PLOS | ONE

RESEARCH ARTICLE

PAFit: A Statistical Method for Measuring Preferential Attachment in Temporal Complex Networks

Thong Pham[1]*, **Paul Sheridan**[2], **Hidetoshi Shimodaira**[1]

1 Division of Mathematical Science, Graduate School of Engineering Science, Osaka University, Osaka, Japan, **2** The Institute of Medical Science, The University of Tokyo, Tokyo, Japan

* thongpham@sigmath.es.osaka-u.ac.jp

OPEN ACCESS

Citation: Pham T, Sheridan P, Shimodaira H (2015) PAFit: A Statistical Method for Measuring Preferential Attachment in Temporal Complex Networks. PLoS ONE 10(9): e0137796. doi:10.1371/journal.pone.0137796

Editor: Zhong-Ke Gao, Tianjin University, CHINA

Received: April 1, 2015

Accepted: August 21, 2015

Published: September 17, 2015

Data Availability Statement: The Flickr dataset is publicly available at the Koblenz Network Collection: http://konect.uni-koblenz.de/networks/flickr-growth.

Funding: This work was supported by grants from Japan Society for the Promotion of Science KAKENHI (26120523 and 24300106 to HS, URL: http://www.jsps.go.jp/english/index.html). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

Abstract

Preferential attachment is a stochastic process that has been proposed to explain certain topological features characteristic of complex networks from diverse domains. The systematic investigation of preferential attachment is an important area of research in network science, not only for the theoretical matter of verifying whether this hypothesized process is operative in real-world networks, but also for the practical insights that follow from knowledge of its functional form. Here we describe a maximum likelihood based estimation method for the measurement of preferential attachment in temporal complex networks. We call the method PAFit, and implement it in an R package of the same name. PAFit constitutes an advance over previous methods primarily because we based it on a nonparametric statistical framework that enables attachment kernel estimation free of any assumptions about its functional form. We show this results in PAFit outperforming the popular methods of Jeong and Newman in Monte Carlo simulations. What is more, we found that the application of PAFit to a publically available Flickr social network dataset yielded clear evidence for a deviation of the attachment kernel from the popularly assumed log-linear form. Independent of our main work, we provide a correction to a consequential error in Newman's original method which had evidently gone unnoticed since its publication over a decade ago.

Introduction

The *de facto* maxim of network science is 'all systems are networks.' For those who are well-acquainted with this network view of reality, recognizing the network nature of virtually any real-world system borders on reflexive. But network scientists are not interested in studying just any old networks. Network scientists habitually confine themselves to the study of complex networks, that is, large-scale networks with emergent topological features that are not found to occur in simple networks [1]. Complex network research turns more or less on the study of two related problems: first, the modelling of dynamical processes taking place on static complex networks in a manner that makes judicious use of known topological features [2], and second, the study of how topological features emerge in temporal complex networks and subsequently evolve over time [3]. It is the latter problem that concerns us in this paper. To be plain: we focus on understanding the extent to which a process know as preferential attachment (PA) explains the emergence of those heavy-tailed degree distributions, best exemplified by power-laws, that are commonly observed in temporal complex networks across nature, society, and technology [3].

It is reasonable to describe a wide variety of real-world systems as temporal (complex) networks. In formal terms, a temporal network represents an evolving system as a sequence of static (complex) networks. This sequence, which we denote by $G_0, G_1, \ldots, G_T$, is best envisioned as a progression of snapshots of a given system taken at discrete time-steps, $t = 0, 1, \ldots, T$. The model at this level of generality is purely descriptive and serves only to frame our thinking about complex systems. Raising it up to the status of an explanatory model with predictive power requires the setting forth of laws governing node generation and the formation of node-to-node connections in the transition from G_{t-1} to G_t. This, however, is a subject to which we must return after a brief digression on some important empirical findings.

By the late 1990s, temporal networks in diverse domains of learning were observed to enjoy some universal topological features [4]. The most publicized of these features at the time was the so-called scale-free property [5]. A network, it will be noted, is scale-free when its degree distribution follows a particular kind of heavy-tailed function called a power-law [6]. The purported universality of scale-free networks was soon, however, called into question when it was pointed out that distinguishing a power-law degree distribution from alternative heavy-tailed functional forms, such as the log-normal distribution, is a matter of considerable subtlety [5–8]. In light of this consideration, we will proceed on the supposition that the temporal networks in the world around us are commonly found to have heavy-tailed degree distributions with those following power-laws constituting an interesting special case.

What was interpreted as mounting empirical evidence for scale-free networks had everybody in network science scurrying to propose explanatory temporal network models that account for them in terms of domain independent processes. The first and most influential of these is known as the Barabási-Albert (BA) model [4]. It is essentially a toy model that generates scale-free networks by appealing to a simple growth law for node generation and a special case of PA to govern the formation of node-to-node connections.

Consider a generalization of the BA model that allows for networks with a wider range of heavy-tailed degree distributions [9]. Let G_t denote a static network at time-step $t = 0, 1, \ldots, T$. Starting from a seed network G_0, for each time-step $1 \leq t < T$, the network G_{t-1} is 'grown' by a new node v' that is subsequently connected to an already existing node to form G_t. Note that, in order to review the precise results about the degree distribution, here we only consider the case in which one node is added at each time-step. Later in the Materials and Methods section, we will describe a model that more closely reflects real-world networks, in which multiple new nodes can appeared at the same time. The PA rule then states that v' connects to a node v_k of degree k in G_{t-1} with probability proportional to A_k:

$$Pr(v' \text{ connects to } v_k) \propto A_k. \tag{1}$$

The function A_k is called the *attachment kernel*. Strictly speaking, PA is only said to occur when A_k is an increasing function on average. It has been shown that the functional form of the attachment kernel can have dramatic effects on temporal network topology. Let us consider the log-linear model $A_k = k^\alpha$ with *attachment exponent* $\alpha > 0$ as an important example. When $A_k = k$ ($\alpha = 1$) the resulting network is scale-free [9]. This is called *linear* PA and it is this form that it is assumed in the BA model. On the other hand, *sub-linear* PA ($0 < \alpha < 1$) gives rise to a class of stretch-exponential degree distributions, while *super-linear* PA ($\alpha > 1$) leads to a winner-take-all phenomena whereby a handful of nodes acquire all the connections [9]. The limiting case $A_k = 1$ for all k ($\alpha = 0$) corresponds to a version of the classical Erdös-Rényi random network model [10]. In this case, the network is also not scale-free.

The point is that the Barabási-Albert family of models explain the emergence of a robust class of heavy-tailed degree distributions in temporal networks in terms of two generic processes: growth and PA. It is, moreover, plausible that these processes operate in many real-world temporal networks. In expanding social networks, for example, it is reasonable to presume that a person with many acquaintances already will tend to make new ones at a faster rate, than a person with few. Or, in citation networks of research papers, to take another example, a paper with many citations may acquire new citations more readily, than a comparatively lesser known paper. The same argument is easily specialized to other sorts of temporal networks by reasoning along similar lines.

Let us suppose for the sake of argument that heavy-tailed degree distributions of this kind prevail in growing temporal networks in the real world. This would not prove that PA was responsible for their generation, because it is always possible that a different process was at work. Rather, the presence of PA in the generative process must be confirmed by observation. In practice, this amounts to estimating the attachment kernel, A_k, from observed data.

The problem of estimating A_k from observed data is one of great importance to the working network scientist. For one thing, there are practical applications in link prediction algorithms [11], and more generally, it gives us valuable insights into the global characteristics of networks [9]. But the core questions surrounding the PA process are these: does PA actually exist in real-world temporal networks; or, in other words, is A_k really increasing in k? If so, then what elementary functional form, if any, does it take? Is it the widely accepted log-linear model $A_k = k^\alpha$, or something else? In order to investigate these questions scientifically, a rigorous statistical method to estimate the functional form of A_k with confidence bounds is necessary. However, to the best of our knowledge, no such method has yet been advanced in the literature.

Nevertheless, detecting the hand of PA in temporal networks has attracted the attention of many researchers, and a number of estimation methods for A_k have been proposed [12–17]. Table 1 shows a summary of the existing methods. The primary drawback of most of these methods is that they explicitly assume the log-linear form $A_k = k^\alpha$ and focus only on estimating the attachment exponent α [14–17]. Massen et al. [14] used a fixed point iterative algorithm for the purpose, Sheridan et al. [15] a Markov chain Monte Carlo method, Gomez et al. [16] a maximum likelihood (ML) approach to estimate the value of α. Lastly, Kunegis et al. [17]

Table 1. Summary of the existing attachment kernel estimation methods.

Method	Form of A_k	Estimation method
Newman [12]	Nonparametric	Weighted sum of multiple histograms
Jeong et al. [13]	Nonparametric	Single histogram
Massen et al. [14]	$A_k = k^\alpha$	Iterative fixed-point algorithm
Sheridan et al. [15]	$A_k = k^\alpha$	Markov chain Monte Carlo
Gomez et al. [16]	$A_k = k^\alpha$	ML by grid search
Kunegis et al. [17]	$A_k = k^\alpha$	Regression

Summary of the existing methods for estimating the attachment kernel A_k. Nonparametric methods are methods that do not assume a functional form for A_k.

doi:10.1371/journal.pone.0137796.t001

estimated α by fitting a function that relates the number of new connections a node obtains in the future with its current degree.

The remaining methods [12, 13] do not assume any functional form for A_k. As invaluable as these methods are, they are ad-hoc methods that are not without their problems. Newman's method [12] is among the first methods proposed to estimate A_k. In real-world examples, Newman's method appears to be able to estimate A_k well for small k, but systematically underestimates A_k when k is large [18, 19]. This is thought to be an artifact of the method [19]. We will explain the reason for this artifact in detail below. On the other hand, while Jeong's method [13] is the most widely-used method in practice [19–31], as we will explain in the Materials and Methods section, it suffers from a kind of bias-variance trade-off.

Our contributions in this paper are twofold.

First and foremost, we propose a nonparametric estimation method called PAFit that employs the ML approach of [32] to fit the functional form of the attachment kernel. PAFit is nonparametric in the sense that it does not make any assumptions on the functional form of A_k. The method works instead by estimating the value of A_k separately for each k. The algorithm underlying PAFit is the Minorize-Maximization [33, 34] (MM) algorithm. We prove that our algorithm converges to a global maximizer of the log-likelihood function and provide a fast approximation to the confidence intervals of the estimated attachment kernel $\hat{A}_k$. This last step is essential for the analysis of large datasets. What is more, we present the results of extensive Monte Carlo simulations that demonstrates how PAFit outperforms the state-of-the-art methods, namely Jeong's method and Newman's method, at the task of estimating A_k for a variety of different functional forms. Lastly, we used PAFit to analyze a publicly available Flickr social network dataset [35], and found that the estimated attachment kernel differed considerably from the functional form $A_k = k^\alpha$. This result immediately suggests that it is important to look beyond the classical log-linear form when modelling the attachment kernel.

A second contribution of ours that should not be overlooked is an explanation of the artifact observed in the Newman's method. In particular, we show that this artifact can be completely eliminated by a simple mathematical correction. This is important because researchers have been using Newman's method to estimate A_k in real-world networks for the better part of 15 years. It will be noted that while PAFit outperformed the corrected version of Newman's method in the simulated examples, the corrected version was found to work better than uncorrected version.

In summary, PAFit is a statistically sound method for estimation of the attachment kernel in temporal complex networks. We have demonstrated its potential in this paper to uncover interesting findings about real-world complex networks. This method is implemented in the `R` package `PAFit` [36].

Before entering into the details of PAFit in the next sections, we note some recent developments in the field of complex networks. In this paper, we estimate the growth mechanism, in particular the PA mechanism, of a network given its growth process, that is, the observed data are the final network and all previous networks in the growth process. There is another line of research that considers a somewhat opposite problem: the underlying network is unknown and the task is to estimate it. This line of research includes, for example, estimating the unknown gene regulatory network from observed omics data [37, 38]. Complex networks have also found applications in many other areas, including economics [39], image analysis [40] and epidemiology [2, 3, 41, 42]. Beside the scale-free property, which only concerns the degree distribution of a network, other topological features which concern higher order of information have also been studied extensively in the literature [43–46].

Materials and Methods

In this section, we begin by introducing the temporal network model underlying the PAFit method. We follow that with an illustrative example to demonstrate the workings of PAFit in comparison with previous methods. From there we explain the mathematical details of PAFit and call attention to its important properties. The similarities between PAFit and previous methods are discussed. It is there that the correction to Newman's method is outlined. Next, fast calculations for the confidence intervals of the estimated values are explained. We then proceed to discuss binning and regularization, which are two important techniques for stabilizing the PAFit estimation. Finally, we mention how to estimate the attachment exponent α of the log-linear model $A_k = k^\alpha$ from $\hat{A}_k$.

The general temporal model

The statistical estimation method we present in this paper is tailored for the following temporal model for directed networks. Note, however, that our method can be easily adapted to work for undirected networks. The model is a simplified version of a model that we introduced in a previous publication [32]. Starting from a seed network at time-step $t_0 = 0$, we grow the network by adding $n(t)$ nodes and $m(t)$ edges at every time-step t, for $t = 1, \ldots, T$. Note that our method allows $m(t)$ to be consisted of both new edges that emanate from the $n(t)$ new nodes and emergent edges between existing nodes. This is important since a large number of real-world networks have emergent edges between existing nodes. At time-step t, the probability that an existing node v with in-degree k acquires a new edge is

$$Pr(v \text{ acquires a new edge}) \propto A_k, \tag{2}$$

where A_k is the value of the attachment kernel at degree k. We call this model the general temporal (GT) model. The GT model includes a number of important network models as special cases. When $A_k = k^\alpha$, the GT model corresponds to Price's model [47, 48] or Barabási-Albert (BA) model [4] in the undirected case. Furthermore, when $A_k = 1$ for all k, then it reduces to the classical Erdös-Rényi random network model [10].

Here we note a quick remark on an assumption about the joint probability distribution of $n(t)$ and $m(t)$. We will return to this assumption later when we discuss the MLE for A_k. Let $\boldsymbol{\theta}(t)$ denote the parameter vector that governs the joint distribution of $n(t)$ and $m(t)$. We assume that $\boldsymbol{\theta}(t)$ does not involve A_k. This very mild assumption allows broad and realistic models for $m(t)$ and $n(t)$. For example, $m(t)$ and $n(t)$ can be random variables whose means and variances depend on G_{t-1}.

A final remark is that, although we do not incorporate deletions of nodes and edges into the formal specification of the GT model, this is purely for simplicity. The PAFit method is able to estimate A_k even when there are deletions of nodes and edges, as long as the deletion mechanism is independent of the addition mechanism and the parameters govern the deletion mechanism do not involve A_k, which are reasonable assumptions. The current R implementation of PAFit [36] can be easily adapted to work in these cases.

An illustrative example

In this section we present a simulated example of a GT model network. The purpose being to demonstrate the workings of our proposed method, PAFit. The network generative process follows the GT model with $A_k = 3(\log(\max(k, 1)))^2 + 1$. Starting from a seed network with 20 nodes, at each time-step, $m(t) = 5$ new edges and $n(t) = 1$ new node were added, until a total of $T = 2000$ nodes is reached. We measure how well a method performs by using the average

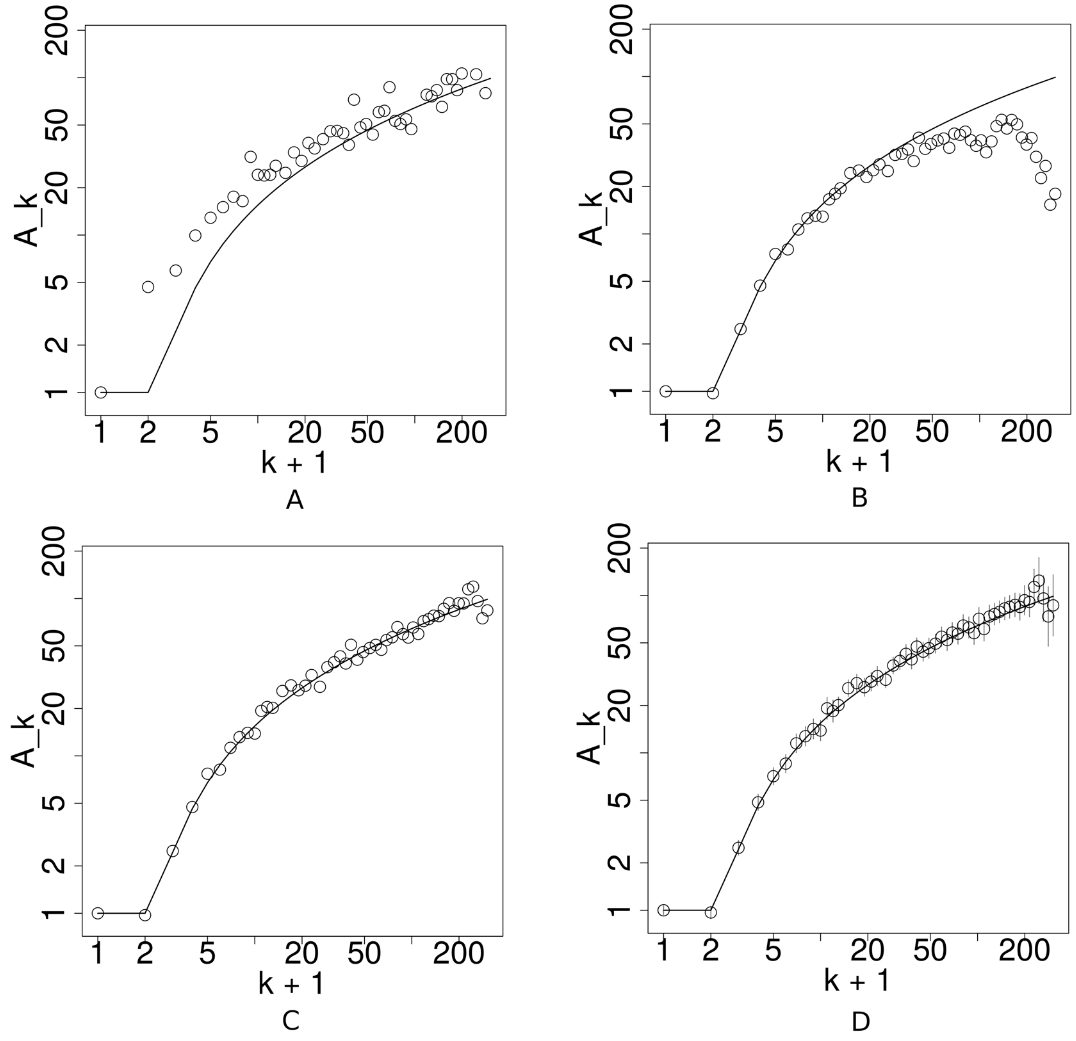

Fig 1. Estimation of the attachment kernel when the true model is $A_k = 3(\log(\max(k, 1)))^2 + 1$. A: Jeong's method. B: Newman's method. C: Corrected Newman's method. D: PAFit. The solid line depicts the true model. The plots are on a log-log scale. The gray vertical lines are the estimated confidence intervals of the estimated values by PAFit. Confidence intervals are not available in the remaining methods.

doi:10.1371/journal.pone.0137796.g001

relative error between the true A_k and the estimated $\hat{A}_k$, defined as $e_A = \frac{1}{K}\sum_{k=1}^{K} \frac{(A_k - \hat{A}_k)^2}{A_k^2}$ where K is the maximum degree that appears in the growth process.

We first apply Jeong's method. Here we choose the time window between when the 1500-th and 2000-th nodes are added. After a suitable normalization, the histogram of the degrees of nodes to which new edges appeared in this window connect will give us $\hat{A}_k$. The average relative error e_A for Jeong's method is 0.337, which is the highest among four methods considered in this example. From Fig 1A, one can see that Jeong's method did capture the shape of A_k, but the estimated function was sparse and fluctuated considerably. These are inherent drawbacks of the method that follow from using only a small time window to estimate A_k.

Second, we apply Newman's method. His method can be interpreted as estimating the attachment kernel by a weighted sum of multiple histograms created at multiple time-steps. In Fig 1B, $\hat{A}_k$ follows the true function very closely, but the value of A_k starts to fall off when the degree k becomes large. This phenomenon can also be seen in other papers [18, 19], and is thought to be an artifact of the method [19]. The value of e_A for Newman's method is 0.168, which is the second highest error among four methods used in this example. With a simple correction, we can eliminate this artifact. As can be seen from Fig 1C, the corrected Newman's method estimated the attachment kernel well the entire range of k. The value of e_A for this corrected version is 0.016, which is much smaller than those of Jeong's method and the original method of Newman.

Unlike all the other methods, PAFit estimates the attachment kernel by maximum likelihood estimation (MLE), and can give quantifications of the uncertainties in the value of $\hat{A}_k$ in the form of confidence intervals. In Fig 1D, the estimated attachment kernel follows the true function comparatively well, even in the high degree region. The average relative error e_A of PAFit is 0.015, which is the smallest error among four method considered here.

Maximum likelihood estimation

Here we derive the MLE for the GT model as described in the previous section. Recall that K is the maximum degree that appears in the growth process. Let $\mathbf{A} = [A_0, A_1, \ldots, A_K]$ be the parameter vector we want to estimate. The likelihood of the data at time-step $t \geq 1$ is the product of $P(m(t), n(t)|G_{t-1}, \boldsymbol{\theta}(t))$ and $P(G_t|G_{t-1}, m(t), n(t), \mathbf{A})$. The likelihood of the data at the starting time $t = 0$ is the product of $P(m(0), n(0)|\boldsymbol{\theta}(0))$ and $P(G_0|m(0), n(0), \boldsymbol{\theta}_*)$, with $\boldsymbol{\theta}_*$ the parameter vector governing the distribution of G_0. Thus the log-likelihood function of the dataset is

$$\begin{aligned} l(\mathbf{A}) \quad = \quad & \sum_{t=1}^{T} \log P(G_t|G_{t-1}, m(t), n(t), \mathbf{A}) + \sum_{t=1}^{T} \log P(m(t), n(t)|G_{t-1}, \boldsymbol{\theta}(t)) + \\ & \log P(m(0), n(0)|\boldsymbol{\theta}(0)) + \log P(G_0|m(0), n(0), \boldsymbol{\theta}_*). \end{aligned} \tag{3}$$

As previously mentioned, we assume that $\boldsymbol{\theta}(t)$, the parameter vector governing the joint distribution of $m(t)$ and $n(t)$, does not involve $\mathbf{A}$. We also assume that $\boldsymbol{\theta}*$ does not involve $\mathbf{A}$. When deriving the MLE of $\mathbf{A}$, these two assumptions allow us to ignore all but the first term in the right hand side of Eq (3). At the onset of time-step t, denote $m_k(t)$ and $n_k(t)$ as the number of new edges that connect to a node with degree k and the number of existing nodes with degree k, respectively. The term $P(G_t|G_{t-1}, m(t), n(t), \mathbf{A})$ corresponds to a multinomial distribution. This follows from the observation that given $m(t)$, the quantities $m_0(t), m_1(t), \ldots, m_K(t)$ follow a multinomial distribution with parameters $p_0(t), p_1(t), \ldots, p_K(t)$, where $p_k(t)$, the probability that a newly added edge at time t connects to a node with degree k, is

$$p_k(t) = \frac{n_k(t)A_k}{\sum_{j=0}^{K} n_j(t)A_j}. \tag{4}$$

After dropping constant terms, we can write down the log-likelihood function in detail:

$$l(\mathbf{A}) = \sum_{t=1}^{T}\sum_{k=0}^{K} m_k(t) \log A_k - \sum_{t=1}^{T} m(t) \log \left(\sum_{j=0}^{K} n_j(t)A_j \right). \tag{5}$$

Note that the A_k are only identifiable up to a multiplicative constant, as can be seen from

Eq (4). We can enforce uniqueness by fixing $A_k = 1$ for some k that satisfies $\sum_{t=1}^{T} m_k(t) > 0$. In practice, we can often set, for example, $A_1 = 1$.

We find the MLE by solving the following likelihood equation.

$$\partial l(\mathbf{A})/\partial \mathbf{A} = 0 \Longleftrightarrow A_k = \frac{\sum_{t=1}^{T} m_k(t)}{\sum_{t=1}^{T} \frac{m(t) n_k(t)}{\sum_{j=0}^{K} n_j(t) A_j}} \qquad k = 1, \ldots, K. \tag{6}$$

But before that we have the following observation.

Proposition 1 *Any* $\mathbf{A}'$ *that satisfies* $\partial l(\mathbf{A})/\partial \mathbf{A} = 0$ *is a global maximizer of* $l(\mathbf{A})$.

The implication of this proposition is that even though the log-likelihood function $l(\mathbf{A})$ might be not concave, any local maximizer is guaranteed to be a global maximizer.

Proof. We consider the one-to-one re-parametrization $\boldsymbol{\beta} = \boldsymbol{\beta}(\mathbf{A}) = [\log A_0, \ldots, \log A_K]^T$. From Eq (5) and the inverse relation $A_k = \exp \beta_k$, the log-likelihood function under this new parametrization is

$$l_{\boldsymbol{\beta}}(\boldsymbol{\beta}) = \sum_{t=1}^{T}\sum_{k=0}^{K} m_k(t)\beta_k - \sum_{t=1}^{T} m(t) \log\left(\sum_{j=0}^{K} n_j(t) \exp \beta_j\right).$$

The second derivative of the log-sum-exponential term $\log\left(\sum_{j=0}^{K} n_j(t) \exp\beta_j\right)$ can be shown to be non-negative semi-definite (a similar calculation can be found on page 74 of [49]). Therefore the second derivative $\Delta^2\, l_{\boldsymbol{\beta}}(\boldsymbol{\beta})$ of $l_{\boldsymbol{\beta}}(\boldsymbol{\beta})$ is non-positive semi-definite. Thus $l_{\boldsymbol{\beta}}(\boldsymbol{\beta})$ is concave. Now the likelihood equation of the re-parametrized log-likelihood function $l_{\boldsymbol{\beta}}$ is

$$\partial l_{\boldsymbol{\beta}}/\partial \boldsymbol{\beta} = 0 \Longleftrightarrow \exp \beta_k = \frac{\sum_{t=1}^{T} m_k(t)}{\sum_{t=1}^{T} \frac{m(t) n_k(t)}{\sum_{j=0}^{K} n_j(t) \exp \beta_j}} \qquad k = 1, \ldots, K. \tag{7}$$

One can see that any $\mathbf{A}'$ satisfying Eq (6) will give a reparametrization $\boldsymbol{\beta}'$ that satisfies Eq (7). This $\boldsymbol{\beta}'$ is a global maximizer since $l_{\boldsymbol{\beta}}(\boldsymbol{\beta})$ is concave. Thus the $\mathbf{A}'$ is also a global maximizer.

Turning now to how to solve for A_k, since A_k appears in both sides of Eq (6), an explicit solution for $\mathbf{A}$ is difficult to obtain. Define $\mathbf{A}^{(i)} = \left[A_1^{(i)}, \ldots, A_K^{(i)}\right]$ as the estimated parameter vector at iteration i. Starting from some initial value $\mathbf{A}^{(0)}$, at iteration $i \geq 1$ we can iteratively update $A_k^{(i)}$ using Eq (6):

$$A_k^{(i)} = \frac{\sum_{t=1}^{T} m_k(t)}{\sum_{t=1}^{T} \frac{m(t) n_k(t)}{\sum_{j=0}^{K} n_j(t) A_j^{(i-1)}}} \qquad k = 1, \ldots, K. \tag{8}$$

We repeat Eq (8) until the following convergence condition is met.

$$\frac{|l(\mathbf{A}^{(i)}) - l(\mathbf{A}^{(i-1)})|}{|l(\mathbf{A}^{(i-1)})| + 1} \leq \epsilon. \tag{9}$$

This iterative algorithm turns out to be an instance of Minorize-Maximization (MM) [33, 34] algorithms, which are used in maximization problems. In MM algorithms, instead of maximizing a complicated objective function, at each iteration we find and maximize a minorize function of the objective function. A minorize function is a function that is equal to the objective function at the current point, and is the lower bound of the objective function at all other points. The minorize function is often chosen so that we can maximize it easier than the original objective function. For our problem, we have the following proposition:

Proposition 2 *Using the aforementioned algorithm, the log-likelihood function is guaranteed to increase with number of iterations. In particular,*

$$l(\mathbf{A}^{(i)}) \leq l(\mathbf{A}^{(i+1)}).$$

Furthermore, the stopping point $\mathbf{A}^{(M)}$ *which satisfies* $\mathbf{A}^{(M)} = \mathbf{A}^{(M-1)}$ *is a global maximizer of* $l(\mathbf{A})$.

Proof. We find a minorize function $Q_i(\mathbf{A})$, which is a function that satisfies $l(\mathbf{A}) \geq Q_i(\mathbf{A})$ for all $\mathbf{A}$, and $l(\mathbf{A}^{(i)}) = Q_i(\mathbf{A}^{(i)})$. Define

$$\begin{aligned} Q_i(\mathbf{A}) \quad = \quad & \sum_{t=1}^{T}\sum_{k=0}^{K} m_k(t)\log A_k - \sum_{t=1}^{T} m(t)\log\left(\sum_{j=0}^{K} n_j(t)A_j^{(i)}\right) - \\ & \sum_{t=1}^{T} m(t)\frac{\sum_{j=0}^{K} n_j(t)A_j}{\sum_{j=0}^{K} n_j(t)A_j^{(i)}} + \sum_{t=1}^{T} m(t). \end{aligned} \tag{10}$$

Applying the inequality $-\log x \geq -\log y - x/y + 1, \forall x, y > 0$ with $x = \sum_{j=0}^{K} n_j(t)A_j$ and $y = \sum_{j=0}^{K} n_j(t)A_j^{(i)}$, one can verify that $l(\mathbf{A}) \geq Q_i(\mathbf{A})$. By substitution, one can check that $l(\mathbf{A}^{(i)}) = Q_i(\mathbf{A}^{(i)})$. It is also easy to verify that the $\mathbf{A}^{(i+1)}$ computed by Eq (8) is the solution of $\partial Q_i/\partial \mathbf{A} = 0$, and thus satisfies $\mathbf{A}^{(i+1)} = \text{argmax } Q_i(\mathbf{A})$. Therefore,

$$l(\mathbf{A}^{(i)}) = Q_i(\mathbf{A}^{(i)}) \leq Q_i(\mathbf{A}^{(i+1)}) \leq l(\mathbf{A}^{(i+1)}).$$

Suppose that the algorithm stops at iteration M such that $\mathbf{A}^{(M)} = \mathbf{A}^{(M-1)}$. From Eqs (8) and (6), we can see that $\mathbf{A}^{(M)}$ satisfies Eq (6). Combining this with Proposition 1, this means that $\mathbf{A}^{(M)}$ is a global maximizer $l(\mathbf{A})$.

It is instructive to look at Eq (8). If we only have one time-step t, then the MLE solution is $A_k = m_k(t)/n_k(t)$, which is an intuitive formula. When there are multiple time-steps, things become less obvious. In this case, the MLE solution for A_k sums up all the $m_k(t)$ then divides this number by a weighted sum of the $n_k(t)$. Looking at the weights, we notice that a weight is larger when the number of new edges $m(t)$ is large and the normalizing factor $\sum_{j=0}^{K} n_j(t)A_j$ is small. This can be interpreted as the MLE solution naturally emphasizing $n_k(t)$ of the time-step t that has a lot of information.

Related work

In this section, we discuss in detail the two most important methods for estimating the attachment kernel: Jeong's method and Newman's method. We then provide a simple mathematical correction to the Newman's method. Finally, we discuss how PAFit fits in the framework of these previous methods.

Jeong's method. Jeong's method [13] estimates A_k by building a histogram of the degrees of nodes to which new edges connect. They choose a time window T_0 and call all nodes in the network at that time T_0-nodes. Next they choose a time window $T_1 > T_0$ and a window length $\Delta T \ll T_1$, then call all nodes added in the interval $[T_1, T_1 + \Delta T]$ the T_1-nodes. For a degree k, they record the number of times a degree k T_0-node is linked to by a T_1-node, and then normalize this number by $n_k(T_0)$ to give $\hat{A}_k$. No guidelines are offered for choosing T_0, T_1 and ΔT.

This method very clearly suffers a type of bias-variance trade-off. In order to negate the effect of fixing the degrees of T_0-nodes at time T_0, one should use a very small window time ΔT. But if the number of new edges in ΔT is small, then the estimation will be unstable. We can increase the number of new edges by increasing the length of ΔT, but this will inevitably introduce a bias into the estimation.

One natural way to overcome this drawback is to use multiple histograms calculated at multiple time-steps, but then the question arises as to how to combine those histograms into a single estimation of A_k while the normalizing factor at each time-step depends on $\mathbf{A}$ (and thus unknown). Newman's method provides an answer to this question.

Newman's method. Instead of using only one histogram at a single time window, Newman's method can be interpreted as estimating A_k by a weighted sum of multiple histograms created at multiple time-steps. Let $N(t)$ denote the number of nodes in the network at the onset of time-step t. Consider creating a histogram at time-step t. Denote $A_k(t)$ as the value of A_k at time t. Newman's argument, written in the notation used here, is that $A_k(t)$ can be estimated as

$$A_k(t) = \frac{m_k(t)N(t)}{n_k(t)m(t)} \tag{11}$$

if $n_k(t) \neq 0$ or $A_k(t) = 0$ if $n_k(t) = 0$, using the following approximations [12].

$$\frac{m_k(t)}{m(t)} \approx p_k(t) \approx A_k(t)\frac{n_k(t)}{N(t)}. \tag{12}$$

The final result of the procedure proposed by Newman [12] is equivalent to the formula

$$A_k = \frac{1}{C}\sum_{t=1}^{T} w_k(t)A_k(t) \tag{13}$$

with the weight $w_k(t) = m(t)1_{n_k(t) \neq 0}$, using the convention $0/0 = 0$. Note that, in Newman's argument, the normalized constant C in Eq (13) is a constant that does not depend on k. In real-world examples, this procedure seems to work well for small k, but the value of A_k starts to fall off when k is large [18, 19] (cf. Fig 1B). Although this phenomenon is often thought to be an artifact of the method [19], an explanation of its cause has yet to be provided.

The corrected Newman's method. Surprisingly, we can completely eliminate the artifact in Newman's original method by a applying a simple mathematical correction (cf. Fig 1C). Immediately from Eq (13), one can recognize that the summation is incorrectly normalized. The constant C should depend on k. The correct normalization is as follows:

$$A_k = \frac{1}{\sum_t w_k(t)}\sum_{t=1}^{T} w_k(t)A_k(t). \tag{14}$$

Newman's original method (see Eq (13)) implicitly assumes that all the degrees already appear in the network from time $t = 0$. This is of course not generally true. In a typical network that grows from a small initial network, the small degrees will appear far sooner than the large degrees. In such cases, the sum $\sum_t m(t)1_{n_k(t) \neq 0}$, which is a weighted sum of the number of time-steps that contain nodes with degree k, is larger for small k and smaller for large k. Forgetting to normalize each A_k in Eq (13) by $\sum_t m(t)1_{n_k(t) \neq 0}$ will cause the waterfall artifact as observed in Fig 1B. In Eq (14), correctly normalizing each A_k by this number completely eliminates the artifact. We will prefer to Eq (14) as the *corrected Newman's method*.

Relation of PAFit to previous methods. Using almost the same route as in deriving Newman's method, one can deduce a similar equation to PAFit's MLE Eq (6). Starting from Eq (12), $m_k(t)$ is equal to $A_k(t)m(t)n_k(t)/N(t)$. Summing over all time-steps t, we see that

$\sum_{t=1}^{T} m_k(t)$ is equal to $\sum_{t=1}^{T} A_k(t)m(t)n_k(t)/N(t)$. Assuming $A_k(t) = A_k$ for all t, we have

$$A_k = \frac{\sum_{t=1}^{T} m_k(t)}{\sum_{t=1}^{T} \frac{m(t)n_k(t)}{N(t)}}. \tag{15}$$

The only difference between Eqs (15) and (6) is the normalizing factor at time t: $\sum_{j=0}^{K} n_j(t)A_j$ versus $N(t)$. This difference is the result of the approximation in Eq (12).

If the ratio $\left(\sum_{j=0}^{K} n_j(t)A_j\right)/N(t)$ is independent of t, then Eqs (15) and (6) are equivalent. Notice that $\sum_{j=0}^{K} n_j(t)A_j/N(t)$ is equal to $\sum_{j=0}^{K} A_j\left(n_j(t)/N(t)\right)$ where $n_j(t)/N(t)$ is the proportion of nodes with degree j at time t. Interestingly, this proportion has been shown to be independent of t when t is large in the case of asymptotically linear or sub-linear attachment kernels, that is, $A_k \sim k^{\alpha}$ with $\alpha \leq 1$ [9]. In such cases, Eqs (15) and (6) are asymptotically equivalent. The two equations are not asymptotically equivalent when A_k is super-linear, that is, $A_k \sim k^{\alpha}$ with $\alpha > 1$, since in this case the proportion $n_j(t)/N(t)$ depends on t [9].

Fast approximation of the confidence intervals

We provide a fast approximation for the confidence intervals of $\hat{A}_k$. Our approximation works quite well in practice. In many real-world datasets, our approximation can provide a huge gain in speed. In standard statistical theory, the confidence intervals of $\hat{\mathbf{A}}$ can be calculated as the diagonal entries of the inverse of $\mathbf{D}$, where $\mathbf{D} = -\partial^2 l(\mathbf{A})/\partial\mathbf{A}\partial\mathbf{A}$. Define the following matrices $\mathbf{B}$,$\mathbf{C}$,$\mathbf{U}$ as:

$$\mathbf{B} = \text{diag}\left(\frac{\sum_{t=1}^{T} m_1(t)}{\hat{A}_1^2}, \ldots, \frac{\sum_{t=1}^{T} m_K(t)}{\hat{A}_K^2}\right), \mathbf{U} = \begin{pmatrix} n_1(1) & \cdots & n_1(T) \\ \vdots & \ddots & \vdots \\ n_K(1) & \cdots & n_K(T) \end{pmatrix},$$

$$\mathbf{C} = \text{diag}\left(\frac{-m(1)}{\sum_{j=1}^{K} n_j(1)\hat{A}_j}, \ldots, \frac{-m(T)}{\sum_{j=1}^{K} n_j(T)\hat{A}_j}\right),$$

then we have

$$\mathbf{D} = \mathbf{B} + \mathbf{U}\mathbf{C}\mathbf{U}^T. \tag{16}$$

The form of Eq (16) makes it quite natural to calculate the inverse of $\mathbf{D}$ by the Woodbury formula [50]:

$$\mathbf{D}^{-1} = (\mathbf{B} + \mathbf{U}\mathbf{C}\mathbf{U}^T)^{-1} = \mathbf{B}^{-1} - \mathbf{B}^{-1}\mathbf{U}(\mathbf{C}^{-1} + \mathbf{U}^T\mathbf{B}^{-1}\mathbf{U})^{-1}\mathbf{U}^T\mathbf{B}^{-1}. \tag{17}$$

Note that $\mathbf{B}$ and $\mathbf{C}$ are both diagonal matrices. This makes inverting them trivial. A direct inversion of $\mathbf{D}$ will require $O(K^3)$ operations, while using Eq (17) will require $O(KT^2)$. We propose to approximate $\mathbf{D}$ by $\mathbf{B}$. This reduces the required number of operations to $O(K)$.

Estimation of the attachment exponent

Given $\hat{A}_k$ and their estimated variances v_k, if we assume that the attachment kernel is indeed $A_k = k^{\alpha}$, it is an important task to correctly estimate the attachment exponent α. Jeong's method proposes to fit $k^{\alpha+1}$ to the cumulative function $\Pi(k) = \sum_{j:j \leq k} A_j$ in order to estimate α.

To take advantage of the estimated variances given by PAFit, we propose to find α by weighted least squares method. The variance of $\log \hat{A}_k$ can be estimated by $var(\log A_k) \approx v_k/\hat{A}_k^2$. The uncertainties in $\log \hat{A}_k$ can be naturally incorporated into the fitting process by using weights that are inversely proportional to variances of $\log \hat{A}_k$. In particular, we minimize the following objective function with respect to α and d:

$$\sum_k \frac{1}{var(\log \hat{A}_k)} (\log \hat{A}_k - \alpha \log k - d)^2.$$

It is important to note that the idea of using estimated variances as weights in weighted least squares methods can also be applied when estimating parameters of models that are different from the model $A_k = k^\alpha$.

Binning

For more stable estimation of the attachment kernel, we use logarithmic binning in order to let the A_k borrow information from the ones that are near them. In logarithmic binning, the length of the i-th bin is c times the length of the $(i - 1)$-th bin. We freely choose the number of bins B, then c is determined from B and the maximum degree K. We set $A_k = \omega_i$ for all k in the i-th bin. Choosing B small helps with stabilizing the estimation result at the risk of loosing fine details in the attachment kernel. For the sake of simplicity we have been using A_k in our equations, but the readers should keep in mind that unless stated otherwise, we always use binning, so in fact A_k should be replaced by ω_i.

Regularization

After binning is performed, another important technique to reduce the variances of the estimated result is regularization. We add the following regularization term

$$-\lambda \frac{1}{\sum_{k=1}^{K-1} w_k} \sum_{k=1}^{K-1} w_k (\log A_{k+1} + \log A_{k-1} - 2 \log A_k)^2 \tag{18}$$

to the log-likelihood function $l(\mathbf{A})$ (see Eq (5)), and maximize the resulted function. This penalty term penalizes the second order differentiation of $\log A_k$, and by doing so, it encourages linearity in $\log A_k$.

The weight w_k can be any positive number. In this paper, we set w_k to be proportional to the number of data points of the degree k: $w_k \propto \sum_{t=1}^{T} m_k(t)$. Intuitively, we want to emphasize the regularization where there is plenty of data (when k is small) and de-emphasize it where the data is scarce (when k is large).

We are still able to derive an MM algorithm for maximizing the penalized log-likelihood function. The details are given in S1 Appendix. As a property of MM algorithms, the penalized log-likelihood function is also guaranteed to increase with number of iterations.

Turning now to how to choose λ, the ratio of the strength of the regularization term (measured by λ) and the number of observed data (measured by $\sum_{t=1}^{T} \sum_{k=0}^{K} m_k(t)$) can be used as a heuristic, reasonable criterion for choosing λ. The 'right' amount of regularization depends on many factors: the number of data points for each degree, the number of time-steps, the shape of the true attachment kernel, etc. We found that a reasonable amount of regularization with ratio ranges from 0 to 1 often leads to satisfactory results.

Table 2. Summary of true attachment kernels used in the Monte Carlo simulation.

Attachment kernel	Parameter	Value
$A_k = \max(k, 1)^\alpha$	α	0.5, 0.6, . . ., 1.5 (11 values)
$A_k = \min(100, \max(k, 1))^\beta$	β	0.8, 1.0, 1.2
$A_k = 3(\log(\max(k, 1)))^b + 1$	b	2, 3

Summary of true attachment kernels used in the Monte Carlo simulation. There is a total of 16 different kernels.

doi:10.1371/journal.pone.0137796.t002

Results and Discussion

Monte Carlo simulation

Here we compare five methods: Jeong's method, Newman's method, the corrected Newman's method, PAFit without regularization (ratio = 0) and PAFit with regularization (ratio = 0.1). We examine two different bin settings: $B = 100$ and $B = 20$. And we perform comparisons in three different functional forms of the true attachment kernels A_k. Table 2 shows the true attachment kernels we used in this experiment. For each true attachment kernel, we generate 100 networks according to the GT model. Starting from a seed network with 20 nodes, at each time-step, $m(t) = 5$ new edges and $n(t) = 1$ new node were added, until a total of 2000 nodes is reached. We compare the average relative error, defined as $\frac{1}{K}\sum_{k=1}^{K}\frac{(A_k - \hat{A}_k)^2}{A_k^2}$, between the true value A_k and $\hat{A}_k$.

We note some remarks about the implementation of the methods. First, due to the inherent sparse nature of Jeong's method, even with binning, sometimes the estimated result of Jeong's method has more 0 than those of the other methods. In such cases, to ensure fair comparison, we perform linear interpolation on the log scale to interpolate the value of those zero-value $\hat{A}_k$. Second, in Jeong's method we choose T_0 as the time when the 1500-th node is added, $T_1 = T_0 + 1$ and $\Delta T = 500$. Third, convergence condition for PAFit is $\epsilon = 10^{-5}$.

The result is shown in Fig 2. Overall, PAFit with regularization outperformed all remaining methods. This suggests that a small amount of regularization is indeed needed to reduce the error in the estimated result. We also notice that binning helps reduced the error in all methods. The fewer number of bins we used, the better the estimated result was found to be.

PAFit without regularization performed better than the corrected Newman's method. In the functional form $A_k = k^\alpha$, when α is up to about 1.2, PAFit without regularization delivered almost the same error as that of the corrected Newman's method, but when $\alpha > 1.2$, PAFit without regularization was better, especially when the number of bins is 20. In the remaining two functional forms, PAFit without regularization also performed better or at least similar to the corrected Newman's method.

When $B = 100$, except the functional form $A_k = \min(100, \max(k, 1))^\beta$, Jeong's method performed reasonably well in comparison with PAFit without regularization and the corrected Newman's method. When $B = 20$, Jeong's method became worse than those two methods. Newman's method (uncorrected) performed the worst in almost all cases, due to its underestimating A_k when k is large.

The Flickr social network

In this section we present the results from our analysis of a publicly available Flickr social network dataset [35]. It consists of a simple directed network of friendship relations between

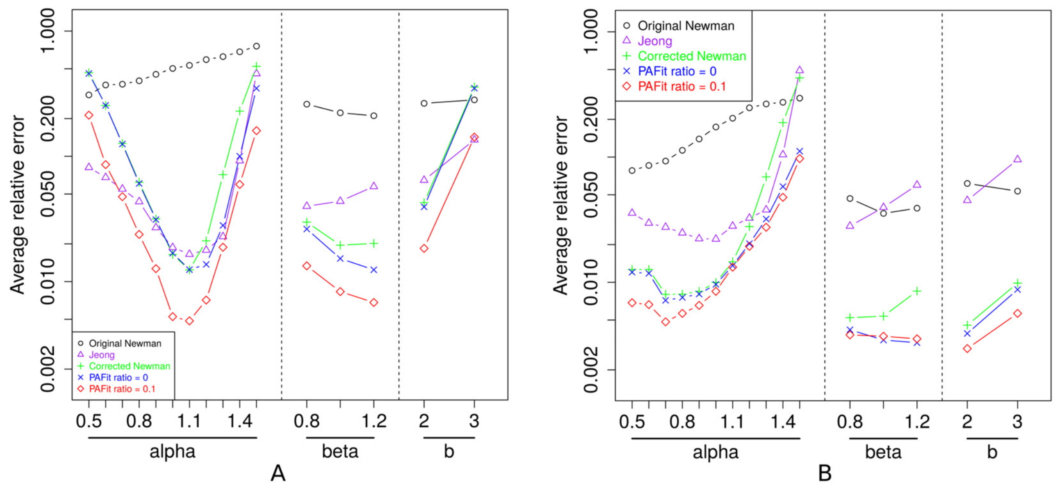

Fig 2. Comparison between five methods in average relative error. A: $B = 100$. B: $B = 20$. See Table 2 for the details of the true attachment kernels A_k used here.

doi:10.1371/journal.pone.0137796.g002

Flickr users. The dataset contains $T = 133$ days of growth. Table 3 shows some important summary statistics of the dataset. The results are shown in Fig 3. We also implemented a quasi-Newton speed-up scheme [51] for the MM algorithms used in PAFit. The convergence condition for PAFit is $\epsilon = 10^{-7}$.

For Jeong's method, we use all the available data by choosing $T_0 = 0$, $T_1 = 1$ and $\Delta T = 132$. It found a sub-linear attachment kernel $\hat{A}_k$. The value of $\hat{A}_k$ fluctuated considerably for large value of k, however. Note that the domain of $\hat{A}_k$ of Jeong's method is smaller than in other methods, since the degrees of T_0-nodes are fixed at T_0.

In the estimated result of Newman's method, we once again spotted the falling off $\hat{A}_k$ when k is large. This phenomenon is, of course, completely eliminated in the corrected Newman's method.

For PAFit, we performed regularization with ratio equal to 0.1. Although the estimation result of PAFit looks very similar to that of the corrected Newman's method, the $\hat{A}_k$ in the high

Table 3. Summary statistics for the Flickr social network dataset.

$\|V\|$	$\|E\|$	T	$\Delta\|V\|$	$\Delta\|E\|$	$\hat{\gamma}$
2302925	33140018	134	815867	16105211	2.15

Summary statistics for the Flickr social network dataset [35]. This is a directed simple network. The numbers $|V|$ and $|E|$ are the total number of nodes and edges in the final network, respectively. Meanwhile, T is the number of observed time-steps, while $\Delta|V|$ and $\Delta|E|$ are the increments of nodes and edges after time $t = 0$, respectively. The value $\hat{\gamma}$ is the power-law exponent of the degree distribution of the final network [6].

doi:10.1371/journal.pone.0137796.t003

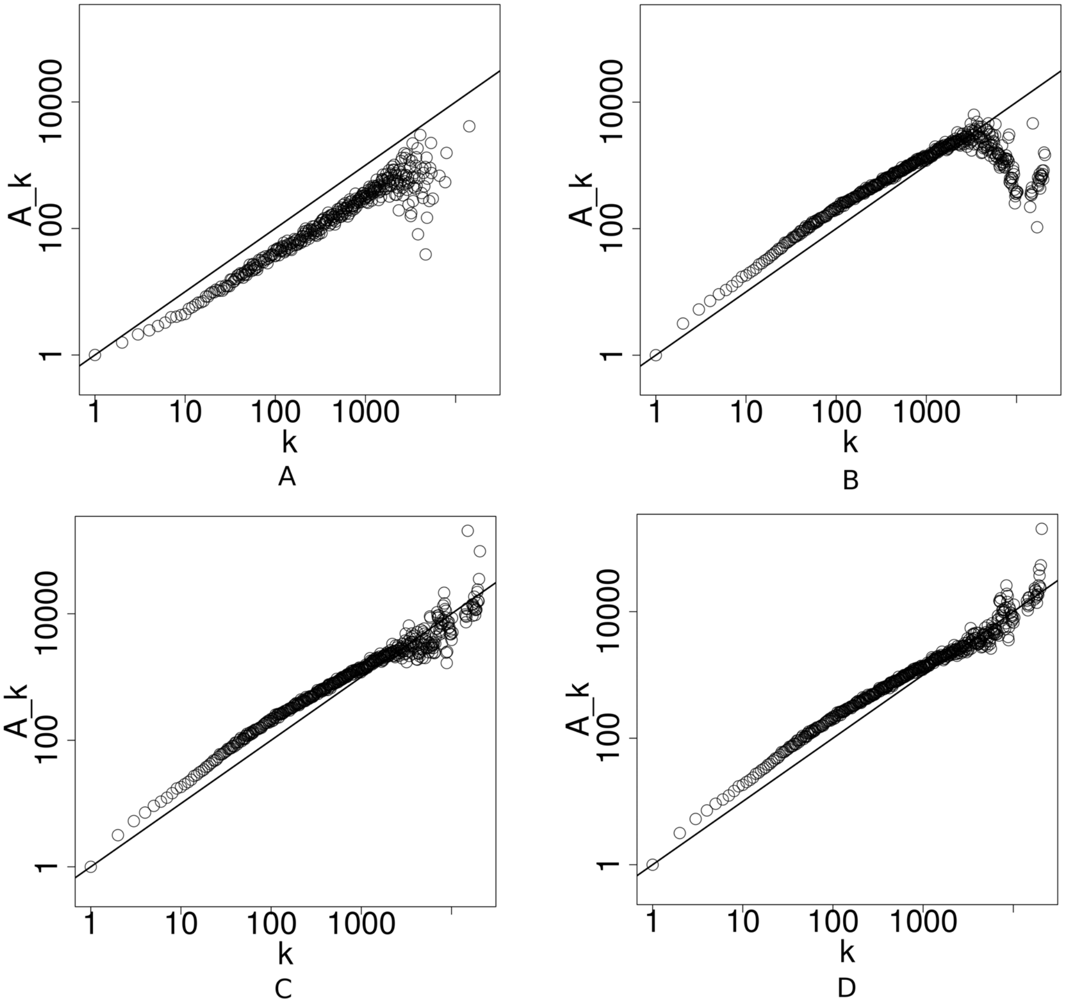

Fig 3. Estimation of the attachment kernel in the Flickr social network dataset. A: Jeong's method. B: Newman's method. C: Corrected Newman's method. D: PAFit. The plots are on a log-log scale. The solid line corresponding to $A_k = k$ is plotted as a visual guide.

doi:10.1371/journal.pone.0137796.g003

degree region of PAFit are less fluctuated and more compact than those of the corrected Newman's method. It is worth noting that we spotted a clear signal of deviation from the log-linear model $A_k = k^{\alpha}$ here.

Conclusion

We proposed a statistically sound method, called PAFit, for estimating the attachment kernel A_k in temporal networks. The method is nonparametric in the sense that it does not assume any particular functional form for A_k. In this way it is able to detect different types of functional forms. We proved that the log-likelihood function is concave under a suitable re-parametrization, and provided a Minorize-Maximization algorithm for its maximization. The proposed algorithm is shown to increase the log-likelihood function monotonically with the number of iterations. It also has the property that if it converges, it will converge to global maximum of the log-likelihood function. We also investigated binning and regularization, and showed that these two simple techniques considerably improve the quality of the estimation. We reported clear evidence for the presence of PA in the Flickr social network. We also found that the functional form of the attachment kernel differs from the classically assumed log-linear form, $A_k = k^\alpha$.

In this paper, we focused on estimating the attachment kernel. Another important ingredient related to temporal network evloution is node fitness [52–55]. In the future, we would like to consider a natural generalization of the method of this paper, in which we can propose a method to estimate jointly the attachment kernel and node fitness. Another potential research direction is theoretical justifications of the superiority of PAFit over Jeong's method and Newman's method. We expect that PAFit and its future extensions will become useful tools for analyzing many types of temporal networks in the real-world.

Supporting Information

S1 Appendix. Derivation of the MM algorithm when the regularization term in Eq (18) is added.
(PDF)

Author Contributions

Conceived and designed the experiments: TP PS HS. Performed the experiments: TP PS HS. Analyzed the data: TP PS HS. Contributed reagents/materials/analysis tools: TP PS HS. Wrote the paper: TP PS HS.

References

1. Caldarelli G. Scale-Free Networks. Oxford Universiy Press; 2007.
2. Pastor-Satorras R, Vespignani A. Epidemic dynamics and endemic states in complex networks. Phys Rev E. 2001 May; 63:066117. Available from: http://link.aps.org/doi/10.1103/PhysRevE.63.066117 doi: 10.1103/PhysRevE.63.066117
3. Newman MEJ. The Structure and Function of Complex Networks. SIAM Review. 2003; 45(2):167–256. Available from: http://dx.doi.org/10.1137/S003614450342480 doi: 10.1137/S003614450342480
4. Albert R, Barabási A. Emergence of scaling in random networks. Science. 1999 October; 286:509–512. doi: 10.1126/science.286.5439.509 PMID: 10521342
5. Newman M. Power laws, Pareto distributions and Zipf's law. Contemporary Physics. 2005; 46:323–351. doi: 10.1080/00107510500052444
6. Clauset A, Shalizi CR, Newman MEJ. Power-Law Distributions in Empirical Data. SIAM Review. 2009; 51(4):661–703. Available from: http://dx.doi.org/10.1137/070710111 doi: 10.1137/070710111
7. Mitzenmacher M. ABrief History of Generative Models for Power Law and Lognormal Distributions. Internet Math. 2003; 1(2):226–251. Available from: http://projecteuclid.org/euclid.im/1089229510 doi: 10.1080/15427951.2004.10129088
8. Lima-Mendez G, van Helden J. The powerful law of the power law and other myths in network biology. Mol BioSyst. 2009; 5:1482–1493. Available from: http://dx.doi.org/10.1039/B908681A doi: 10.1039/b908681a PMID: 20023717
9. Krapivsky P, Rodgers G, Redner S. Organization of growing networks. Physical Review E. 2001;p. 066123. doi: 10.1103/PhysRevE.63.066123
10. Erdös P, Rényi A. On random graphs. Publicationes Mathematicae Debrecen. 1959; 6:290–297.
11. Lü L, Zhou T. Link prediction in complex networks: A survey. Physica A: Statistical Mechanics and its Applications. 2011; 390(6):1150–1170. Available from: http://www.sciencedirect.com/science/article/pii/S037843711000991X doi: 10.1016/j.physa.2010.11.027

12. Newman M. Clustering and preferential attachment in growing networks. Physical Review E. 2001; 64 (2):025102. doi: 10.1103/PhysRevE.64.025102

13. Jeong H, Néda Z, Barabási A. Measuring preferential attachment in evolving networks. Europhysics Letters. 2003; 61(61):567–572. doi: 10.1209/epl/i2003-00166-9

14. Massen C, Jonathan P. Preferential attachment during the evolution of a potential energy landscape. The Journal of Chemical Physics. 2007; 127:114306. doi: 10.1063/1.2773721 PMID: 17887836

15. Sheridan P, Yagahara Y, Shimodaira H. Measuring preferential attachment in growing networks with missing-timelines using Markov chain Monte Carlo. Physica A Statistical Mechanics and its Applications. 2012 October; 391:5031–5040. doi: 10.1016/j.physa.2012.05.041

16. Gómez V, Kappen HJ, Kaltenbrunner A. Modeling the Structure and Evolution of Discussion Cascades. In: Proceedings of the 22nd ACM Conference on Hypertext and Hypermedia. HT'11. New York, NY, USA: ACM; 2011. p. 181–190. Available from: http://doi.acm.org/10.1145/1995966.1995992

17. Kunegis J, Blattner M, Moser C. Preferential Attachment in Online Networks: Measurement and Explanations. In: WebSci'13. France; 2013.

18. Capocci A, Servedio V, Colaiori F, Buriol L, Donato D, Leonardi S, et al. Preferential attachment in the growth of social networks: The internet encyclopedia Wikipedia. Phys Rev E. 2006 Sep; 74:036116. Available from: http://link.aps.org/doi/10.1103/PhysRevE.74.036116 doi: 10.1103/PhysRevE.74.036116

19. Herdagdelen A, Aygün E, Bingol H. A formal treatment of generalized preferential attachment and its empirical validation. EPL (Europhysics Letters). 2007; 78(6):60007. Available from: http://stacks.iop.org/0295-5075/78/i = 6/a = 60007 doi: 10.1209/0295-5075/78/60007

20. Kou Z, Zhang C. Reply networks on a bulletin board system. Physical Review E. 2003; 67:036117. doi: 10.1103/PhysRevE.67.036117

21. Gong P, van Leeuwen C. Emergence of scale-free network with chaotic units. Physica A: Statistical Mechanics and its Applications. 2003; 321(3):679–688. Available from: http://EconPapers.repec.org/RePEc:eee:phsmap:v:321:y:2003:i:3:p:679-688 doi: 10.1016/S0378-4371(02)01735-1

22. Eisenberg E, Levanon EY. Preferential Attachment in the Protein Network Evolution. Phys Rev Lett. 2003 Sep; 91:138701. Available from: http://link.aps.org/doi/10.1103/PhysRevLett.91.138701 doi: 10.1103/PhysRevLett.91.138701 PMID: 14525344

23. Redner S. Citation statistics from 110 years of physical review. Physics Today. 2005; 58 (6):49–54. doi: 10.1063/1.1996475

24. Valverde S, Solé RV, Bedau MA, Packard N. Topology and evolution of technology innovation networks. Phys Rev E. 2007 Nov; 76:056118. Available from: http://link.aps.org/doi/10.1103/PhysRevE.76.056118 doi: 10.1103/PhysRevE.76.056118

25. Eom YH, Jeon C, Jeong H, Kahng B. Evolution of weighted scale-free networks in empirical data. Phys Rev E. 2008 May; 77:056105. Available from: http://link.aps.org/doi/10.1103/PhysRevE.77.056105 doi: 10.1103/PhysRevE.77.056105

26. Kaza S, Xu J, Marshall B, Chen H. Topological Analysis of Criminal Activity Networks: Enhancing Transportation Security. Intelligent Transportation Systems, IEEE Transactions on. 2009 March; 10 (1):83–91. doi: 10.1109/TITS.2008.2011695

27. Menghui L, Liang G, Ying F, Jinshan W, Zengru D. Emergence of global preferential attachment from local interaction. New Journal of Physics. 2010; 12(4):043029. Available from: http://stacks.iop.org/1367-2630/12/i = 4/a = 043029 doi: 10.1088/1367-2630/12/4/043029

28. Eom YH, Fortunato S. Characterizing and Modeling Citation Dynamics. PLoS ONE. 2011 09; 6(9): e24926. Available from: http://dx.doi.org/10.1371%2Fjournal.pone.0024926 doi: 10.1371/journal.pone.0024926 PMID: 21966387

29. Perc M. Evolution of the most common English words and phrases over the centuries. J R Soc Interface. 2012; 9:3323–3328. doi: 10.1098/rsif.2012.0491 PMID: 22832364

30. Wu X, Yu K, Xin W. On the growth of Internet application flows: A complex network perspective. In: INFOCOM. IEEE; 2011. p. 2096–2104.

31. Mingyang W, Guang Y, Daren Y. Measuring the preferential attachment mechanism in citation networks. Physica A: Statistical Mechanics and its Applications. 2008; 387(18):4692–4698. Available from: http://www.sciencedirect.com/science/article/pii/S0378437108003208 doi: 10.1016/j.physa.2008.03.017

32. Pham T, Sheridan P, Shimodaira H. Nonparametric estimation of the preferential attachment function in complex networks: evidence of deviations from log linearity. In: Proceedings of ECCS'14, Springer Proceedings in Complexity (in press); 2015.

33. Hunter D, Lange K. Quantile regression via an MM algorithm. J Comput Graphical Stat. 2000;p. 60–77. doi: 10.2307/1390613

34. Hunter D, Lange K. A Tutorial on MM Algorithms. The American Statistician. 2004; 58:30–37. doi: 10.1198/0003130042836

35. Mislove A, Koppula H, Gummadi K, Druschel P, Bhattacharjee B. Growth of the Flickr social network. In: Proc. Workshop on Online Social Networks; 2008. p. 25–30.

36. Pham T, Sheridan P, Shimodaira H. PAFit: Nonparametric Estimation of Preferential Attachment and Node Fitness in Temporal Complex Networks; 2015. R package version 0.7.4.

37. Sheridan P, Kamimura T, Shimodaira H. A Scale-Free Structure Prior for Graphical Models with Applications in Functional Genomics. PLoS ONE. 2010 11; 5(11):e13580. Available from: http://dx.doi.org/10.1371%2Fjournal.pone.0013580 doi: 10.1371/journal.pone.0013580 PMID: 21079769

38. Lopes FM, Jr DCM, Barrera J, Jr RMC. A feature selection technique for inference of graphs from their known topological properties: Revealing scale-free gene regulatory networks. Information Sciences. 2014; 272(0):1–15. Available from: http://www.sciencedirect.com/science/article/pii/S0020025514002023 doi: 10.1016/j.ins.2014.02.096

39. Eagle N, Macy M, Claxton R. Network Diversity and Economic Development. Science. 2010; 328 (5981):1029–1031. Available from: http://www.sciencemag.org/content/328/5981/1029.abstract doi: 10.1126/science.1186605 PMID: 20489022

40. Backes AR, Casanova D, Bruno OM. Texture analysis and classification: A complex network-based approach. Information Sciences. 2013; 219(0):168–180. Available from: http://www.sciencedirect.com/science/article/pii/S0020025512004677 doi: 10.1016/j.ins.2012.07.003

41. Pastor-Satorras R, Castellano C, Van Mieghem P, Vespignani A. Epidemic processes in complex networks. arXiv preprint arXiv:14082701. 2014;.

42. Xia C, Wang L, Sun S, Wang J. An SIR model with infection delay and propagation vector in complex networks. Nonlinear Dynamics. 2012; 69(3):927–934. Available from: http://dx.doi.org/10.1007/s11071-011-0313-y doi: 10.1007/s11071-011-0313-y

43. Dehmer M, Mowshowitz A. A history of graph entropy measures. Information Sciences. 2011; 181 (1):57–78. Available from: http://www.sciencedirect.com/science/article/pii/S0020025510004147 doi: 10.1016/j.ins.2010.08.041

44. Kraus V, Dehmer M, Schutte M. On Sphere-Regular Graphs and the Extremality of Information-Theoretic Network Measures. MATCH Communications in Mathematical and in Computer Chemistry. 2013; 70:885–900.

45. Dehmer M, Grabner M. The Discrimination Power of Molecular Identification Numbers Revisited. MATCH Communications in Mathematical and in Computer Chemistry. 2013; 69:785–794.

46. Cao S, Dehmer M, Shi Y. Extremality of degree-based graph entropies. Information Sciences. 2014; 278(0):22–33. Available from: http://www.sciencedirect.com/science/article/pii/S0020025514004393 doi: 10.1016/j.ins.2014.03.133

47. Price DJdS. A general theory of bibliometric and other cumulative advantage processes. Journal of the American Society for Information Science. 1976; 27:292–306. doi: 10.1002/asi.4630270505

48. Price DJdS. Networks of Scientific Papers. Science. 1965; 149(3683):510–515. Available from: http://www.sciencemag.org/content/149/3683/510.short doi: 10.1126/science.149.3683.510 PMID: 14325149

49. Boyd S, Vandenberghe L. In: Convex optimization. Cambridge University Press; 2000.

50. Woodbury MA. Inverting modified matrices. Statistical Research Group, Memo. Rep. no. 42. Princeton University, Princeton, N. J.; 1950.

51. Zhou H, Alexander D, Lange K. A quasi-Newton acceleration for high-dimensional optimization algorithms. Statistics and Computing. 2011; 21:261–273. doi: 10.1007/s11222-009-9166-3 PMID: 21359052

52. Bianconi G, Barabási A. Competition and multiscaling in evolving networks. Europhys Lett. 2001; 54:436. doi: 10.1209/epl/i2001-00260-6

53. Caldarelli G, Capocci A, De Los Rios P, Muñoz MA. Scale-Free Networks from Varying Vertex Intrinsic Fitness. Phys Rev Lett. 2002 Dec; 89:258702. Available from: http://link.aps.org/doi/10.1103/PhysRevLett.89.258702 doi: 10.1103/PhysRevLett.89.258702 PMID: 12484927

54. Kong J, Sarshar N, Roychowdhury V. Experience versus Talent Shapes the Structure of the Web. Proceedings of the National Academy of Sciences of the USA. 2008; 37:105.

55. Nguyen K, Tran DA. Fitness-Based Generative Models for Power-Law Networks. In: Thai MT, Pardalos PM, editors. Handbook of Optimization in Complex Networks. Springer Optimization and Its Applications. Springer US; 2012. p. 39–53.