ECAI 2008
M. Ghallab et al. (Eds.)
IOS Press, 2008

doi:10.3233/978-1-58603-891-5-162

A Fast Method for Property Prediction in Graph-Structured Data from Positive and Unlabelled Examples

Susanne Hoche[1] and **Peter Flach**[2] and **David Hardcastle**[3]

Abstract.
The analysis of large and complex networks, or graphs, is becoming increasingly important in many scientific areas including machine learning, social network analysis and bioinformatics. One natural type of question that can be asked in network analysis is "Given two sets R and T of individuals in a graph with complete and missing knowledge, respectively, about a property of interest, which individuals in T are closest to R with respect to this property?". To answer this question, we can rank the individuals in T such that the individuals ranked highest are most likely to exhibit the property of interest. Several methods based on weighted paths in the graph and Markov chain models have been proposed to solve this task. In this paper, we show that we can improve previously published approaches by rephrasing this problem as the task of property prediction in graph-structured data from positive examples, the individuals in R, and unlabelled data, the individuals in T, and applying an inexpensive iterative neighbourhood's majority vote based prediction algorithm ("iNMV") to this task. We evaluate our iNMV prediction algorithm and two previously proposed methods using Markov chains on three real world graphs in terms of ROC AUC statistic. iNMV obtains rankings that are either significantly better or not significantly worse than the rankings obtained from the more complex Markov chain based algorithms, while achieving a reduction in run time of one order of magnitude on large graphs.

1 Introduction

The analysis of large and complex networks or graphs is becoming increasingly important in a variety of scientific disciplines. Graphs allow us to model various tasks for graph-structured data which consist of individuals that are connected to each other in terms of, e.g., a shared interest or common function. In a *graph* $G = (V, E)$, the individuals are modelled as *nodes* $v \in V$, and the connection between the individuals as *links* $e \in E \subseteq V \times V$ between the nodes.

One prominent task in the analysis of graph-structured data is to rank one fraction $T \subset V$ of *target* nodes in a graph relative to another fraction $R \subset V$ of *root* nodes exhibiting a certain property of interest ϕ, in order to answer the question how close or similar they are to the ones in R with respect to ϕ. Here, we focus on *co-authorship graphs* where the nodes are papers which are linked to each other by an undirected weighted edge iff the papers have one or more author in common; $R \subset V$ is a set of papers having scientific topic ϕ, and $T \subset V$ is a set of papers with unknown topics for which we want to know how similar they are to the papers in R with observed topic ϕ.

To answer such a question, we can attempt to rank the nodes in T such that the nodes ranked highest are most likely to exhibit ϕ and can thus be assumed to be closest to R with respect to ϕ. A number of approaches have been proposed in different scientific areas to determine a node's importance in a graph, such as, e.g., numerous node centrality measures in social network analysis [19], and ranking algorithms motivated by the necessity to sort Web pages in a specific Web search task (e.g., HITS [11] and PageRank [3]). However, while these algorithms operate on a global level, the task we are interested in is to rank nodes on a local level, i.e., with respect to a given set R of nodes exhibiting property ϕ which can be interpreted as existing background knowledge, or ranking bias.

Several such local ranking methods which answer the question of relative importance for graph structured data have been proposed in [20]. These methods are based on weighted paths and Markov chain models and thus computationally expensive which makes their application for large graphs inefficient. We can improve these approaches by rephrasing the ranking problem as the task of property prediction in graph-structured data from positive examples, the nodes in R, and unlabelled data, the nodes in T, and applying an inexpensive iterative neighbourhood's majority vote based prediction algorithm ("iNMV") that allows an effective and efficient ranking of the nodes in T with respect to the nodes in R. Given a set $R \subset V$ of papers in a co-authorship graph G with an observed topic $\phi \in \Phi$, one can predict – on the basis of the known topics and the graph's link structure – the probability that for a given set T of papers with unknown topics, $t \in T$ has topic ϕ, and rank the nodes in T according to this predicted probability, i.e., according to their similarity to R with respect to ϕ.

The remainder of the paper is organised as follows. We discuss two Markov chain based methods proposed in [20] for ranking individuals in graphs in Section 2. In Section 3, we present our iNMV prediction algorithm and detail how we obtain a ranking of T. In Section 4, we show that on three real world graphs the iNMV prediction algorithm achieves rankings that are either significantly better or not significantly worse than the rankings obtained from the two methods described in Section 2, and at the same time reduces the run time on large graphs by one order of magnitude. We review related work in Section 5 and conclude in Section 6.

[1] University of Bristol, Department of Computer Science, UK, email: hoche@cs.bris.ac.uk

[2] University of Bristol, Department of Computer Science, UK, email: Peter.Flach@cs.bris.ac.uk

[3] University of Bristol, Department of Computer Science, UK, email: Hardcastle.David@yahoo.co.uk

2 Local Ranking Methods based on Markov Chains

White and Smyth propose in [20] several local ranking methods – based on weighted paths and Markov chain models – which answer the question of the relative importance of a set T of nodes in a graph G with respect to another set R in G. Here, we discuss two of their proposed methods that are based on Markov chains. In a Markov chain based approach G is viewed as representing a first-order Markov chain. The idea is to traverse the graph in a Markov random walk, i.e., to start at some node and then randomly follow an outgoing edge to the next node from where the process then repeats itself. The first-order Markov chain, or the transitions between the nodes, is characterized by a transition probability matrix P. The descriptions in the next two sections are based on [20].

2.1 Inverse Average Mean First Passage Time

The *mean first passage time* m_{rt} from a node r to a node t in a first-order Markov chain is defined as the expected number of steps in an infinite-length Markov random walk starting at r until the first arrival at t, i.e., as

$$m_{rt} = \sum_{n=1}^{\infty} n f_{rt}^{(n)}, \tag{1}$$

where $f_{rt}^{(n)}$ denotes the probability that the random walk starting at r reaches t after exactly n steps. [20] defines the importance $I_1(t|R)$ of a node t with respect to a set R in terms of the *inverse average mean first passage time*, i.e., as

$$I_1(t|R) = \frac{1}{\frac{1}{|R|}\sum_{r\in R} m_{rt}} \tag{2}$$

That is, important nodes are relatively close to all the nodes in R.

A so-called *mean first passage time matrix* M with entries m_{ij} for all pairs of nodes (v_i, v_j) in the graph can be obtained as follows. The fundamental matrix is defined as $Z = (I - P - e\pi^T)^{-1}$, where P is the Markov transition probability matrix, e a column vector containing all ones, and π a column vector of the stationary distribution for the Markov chain. The mean first passage time matrix is then obtained as

$$M = (I - Z + EZ_{dg})D, \tag{3}$$

where I is the identity matrix, E a matrix containing all ones, Z_{dg} the matrix that agrees with Z on the diagonal but is 0 elsewhere, and D the diagonal matrix with elements $d_{ii} = \frac{1}{\pi(i)}$ for node i's stationary distribution $\pi(i)$ for the Markov chain.

2.2 *K*-Step Markov Approach

An alternative approach investigated in [20] defines the importance $I_2(t|R)$ of a node t with respect to a set R on the basis of a Markov random walk of fixed length K, i.e., as the probability that the Markov random walk starting at r and ending after exactly K steps reaches t. The value K determines the bias towards the set R: the smaller K the larger is R's influence, the larger K the more we approach the Markov chain's stationary distribution.

$I_2(t|R)$ can be computed as

$$I_2(t|R) = [Pp_R + P^2 p_R + \cdots + P^K p_R]_t, \tag{4}$$

where P is the Markov transition probability matrix, p_R is a column vector containing the initial probabilities for the set R, and $[X]_t$ denotes the t-th entry of the column vector X.

3 Rephrasing the Task of Local Ranking in Terms of Property Prediction

Our main contribution in this paper is to show that we can solve the local ranking problem more efficiently by rephrasing it as the task of property prediction from positive and unlabelled examples. Specifically, let $G = (V, E)$ be a given co-authorship graph with a set of nodes (papers) V and a set $E \subseteq V \times V$ of undirected (co-authorship) edges (v_i, v_j) with weight w_{ij}, and let Φ be a set of topics that each paper can have (we assume that a paper can have several topics). Furthermore, let $V = R \cup T, R \cap T = \emptyset$, where R is a set of *root nodes*, or positive examples, for which we have observed the topics, and T is a set of *target nodes*, or unlabelled examples, for which we do not know the topics. The task is to rank the nodes in T for each $\phi_k \in \Phi$ separately on the basis of the set R of root nodes and the graph's link structure given by E according to their probability of exhibiting topic ϕ_k.

3.1 Iterative Neighbourhood's Majority Vote based Property Prediction

To this end, we apply our iterative neighbourhood's majority vote prediction algorithm iNMV which is based on a simple majority vote of directly linked nodes, or neighbours, and which consists of an *initialisation step* and an *update step* which can be applied iteratively.

In the initialisation step, we assign for each target node an initial estimate to its topic probability on the basis of the topics observed for the root set R. In an update step, a node's existing estimate is modified based on the neighbouring nodes' current estimates. This way, entities are classified in dependence of each other, and mutual influence of the predictions is accounted for. The more often the update step is iterated, the more the predictions are propagated through the graph.

Since papers can have multiple topics, we consider for each topic $\phi_k \in \Phi$ a binary learning problem where nodes having topic ϕ_k constitute the *positive* examples. For each topic $\phi_k \in \Phi$ separately, iNMV derives for each target node $v_i \in T$, an estimate of the probability of observing ϕ_k for v_i. We denote the set of topics of paper v_i as its *topic set* $y_i \subseteq \Phi$.

Our approach assumes that nodes in the same neighbourhood of the graph tend to have similar properties, and that the predicted topic for one node in the graph depends on the topic of the nodes directly linked to it. Therefore, we assume that the probability of observing topic ϕ_k for node $v_i \in T$ given G is equal to the probability of observing ϕ_k for v_i given v_i's neighbourhood $N_i := \{v_j \in V | (v_i, v_j) \in E\}$ consisting of those nodes in V that are directly linked to v_i. We base the prediction of an unlabelled node's topic probability both on labelled and unlabelled neighbours in the graph, and thus derive a topic probability estimate from the known topics and topic probability estimates of directly linked root and target nodes, respectively.

To predict the probability of observing ϕ_k for a node $v_i \in T$ with unknown topic set y_i, we assign to v_i an initial estimate $p_{ik}^{(1)} := P(\phi_k \in y_i|R)$, where $P(\phi_k \in y_i|R)$ denotes the probability that paper v_i has topic ϕ_k, conditioned on the topics observed in R. This estimate is based on the number n_k of times that ϕ_k is observed in R using the maximum likelihood based m-estimate where the observations are augmented by m additional samples which are assumed to be distributed according to p:

$$p_{ik}^{(1)} := P(y_i = \phi_k|R) = \frac{n_k + p \cdot m}{|R| + m}, \tag{5}$$

where $|R|$ denotes the cardinality of set R. We choose $m = 1$ and $p = 0.5$ (each topic is equally likely to be present or absent).

For a node $v_i \in R$ with observed topic, let $p_{ik}^{(1)} := 1$ for every topic ϕ_k that is observed for v_i.

For each topic ϕ_k, we update the initial probability estimates $p_{ik}^{(1)}$ for each node $v_i \in T$ based on its neighbourhood's estimates: the modified estimate $p_{ik}^{(t+1)} := P^{(t+1)}(y_i = \phi_k|N_i)$ is derived on the basis of the estimates $p_{jk}^{(t)} := P^{(t)}(y_j = \phi_k|N_j)$ for observing ϕ_k for v_i's neighbours $v_j \in N_i$ in the t-th update step:

$$p_{ik}^{(t+1)} := P^{(t+1)}(y_i = \phi_k|N_i) = \frac{1}{\sum_{n_j \in N_i} w_{ij}} \sum_{n_j \in N_i} w_{ij} p_{jk}^{(t)}, \tag{6}$$

where w_{ij} is the weight of the edge between the nodes v_i and v_j.

As we are dealing with an undirected graph, equation (6) is recursive. To account for the mutual influence between linked nodes, the estimates can be propagated through the graph by iterating equation (6) several times. With more iterations, predictions are propagated further through the graph.

3.2 Ranking the Target Set using ROC Analysis

iNMV obtains for every topic $\phi_k \in \Phi$ and every node $v_i \in T$ an estimate p_{ik} of the probability of observing ϕ_k for v_i. We interpret p_{ik} as a score which we use to order the target nodes T. iNMV learns from positive and unlabelled examples, i.e., from root and target nodes. However, for each topic $\phi_k \in \Phi$ we have originally positive and negative examples, i.e., those examples which exhibit ϕ_k and those which do not. To generate unlabelled examples, we delete for each topic and each target node the label indicating to which topic the paper belongs, but use it, after we have obtained the ranking of the nodes, to compute the ranking's *AUC*.

The area under the ROC Curve statistic, or AUC, is a measure based on the pairwise comparisons between the results of a binary prediction problem, and is often used to evaluate the performance of a prediction or ranking algorithm. It can be interpreted as the probability that for a pair $(+,-)$ of a positive and a negative example that are both drawn uniformly at random, a higher score will be assigned to the positive example than to the negative (which means that these two examples are ranked correctly relative to each other). An algorithm's AUC is the fraction of $(+,-)$-pairs that it correctly ranks relative to each other, and is defined as

$$AUC = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} 1_{(+_i > -_j)}}{m \cdot n}, \tag{7}$$

where $+_1, \cdots, +_m$ are the scores assigned to the m positive examples, $-_1, \cdots, -_n$ are the scores assigned to the n negative examples, and $1_{(+_i > -_j)}$ is the indicator function which is equal to 1 if $+_i > -_j$, and 0 otherwise. An algorithm's AUC is maximal, i.e., equal to 1, iff it ranks all positive examples higher than the negative examples. Any misranked $(+,-)$-tuple decreases the AUC.

4 Empirical Evaluation

We evaluate the three methods described in Sections 2 and 3 on co-authorship graphs induced from the bibliographic data sets "IPLNet2" [1] and "Cora" [14]. The weighted links between the nodes are modelled in terms of an adjacency matrix A which holds for each pair (v_i, v_j) of connected nodes $v_i, v_j \in V$ a non-zero entry w_{ij} according to the overlap of the papers' author lists. We obtain the Markov transition probability matrix P from A by normalising the rows in A.

4.1 Data and Experimental Setup

The ILPNet2 bibliographic database contains hand-selected ILP-related references from 1970 onwards. Our co-authorship graph consists of the largest connected component of 406 nodes with known topics and 6354 links (on average ≈ 15 links per node). We restrict our evaluation to the 10 topics that include at least 20 papers each.

For each topic ϕ, we generate in 10 trials 4 distinct root and target set partitions. In each partition, the root set consists of 75% of the positive examples, i.e., the papers which have topic ϕ. The target set contains the remaining 25% of the positive examples and all negative examples, i.e., the papers which do not have topic ϕ.

The target nodes are distinct in each of the 4 root and target set partitions, and their union results in the complete set of nodes. Thus, each node serves for each topic and trial exactly once as an unlabelled example, or target node. For each topic, we apply the three methods to the 40 distinct data partitions. From this we yield for each topic ϕ and each node $v \in T$ an estimated degree to which v belongs to ϕ. We interpreted these values as scores and use them to rank the nodes as detailed in Section 3.2, where a higher score indicates a higher probability of exhibiting ϕ.

Cora is a collection of $\approx 34,000$ computer science research papers that have been automatically collected from the web [14]. Our co-authorship graph consists of the largest connected component of 10,513 nodes with known topics and 87,438 links (on average ≈ 8 links per node). The topics establish a hierarchy with general computer science topics at the top level which branch out into several sub-levels. We restrict our evaluation to the 6 top-level topics with the highest number of positive examples ("6 Top"), and to the 7 Machine Learning sub-topics on the lowest hierarchy level ("7 ML").

For each topic ϕ, we generate in 5 trials 2 distinct root and target set partitions, where a root set consists of 50% of the positive examples, and a target set of the remaining 50% of the positive examples and all negative examples. For each topic, we apply the three methods to 10 "6 Top" and "7 ML" root and target set partitions, respectively, and use the resulting scores to generate rankings of the target nodes which we evaluate in terms of ROC AUC statistic.

4.2 Results

In Figure 1, we show for the three methods described in Sections 2 and 3 and the three domains described in Section 4.1 boxplots of the AUCs for all topics averaged over all partitions and trials. We show for the ILPNet2 data from left to right boxplots for the AUCs obtained from the inverse average mean first passage time (iaMFPT) method, iNMV with 1, 5, and 10 iterations, respectively, and the K-Step Markov method for $K = 1, 2, 5, 10, 25$. Each boxplot shows the median, lower and upper quartile, and the lower and upper limit of the AUCs for the single topics, for one method.

Since the iaMFPT method has been found numerically too complex for the large Cora graph, results for this method are only shown for the small ILPNet2 graph. We think that this is justified since the ranking of this method is significantly worse than the rankings of all other methods (see below). We have also performed experiments for the K-Step Markov method for $K > 25$ but found that the AUCs are further decreasing and significantly lower than those for iNMV with 1, 5 or 20 iterations, and thus omit these results.

For the two Cora domains, we show in Figure 1 from left to right boxplots for the AUCs obtained from iNMV with 1, 5, and 10 iterations, respectively, and the K-Step Markov method for $K = 1, 2, 5, 10, 25$. For the two Cora domains and all methods, the single topics' AUCs are in close range to each other. In contrast, the AUCs of the ILPNet2 topics exhibit large differences for all methods. In all the domains, nodes belonging to some topics form heterogeneous clusters in the graph, while nodes belonging to others topics are spread more widely over the graph. This seems to be more problematic when only a small number of positive examples exists.

We perform a significance test to answer the question whether the results are significantly different. When comparing more than two classifiers, the non-parametric Friedman test [9] is widely recommended [6]. The Friedman test compares k algorithms over N data sets by ranking each algorithm on each data set separately, with the best result receiving rank 1, etc., and assigning average ranks in case of ties. The test then compares the average ranks of all algorithms on all data sets. If the null-hypothesis – that all algorithms are performing equivalently – is rejected under the Friedman test statistic, post-hoc tests such as the Nemenyi test [15] can be used to determine which algorithms perform statistically different. Note that for each topic ϕ, distinct root and target set partitions are generated, and that the Friedman test can thus be applied to these $N = |\phi|$ mutually independent data sets.

According to the Friedman test, the AUCs averaged over all trials and partitions for the ILPNet2 data set obtained from the iaMFPT method are significantly worse than the rankings obtained from any other method. The AUC of the ranking obtained from the iaMFPT is most likely so much smaller because a target node t's importance $I_1(t|R)$ is equally influenced by all root nodes in R. By contrast, a target node's ranking obtained from iNMV or the K-Step Markov method for small K depends on a much smaller neighbourhood. This seems to indicate that the set of root nodes has to be rather coherent in order for the iaMFPT to produce a good ranking as, e.g., in the data sets evaluated in [20] (e.g., a set of collaborating authors, or interacting terrorists, where $|R| = 2$). In the ILPNet2 data, where the root set consists of a set of papers which have the topic of interest but which most likely belong to different "co-authorship cliques", this assumption does not seem to hold, but rather the neighbourhood assumption that directly linked papers tend to be on the same topic.

For the Cora "6 Top" data, the Friedman test reports for the AUCs averaged over all trials and partitions that both iNMV with 5 and 20 iterations are significantly better than the K-Step Markov method for both $K = 1$ and $K = 25$. No significant differences have been found for the rankings on the Cora "7 ML" data.

4.3 Discussion

For iNMV, we obtain with 5 iterations on all three domains rankings with the highest AUCs. Equally, the K-Step Markov method yields for small K (2 or 5) the best AUCs. This indicates that on the domains we are investigating, the rankings benefit from a mixture of local patterns from small neighbourhoods in the graph rather than from a global method that considers information from large areas of the graph (as, e.g., the K-step Markov with larger K, or iaMFPT).

The K-Step Markov method considers for a target node $t \in T$ all nodes $r \in R$ that are K hops in G away from t. In contrast, iNMV with K iterations of the update step considers for the estimate of t's topic probability all nodes $r \in R$ that are K hops in G away from t, and additionally all nodes $t' \in T$ that are K hops in G away from t, where the topic probability estimate of t' itself is modified in each iteration of the update step on the basis of its direct neighbourhood. This way, mutual influence of the unlabelled nodes is also taken into account which seems to be advantageous for the ranking of T with respect to R and ϕ.

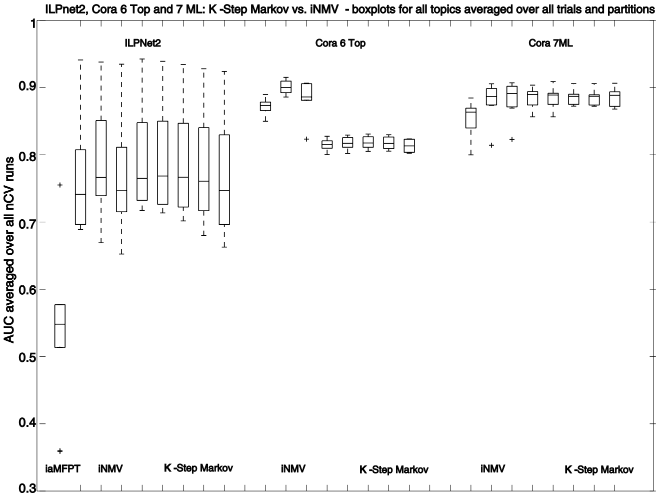

Figure 1. Boxplots for the AUCs of the rankings resulting from the methods described in Sections 2 and 3 on the ILPNet2, Cora "6 Top" and "7 ML" data sets for all topics averaged over all partitions and trials. For each domain, we show – from left to right – a boxplot for iNMV with 1, 5, and 20 Iterations, and for the K-Step Markov method for $K = 1, 2, 5, 10, 25$, respectively. For the ILPNet2 data, the leftmost boxplot is for the iaMFPT method. Each boxplot shows the median, lower and upper quartile, and the lower and upper limit of the data points (not considered to be outliers), i.e., the AUCs for the single topics, for one method. An outlier is depicted as "+".

For the domains investigated in this paper, the obtained AUCs do not seem to depend on the percentage of positive examples for a topic. Rather, the main factors seem to be the number of intra- and inter-topic neighbours, respectively, that a node is linked to, and the way that the nodes with the same topic are positioned in the graph G. The more the nodes in G establish areas homogeneous with respect to their topics the more successful can a method be that assumes similar nodes in the neighbourhood of each other and thus bases its prediction for a node v on a small region around v in the graph.

	iNMV 1It	iNMV 5 Its	iNMV 20 Its	1-Step Markov	2-Step Markov	5-Step Markov	10-Step Markov	25-Step Markov	inv. avg MFPT
ILPNet2	2.3±0.06	13.4±0.7	34±1.6	7.5±0.6	7.5±0.6	7.6±0.6	7.9±0.7	8.6±0.6	17.5±1.6
Cora6Top	216±12	252±15	414±29	1477±2	1479±2	1638±27	2309±23	4446±6	n/a
Cora7ML	218±6	266±7	465±16	1508±27	1555±33	1649±29	2312±21	4460±19	n/a

Figure 2. Run time complexity and standard deviations of the compared methods in seconds on a Intel(R) Xeon(TM) MP CPU 3.16GHz processor.

In Figure 2, we report the run time complexity for the iNMV and K-Step Markov methods and all domains, and that of the iaMFPT method for ILPNet2. On the small ILPNet2 co-authorship graph, iNMV is with 5 and 20 iterations 2 to 5 times slower than the K-Step Markov method. However, all methods' run time lies in the range of a a few seconds only. For the large graphs, the K-Step Markov method's run time is 6 to 10 times larger than that of iNMV, i.e., in the range of hours rather than minutes.

5 Related Work

Closely related to our work with respect to prediction methods in graph-structured data are the publications in the fields of link-based object classification, collective inference, and iterative classification. [4] and [17] were among the first to study the effects of using related objects' attributes to enhance classification in graph-structured domains. [4] proposes a relaxation-labelling based method for topic prediction in hyperlinked domains. [17] incrementally classifies a collection of encyclopedia articles and take into account the classes of unlabelled documents only after they have been classified on the basis of neighbouring documents. [2] introduces conditional random fields for link-based object classification, e.g. for part-of-speech tagging, while [18] extends this approach to a setting of arbitrary graphs instead of chains. [16] proposes the use of relational dependency networks and Gibbs sampling to collectively infer labels for linked instances. [12] proposes an iterative link-based object classification method based on modelling link distributions which describe the neighbourhood of directed links around an object. [13] investigates the effectiveness of relaxation labelling based methods for classification of graph-structured data similar to the one proposed in [4].

However, none of these works consider the task of ranking a set of target nodes with respect to a set of root nodes exhibiting a specific property. Although we have for all domains that we investigate in this paper both positive and negative labelled examples, we only consider the positive examples as labelled. We argue that it is realistic to assume a paper that is not labelled as belonging to a specific topic to be unlabelled rather than to be a negative example.

In the areas of social network analysis and Web mining, several approaches have been proposed to determine a node's importance in a graph. Freeman developed several measures of node centrality which express how important a node is in a graph [7, 8]. A comprehensive overview about centrality measures in graphs is given in [19].

Several algorithms have been proposed to rank the nodes in a graph of Web pages. Well known examples are HITS [11] and PageRank [3] – which operate on a global level – and personalised variants thereof, e.g., a topic-sensitive PageRank [10] where the ranking of Web pages is biased towards a set of specific topics, and a personalised version of HITS [5] which adjusts the measure of an authoritative source on the basis of incorporating user feedback. These personalised variants bias the standard ranking towards a set of a-priori defined root nodes. However, they have been designed specifically for the context of Web queries.

6 Conclusion

We presented an effective and efficient algorithm to solve the task of ranking a set of target nodes in a graph with respect to a pre-defined set of root nodes which exhibit a specific property of interest. To this end, we rephrased the ranking problem as the task of property prediction in graph-structured data from positive and unlabelled examples, and proposed an inexpensive iterative neighbourhood's majority vote based prediction algorithm, iNMV. On three real-world co-authorship networks, iNMV obtains rankings that are either significantly better or not significantly worse with respect to AUC than the rankings obtained from two previously published Markov chain based algorithms, and at the same time achieves a reduction in run time of one order of magnitude on large graphs. For a local ranking method, it seems to be advantageous to not only account for the root nodes' influence on the prediction for a target node but to also consider, as iNMV with several iterations of the update step does, the mutual influence of linked target nodes.

In future work we plan to investigate whether there are benefits in learning a joint model for two or more topics. Topics are likely to be correlated (overlapping or disjoint), and we may be able to take advantage of that. We are furthermore investigating the time dependency of co-authorship networks and paper topics.

Acknowledgements

The authors would like to acknowledge funding and support for this work from GCHQ in Cheltenham in the UK. and would like to thank Jörg Kaduk for numerous interesting discussions.

REFERENCES

[1] ILPnet2 on-line library. `http://www.cs.bris.ac.uk/~ILPnet2/Tools/Reports`.

[2] J. Lafferty, A. McCallum, and F. Pereira, 'Conditional random fields: Probabilistic models for segmenting and labeling sequence data', in *Proceedings of the 18th International Conference on Machine Learning*, pp. 282–289 (2001).

[3] S. Brin and L. Page, 'The anatomy of a large-scale hypertextual web search engine', in *Proceedings of the 7th International World Wide Web Conference*, pp. 107–117 (1998).

[4] S. Chakrabarti, B.E. Dom, and P. Indyk, 'Enhanced hypertext categorization using hyperlinks', in *Proceedings of the SIGMOD-98 ACM International Conference on Management of Data*, pp. 307–318 (1998).

[5] H. Chang, D. Cohn, and A. McCallum, 'Creating customized authority lists', in *Proceedings of the 17th International Conference on Machine Learning*, pp. 167–174 (2000).

[6] J. Demšar, 'Statistical comparisons of classifiers over multiple data sets', *Journal of Machine Learning Research*, **7**, 1–30 (2006).

[7] L. C. Freeman, 'A set of measures of centrality based on betweenness', *Sociometry*, **40**, 35–41 (1977).

[8] L. C. Freeman, 'Centrality in social networks: I. conceptual clarification', *Social Networks*, **1**(3), 215–239 (1979).

[9] M. Friedman, 'The use of ranks to avoid the assumption of normality implicit in the analysis of variance', *Journal of American Statistical Association*, **32**, 675–701 (1937).

[10] T. Haveliwala, 'Topic-sensitive PageRank', in *Proceedings of the 11th International World Wide Web Conference*, pp. 517–526 (2002).

[11] J. Kleinberg, 'Authoritative sources in a hyperlinked environment', in *Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms* (1998).

[12] Q. Lu and L. Getoor, 'Link based classification', in *Proceedings of the 20th International Conference on Machine Learning*, pp. 496–503 (2003).

[13] S.A. Macskassy and F. Provost, 'Classification in networked data: A toolkit and a univariate case study', *Journal of Machine Learning*, **8**, 935–983 (2007).

[14] A. McCallum, K. Nigam, J. Rennie, and K. Seymore, 'Automating the construction of internet portals with machine learning', *Information Retrieval*, **3**(2), 127–163 (2000).

[15] P. B. Nemenyi, *Distribution-free multiple comparisons*, Ph.D. dissertation, Princeton University, 1963.

[16] J. Neville and D. Jensen, 'Iterative classification in relational data', in *Proceedings of the AAAI-2000 Workshop on Learning Statistical Models from Relational Data*, pp. 13–20 (2000).

[17] H.-J. Oh, S. H. Myaeng, and M.-H. Lee, 'A practical hypertext categorization method using links and incrementally available class information', in *Proceedings of the 23rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 264–271 (2000).

[18] B. Taskar, P. Abbeel, and D. Koller, 'Discriminative probabilistic models for relational data', in *Proceedings of the 18th International Conference on Uncertainty in Artificial Intelligence*, pp. 485 – 492 (2002).

[19] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*, Cambridge University Press, 1994.

[20] S. White and P. Smyth, 'Algorithms for estimating relative importance in networks', in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 266–275 (2003).