Relational Classification Through Three–State Epidemic Dynamics

Aram Galstyan
Information Sciences Institute
University of Southern California
Marina del Rey, CA, U.S.A.
galstyan@isi.edu

Paul Cohen
Information Sciences Institute
University of Southern California
Marina del Rey, CA, U.S.A.
cohen@isi.edu

Abstract - ***Relational classification in networked data plays an important role in many problems such as text categorization, classification of web pages, group finding in peer networks,*** *etc.* ***We have previously demonstrated that for a class of label propagating algorithms the underlying dynamics can be modeled as a two-state epidemic process on heterogeneous networks, where infected nodes correspond to classified data instances. We have also suggested a binary classification algorithm that utilizes non–trivial characteristics of epidemic dynamics. In this paper we extend our previous work by considering a three–state epidemic model for label propagation. Specifically, we introduce a new, intermediate state that corresponds to "susceptible" data instances. The utility of the added state is that it allows to control the rates of epidemic spreading, hence making the algorithm more flexible. We show empirically that this extension improves significantly the performance of the algorithm. In particular, we demonstrate that the new algorithm achieves good classification accuracy even for relatively large overlap across the classes.***

Keywords: Relational learning, binary classification.

1 Introduction

Building career histories of scientists is a challenging task that requires fusing information from various heterogeneous sources, finding publication patterns, tracking changes in these patterns through time, *etc.* Another relevant problem is disentangling different threads or topics in a publication history of a scientist. This would be trivial if each publication contained a keyword and/or category label describing the thread, such in the Cora data–set of categorized computer science papers [6]. However, many databases lack such information or provide only partial categorization of publications. Moreover, different databases might describe the same topic using slightly different keywords, so one has to account for this ambiguity while combining records from several sources.

From the perspective of relational learning, the thread disentanglement problem can be reduced to classifying publications into one of several categories, given relational data that describes papers and relationship between them. In contrast to traditional machine learning where data instances are assumed to be independent and identically distributed, relational learning techniques explicitly take into account interdependence of various instances. This allows them to make inferences based on not only intrinsic attributes of data but also its relational structure, thus enhancing inference accuracy. To illustrate the potential advantages of relational learning over more traditional approaches, consider the problem of categorizing the publication record of a scientist into topics. One can represent the relational database as a graph where each paper is represented by a node, and links between two nodes describe relationships between corresponding papers. Examples of such relationships include common authors, shared references, cross–reference, etc. Consider now a particular paper in this relational structure by a certain author. Even if this paper does not have a keyword, one might still be able infer its category by looking at the category labels of the papers that it is most strongly connected with. Indeed, recently a number of authors[10, 5, 1] have successfully used relational learning algorithms for classifying papers in CORA data–set [6] into topics. Other relevant applications of relational learning techniques include hypertext classification [4], link prediction[11], classification of web pages [9, 5], studying relational structures in scientific publications [7], etc.

Most relational learning algorithms are iterative and work though propagating either class labels or corresponding probabilities [8, 5]. One important issue with iterative classifiers is that false inferences made at some point in iteration might propagate further causing an "avalanche" [8]. Hence, it is very desirable to have some indicators showing when this happens. In our previous work we have shown that for a class of label propagating algorithms such an indicator can be obtained by looking at the dynamics of newly classified instances [1, 2]. We have demonstrated that these dynamics can be modeled as an epidemic spreading in heterogeneous population. Furthermore, if the coupling between two sub–populations is sufficiently weak, then the epidemics has a non-trivial two–tier structure. This is due to the fact that *true* class labels propagate at faster rate than $false$ ones. We have also indicated

how to use this dynamical signature for obtaining a robust and virtually paramete–free classifier.

Although the two–tier based classifier works well for relatively weak coupling between the classes, its performance deteriorates drastically with increasing the overlap between them. This is because for large overlap there is a high probability that the infection will spread outside the class at the early stages of iterative process. In this paper we address this shortcoming by adding a new, intermediate state and extending the analogy of label propagation scheme to an epidemic system with three states: "healthy", "susceptible", and " infected". Initially all the nodes (e.g., data instances) are in the "healthy" state, except the A nodes with known class labels that are in the "infected" state. At each iteration, a node will become susceptible if it is connected to super–threshold number of infected nodes. In contrast to the previous algorithm, however, not all susceptible nodes will make a transition to "infected" state at once. Instead, at each time step only a certain fraction of susceptible nodes will actually become infected. The main utility of the added state is that it slows down the rate of epidemic spreading, hence allowing to control the spread of infection from one sub–population to the other. We demonstrate that this added flexibility provides significant improvement over the original algorithm. In particular, we present empirical evidence that the new algorithm consistently outperforms the previous one, and achieves a good classification accuracy even when the overlap across two classes is relatively large.

2 Problem Settings

Generally speaking, relational learning algorithms use both intrinsic and relational characteristics of the data for inference. In this paper, however, we will neglect any intrinsic attributes and concentrate on solely relational aspect of classification. In this context, the relational data–set can be represented as a graph, where nodes represent data instances, and edges (possibly weighted) describe relationship between them. For instance, in CORA data–set of categorized computer science papers [6], each node represents a paper, while a link between two papers describes their relationship (e.g., common authors, cross–references, etc.). The main assumption of relational classification on such data is *homophily*, i.e., the notion that the data instances that are similar tend to be better connected (e.g., for the CORA data–set the homophily assumption means that papers that share authors and/or common references are likely to be similar).

Throughout this paper we will evaluate our algorithms on synthetic data–sets as one schematically depicted in Fig. 1. Namely, each data instance belongs to one of two possible classes, A and B, with N_a and N_b nodes in each class, respectively. These two classes are characterized by two loosely coupled subgraphs in Fig. 1. Initially we are given the class labels of small fraction of data instances of type A (red nodes). The problem is then to find other members of A based on the pattern of links in the graph. The graph itself is constructed as follows. Within each group, we randomly establish a link between two nodes with probability $p_{in}^{a,b}$. Probability of a link between two nodes across the groups is p_{out}. We also define average connectivities between respective types of nodes, $z_{aa} = p_{in}^{a} N_a$, $z_{bb} = p_{in}^{b} N_b$, $z_{ab} = p_{out} N_b$ and $z_{ba} = p_{out} N_a$. Note that generally speaking $z_{ab} \neq z_{ba}$ if the sizes of two groups are not equal, $N_a \neq N_b$. In this paper, we will characterize the intra– and inter–group connectivities through $z_{aa} = z_{bb} \equiv z_{in}$, and $z_{ab} \equiv z_{out}$. Clearly, the ratio z_{out}/z_{in} characterizes the difficulty of the classification task. For small values of this ratio, the coupling between two sub–graphs is weak so most classification algorithms should do a good job of assigning correct class labels. If one increases this ratio, however, then the difference between in– and out–group link patterns decreases, hence making it difficult to classify nodes correctly.

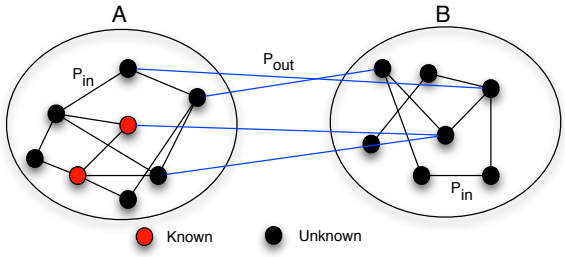

Figure 1: Schematic representation of relational data–set

3 Binary Classification Through Two–Tier Dynamics

The idea behind iterative classification is that inferences made at some point might be used for drawing further inferences. Although iterative classifiers are superior to one–shot classification techniques, there is a certain risk involved. Indeed, if one makes incorrect inferences at some point then there is a possibility that it will "snowball" and cause an avalanche of further incorrect inferences [8]. Moreover, since most of the algorithm rely on parameters, then the issue of parameter sensitivity becomes very important. Indeed, if small adjustment in parameters result in even a small number of incorrect inferences, then there is a chance that the iterative procedure will propagate these erroneous inferences further, causing instabilities. Hence, it is very important to be able to detect such instabilities, and prevent them from happening.

In our previous work we have suggested heuristics for detecting such undesired avalanches [1, 2]. This is done by looking at the dynamics of newly classified instances. The key for detection the instability is a phenomenon that we call two–tier dynamics. Namely, we characterize the dynamics of an iterative classifier by fraction of newly classified instances at each time step., e.g., if $\rho(t)$ is the fraction of classified instances at

time t, then the relevant variable is $\Delta\rho(t) = \rho(t) - \rho(t-1)$. As it will be clear later, two–tiered dynamics arises whenever $\Delta\rho(t)$ has two temporally separated peaks, that characterize epidemic spreading in separate sub–populations of nodes.

To be concrete, let us consider a relational data–set where each data instance belongs to one of two classes A and B, as one depicted in Fig. 1 . We assume that the relational structure is fully characterized by the adjacency matrix M so that the entry M_{ij} describes the strength of the relationship between the i–th and j–th instances. Our algorithm relies on a threshold to decide when to propagate labels. Namely, a node will be classified as type A if it is connected to super–threshold number of type A nodes. Let us associate a state variable with each node, $s_i = 0, 1$ so that the state value $s_i = 1$ corresponds to type A. Initially, only the nodes with known class labels have $s_i = 1$ while the rest are in state $s = 0$. At each iteration step, for each non–classified node we calculate the cumulative weight of the links of that instance with known instances of type A. If this cumulative weight is greater or equal than a certain threshold H , that node will be classified as a type A itself. The pseudo–code for this iterative scheme is shown in Fig. 2. Note that this mechanism asymmetric in the sense that if a node was classified as type A it will remain in that class until the end of iteration. This implies that the total number of classified A node is a monotonically non–decreasing function of time. If the duration of iteration, T_{max}, is sufficiently

input adjacency matrix $\mathbf{M}$
initialize $s_i = 1$, for initially known instances,
initialize $s_i = 0$ for unknown instances
initialize a threshold H
iterate $t = 0 : T_{max}$
    for i–th node with $s_i(t) = 0$
        calculate $w_i = \sum M_{ij} s_j(t)$
        if $w_i \geq H \Rightarrow s_i(t+1) = 1$
    end for loop
end

Figure 2: Pseudo–code of the iterative procedure

long, then a steady state will be achieved, e.g., none of the nodes changes its state upon further iterations. Obviously, the final state of the system will depend on the threshold value and the graph properties as characterized by adjacency matrix. Specifically, if the threshold H is set sufficiently low then the system will evolve to a state where every node is in state $s = 1$, i.e., every instance has been classified as type A. On the other hand, if threshold is too high, then no additional node will change its state to $s = 1$ at all.

Before proceeding further, we note that our iterative procedure can be viewed as an epidemic process on a network. Indeed, let us treat the initially known data instances of type A as "infected". Then the dynamical scheme above describes an epidemic spreading throughout the network. If there were no links between data instances of different type, then clearly the epidemics would be contained in the A sub–population. Hence, one could relax the classification criterion by reducing the threshold H so that all the data instances of type A will be infected, i.e., correctly classified. If there is a non–zero coupling between two classes, however, then there is a chance that the epidemic will "leak" to the second sub–population too, hence causing an avalanche of wrong inferences. Our main observation is that if the coupling is not very strong, then one can choose a threshold value so that the epidemic spreading in separate sub–populations is well–separated in time. In other words, the epidemic infects most of the nodes in population A before spreading through the second population. Then, one can look at the dynamics of newly classified instances and detect the onset of the avalanche.

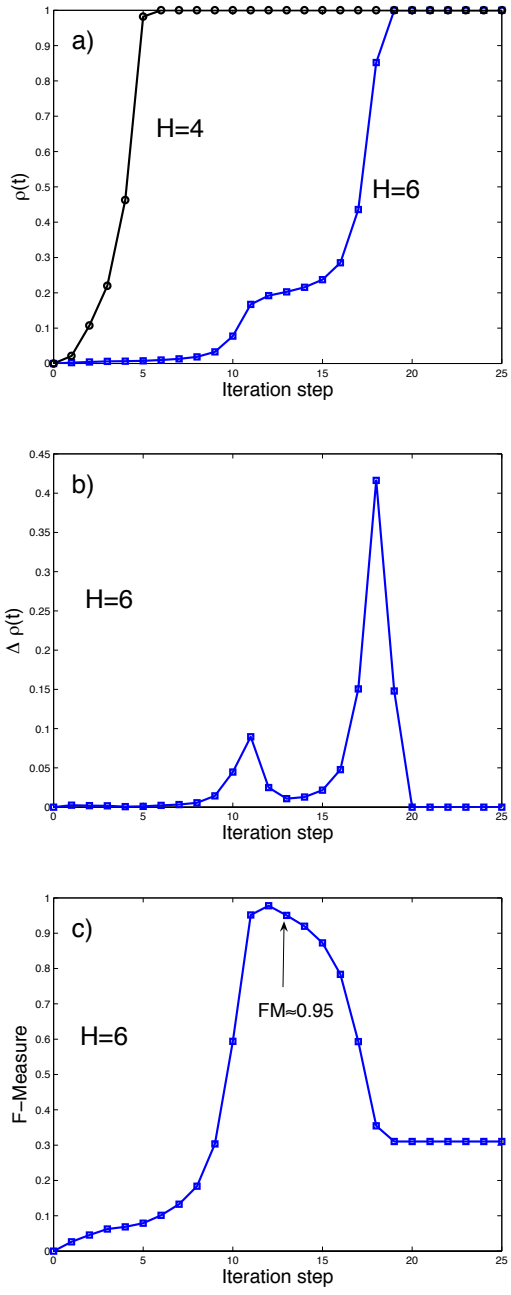

Figure 3: Simulation results for (a)$\rho(t)$ and (b) $\Delta\rho(t) = \rho(t) - \rho(t-1)$ for a random network.

To demonstrate this, in Fig. 3 we present the results of the iterative procedure on randomly generated graphs for the same network parameters but two different values of the threshold parameter. The parameters of the network are $N_a = 1000$, $N_b = 4000$, $z_{aa} = z_{bb} = 20$, $z_{ab} = 8$. Initially, only 10% of A nodes are classified. For $H = 4$ all of the nodes are classified as type A after a short period of time. For $H = 6$, however, the dynamics is drastically different as it has a distinctly bimodal shape. Indeed, after a sharp initial spread the dynamics seems to be saturating around $t = 13$. However, upon further iteration the number of classified nodes increases rapidly and after short transient all the nodes are infected. Clearly, this corresponds to the onset of the "avalanche" where certain wrong inferences propagate and infect rest of the system. This is especially clear in Fig. 3 (b) where we plot the the fraction of newly classified instances vs. time, and observe two well separated maxima, which characterize the peak infection rates in respective population.

Note that this bimodal shape suggest a natural criterion for stopping the iteration. Namely, the iteration should be stopped when the second peaks starts to develop, e.g., before infection starts to spread into the second population. Indeed, in Fig. 3 (c) we plot the F measure of classification accuracy vs. iteration step. One can see that at the point where the second peak starts to develop $F-Measure \approx 0.95$, which is slightly less than the maximum value 0.97.

We now consider the effect of the overlap between two populations by increasing the inter–group connectivity z_{ab}. As we mentioned before, increasing the overlap should make the classification task more difficult. In particular, we expect that for large z_{ab} the two–tier dynamics should be less pronounced. Indeed, in Fig. 4 a) we plot the fraction of newly infected nodes for $z_{ab} = 12$ and $z_{ab} = 16$. One can see that for $z_{ab} = 12$, although there are still two peaks, the separation between them has decreased drastically. Increasing z_{ab} further leads to gradual disappearance of two–tier structure, as it shown for $z_{ab} = 16$. In the terminology of epidemic dynamics, this is due the fact that for large inter–group connectivity the epidemic starts to spread into the B nodes before infecting majority of A nodes.

To explain this phenomenon, we now provide a qualitative assessment of two–tier dynamics (a more detailed study will be presented elsewhere [3]). Let us first consider epidemic spreading in a single population, e.g., among A nodes, and neglect inter–group links. It can be demonstrated that for a fixed fraction of initially infected nodes, there is a critical intra–group connectivity z_{in}^c so that for $z_{in} < z_{in}^c$ the epidemic will be contained within a small fraction of nodes, while for $z_{in} > z_{in}^c$ it will spread throughout a system. Put conversely, we can say that for any fixed connectivity z_{in}, there is a critical fraction of initially infected nodes ρ_0^c so that for $\rho_0 > \rho_0^c$ the epidemic will spread globally. We also note that at the critical point, the transient time (i.e., time to reach the steady state) of the epidemic process diverges.

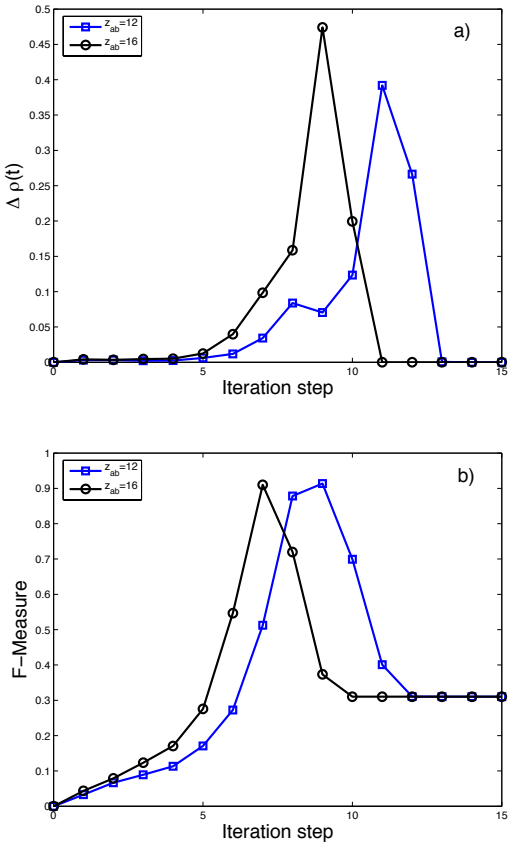

Figure 4: (a)The fraction of newly classified nodes and (b) F measure vs. iteration step for $z_{ab} = 12$ and $z_{ab} = 16$.

Now let us consider the full system with two populations. Let us again assume that all initially infected nodes are contained in the A population. For sufficiently weak coupling between the groups, the the epidemic dynamics among A nodes will be virtually unaffected by the B nodes. In particular, whether the epidemic will infect all of the A nodes will depend on the fraction of initially infected nodes and the connectivity z_{in}. Assume that these parameters are chosen such that the epidemic process indeed infects all of the A nodes. Let us now ask under what conditions the infection will spread through B population. It is easy to see that the infected A nodes play the role of infection "seeds" for B nodes. Moreover, the effective fraction of these seed nodes depend on the inter–group connectivity z_{out}. Hence, by extending the reasoning about critical phenomenon to B nodes, one can demonstrate that for a given fraction of infected A nodes ρ_a and an intra–group connectivity z_{in}, there is a critical inter–group connectivity $z_{out}^c(\rho_a, z_{in})$ so that for $z_{out} > z_{out}^c$ the infection will spread globally to B nodes. Assuming that z_{in} is fixed, this relation is characterized by a critical line in $z_{out} - \rho_a$ plane. Now we can assess when the two tier–dynamics will be most pronounced. Indeed, if we take $\rho_a = 1$ (meaning that all A nodes have been affected) then the maximum separation between two activity peaks will be infinite for $z_{out} = z_{out}^c(1, z_{in})$,

since the transient time of epidemic among B nodes diverges at the critical point. Moreover, for values of z_{out} slightly above the critical value, one should still expect a significant separation between two peaks.

We now consider the effect of increasing the inter–group connectivity z_{out}. Clearly, this will decrease the critical fraction of A nodes for which the epidemics will spread to B nodes. Specifically, let us define $\rho_a^c(z_{out})$ as the minimum fraction of infected A nodes required to cause a global epidemic among B nodes. In other words, for a fixed z_{out} the infection will spread to B population only after infecting the fraction $\rho_a^c(z_{out})$ of A nodes. Now, if this fraction is close to 1, then the epidemic will spread to B nodes only after infecting the majority of A nodes, hence, giving rise to two–tier dynamics. On the other hand, if this fraction is considerably less than one, the the epidemic will leak to B nodes prematurely. To be more precise, consider again Fig. 3 (b). If at the height of the first peak the density of infected A nodes is considerably greater than the threshold $\rho_a^c(z_{out})$, then at the next iteration there will be a large number of infected B nodes . As a consequence, the fraction of newly infected nodes will still increase, and there will be no two–tier dynamics.

It is worthwhile to note that even in the absence of two–tier dynamics, our main assumption that *true* class labels propagate faster than the $false$ ones still holds to some degree. Indeed, in Fig. 4 we plot the F measure vs. iteration step, and note that it still attains a maximum value that is close to 0.95. Hence, if we somehow knew where to stop the iteration, we would still be able to obtain good classification accuracy. The problem is, however, that without a clear two–tier signature we do not have any criterion as when to stop the iteration.

4 Three–State Epidemic Model for Label Propagation

As we explained above, the two-tier dynamics is not present for sufficiently strong coupling between two populations. At the same time, we saw that the true class labels still propagate faster than the false ones. This suggests that the two–state epidemic mechanism is somehow rigid as it does not allow one to control the rate of epidemic spreading. Indeed, recall that there is a threshold fraction of A nodes ρ_a^c so that for $\rho_a > \rho_a^c$ the epidemic starts to spread among B nodes. Thus, if we could control the rate of epidemic among A nodes, we should in principle be able to infect up to ρ_a^c fraction of A nodes without worrying that the infection will leak to B nodes . However, in the absence of such a control mechanism, there is a chance that at some point in the iteration the fraction of infected A nodes will "overshoot" this threshold significantly, hence causing epidemic among B nodes.

To account for this shortcoming, we now consider an extension of the previous algorithm by adding another, intermediate state. In the terminology of relational classification this intermediate state describes a node that has been "marked" for infection, but has not been infected yet. We term this intermediate state as "susceptible". At a given iteration step, a node will become susceptible if it is connected to super-threshold number of infected nodes. However, we will now allow only a maximum number, N_{max}, of susceptible nodes to become infected at each iteration step. By choosing N_{max} sufficiently small, we can expect that the fraction of infected A nodes will approach its critical value ρ_a^c smoothly, without the risk of overshooting[1]. Note that in practice this can be done by keeping a queue of susceptible nodes, and processing the queue according to some priority mechanism. The priority mechanism used in this paper is FIFO (first–in–first–out). However, other mechanism can be used too.

Before presenting our results, we now address the question of what kind of criterion should be used for stopping the iteration. Clearly, since at each time step only a handful of nodes are classified as infected, one should not expect any two–tier dynamics in the number of infected nodes. However, as we will see below, the number of newly susceptible nodes does in fact have the two–tier structure, e.g., if $s(t)$ is the fraction of susceptible nodes at time t, then the relevant quantity is $\Delta s(t) = s(t) - s(t-1)$. As in the case of previous two-state algorithm, the onset of the "avalanche" is then indicated by a second developing peak. Once this onset is found (e.g., by finding the iteration step for which the curve has its minimum), then we can backtrack trough class–label assignments, and "un–impeach" nodes that became susceptible after the onset.

To test our new algorithm, we have performed extensive empirical studies of new classification scheme for data-sets of varying overlap. We found that the new algorithm consistently outperforms the old one for all considered data–sets. Even for relatively small overlap across the classes, when the two–tier dynamics in two–state model is present, the classification accuracy of the new algorithm is significantly better. Indeed, in Fig. 5 (a) we plot the time series of F measure for the new classification scheme, for a inter–group connectivities $z_{ab} = 8$ and $z_{ab} = 20$. For $z_{ab} = 8$, the F measure attains a maximum value very close to 1, while for $z_{ab} = 20$ the maximum is close to 0.9. Note that the behavior of both curves is relatively smooth around the maxima. This suggests that even if one stops the iteration within some interval around the optimal stopping time, one can still obtain relatively good classification. Indeed, in Fig. 5 (b) we plot the differential fraction of susceptible nodes, $\Delta s(t)$. Because of large dispersion, we also plot the running averages of each curve using a window of size 20, and use this average for examining two–tier structure. Note that for $z_{ab} = 8$, $\Delta s(t)$ is rather flat for a time interval between $t = 50$ and $t = 80$. Remarkably, we found that even if we choose the stopping time randomly from this interval, the resulting F measure will be contained in the interval $0.985 - 0.99$, which is higher than the F measure

[1] Our experiments suggest that the exact value of N_{max} does not affect the results much, if it is not chosen too high. In the results reported below, we used $N_{max} = 10$.

0.95 achieved by the previous algorithm. Most importantly, however, the new algorithm allows to achieve significant classification accuracy even when the previous two-state scheme fails, e.g., does not demonstrate two–tier dynamics. Indeed, determining the onset for $z_{ab} = 20$ at $t \approx 50$, one is able to achieve an F measure around 0.85.

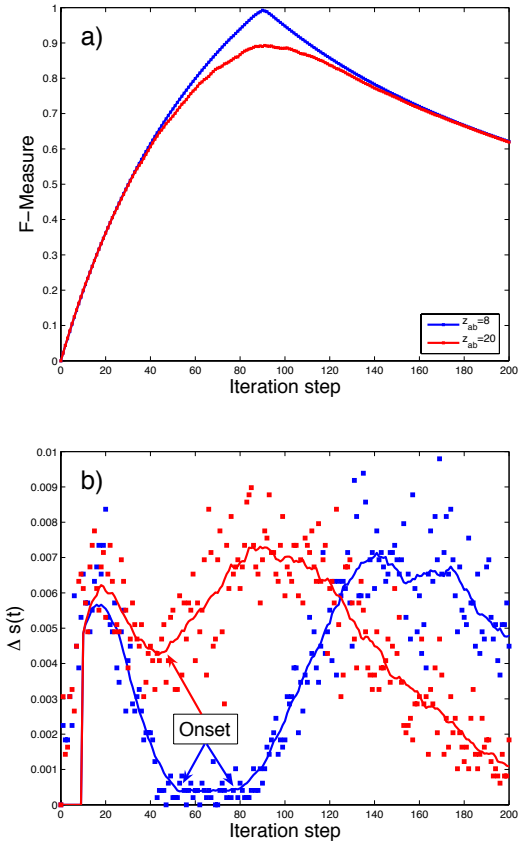

Figure 5: The F measure (a) and the fraction of newly "susceptible" nodes (b) vs. iteration step for $z_{ab} = 8$ (blue) and $z_{ab} = 20$ (red). The solid line in (b) is the moving average of respective scatter–plots using a window of size 20

.

5 Conclusion and Future Work

In conclusion, we have presented a relational classification algorithm based on a three state epidemic process. This algorithm extends our previous two–state model by by adding a new, intermediate state. The addition of this state allows us to control the rate of epidemic spreading across the networked data, hence preventing premature leak of epidemic into the second sub–population. Our experiments demonstrate that this extension improves significantly the performance of the algorithm. In particular, the algorithm is remarkably accurate even when the overlap between two classes in relational data is relatively large.

As future work, we intend to test different priority mechanisms while processing the queue of susceptible nodes. For instance, one could define a priority scheme that depends on the degree of the nodes. The reason for this is as follows: if a certain nodes has super–threshold number of links to infected nodes, but has less links in total as other susceptible nodes, then it should be a better candidate for classification. We do not think that such a priority mechanism would make difference on the empirical studies on random graphs presented here. Indeed, the number of links of a node within and outside of a group are uncorrelated by our construction. However, this mechanism might be important in other scenarios where such a correlation exists.

We also intend to validate our algorithm on real world data–sets. We have previously demonstrated that the two–tier dynamics based algorithm performs well on certain subtopics from CORA archive of categorized computer science papers. However, one of the issues with the previous algorithms was that it did not perform well for classes with large number of members. The reason for poor performance is that for a large class the chances that the epidemic will leak outside before infecting the correct class instances are greater. Since our new algorithm allows better control over the rate of epidemic spredading, we we expect that it will perform well on large classes too.

References

[1] A. Galstyan and P. R. Cohen, "Inferring Useful Heuristics from the Dynamics of Iterative Relational Classifiers," International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, (2005).

[2] A. Galstyan and P. R. Cohen, "Identifying Covert Sub-Networks Through Iterative Node Classification", in Proc. of International Conference on Intelligence Analysis, McLean, VA, (2005).

[3] A. Galstyan and P. R. Cohen, "Global Cascades in Modular Networks", working paper, (2006).

[4] L. Getoor, E. Segal, B. Taskar, and D. Koller, "Probabilistic models of text and link structure for hypertext classification", In IJCAI Workshop on Text Learning: Beyond Supervision, 2001.

[5] S. A. Macskassy, and F. J. Provost, "A Simple Relational Classifier", Workshop on Multi-Relational Data Mining in conjunction with KDD-2003, (2003).

[6] Andrew McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore, "Automating the construction of internet portals with machine learning", *Information Retrieval Journal*, 3:127–163, 2000.

[7] A. McGovern, L. Friedland, M. Hay, B. Gallagher, A. Fast, J. Neville, and D. Jensen, "Exploiting relational structure to understand publication", 2003.

[8] J. Neville and D. Jensen, "Iterative classification in relational data", Proc. AAAI-2000 Workshop

on Learning Statistical Models from Relational Data, pages 13–20. AAAI Press, 2000.

[9] Seán Slattery and Mark Craven, "Discovering test set regularities in relational domains", In Pat Langley, editor, *Proceedings of ICML-00, 17th International Conference on Machine Learning*, pages 895–902, Stanford, US, 2000. Morgan Kaufmann Publishers, San Francisco, US.

[10] B. Taskar, E. Segal and D. Koller, "Probabilistic Classification and Clustering in Relational Data", In Proceedings of IJCAI-01, 17th International Joint Conference on Artificial Intelligence, Seattle, US, 2001.

[11] B. Taskar, M. Wong, P. Abbeel, and D. Koller, "Link prediction in relational data", In Proceedings of Neural Information Processing Systems, 2004.