Modelling the evolution of a bipartite graph
CS224W Project

Clement Ntwari Nshuti
Stanford - MS EE
cntwarin@stanford.edu

Nicolas Ehrhardt
Stanford - MS CS
ehrhardn@cs.stanford.edu

1 Introduction

Google Local, Amazon, Ebay and Yelp are examples of businesses whose quality rely heavily on information provided by users in the form of reviews of businesses or of products. It is of utmost importance to be able to store this information and efficient storing can be helped by having a good estimate of how the number of reviews, reviewers and reviewees will evolve over time. In each case, the triplets (review, reviewer, reviewee) can be seen as elements of a bipartite graph in which reviewers constitute one set of nodes, reviewees constitute the other. A link exist between two nodes of the two groups if the reviewee-node has been reviewed by the reviewer-node.

Extensive research has been done on modelling the evolution of classical social networks [4], [6], [5]. Bipartite graphs have not had such a great success. This is due on the one hand to the fact that the evolution of some features is trivially predicted for bipartite graphs : for example the clustering coefficient in a bipartite graph is always zero. On the other hand it is most likely assumed that what holds true for a regular multi-partite graph also holds for a bipartite graph. However as we will we see in this work, it is important to build specific models for some types of bipartite graphs.

In this paper we focus on the bipartite graph of user-business reviews from Yelp. We present a model for the evolution of this graph. Our work is strongly builds on Leskovec et al.'s work in modelling the microscopic evolution of social networks.

We begin by briefly summarizing previous work done on this topic in section 2 before presenting our approach to the problem in section 3. There we describe the dataset at hand in detail and present the preliminary analysis that motivated the design of the evolution algorithm described in 4. Section 6 is dedicated to the analysis of the results before concluding in section 8.

2 Relevant work

The most relevant work we found in the literature about the evolution of bipartite graphs is Kunegis, Luca, and Albayrak's work on modelling the edge creation probability using infinite weighted sums of powers of the adjacency matrix A. Despite the beauty of the mathematical model proposed by the authors, it's performances were often times not better than a classical preferential attachement model.

Leskovec et al.'s work in "Microscopic Evolution of Social Networks" more closely resembles what we are aiming for with our work. In this 2008 paper the authors begin by measuring several graph metrics such as the degree distribution, the evolution of clustering coefficients, the distribution of the time between the creation of two edges, the arrival rate of nodes and edges and many more. Whenever possible they build analytical models for the metrics measured. Some features, such as nodes arrival rate, turned out to be more complicated to model as their evolution depends very much on the graphs. Building up on these metrics the authors presented a full algorithm for evolving a graph. The particularity of their method is that the evolution is modelled at a microscopic level. Specifically each node is assigned a few properties such as a lifetime (time between its arrival and the creation of its last node) or time between the creation of each of its nodes. And as the time passes, these values are used to model the probability that a node adds an edge to the graph. The decisions to add an edge are made independently for each node instead of following a general law for the whole graph.

Though being of utmost interest for our work, this paper had the drawback of not analyzing the performances of the model. Moreover, when a node adds an edge to the graph it randomly

Feature	Value
N	295051
N_u	252898
N_b	42153
E	1125458
ρ	1.11
α_u	1.75
α_b	2.17

Table 1: A few network statistics : N is the number of nodes, N_u is the number of users, N_b the number of businesses, E is the number of edges, ρ is the densification exponent $E \propto N^\rho$, α is the degree exponent in the degree distribution $p(d) \propto d^{-\alpha}$, α_u and α_b are the degree exponents in the degree distributions of the user and business nodes respectively

chooses among nodes 2 hops away, the destination node for the edge. In a bipartite graph this strategy would not be relevant.

These two flaws in Leskovec et al. motivated the approach that we describe in the next section.

3 Approach

3.1 Dataset

We will be using the Yelp Academic Dataset[1] making our nodes `user` and `business`, an edge representing a `review`. The data is already curated and accessible so no work has been done on collecting/gathering. Our graph contains $N = 295051$ nodes and $E = 1125458$. Table 1 presents even more network statistics about the graph.

In our final graph, each node is attributed a type and a creation timestamp. Similarly edges are attributed a timestamp matching the review creation date. It is important to note that reviews can be updated on Yelp. To simplify our graph we decided to discard any review update.

We created a graph generator that yields snapshots of the graph for each given month. We will be leveraging this code to compute metrics on the evolution of the graph at the month level.

3.2 Statistics

In this paragraph we will look closer to some of the network statistics in table 1.

3.2.1 Degree distribution

Unremarkably, our graph follows a power law degree distribution. Indeed, despite a slightly off degree trend for businesses, we can see on figure 1 that Yelp user-business graph follows a power law. From an internal source from Yelp we know of two strategies of Yelp that probably explain the distribution for businesses. On the one hand users are encouraged to review businesses that have not yet been reviewed by being rewarded with special badges. They are also guaranteed that their reviews will be seen first in this case. Moreover when recommending businesses to users, Yelp gives priority to businesses that they have not yet reviewed.

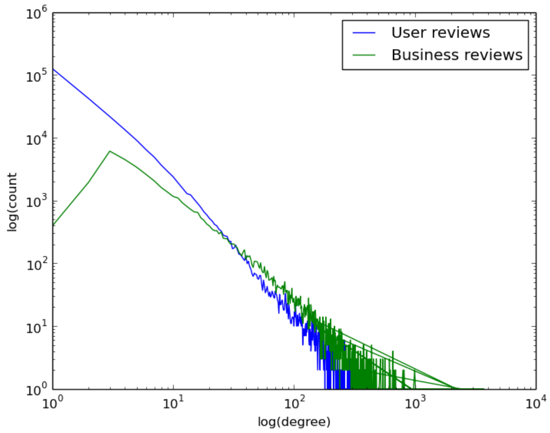

Figure 1: Power law degree distribution

3.2.2 Nodes and edges arrival

We know from Leskovec et al.'s work that the node arrival rate $N(t)$ varies greatly from one graph to the other and follows little pattern. From a simple analysis presented on figure 2 we see that $N_u(t) \propto t^{\gamma_u}$ and $N_b(t) \propto t^{\gamma_b}$. This results in $N(t) \approx \beta_u t^{\gamma_u} + \beta_b t^{\gamma_b}$. However this equation does not yield a very good approximation of $N(t)$. We believe that the estimator $\hat{N}(t) = \beta_u t^{\gamma_u} + \beta_b t^{\gamma_b}$ performs poorly because it assumes that there are only two types of nodes : users and businesses. However in reality there are several types of nodes: male users,female users,users in America, users in Europe, users living in California,restaurants, cinemas and many more. If we assume that each of these nodes in each of these categories follows the law $N_i(t) = \beta_i t^{\gamma_i}$ then

$$N(t) = \sum_{i=1}^{\infty} \beta_i t^{\gamma_i}$$

. The values of γ_i being unknown we can make a simple assumption that they are random variables

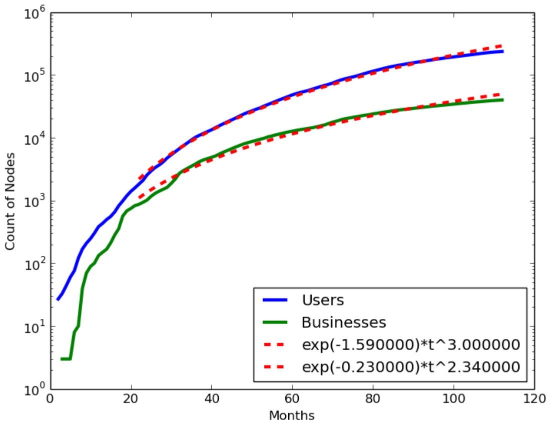

Figure 2: Node counts with time

following a distribution $\gamma_i \sim \mathcal{N}(\mu_N, \sigma_N^2)$. This gives a closed form for the nodes arrival rate

$$N(t) = \beta_N \, t^{\mu_N} \, \exp\left(\frac{\sigma_N^2 \log^2 t}{2}\right) \qquad (1)$$

as proven in theorem 1

Theorem 1. *Let's assume that nodes in Yelp's user-business review bipartite graph can be broken down in several independent groups each having a node arrival rate $n_i(t) = \beta_i t_{\gamma_i}$ where $\gamma_i \sim \mathcal{N}(\mu_N, \sigma_N^2)$ and $\sum_{i=1}^{\infty} = \beta_N$. Then the expected number of nodes Yelp's user-business graph is*

$$\begin{aligned} N(t) = \mathbb{E}[n(t)] &= \mathbb{E}\left[\sum_{i=1}^{\infty} \beta_i t^{\gamma_i}\right] && (2) \\ &= \sum_{i=1}^{\infty} \beta_i \mathbb{E}\left[t^{\gamma_i}\right] && (3) \\ &= \beta_N \, t^{\mu_N} \, \exp\left(\frac{\sigma_N^2 \log^2 t}{2}\right). \end{aligned}$$

Proof. Let X be a random variable following a distribution $\mathcal{N}(\mu, \sigma^2)$. The it's moment generating function $M_X(t)$ is known to be

$$\begin{aligned} M_X(t) &= \mathbb{E}\left[e^{tX}\right] \\ &= \exp\left(\mu t + \frac{\sigma^2 t^2}{2}\right) \end{aligned}$$

Now let's observe that

$$\begin{aligned} \mathbb{E}\left[t^{\alpha_i}\right] &= \mathbb{E}\left[e^{\alpha_i \log t}\right] \\ &= M_{\alpha_i}(\log t) \\ &= \exp\left(\mu \log t + \frac{\sigma^2 \log^2 t}{2}\right) \\ &= t^{\mu} \exp\left(\frac{\sigma^2 \log^2 t}{2}\right). \end{aligned}$$

Plugging this result back into equation 3 gives the desired result. □

The same analysis about the evolution of edges, lead us to make the same assumptions about the edge arrival rate $E(t)$. As we can see on figures 3 and 4 the resulting models predict quite well the evolution of the graph.

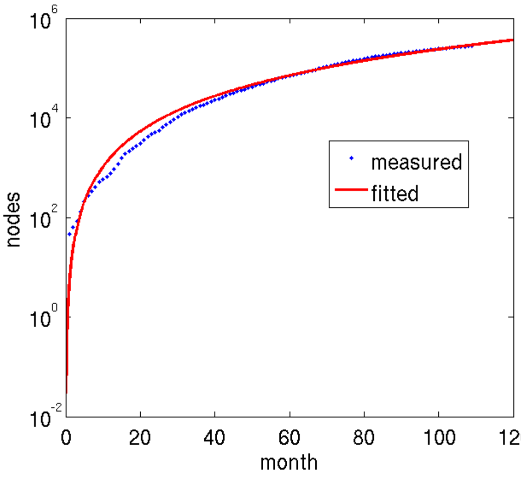

Figure 3: Approximation of the nodes evolution

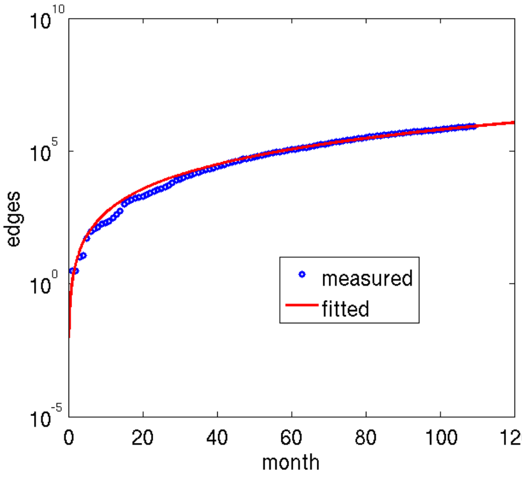

Figure 4: Approximation of edges evolution

Contrary to what the theory tells us $\lim_{t\to\infty} \frac{\log E(t)}{\log N(t)}$ is not constant for this graph as illustrated on figure 5. However if we fit the model in equation 1 to the edges and nodes evolution of the graph. The estimated densification exponent $\hat{\rho}(t) = \log \hat{E}(t) / \log \hat{N}(t)$ closely matches the observed values. In addition to having an estimation of the evolution of the number of nodes in the graph, it is important to know what fraction of these nodes are users, and which fraction are businesses. One approach would be to fit the model described in equation 1 to the number of user and business nodes. We found an alternative method by looking at the experimental values of $\log N_u(t)/\log N_b(t)$. Indeed we observed that $N_u(t) \propto (N_b(t))^{\alpha_{ub}}$ with $\alpha_{ub} \simeq 0.85$. Therefore another approach in finding $N_u(t)$ and $N_b(t)$ is to find the root of $f(N_u(t)) = N_u(t) + N_u(t)^{\alpha_{ub}} - N(t)$. In our final algorithm we chose to use this approach as it can be easily adapted to a situation in which a better model for $N(t)$ is found.

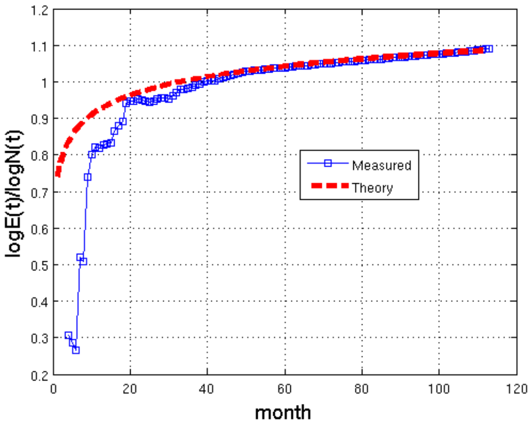

Figure 5: The densification exponent of the graph evolves over time as more and more *communities* join the Yelp user and business base.

3.2.3 Link creation

We have explored the rates at which nodes and edges arrive in the network, but we need to model how the links between the nodes happen. For unipartite graphs, it is a well known fact that nodes create links to other nodes following a preferential attachement (PA) model. By definition of this model, the probability of an edge linking to a node is proportionnal to the degree of the node. However the experimental results tell a completely different story for the Yelp graph. Indeed as illustrated on figure 6 the probability that a link created links to a user or business of degree d is a decreasing function of the node degree. This is due to the fact that in this type of network, it is much more valuable for a user to be the first one to review a business because on the one hand Yelp gives *badges* to first reviewers of businesses and on the other hand there is probably a pleasure in being the first one to give a review. This explains the trend for the businesses. As for the users, new comers are still very motivated to review businesses but as time passes, the dedication might fade off, hence the decreasing trend for the users as well.

The distributions in figure 6 do not follow any power law model that we could find. In our simulations we therefore chose to use the experimental values.

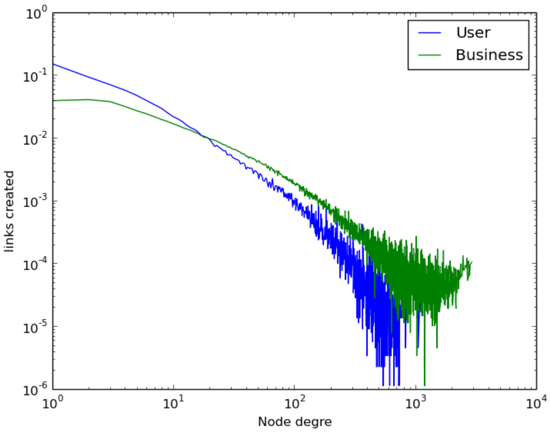

Figure 6: On the contrary of social networks, the probability of edge linking to a node of degree d does not increase as function of the node degree.

In the process of understanding the underlying phenomenon under which links are created it is also important to know the time between two review creation and the time between the first and last reviews of a user. Let $t_0(u)$ be the time when node u was created and $t_i(u)$ the time when the i-th review of u was created:

- The *node lifetime* $a(u) = t_{deg(u)}(u) - t_1(u)$ is defined as the time between the node creation and the last edge that it created.
- The *time gap* $\delta^{(i)}(u) = t_i(u) - t_{i-1}(u)$ between edge creation is defined as the time between the creation of two edges of a node.

We computed these metrics for both types of nodes and the results are summarized on figures 7 and 8. We obtain a decreasing distribution for the lifetimes of user nodes which seems to follow a power-law with exponential cut-off. On the other hand the distribution for the business type seems to be flat for the most part.

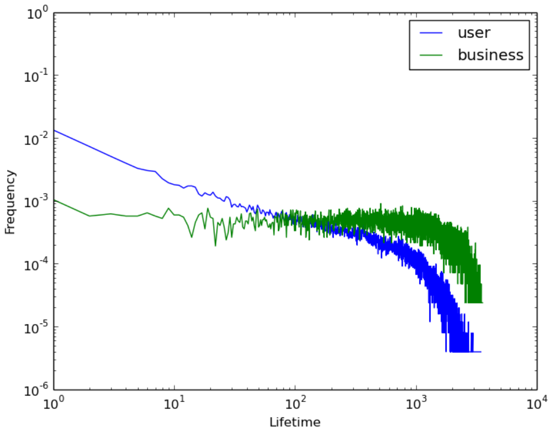

Figure 7: Lifetime (a) distribution

Similarly, the distribution for the time gap users has the shape of a power law with exponential cut-off. The small irregularities in the curve (small spikes) are nicely mapping to 7 days gap which is fully relevant because users tend to review more during weekends.

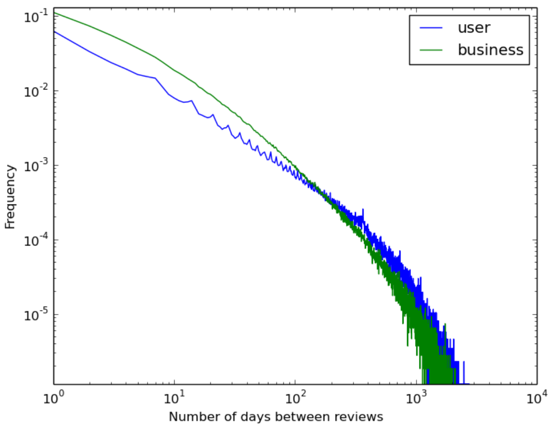

Figure 8: Days between reviews (δ) aggregated distribution

Finally we can note that the edge creation also follows an law decreasing faster than a power-law. Since we know that models based on preferential attachment are generating similar distributions we will try to derive an approach that explains this distribution.

4 Proposed Algorithm

We propose the following algorithm to model the graph evolution

1. Get the number of new nodes using $N(\cdot)$
2. User node u arrives and samples $a(u)$ according to the distribution fitted to the results on figure 7
3. User node u links to business node b with probability proporitional to its degree (preferential attachment)
4. Node u samples a time gap $\delta(u)$ following a distribution fitted to the data on figure 8 and goes to sleep during this period.
5. A node wakes up when $\delta(u)$ iterations of the algorithm have elapsed and its lifetime is below the total number of steps done by the algorithm.
6. When a node u wakes up, it links to a business following a `random`[3] process then goes back to step 4.

The `random`[3] is a simple random markov decision process strategy in which a node chooses one of it's neighbors at random and keeps doing so starting at this neighbor three times. We believe that this strategy will still give us a power law distribution but will also keep the local substructure of the graph.

It is important to note that our businesses are completely passive during this process. This makes sense given the fact that this is also the case in real life. Furthermore, we are losing the locality of the edge creation that makes less sense in our context. In fact, using user to user distances seem to have fail to improve state of the art algorithm[2].

5 Evaluation

5.1 Method

Let T be the number of time periods in which a graph has evolved. We define G_T as the full graph that we are trying to model, and $G_{T/2}$ as the graph at time $T/2$. Our evaluation method will consist of the following process: take $G_{T/2}$ and make it evolve following the proposed algorithm for $T/2$ time step, call the resulting graph G'_T. We will then compare the different metrics of G_T and G'_T to figure out the quality of our algorithm.

5.2 Parameters

To obtain the full algorithm, we need to estimate the distributions introduced in point 2. (lifetime a of a node) and point 4. (time gap between reviews $\delta(u)$).

We modeled the lifetime as a power law distribution $p(a;\lambda) \propto a^{-\lambda}$ since the values in the tail of figure 7 are matching to the maximum value of T. When estimating the exponent of the power law we obtained the value of 0.70 using a maximum likelihood estimator on a truncated distribution to avoid the artifact tail.

As for the parameter δ, we reached to the same conclusion as the authors of [3]. Not only are the time gaps following a power law with heavy tail, but the tail is heavier when the degree of the node increases $p(\delta, deg(u); \alpha, \beta) \propto \delta^{-\alpha}\exp(-\beta\delta deg(u))$. An other way of saying this is that the more a user has reviews, the sooner he will review another business. We obtained the values of $\alpha = 0.80$ and $\beta = 0.00017$ using a maximum likelihook estimator.

6 Results

A first important metric is edge creation overtime. We expect our model to perform decently until a certain point in time in which the parameters governing dynamics of the network might have changed too much. In the following graph, the simulation of the graph evolution kicks in at month 20.

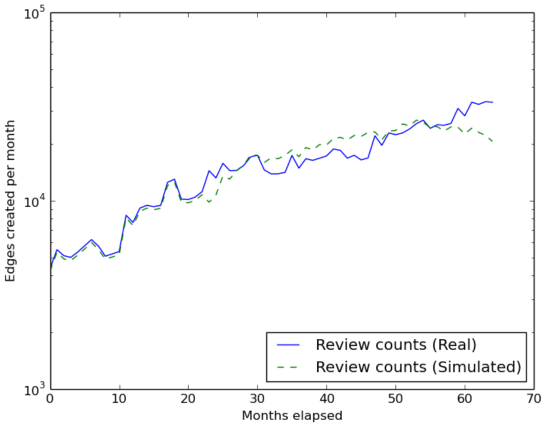

Figure 9: Edge creation per month

We can see that the model performs really well until the last six months where it starts to fall behind. This tells us that the fitted parameter for node arrival, and review creation were initiated reasonable values.

On the disappointing side, the densification exponent evolution of our graph has sees a noticeable gap right at the time when the algorithm starts simulating. This indicates that our model does not perform well in terms of degree distribution preservation. However, the curves still seem to follow a dynamic similar to the real graph (a slow linear decrease).

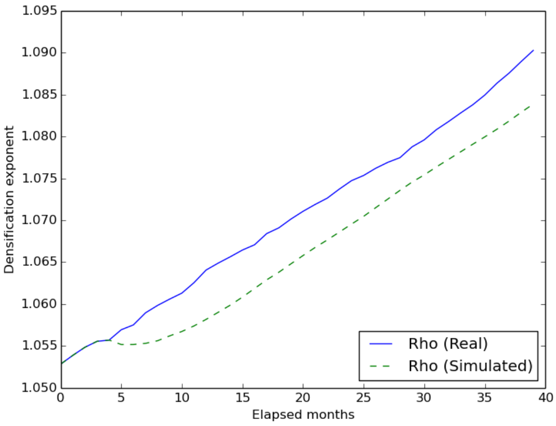

Figure 10: Densification exponent with time

Following this result it is now worth looking at the degree distribution. If we consider the degree distribution to follow a power law and we plot its parameter over time we notice an even bigger discrepancy between the simulated graph and the real graph. (note: in 11, the alpha indicated in the legend is not linked to the parameter alpha of the time gaps introduced previously). From this we conclude that the `random`^{3} strategy to create edges is not adapated to our type of graph.

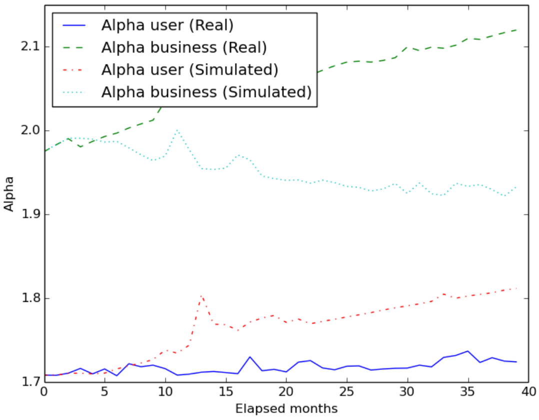

Figure 11: Degree distribution parameter

From these results we can validate our fitted parameters for the lifetime and time gaps distribution but some work remains for choosing how to assign new reviews.

7 Future work

We have clearly identified that the edge creation process following an extended random walk (`random`[3]) is not well suited for bipartite graphs. Therefore, other techniques and distributions could be used and try in point 6. of the algorithm.

The dataset studied has some weaknesses: it is composed of five clusters (cities) that have very few interactions with each other but still enough that it is hard for us to split them appart, we also noticed some data discrepancy and mislabelling which made $\delta^{(0)}$ impossible to derive for example. A good analysis would implement the suggested algorithm on a dataset without too much sparsity with dates at least at a daily granularity.

8 Conclusion

Our study shows that review creation can still be captured by the lifetime and time gap idea. We've also seen that time gaps are decreasing with node degrees which confirms other studdies.

We introduced a generic algorithm for edge creation of bipartite graphs where one type of edge is inactive and the other active. This algorithm could be applied to many different kind of graphs following the pattern (`user` to `object`) which makes it quite powerful.

It has shown reasonable results on the tested dataset as we were able to match the edge creation rate up to 3 years after the starting point of the optimization. However, the last step remains debatable since we are seeing divergence of the degree distribution, although its strenght is that it preserves a local structure of the edge creation that stays contains in a close neighborhood of the starting node.

References

[1] Yelp. *Yelp Phoenix academic dataset*. URL: http://www.yelp.com/dataset_challenge/ (visited on 2014).

[2] Jérôme Kunegis, Ernesto William De Luca, and Sahin Albayrak. “The Link Prediction Problem in Bipartite Networks”. In: *CoRR* abs/1006.5367 (2010). URL: http://arxiv.org/abs/1006.5367.

[3] Jure Leskovec et al. “Microscopic Evolution of Social Networks”. In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD ’08. Las Vegas, Nevada, USA: ACM, 2008, pp. 462–470. ISBN: 978-1-60558-193-4. DOI: 10.1145/1401890.1401948. URL: http://doi.acm.org/10.1145/1401890.1401948.

[4] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. “Graph Evolution: Densification and Shrinking Diameters”. In: *ACM Trans. Knowl. Discov. Data* 1.1 (Mar. 2007). ISSN: 1556-4681. DOI: 10.1145/1217299.1217301. URL: http://doi.acm.org/10.1145/1217299.1217301.

[5] G Kossinets and D Watts. “Empirical Analysis of an Evolving Social Network”. In: *Science* 311.5757 (2006), pp. 88–90.

[6] A. L. Barabasi et al. *Evolution of the social network of scientific collaborations*. 2002.