Link prediction on a Wikipedia dataset based on triadic closure

Robert Geroge
georgr2@rpi.edu

Abstract—**In this paper, a model for link prediction on a dataset representing changes to the Wikipedia website is presented. We first analyze the properties of this dataset, and provide a brief comparison against synthetic datasets with similar controlled properties; the differences are very much significant so only a cursory examination is done. After analyzing the properties of this dataset, we then propose a model for link prediction on the dataset, and give a brief description of the motivations for each algorithm which is to be used for the prediction. Our criteria for measuring the frequency and validity of predictions are then presented, after which the results of the link prediction models are discussed. We compare the results of unweighted and weighted predictors, and discuss potential explanations for our results.**

I. Introduction

The recent rise of social networking sites has led to a large scale creation of massive graph datasets that are on a scale not seen before. Such sets of data present both unique opportunities to researchers, as well as new challenges. An important area of research on these datasets deals with predicting how an agent in our world will act in the future given historical information. In essence, we want to be able to predict how links in the network will form before they form. Being able to successfuly do so has practical applications in several situations, the most intuitive of which is likely reccomendation systems. One can imagine a graph where nodes are products and edges represent shared purchases; if one can predict which links are likely to form given a chance, context can be changed to support this formation. On this specific dataset, we could consider the link prediction methods as a form of friend suggestion system. Additionally, as most link prediction algorithms, including those used in this paper, focus on triadic closure, we also will investigate whether or not this graph appears to experience triadic closure.

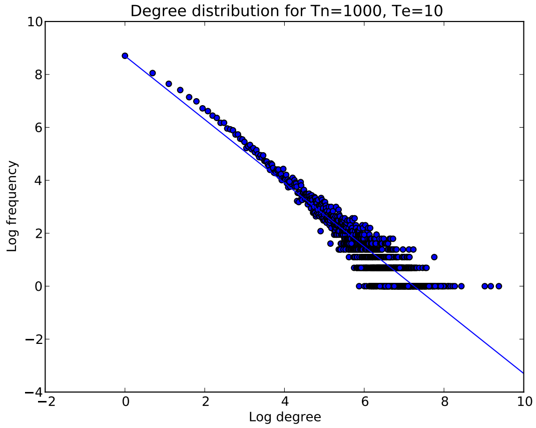
A. Prior work

Prior works which have dealt with either similar data or similar problems do exist in the literature. Here we examine selected articles which are relevant to the work presented here. Perhaps the paper most important to this is [**?**], which described and made popular the K-clique percolation clustering algorithm. In essence, when one wants to find overlapping communities, this paper shows that if one accepts a community to be the k-cliques of a graph, there can be an overlapping structure. Another important paper is by [**?**], which is the algorithm used to generate the cliques themselves. As the nave solution requires exponential time which grows as the graph grows, the K-clique percolation method would not be viable on any large dataset. However, in this paper, the authors describe an algorithm which enables cliques to be found in a graph only requiring time that grows with the highest degree node of every possible connected subgraph internal to the subgraph. This term is also called degeneracy. This method opens up larger sparse datasets to clique detectiona fact that is exploited on our dataset. Additionally, there have been several papers on link prediction; most useful to this was a literature review [**?**] in which most of the major link prediction algorithms are described in some detail.

II. Dataset

The raw data that was used in generating our working dataset consisted of edits to individual articles within the Wikipedia project. Entries had a time resolution of one week; each entry contained a date, an article ID, and the IDs of every user which has made an edit to the article in the prior week. The data available allows us to examine Wikipedia throughout its formation and growth. We note that the scale of this dataset is quite large; there are

Fig. 1. Shows the polynomial backoff, with α approximately equal to -1.1995



approximately 25,000,000 entries for changes to articles, and most every change has a record of several users editing the same article.

A. Graph generation

In order to generate a tractable dataset from this very large edit history, a threshold was imposed on the number of times a user has edited an article, T_n. If the user has made more edits than this number, they are added to the graph. Additionally, there is a threshold for edge formation, T_e. If a user has edited more articles than this threshold since being added to the graph with another user, then the two users will have an edge formed between them. This generates a sparse, symmetric graph. It is noted that as the working dataset is generated in one pass, we do not consider common edits prior to the addition of users to the network. Were this to be desired, a hash map of users satisfying the user threshold would first have to be generatedand then the edge formation could be tracked. There are no theoretical issues with doing all this in one pass, but the memory required would be that representing a threshold value of zero for bothwhich approaches N(N-1)/2 rapidly enough that it is severely impractical in practice. Extensions to this process for later results include assigning weighted edges to nodes, where the weight is the number of shared edits the two users have made.

B. Graph properties

We now consider some of the interesting basic properties of the dataset the above process has generated. Most aspects of this dataset are more or less unsurprising; it has polynomial degree decay as seen in Figure **??**, similar to most other social networks analyzed in the literature; it has a surprisingly high clustering coefficient (see Figures **??**, **??**), which is likely biased by the criteria we used to generate the dataset; and it has an interesting rate of growth, as seen in Figure **??**. Perhaps the property which is most intriguing is the clustering coefficient. It remains high throughout the existence of the network, but is exceptionally high for the first several months. While this paper does not attempt to explain why this property is observed, we posit a possible explanation that the vastly smaller scale and community of Wikipedia editors prior to its mainstream success had more common interests than editors whom began editing it as the content drastically expanded. This explanation is also assisted by the observed rate of growth for new users in the network; it strongly follows a logistic growth model. However, the number of edges in the graph appears to be growing at an unbounded rate. A potential explanation for this is that one factor which likely motivated users to become editorsthe opportunity to add new content to Wikipediais becoming less of a factor as the amount of content not in Wikipedia which has a viable place in Wikipedia is diminishing. In addition to the base graph, K-clique percolation was used to identify overlapping communities in the graph. We also observed that the vast majority of all predictions were made within a preexisting community, and as our predictive measures were significantly less accurate at itentifying extracommunity links, restricted link formation to within preexisting communities.

III. Link prediction

The link-prediction methodologies that were used to predict link formation on the graph are as follows: Adamic-Adar, Common Neighbors, Jaccard Similarity, and Preferential Attachment; for a brief equation describing each method, please see Figure **??**. A brief description of each algorithm follows. In Adamic-Adar, for each common neighbor of node x and y, we increment the heuristic indicator for link

Fig. 2. Shows the rate at which new nodes are added to the graph over time

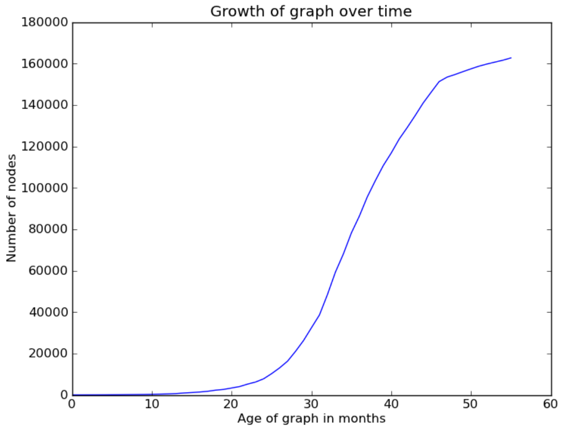


Fig. 3. Shows the change in clustering coefficient over time with a 5000 threshold for node addition

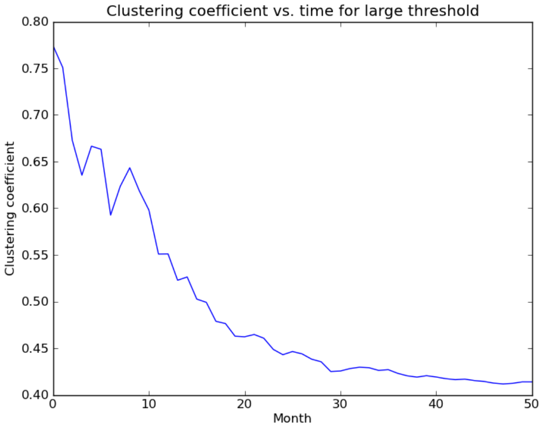


Fig. 4. Shows the change in clustering coefficient over time with a 1000 threshold for node addition

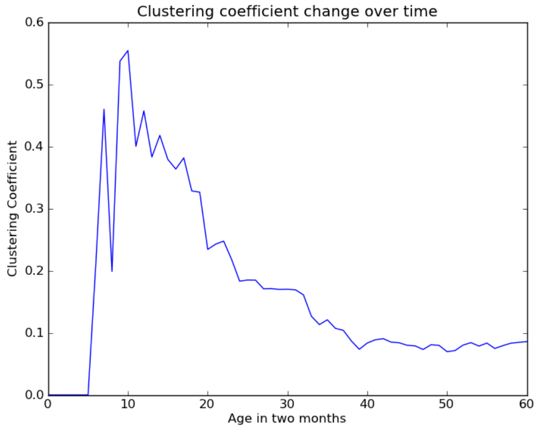


Fig. 5. Contains pertient equations to preliminary link prediction methodology

Predictor	**Equation**
Common Neighbors	$\|\Gamma(X)\cap\Gamma(Y)\|$
Jaccard Similarity	$\|\Gamma(X)\cap\Gamma(Y)\|/\Gamma(X)\cup\Gamma(Y)\|$
Adamic/Adar	$\Sigma_{z\in\Gamma(X)\cap\Gamma(Y)}1/\log(z)$
Preferential Attachment	$\|\Gamma(X)\|*\|\Gamma(Y)\|$

formation by $1/log(|z|)$. In effect, unique common neighbors bear more weight than less unique common neighbors. For a common neighbors scheme, one merely considers the number of common neighbors between x and y. Jaccard Similarity is essentially the fraction of neighbors of x and y which are shared; it can be viewed as a sort of normalized common neighbors approach. Lastly, preferential attachment considers the product of the number of neighbors of x and y. Each heuristic method will use a cutoff value for the necessary weight required to add a node. Comparisons will be done between common neighbors, Adamic/Adar, and preferential attachment models directly. As the threshold value for Jaccard similarity does not directly relate to any of the other models, we will briefly discuss the results obtained by using Jaccard Similarity, but focus on the three other methods. While it is implied by the equations, we are only considering nodes which have at least one common neighbor; the value of all heuristics save for preferential attachment will be zero otherwise. This essentially lmits our link prediction to predicting which triangles in the graph will close.

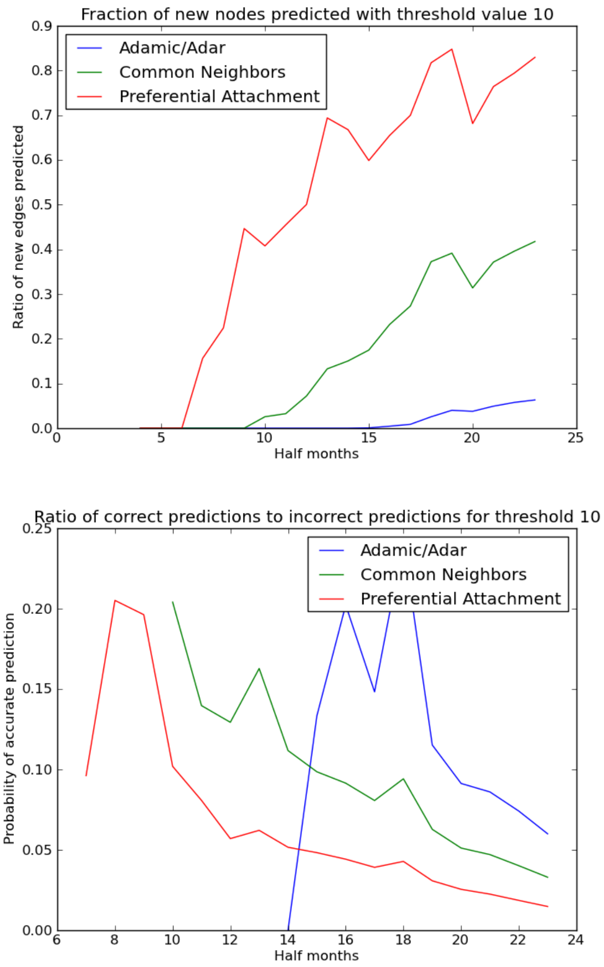
Fig. 6. Contains revised equations incorporating edge weight into calculation. Note $w(p, z)$ denotes average weight of e(x,z), e(y,z); ignoring zero values.

Predictor	**Equation**
Common Neighbors	$\Sigma_{z\in\Gamma(X)\cap\Gamma(Y)}w(p,z)$
Jaccard Similarity	$\|\Sigma_{z\in\Gamma(X)\cap\Gamma(Y)}w(p,z)/\|\Gamma(X)\cup\Gamma($
Adamic/Adar	$\Sigma_{z\in\Gamma(X)\cap\Gamma(Y)}w(p,z)/\log(z)$
Preferential Attachment	$\|\Gamma(X)\|*\|\Gamma(Y)\|$

A. *Preliminary results*

In analyzing the result of these methods, we define the following values. Accuracy will be the fraction of new nodes which are correctly predicted. Relevance will be the fraction of incorrect positive predictions to correct positive predictions. Weighted relevance will be the fraction of incorrect predictions to possible incorrect predictions $N(N-1)/2-|Edges|$. Our results using the above methodology were less than impressive. Out of all the methods, Adamic/Adar performed the best in terms of accuracy, but there was always a tradeoff as it found fewer new edges than any other method, as seen in Figure **??**. Most predictions were still wrong, and while it managed to accurately predict a fair portion of all new links, the majority of new links went undetected. All other predictive methods had similarly poor performance in making relevant predictions at lower threshold values, but did find a higher proportion of new edges. While the relevance of the results appears, out of the context of the graph, to be quite terrible, we do note that while the frequency to type two errors increases as the graph grows, the graph itself is growing at such a rate that compared to the number of incorrect predictions we could have made, $N(N-1)/2-|E|$, our predictions are not getting substantially less accurate compared to a random prediction. However, the most interesting result upon cursory examination is that preferential attachment is substantially better than the other link prediction methods in terms of the fraction of nodes formed which are predicted. Even when the threshold is increased past the point at which both common neighbors and Adamic/Adar fail to predict more than a few nodes, preferential attachment predicts a majority of the nodes with increasing relevancy as the threshold values increase. However, this is likely due to the relatively high degrees of nodes within the graph; until we choose very high cutoff points, preferential attachment is extremely aggressive in choosing edges to add, and as a consequence fares the worst at choosing highly relevant nodes. Jaccard similarity performs similarly to the common neighbors method; while we cannot directly compare results due to Jaccard having normalized parameters, the relationship between the fraction of nodes predicted and the probability of a prediction beign correct appears to be similar for the two. Lastly, we consider the effect that treating edges as unweighted has had on our results. In order to do this, we regenerated our dataset, and assigned weights to each edge which were equal to the number of times two users edited together.

Fig. 7. Preliminary results with heuristics given in Table 1



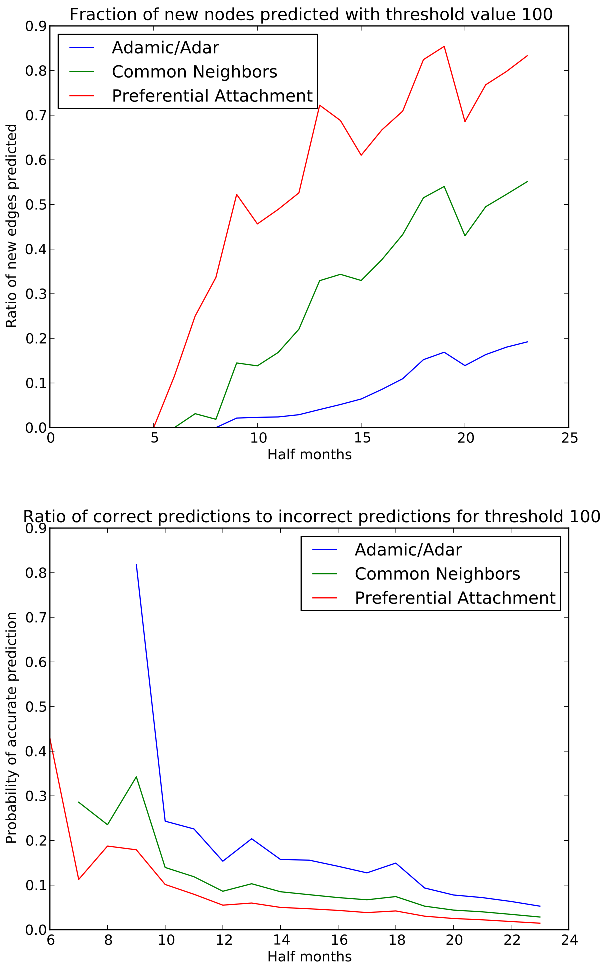
B. Link prediction revisited

By changing to a weighted graph, the link prediction algorithms which have been discussed so far generally have the average weight of each node to the neighbor in the numerator for the heuristics. The updated equations are in Figure **??** By rerunning the link prediction on a dataset which incorporates the frequency of edits, the results that we obtained appeared to be slightly better at predicting new nodes, but this comes at a cost of less relevant predictions as can be seen in Figure **??**) . Figure **??** shows the ratio of correct guesses of the weighted graph over the correct guesses of the unweighted graph, with thresholds of 100 and 10, respectively. A higher fraction of nodes were correctly predicted, and a higher fraction of predicted nodes were taken as compared to the unweighted predictor models. Preferential attachment takes on values so high that thresholds which are reasonalbe for the other heuristics are not reasonable for preferential attachment; most links that are at all likely will be predicted until we get into extremely high thresholds. We note, however, that as each method has strictly higher values, as the weight is at least one, the results for very low thresholds essentially predicted that every triangle in the graph will close As we move into the dramatically hgihger threshold values, the inclusion of weight in our heuristics causes our predictions to become substantially more relevant as the graph ages, but this comes at an extreme cost in that the number of edges predicted drops dramatically. Memory constraints prevented the analysis of later portions of the graph; it is possible that the weighted predictors will make up for their poor performance as the graph grows, but from our results on this dataset it appears that including weighted edges does not help with prediction. The fact that Preferential Attachment predicts most of the new edges suggests that triadic closure is strongly favored in this dataset.

IV. Conclusions

From the results discussed in prior sections, it appears that members of the Wikipedia network which fit the selection criteria used to generate our working dataset tend to form links in a way best explained by the level of activity for each user. The preferential attachment model for link formation substantially outperforms methods typically considered to be superior, such as Adamic/Adar, in terms of how many new edges are successfuly predicted. This result was surprising, but is likely due to the drastically higher values for the preferential attachment heuristic. Adamic/Adar has the highest ratio of correct to incorrect positive predictions, although this comes at the cost of predictive bredth. Additionally, link formation, by the definition used in this paper, appears to have been occuring at a significantly higher rate early on in Wikipedia's development; this appears to cause the network to both have an extremely high clustering coefficient and also causes predictive methods to be both more accurate and more relevant while the network is small and the clustering coefficient is high. Additionally, this lends us to believe that triadic closure, while still occurring in Wikipedia, is happening at a slower pace now than before–likely due to the influx of less active editors. Ultimately when attempting to predict links, one has to decide between having fewer valid results but even fewer invalid results, and having more valid results but even more invalid results.

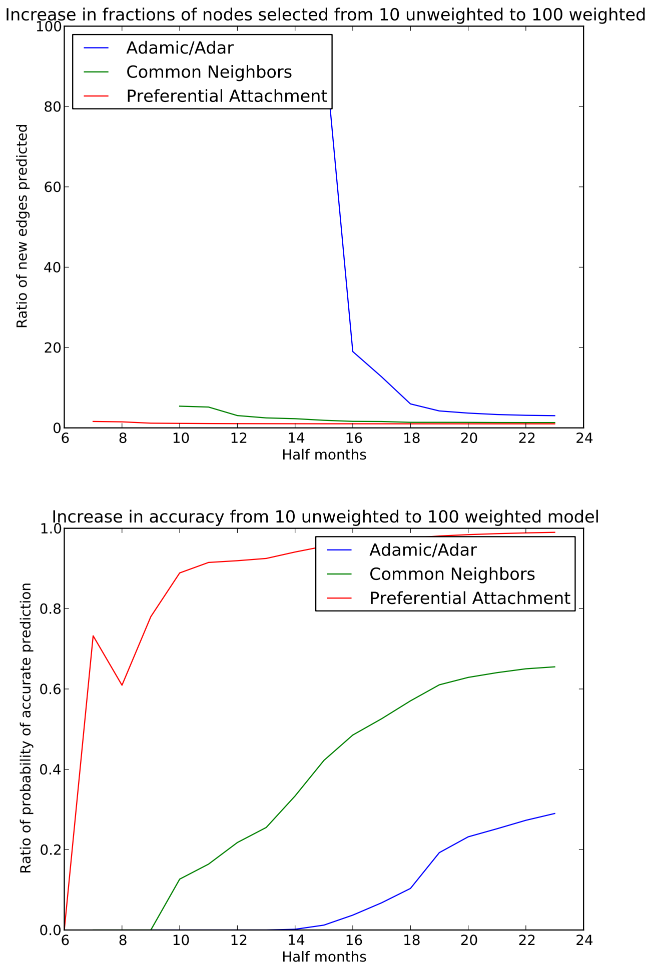
Fig. 8. Results with heuristics given in Table 2



V. Future work

There are a number of ways in which we could potentially improve upon our results. Perhaps the most intuitive extension would be to have decay in the relations between users. Additionally, it would be interesting to obtain results on a less selective portion of the dataset; the available computational resources restricted structures in memory to about

Fig. 9. A comparison between unweighted and weighted results



one gigabyte, which severely limited options for dataset generation. Lastly, it is possible that incorporating a representation, such as the vector obtained through latent semantic analysis, of what an article represents could allow us to better predict how users will form links. We would expect that editors of similar articles will more readily form bonds than those who edit unrelated articles. We also believe that examining different requirements for accuracy could produce different results; if we allow for edges to form in a larger timescale than the two weeks which we require for a prediction to be marked as accurate, it is possible that the results would differ substantially. However, what parameters would best represent real world situations is not enitrely clear; the lack of quantifiable meaning in many parameters we must use is another place in which we could improve upon this work.