Noname manuscript No.
(will be inserted by the editor)

Detection of user roles with thread growth models

Alberto Lumbreras · Julien Velcin ·
Marie Guégan · Bertrand Jouve

Received: date / Accepted: date

1 Introduction

Random graph models are stochastic generators of graphs that try to reproduce the properties of a some real-world graphs. Ideally, these models should reproduce a large set of properties using a minimum number of assumptions and parameters. If the generated graphs and the real-world graphs share some relevant properties, then the proposed growth mechanism might be a reasonable approximation of the growth laws under which the real-world graphs evolve (Kolaczyk, 2009). Formally, a growth model is a probability distribution that quantifies the probability of an existing vertex i of being chosen as the parent for a new vertex x_t:

$$p(x_t \sim i|G_{t-1}; \boldsymbol{\theta})$$

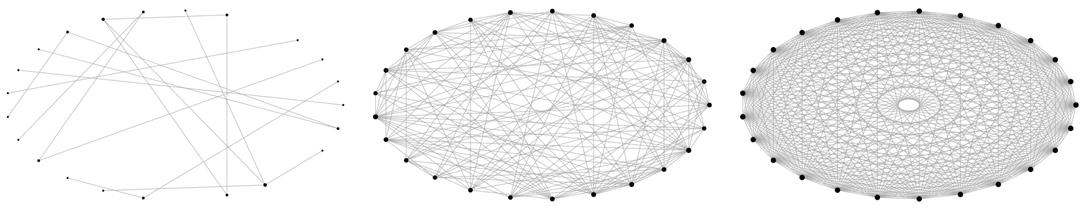
where G_{t-1} is the state of the graph before x_t is attached and $\boldsymbol{\theta}$ is the vector of model parameters.

Online discussions can be regarded as evolving tree graphs where vertices represent messages and a directed edge indicates that a message is a reply to another message. The tree starts with the root message that starts the conversation, and then evolves towards some form of tree. Different models have been proposed to account for both the way how a tree evolves and the final

Alberto Lumbreras · Marie Guégan
Technicolor
975 Avenue des Champs Blancs,
35576 Cesson-Sévigné,
France
E-mail: alberto.lumbreras@technicolor.com
E-mail: marie.guegan@technicolor.com

Julien Velcin
Laboratoire ERIC, Université de Lyon,
5, avenue Pierre Mendès France, 69676 Bron,
France
E-mail: julien.velcin@univ-lyon2.fr

Bertrand Jouve
Université de Toulouse; UT2; FRAMESPA/IMT; 5 allée Antonio Machado, 31058 Toulouse, cedex 9
CNRS; FRAMESPA; F-31000 Toulouse
CNRS; IMT; F-31000 Toulouse
France
E-mail: jouve@univ-tlse2.fr



(a) $p = 0.1$ **(b)** $p = 0.5$ **(c)** $p = 1$

Fig. 1 Erdös-Rényi graphs

properties of the tree. The parameters of these models are fixed and every new vertex is assumed to chose its parent according to the current state of the graph G_{t-1} and a set of fixed parameters that govern the whole process. These parameters regulate, for instance, the tendency to reply to the root, or how fast a vertex with more replies attracts more replies. Since the parameters are fixed, these models implicitly assume that the choice of a parent is independent of the user who writes the new post.

In this chapter, we explicitly assume that posts written by different users may have different parameters. Some users, for instance, might tend to reply to the root and avoid conversations deeper in the tree. Others might tend to ignore old posts. Others might be specially attracted by popular posts. Formally, we assume that there are K latent types of users and that users of type k behave according to their own group parameters $\boldsymbol{\theta}_k$. Equation 1 depends now on the author of post x_t:

$$p(x_t \sim i | G_{t-1}; \boldsymbol{\theta}_{z_u})$$

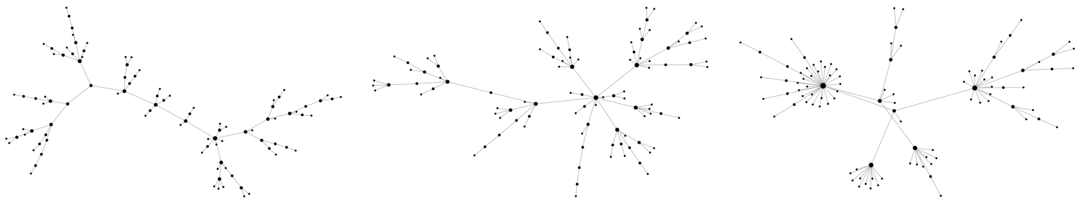
where z_u is the group of user u and $\boldsymbol{\theta}_{z_u}$ are the parameters of that group.

The remaining of this chapters is structure as follows: first, we recall the Preferential Attachment model and we present the thread models. Then we present our model, which finds k sets of parameters for k types of user. Finally, we compare both models and show that, while there is no a remarkable difference on how they reproduce structural properties of real threads, our model can be used for recommendation of posts.

2 Network Growth model

Growth models try to reproduce not only the final properties of the network but also how the network is built. The *preferential attachment* model proposed by Barabási and Albert (1999) is the best well-known member of this family. The Barabasi-Albert model builds a graph by sequentially adding its vertices; once a new vertex x is added to the graph it decides whether to create an edge to an existing vertex i with probability

$$p(x \sim i | G) = \frac{d_i^\alpha}{Z}; \qquad Z = \sum_{j=1}^{|V(G)|} d_j^\alpha \tag{1}$$



(a) $\alpha = 0$ (b) $\alpha = 1$ (c) $\alpha = 1.8$

Fig. 2 Barabasi-Albert graphs with one edge created at every step.

where d_i is the degree of the vertex i before vertex x is added[1]. No This model reproduces a rich-get-richer phenomena controlled by the parameter α. The particular case of $\alpha = 1$ is called *linear preferential-attachment* since the probabilities increase linearly with the number of degrees. Figure 2 shows examples of Barabasi-Albert graphs generated with different α. The Barabasi-Albert explains very well the power-law degree distributions observed in many real graphs.

During the recent years, some authors have proposed models to explain the growth of online conversations. (Kumar et al, 2010; Gómez et al, 2010; Wang et al, 2012; Gómez et al, 2012). We describe this models in the following sections.

2.1 Preferential attachment and recency

Kumar et al (2010) proposed a model that combines both *preferential-attachment* and *recency*. The higher the degree of a post and the later it was published, the easier for this post to attract the incoming replies. At every time step, a decision is made to stop the thread or to add a new post. Every new post choses its parent according to:

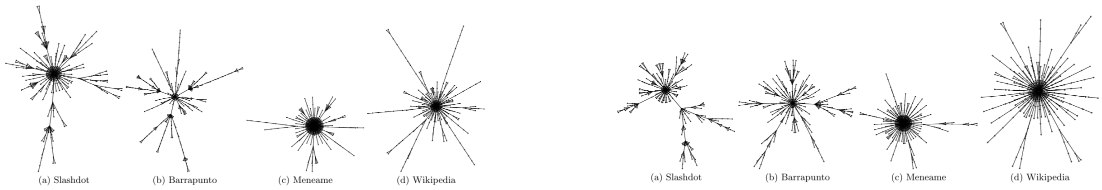
$$p(x \sim i|G) = \frac{h(d_u, r_u)}{Z}; \qquad h(d_u, r_u) = \alpha d_u + \tau^{r_u}; \qquad Z = \sum_{n=1}^{|V(G)|} h(d_n, r_n) + \delta \tag{2}$$

where r_u is the number of time steps since u was added to the thread. The authors report that when the alternative function $h(d_u, r_u) = d_u \tau^{r_u}$ is used, the recency factor prevents the preferential attachment factor from generating heavy-tailed degree distributions. The choice of placing α as a coefficient instead of an exponent is made for mathematical convenience so that Z does not depend on the graph structure at that particular moment.

The authors also propose an improvement of the model to account for the identity of posts authors. For a new post v replying to a post u, its author $a(v)$ can be either $a(u)$ (a self-reply), another author $a(w)$ that has already participated in the chain from u to the root, or some other new author belonging to the set of authors A that have not participated in the chain:

$$a(v) = \begin{cases} a(w) & \text{with probability } \gamma \\ a(u) & \text{with probability } \epsilon \\ a \in A & \text{with probability } 1 - \gamma - \epsilon \end{cases} \tag{3}$$

[1] To avoid loaded notations we avoid writing G_{t-1} $d_{i,t-1}$ and Z_{t-1} when it is clear by the context that they correspond to the last state of the graph before adding the new node x.



(a) real **(b)** synthetic

Fig. 3 Random grahs for discussion threads. Gómez-Kappen-Kaltenbrunner

Parameter estimation

The parameters are estimated by a grid search computing the maximum likelihood estimation of different $\alpha, \tau, \gamma, \epsilon$.

2.2 Preferential attachment and root bias

In Gómez et al (2010) the authors combine *preferential-attachment* with a *bias towards the root*. The probability of choosing an existing parent k is:

$$p(x \sim k|G) \propto (\beta_k d_k)^{\alpha_k}$$

where

$$\alpha_k = \begin{cases} \alpha_1 & \text{for } k = 1 \\ \alpha_c & \text{for } k \in \{2, ..., t\} \end{cases}$$
$$\beta_k = \begin{cases} \beta & \text{for } k = 1 \\ 1 & \text{for } k \in \{2, ..., t\} \end{cases} \tag{4}$$

Note that α_k is the preferential attachment exponent and that if $\alpha_1 = \alpha_c$ and $\beta = 1$ we recover the Barabasi-Albert model of preferential attachment.

Parameter estimation

Maximum Likelihood Estimation of α_1, α_c and β is done by minimizing the negative log-likelihood:

$$\log \mathcal{L}(\mathbf{G}|\alpha_k, \beta_k) = \sum_{i=1}^{N} \sum_{t=2}^{|V(G_i|} \alpha_k (\log \beta_k + \log d_{k,(t-1)}) - \log \sum_{l=1}^{t} (\beta_l d_{l,(t-1)}) \tag{5}$$

Since this is a convex function, the minimization is done with the Nelder-Mead algorithm (`fminsearch` in Matlab).The authors fitted the parameters to several datasets and then generate graphs that resemble the original conversations (see Figure 3).

2.3 Preferential attachment, root bias and recency

In Gómez et al (2012) the authors combine *preferential-attachment*, a *bias towards the root* and *novelty*. Unlike in their former model in Gómez et al (2010), here they sum these factors instead of multiplying them:

$$p(x \sim k|G) \propto \beta_k + \alpha d_{k,(t-1)} + \tau^{t-k} \tag{6}$$

Parameter estimation

The negative log-likelihood to be minimized is:

$$\log \mathcal{L}(\mathbf{G}|\alpha, \beta_k, \tau) = \sum_{i=1}^{N} \sum_{t=2}^{|V(G_i)|} \log \left(\beta_k + \alpha d_{k,(t-1)} + \tau^{t-k}\right) - \log \sum_{l=1}^{t} \left(\beta_l + \alpha d_{l,(t-1)} + \tau^{t-l}\right) \tag{7}$$

As in Gómez et al (2010), parameters are optimized through numerical methods.

2.4 Time-sensitive preferential attachment

In Wang et al (2012) authors make two observations. On the one hand, that distance between its posts follows an upper-truncated Pareto distribution. On the other hand, that threads grow faster when they are featured in the front page (or some sections showing the top discussions at that moment) and they slow down once they disappear from the front page. From this, they propose a model that models the growth of a thread in a given forum. As for the structure, they use preferential-attachment. It is the only existing model that includes real time.

Parameter estimation

Their model has two parts: a upper-truncated distribution to model the time of response and the preferential attachment to model the structure. Authors use their Maximum Likelihood estimators, which are both known in the literature.

2.5 Limitations of current models

There are two aspects that might me improved:

- Time is only poorly combined with the structure in Wang et al (2012). Actually in this model time is independent of the structure and vicecersa. As suggested in Gómez et al (2012) (Conclusions) combining both time and structure is an interesting line of research. Besides, there are probably other ways of consider time.
- These models estimate their parameters once and therefore very different threads are summarized with common parameters. However, imagine that we learned our parameters from a set of threads $\mathbf{G}$ and now we want to make predictions on a particular new thread G^*, that is, we want to compute $p(x \sim i|G^*_{1:t-1}, \mathbf{G})$. There is a lot to be learned from the particular ongoing dynamics of the conversation until time t, and making predictions based on the globally estimated parameters will not be flexible enough to adapt the prediction to the last observations. Bayesian inference is a natural way to do this since we will be constantly updating our believes every time a new observation (post) arrives. The challenge here is its computational cost, so we should probably work with fast approximations to the posterior

Growth graph models are stochastic processes governed by a set of parameters. Once these parameters are estimated, the model does not change anymore. Yet, threads can vary a lot and therefore the parameters that explain the global dataset do not usually explain the specific dynamics of a particular discussion.

3 Mixture-based model

We build our model over Gómez et al (2012). We consider the there are N latent groups of users and that each group i has its own parameters $\alpha_i, \beta_i, \tau_i$. Our task is to detect the latent clusters and the parameters associated to each cluster. For this, we will make use of the Expectation-Maximization algorithm.

3.1 General EM

$$\begin{aligned}
\ln p(\mathbf{X}|\boldsymbol{\theta}) &= \ln \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}) && (8)\\
&= \ln \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})}{q(\mathbf{Z})} && (9)\\
&\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln \frac{p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})}{q(\mathbf{Z})} && (10)
\end{aligned}$$

The equality holds when $q(\mathbf{Z})$ is the posterior $p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta})$:

$$\ln p(\mathbf{X}|\boldsymbol{\theta}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}) \ln \frac{p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})}{p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta})} \qquad \blacksquare \tag{11}$$

3.2 EM for current model

Given a post n written at time step t, we consider the three following observations: the degree of its parent at time $t-1$, denoted as d_n; whether its parent is the root post, denoted as r_n; and the lag, in time steps, between the post and its parent, denoted as l_n. Let $\mathbf{X} = \{x_1, ..., x_N\}$ be a matrix where $x_i = \{t_i, d_i, r_i, l_i\}$

The model of Gómez et al (2012) can be expressed as:

$$p(\mathbf{X}|\boldsymbol{\theta}) = \prod_{n=1}^{N} \frac{\alpha d_n + \beta r_n + \tau^{l_n}}{\alpha(t_n - 1) + \beta + \frac{\tau(\tau^{t_n} - 1)}{\tau - 1}} \tag{12}$$

where instead of looping through all the posts in all the trees, we just look through all the posts. The reason to change the notation is that this will make or equations more uncluttered, it makes it more universal (the total likelihood expressed a single product individual likelihoods), and that is friendlier to code, since we think of the data as a matrix rather than as a set of trees .

We consider that there exists latent groups of users where every group has its own parameters $\theta_k = \{\alpha_k, \beta_k, \tau_k\}$. Our task, then, is to estimate the parameters of each group and the group where each user belongs. Let $\mathbf{X}_u$ the submatrix of $\mathbf{X}$ composed of all posts written by user u.

Let $\mathbf{Z} = \{z_1, ..., z_U\}$ be the indicators matrix where $z_i = \{z_{i1}, ..., z_{iK}\}$ and where z_{ik} is one if user i belongs to group k and zero otherwise.

In the following, we will prepare the elements needed in Equation 11. First, we need the complete loglikelihood. The complete likelihood is:

$$p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}) = \prod_{n=1}^{N}\prod_{k=1}^{K} \pi_k^{z_{nk}} \left(\frac{\alpha_k d_n + \beta_k r_n + \tau_k^{l_n}}{\alpha_k(t_n - 1) + \beta_k + \frac{\tau_k(\tau_k^{t_n} - 1)}{\tau_k - 1}} \right)^{z_{nk}} \tag{13}$$

and its logarithm:

$$\ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}) = \sum_{n=1}^{N}\sum_{k=1}^{K} z_{nk} \left(\ln \pi_k + ln \frac{\alpha_k d_n + \beta_k r_n + \tau_k^{l_n}}{\alpha_k(t_n - 1) + \beta_k + \frac{\tau_k(\tau_k^{t_n} - 1)}{\tau_k - 1}} \right) \tag{14}$$

Note that we use indicator variables for the posts to avoid introducing the users for now. A post has the indicator of the user who wrote it. On the other hand, the posterior distribution of the latent factors is:

$$p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}) \propto p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$$

that can be factorized in $\mathbf{z}_1, ...\mathbf{z}_U$. For a given user u:

$$p(z_u|\mathbf{X}_u, \boldsymbol{\theta}) \propto p(\mathbf{X}_u, z_u|\boldsymbol{\theta})$$

The expected value of the indicator variable $z_u k$ under this posterior distribution is given by:

$$\mathbb{E}[z_{uk}] = \frac{\sum_{z_u} z_{uk} \prod_c \left[\pi_c \frac{\alpha_c d_n + \beta_c r_n + \tau_c^{l_n}}{\alpha_c(t_n - 1) + \beta_c + \frac{\tau_c(\tau_c^{t_n} - 1)}{\tau_c - 1}} \right]^{z_{uc}}}{\sum_{z_u} \prod_c \left[\pi_c \frac{\alpha_c d_n + \beta_c r_n + \tau_c^{l_n}}{\alpha_c(t_n - 1) + \beta + \frac{\tau_c(\tau_c^{t_n} - 1)}{\tau_c - 1}} \right]^{z_{uc}}} = \frac{\pi_k \frac{\alpha_c d_n + \beta_c r_n + \tau_c^{l_n}}{\alpha_c(t_n - 1) + \beta_c + \frac{\tau_c(\tau_c^{t_n} - 1)}{\tau_c - 1}}}{\sum_{j=1}^{K} \pi_j \frac{\alpha d_n + \beta r_n + \tau^{l_n}}{\alpha(t_n - 1) + \beta + \frac{\tau(\tau^{t_n} - 1)}{\tau - 1}}} = \gamma(z_{uk}) \tag{15}$$

[2] Finally, we can obtain the expected value of the complete data log likelihood:

$$\mathbb{E}_{\mathbf{Z}}[\ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})] = \sum_{n=1}^{N}\sum_{k=1}^{K} \gamma(z_{nk}) \left(\ln \pi_k + ln \frac{\alpha_k d_n + \beta_k r_n + \tau_k^{l_n}}{\alpha_k(t_n - 1) + \beta_k + \frac{\tau_k(\tau_k^{t_n} - 1)}{\tau_k - 1}} \right) \tag{16}$$

which we can optimize iteratively.

[2] TODO: This has to be done by user, not by post. there is a product missing. Actually use logs, sum and finally get back with exp

Appendices

(Only for internal discussion)

References

Barabási AL, Albert R (1999) Emergence of scaling in random networks. Science 286(October)

Gómez V, Kappen HJ, Kaltenbrunner A (2010) Modeling the structure and evolution of discussion cascades. In: Proceedings of the 22nd ACM conference on Hypertext and hypermedia, pp 181–190

Gómez V, Kappen HJ, Litvak N, Kaltenbrunner A (2012) A likelihood-based framework for the analysis of discussion threads. World Wide Web p 31

Kolaczyk ED (2009) Statistical Analysis of Network Data: Methods and Models, 1st edn. Springer Publishing Company, Incorporated

Kumar R, Mahdian M, McGlohon M (2010) Dynamics of Conversations. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 553–562

Wang C, Ye M, Huberman Ba (2012) From user comments to on-line conversations. Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '12 p 244