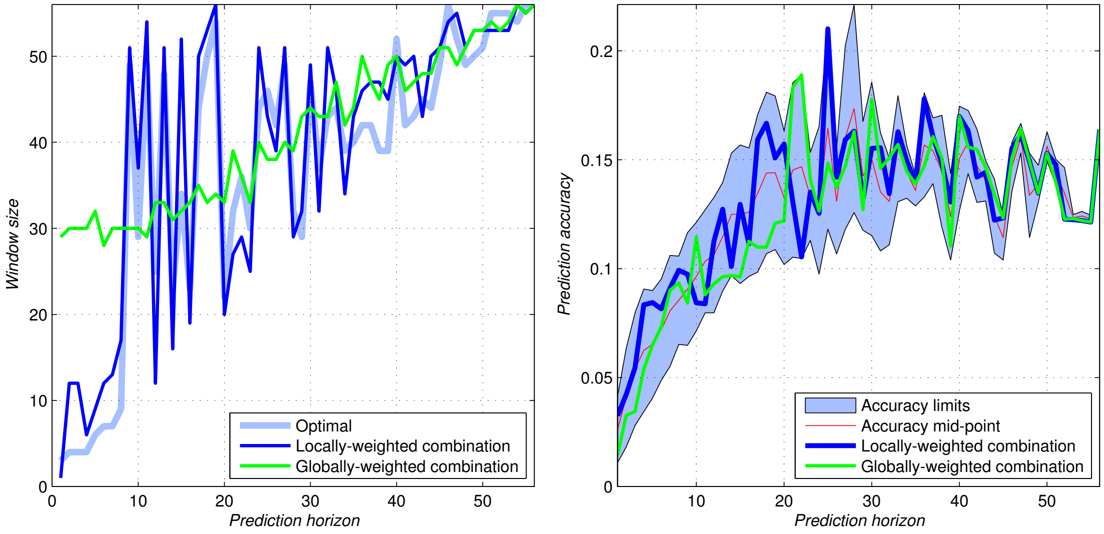

Fig. 7. TTM: optimal and WIM-recommended window sizes (left) and predictions (right)

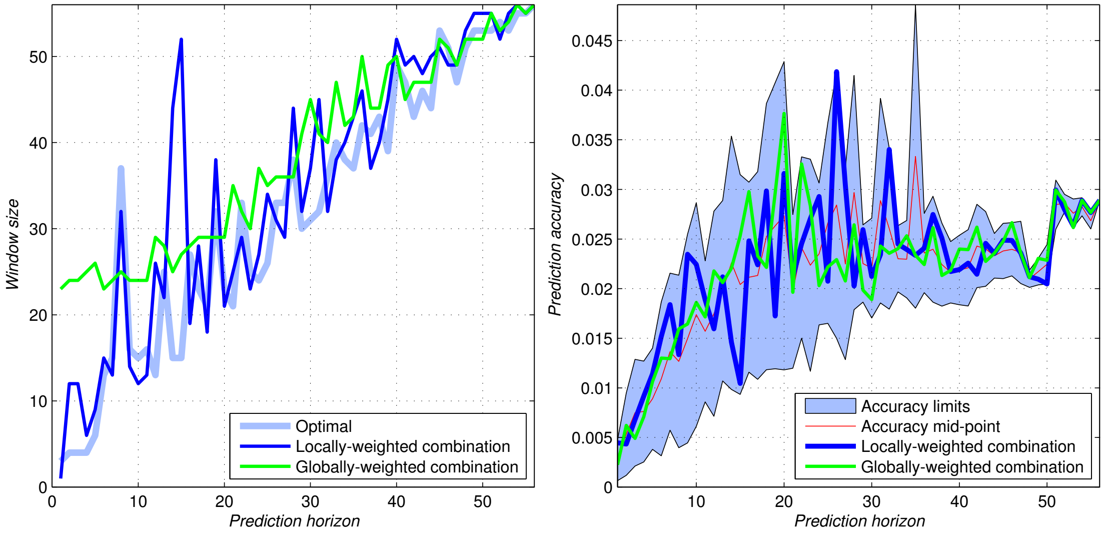

Fig. 8. PA: optimal and WIM-recommended window sizes (left) and predictions (right)

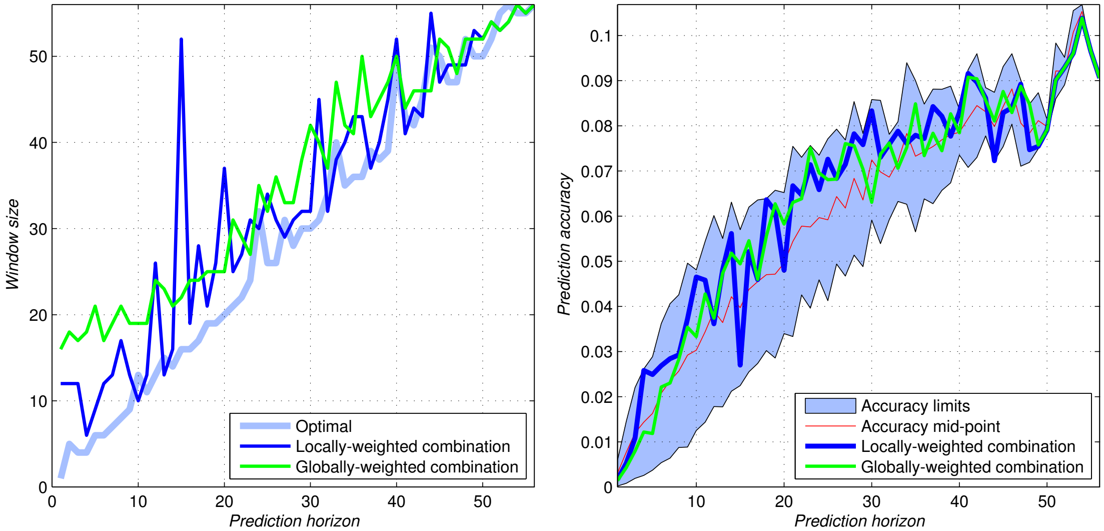

Fig. 9. CN: optimal and WIM-recommended window sizes (left) and predictions (right)

apparent in the case of the TTM predictor. As it can be seen, GWC tends to recommend windows which are bigger than optimal, especially for small horizons (between 1 and 10 days). This is a result of the averaging effect – the WIMs which are most suitable in this range of horizon sizes (i.e. *BCw* and *SPw* – see Figure 6) have relatively low global weights due to their underperformance for bigger values of H.

The effect of the recommendations on the prediction accuracy has been depicted in the right parts of Figures 7, 8 and 9, where the shaded area represents the boundaries of the accuracy obtained by taking best and worst predictions for each prediction horizon. In the case of TTM (Figure 7) LWC in general also outperforms GWC except for a small number of horizon sizes. The same trend can be observed in the case of PA and CN, where it is also visible that better recommendations correspond with better prediction accuracy. It is also worth noticing that in all the cases the prediction accuracy achieved due to using the recommended window size is not only considerably above the lower accuracy limit but is usually also above the average accuracy ('accuracy mid-point' – the center of the shaded region), often approaching the best achievable performance.

VI. Conclusions and Future Work

The presented study is the first step towards a systematic approach to the problem of meaningful, data-driven, window size selection for prediction of network evolution. The study showed that by minimizing the proposed Window Incoherence Measures, and hence effectively choosing window size in a way that the properties of a network within each window are as close as possible to the characteristics of the global network, the link prediction accuracy can be increased. This has been achieved irrespective of the actual predictive method used. The proposed approach can also be used for recommendation of the optimal size of the prediction horizon, if the actual predictive applications allows for such flexibility.

The encouraging results of this work have allowed us to identify many interesting and challenging open issues, which will be the subject of our future research. We intend to further validate the proposed method on other, larger datasets, where alternative definitions and types of relation could be employed (e.g. relations based on the contents of communication using a text-mining approach). We also aim at extending this research by investigating additional Window Incoherence Measures, based on other properties of the networks, as well as by employing multi-objective optimization techniques, where a carefully chosen subset of the proposed measures would be optimized concurrently. As mentioned before, we are also going to explore the usability of variable–sized windows for better prediction accuracy, which to the best of our knowledge, will be a highly innovative approach to this problem.

Acknowledgments

The research leading to these results has received funding from the European Union 7^{th} Framework Programme (FP7/2007-2013) under grant agreement no. 251617.

The research presented in this paper has been partially supported by the European Union within the European Regional Development Fund program no. POIG.01.03.01-00-008/08.

References

[1] R. Albert and A.-L. Barabasi. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74:47–97, 2002.

[2] R. Albert, H. Jeong, and A. L. Barabási. The diameter of the world wide web. *Nature*, 401:130–131, 1999.

[3] R. Albert, H. Jeong, and A.-L. Barabási. Error and attack tolerance of complex networks. *Nature*, 406(6794):378–382, 2000.

[4] A. Barabási. *Bursts: The Hidden Pattern Behind Everything We Do*. Dutton, 2010.

[5] A.-L. Barabasi. *Linked: How Everything Is Connected to Everything Else and What It Means*. Plume, reissue edition, April 2003.

[6] A. Barrat, M. Barthelemy, and A. Vespignani. *Dynamical processes on complex networks*. Cambridge University Press, 2008.

[7] V. Batagelj and A. Mrvar. A subquadratic triad census algorithm for large sparse networks with small maximum degree. *Social networks*, 23(3):237–243, 2001.

[8] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. Hwang. Complex networks: Structure and dynamics. *Physics Reports*, 424(4-5):175–308, February 2006.

[9] B. Bollobas. *Random Graphs*. Academic, London, UK, 1985.

[10] B. Bringmann, M. Berlingero, F. Bonch, and A. Gionis. Learning and predicting the evolution of social networks. *IEEE Intelligent Systems*, 25(4):26–35, 2010.

[11] M. Budka, B. Gabrys, and K. Musial. On accuracy of PDF divergence estimators and their applicability to representative data sampling. *Entropy*, 13(7):1229–1266, 2011.

[12] D. Davis, R. Lichtenwalter, and N. Chawla. Multi-relational link prediction in heterogeneous information networks. In *2011 International Conference on Advances in Social Networks Analysis and Mining*, pages 281–288. IEEE, 2011.

[13] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. In *SIGCOMM '99: Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication*, volume 29, pages 251–262, New York, NY, USA, October 1999. ACM.

[14] B. A. Huberman and L. A. Adamic. Growth dynamics of the World-Wide Web. *Nature*, 401(6749):131, 1999.

[15] D. Johnson. Efficient algorithms for shortest paths in sparse networks. *Journal of the ACM (JACM)*, 24(1):1–13, 1977.

[16] K. Juszczyszyn, M. Budka, and K. Musial. The dynamic structural patterns of social networks based on triad transitions. In *Advances in Social Networks Analysis and Mining (ASONAM), 2011 International Conference on*, pages 581–586, July 2011.

[17] K. Juszczyszyn, K. Musial, and M. Budka. Link prediction based on subgraph evolution in dynamic social networks. In *Privacy, Security, Risk and Trust (PASSAT), 2011 IEEE Third International Conference on and 2011 IEEE Third International Confernece on Social Computing (SocialCom)*, pages 27–34. IEEE, 2011.

[18] K. Juszczyszyn, K. Musial, and M. Budka. On analysis of complex network dynamics changes in local topology. In *The fifth SNAKDD Workshop 2011 on Social Network Mining and Analysis held in conjunction with SIGKDD conference*, pages 61–70, 2011.

[19] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[20] S. Kullback and R. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[21] J. Leskovec, L. Backstrom, R. Kumar, and A. Tomkins. Microscopic evolution of social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 462–470, New York, NY, USA, 2008. ACM.

[22] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology*, 58(7):1019–1031, 2007.

[23] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031, 2007.

[24] J. Lin. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 1991.

[25] G. Linoff and M. Berry. *Data mining techniques: for marketing, sales, and customer relationship management*. Wiley, 2011.

[26] K. Musial, M. Budka, and K. Juszczyszyn. Creation and Growth of Online Social Network - How do social networks evolve? *World Wide Web Journal*, DOI: 10.1007/s11280-012-0177-1, 2012.

[27] M. Newman. Clustering and preferential attachment in growing networks. *Physical Review E*, 64(2):025102, 2001.

[28] M. Newman. *Networks: An Introduction*. Oxford University Press, 2010.

[29] E. P. and R. A. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5:17–60, 1960.

[30] S. Redner. How popular is your paper? an empirical study of the citation distribution. *Eur. Phys. J.*, B(4):131–138, April 1998.

[31] S. H. Strogatz. Exploring complex networks. *Nature*, 410(6825):268–276, March 2001. ch: Complex Networks.

[32] R. Sulo, T. Berger-Wolf, and R. Grossman. Meaningful selection of temporal resolution for dynamic networks. In *Proceedings of the Eighth Workshop on Mining and Learning with Graphs*, pages 127–136. ACM, 2010.

[33] D. Watts and S. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–444, 1998.