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

Panning for Gold - Using Variograms to Select Useful Connections in a Temporal Multigraph Setting

Alexey Uversky · Dušan Ramljak · Vladan Radosavljević · Kosta Ristovski · Zoran Obradović

Received: date / Accepted: date

Abstract As social networks continue growing in size and popularity, so does the amount of data that is publically available pertaining to the members of these networks, which results in a rapid growth of the types of connections that can potentially be established between members of the network. With this increasing amount of available information, the task of examining social networks becomes more exciting and more difficult at the same time. Researchers now have to filter the large field of potentially useful information to find the golden nuggets that will serve their needs, while ignoring data that would actually hamper their experiments. To address this issue, we propose a novel integration of a variogram-based system for selecting potentially informative relationships with a Gaussian Conditional Random Field (GCRF) model that is able to perform node attribute prediction in social networks that are both temporal and contain a wide variety of relationship types. By first evaluating the large pool of relationships via variograms, which can be done very quickly, we are able to fully utilize the power of the GCRF model, which becomes computationally expensive when the network size or the number of relationships examined grows. Our experiments show that we can use variograms to narrow the potential field of relationships drastically and consequently make the GCRF model run in a reasonable amount of time and remain accurate.

Keywords conditional random fields · temporal social networks · citation networks · variograms

This work is supported in part by DARPA Grant FA9550-12-1-0406 negotiated by AFOSR.

Corresponding author: Zoran Obradović
Center for Data Analytics and Biomedical Informatics,
Department of Computer and Information Sciences,
College of Science and Technology,
Temple University, Philadelphia, PA, USA
E-mail: zoran.obradovic@temple.edu

1 Introduction

Social networks are becoming increasingly more popular as researchers realize the wealth of information that can be extracted from, and consequently studied, from these networks. In addition, these networks come in all shapes and sizes, meaning that regardless of the area of study, an appropriate network can most likely be found and studied. Various studies of social networks have been done, such as examining delinquency and alcohol consumption in students based on social interactions [Ouzienko et al., 2011] and creating a recommender system for online marketplaces based on social interactions amongst customers and the products they have purchased in the past [Walter et al., 2008]. Social networks are also utilized for more commercial purposes, such as identifying relevant ad placements for customers based on their social interactions and purchasing habits [Yang et al., 2006]. Usually these studies focus on a very select few types of relationships, however, and ignore the other connections that aren't immediately obvious as being useful to the study in question.

In this project we intend to address this issue by examining a social network with a rich variety of relationships, some of which evolve over time and others not having this characteristic. Note here that we use the terms "relationship" and "connection" interchangeably throughout the paper, but both simply refer to a link between two nodes in a network that can either be explicitly defined in the network, or computed according to criteria appropriate for the network. We explore how the information in such a network can be utilized in the context of a state-of-the-art predictive model: Gaussian Conditional Random Fields (GCRF). This model was first used in computer vision [Liu et al., 2007] but has since then been integrated for various other purposes, including regression in social networks. Since the model augments the output of multiple predictors that don't require network structure (and hence can rely solely on an individual node's history of node attribute values) with multiple similarity measures that can be defined based on network structure, it is very useful for predicting future node attribute values in a complex multigraph social network whose elements can change over time. In the past, simpler models have been implemented to perform analysis of social networks, but they have relied on significantly smaller subsets of potential relationships that were selected according to intuition. We hope to establish a more concrete set of steps for choosing connections that can be useful for a particular purpose by combining the use of variograms with a GCRF model. This choosing process is doubly important because, in addition to removing connections which may reduce model accuracy due to having bad information, it helps to drastically reduce the running time of the model, which becomes very slow when multiple connections are used concurrently, by reducing the pool of connections that are considered for use in the model. This is an extension of our preliminary results submitted to the ASONAM conference, which we expand by automating the previously manual similarity selection method, and further confirm the

effectiveness of the selection method that is done prior to utilizing the GCRF model.

Due to their availability and nature, bibliographic networks serve as very good candidates for being analyzed with our new approach. Bibliographies of all kinds are available online for free (two of the more popular examples of this include DBLP and CiteSeer), and inherently contain a rich multigraph structure, since one can construct relationships between authors, venues, papers, or terms. Furthermore, these relationships can be easily extracted from a collection of documents with simple text-mining techniques, even if they are not explicitly defined in the collection. If we treat a bibliography as our social network of choice, we can extract relationships based on the different aspects of the network described previously, and examine how the relationships between nodes in the network are affected by these connection types.

In this paper we focus on a network of high energy particle physics papers (HepTh)[1], and treat papers as the nodes we want to examine over time. We consider citation counts for each paper as the attribute of interest, and aim to predict future citation counts given a history of citation counts as well as a history of the evolution of relationships amongst nodes in the network, which results in a multigraph type structure. By using the GCRF model we are able to observe the impact of the past citation counts for each paper regardless of the network structure via the unstructured predictors, evaluate different similarity measures based on different connection types with our variogram technique, and finally examine the utility of these similarities on improving the unstructured predictor performance by adding the impact of network relationships into the mix.

The main goal of this paper is to show that we can quickly and effectively analyze a large pool of relationships that can be defined on a social network, and drastically reduce the number of relationships that should be considered for use in a predictive model. We analyze these links individually rather than grouping similar types together, since it is possible that links of the same type may contain enough differing information to be useful by themselves. Although there are many various types of social networks we could have examined in this context, we chose the HepTh citation dataset as it is of manageable size, is publically available, contains the raw text data that can be used to extract additional relationships, and was also used in a KDD Cup competition whose winner was able to predict future citation counts with a very simple predictor that did not utilize network structure (and hence we could see the effects of adding network by comparing against this predictor). We focus on the same task, namely, future node attribute prediction, as it is challenging and has a wide variety of applications, such as predicting future citation counts of scientific papers to help identify rising trends in research or find influential authors to keep track of, or find future interest in a product (which can then be used for advertisement, for example) by predicting the amount that will be purchased by a set of customers in the future. To evaluate the effects of the

[1] http://www.arxiv.org

relationship selection process we propose here, we compare the performance of a model that predicts future node attributes of a citation network without using any relationships defined on the network to its performance when these relationships are used, and show that the relationships we chose do in fact improve the accuracy of the model.

The rest of the paper is organized as follows: in Section 2 we discuss some of the work that has already been done in the field of node and link prediction in a bibliographic network setting, as well as the improvements we have made to our method since reporting the preliminary results at ASONAM conference. Section 3 provides the formal description of the GCRF model we use, and defines how we use variograms to reduce the computational complexity of the model by eliminating useless relationships beforehand. In Section 4 we describe the dataset used in our experiments, as well as explain all of the similarity measures that we used and evaluated using variograms and the GCRF model. Section 5 provides an analysis of our experimental setup, our usage of variograms, the evaluation measures we use, and the results we obtained. Finally, Section 6 consists of a summary of our findings, as well as future directions we intend to undertake with this project.

2 Related work

Bibliographic networks have been analyzed in the past, with different avenues of exploration in mind. The 2003 KDD Cup competition included a citation prediction portion for a set of physics papers, and the winning method was surprisingly simple as it ignored network structure completely, and made predictions as the average of a set of nearest neighbors of a paper, determined by the likeness of their citation histories [Manjunatha et al., 2003]. Castillo et al. [Castillo et al., 2007] extracted primarily author-related data from bibliographic networks and used them for predicting future citation counts, while ignoring the other types of data that can be taken from such networks with minimal effort. Long-term citation prediction for papers was also carried out by Yan et al. [Yan et al., 2012], who utilized several types of features extracted from bibliographic networks in the context of Gaussian processes and classification trees. Our work intends to examine even more types of data that can be extracted from a bibliographic network, as well as their combinations, and furthermore do so on a finer temporal granularity, focusing on prediction in a future month rather than a future year.

Citation prediction has also been incorporated into research focusing on link prediction rather than node attribute prediction, aiming to answer the more difficult question of who will cite a paper rather than simply how many times it will be cited. While we focus on the node attribute prediction aspect of analysis, we feel it is important to highlight some of the work that has been done in this field here as they are primarily based on the idea of computing similarities between pairs of papers to discern future relationships, rather than focusing on features for papers independently of other papers in

the network. This is the same approach that we take for our work with variograms and the GCRF model, and we provide a very brief overview of some of the work in link prediction here. Shibata et al. [Shibata et al., 2012] implement a support vector machine method utilizing connections between pairs of papers based on various general network attributes of each paper to predict whether a link will form between the pair in the future. Bethard and Jurafsky [Bethard and Jurafsky, 2010] compute a variety of inter-paper relationships that include connections based on common topics, shared authors, and similar content. Yu et al. [Yu et al., 2012] develop a novel meta-path-based feature set that measures similarity between paper pairs based on their distance using different kinds of meta-paths, which use different kinds of network information.

The Conditional Random Field model was first introduced by Lafferty et al. [Lafferty et al., 2001], and was outlined nicely in [Sutton and Mccallum, 2007]. It was improved to allow for faster inference and consequently faster learning in [Liu et al., 2007], to accommodate continuous values rather than being restricted to discrete values in [Qin et al., 2008], and was recently used for regression of remote sensing data in [Radosavljevic et al., 2010]. Our work further extends the Conditional Random Field model adapted for regression, by tailoring the model to work with a bibliographic network in which the output can be regarded as a multivariate Gaussian distribution, allowing us to perform computations in a more computationally feasible way.

Ristovski et al. [Ristovski et al., 2013] have used a similar approach to create a fast high-dimensional Gaussian filtering method applied on an approximation of the CCRF model to dramatically reduce computation time for predicting the set of behaviors that are impacted by both dynamic individual-level evolution and effects from the entire grid for Aerosol dispersal. Their method scales very well with size and can handle significantly larger graphs than the standard GCRF method, but it is also rather limited as it only works with relationships that are in Euclidean space, and cannot handle a large number of relationships. Our approach is an ideal addition to this model as it can pre-emptively assess any number of relationships, and determine if any thresholds need to be imposed on the similarity types in order for them to be useful for the model. We intend to explore the integration of our approach with this state-of-the-art in future experiments.

In our preliminary work [Uversky et al., 2013] we applied a rudimentary variogram-based approach to pre-emptively analyze similarity measures before using them in the GCRF model. This involved analyzing variogram plots by hand, and choosing a small subset of similarities to include in the model. In this extension we automate this procedure by using a simple classifier to label similarities as good or bad, drastically reducing the amount of time required to apply the labels. We also improve on this selection method further by examining the correlations between the similarities that were labeled as "good" and removing those that are highly correlated, reducing the amount of redundant information being fed to the model. Finally, we conducted two additional sets of experiments to explore how our method and model perform

in the presence of noisy or missing data, both of which are problems frequently encountered in social networks.

3 Model Description

3.1 Multigraph Networks

In this section we broadly define the structure of a multigraph network, and how this structure fits into the context of our model. Say that we want to observe a set of N nodes over T time steps, with each node possessing a real-valued attribute whose value changes over time, and we are able to observe this value at each time step. In addition to this attribute, we are also able to observe different kinds of connections among the N nodes, with these connections also changing over time. Since there are multiple connections between each pair of nodes at each time step, the network possesses a multigraph structure. We now define our central problem as:

Given a set of nodes N, a history of node attribute values for each node up to time step t, and a history of connections among nodes up to time step t, predict the node attribute values for all N nodes at time step $t + 1$.

In our experiments N pertains to the set of papers we are interested in, T pertains to the snapshots of the network we are considering taken in monthly intervals, the node attributes refer to the number of times each paper has been cited at the given month, and the connections refer to different similarities, formally defined in Section 5, that exist between each pair of papers. Our ultimate goal is hence to predict the citation count for each paper in a set of papers in a future month with the highest accuracy, which involves first eliminating useless or even potentially harmful similarities with a variogram-based model.

3.2 Conditional Random Fields

By using a Conditional Random Field model we can model the distribution of an output variable conditioned on an input variable by incorporating different types of dependencies between the input and output variables, as well as between the output variable values for different nodes. The former dependency is modeled via association potential functions that are traditionally used in regression problems to connect input and output variables to predict output values in future time steps. In our model these association potential functions are represented with two unstructured predictors that connect the inputs and outputs.

In addition to examining the relationship between input and output variables, the Conditional Random Field model also allows for incorporating dependencies between output variables, which are represented with interaction potential functions. These functions may or may not also depend on the input

variable, and can be defined in a wide variety of ways depending on the context of the problem. In our case, these interaction potential functions measure the similarity between a pair of papers based on citation history similarities, citing author similarities, common terms used in the papers, and so on. Figure 1 illustrates a simplified graphical representation of the relationship types that are used by a Conditional Random Fields model.

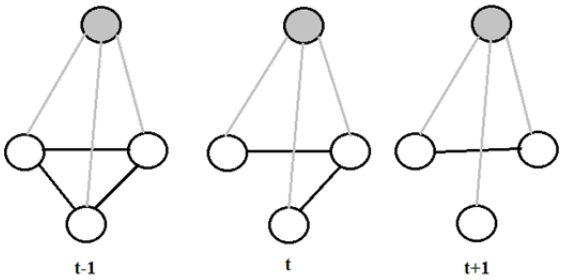

Fig. 1 Graphical representation of the relationships observed by a Conditional Random Field model over 3 time points. The gray nodes represent an input variable, the white nodes represent the corresponding output variables, the black links represent the interaction potential, and the gray links represent the association potential. Note that the strength of the ties among both inputs and outputs varies over time, and the ties can disappear completely if the strength is zero at a particular time point.

In order to model the conditional distribution of output vectors $y = (y_1 \dots y_N)$ on a set of input vectors $x = (x_1 \dots x_N)$, with association potential function $A(\alpha, y_i, x)$ where α is a K-dimensional set of parameters and interaction potential function $I(\beta, y_i, y_j, x)$ where β is a L-dimensional set of parameters, we represent the distribution as

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta})} exp(\sum_{i=1}^{N} A(\boldsymbol{\alpha}, y, \mathbf{x}) + \sum_{j \sim i} I(\boldsymbol{\beta}, y_i, y_j, \mathbf{x})) \tag{1}$$

where $j \sim i$ denotes the connected outputs y_i and y_j (connected with a black line at Figure 1) and where $Z(\mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ is the normalization function defined as

$$Z(\mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \int_y exp(\sum_{i=1}^{N} A(\boldsymbol{\alpha}, y, \mathbf{x}) + \sum_{j \sim i} I(\boldsymbol{\beta}, y_i, y_j, \mathbf{x}))dy \tag{2}$$

As already noted, A and I could be conveniently defined as linear combinations of a set of fixed features in terms of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$

$$A(\boldsymbol{\alpha}, y_i, \mathbf{x}) = \sum_{k=1}^{K} \boldsymbol{\alpha}_k f_k(y_i, \mathbf{x}) \tag{3}$$

$$I(\boldsymbol{\beta}, y_i, y_j, \mathbf{x}) = \sum_{l=1}^{L} \boldsymbol{\beta}_l g_l(y_i, y_j, \mathbf{x}) \tag{4}$$

We use features to define the model because it allows for a virtually inexhaustible set of connections between the input and output variables. Any feature or connection that we consider to be potentially useful can be included, and the parameter estimation portion of the model automatically assigns a weight to each feature to exhibit how useful it actually is for output prediction.

The learning task is to choose values of parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ to maximize the conditional log-likelihood of the set of training examples (we assume that interactions among outputs are defined over whole training set)

$$L(\boldsymbol{\alpha}, \boldsymbol{\beta}) = logP(\mathbf{y}|\mathbf{x}) \tag{5}$$

$$(\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}) = \arg\max_{\boldsymbol{\alpha},\boldsymbol{\beta}}(L(\boldsymbol{\alpha}, \boldsymbol{\beta})) \tag{6}$$

The inference task is to find the outputs y for a given set of observations x and estimated parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ such that the conditional probability $P(\mathbf{y}|\mathbf{x})$ is maximized

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}}(P(\mathbf{y}|\mathbf{x})) \tag{7}$$

The initial iteration of the Conditional Random Field model was used for classification, which is computationally a much simpler task since the required normalization function Z had to be a sum over a finite set of possible values, rather than an integral over a set of functions that have different definitions in different contexts. Since this requires Z to be an integrable function, using the model can prove to be difficult since this property can be exceedingly difficult or expensive to prove due to the complexity of the potential functions that may be used. In order to alleviate this concern, the underlying conditional probability $P(y|x)$ can be represented as a multivariate Gaussian distribution, resulting in Gaussian Conditional Random Fields that are outlined in the next section.

3.3 Gaussian Conditional Random Fields

The exponent portion E of the CRF model can be rewritten in Gaussian form as:

$$E = -\frac{1}{2}((\mathbf{y}-\mu)^T Q(\mathbf{y}-\mu)) = -\frac{1}{2}(\mathbf{y}^T Q\mathbf{y}) + \mathbf{y}^T Q\mu + const, \tag{8}$$

where $Q = \Sigma^{-1}$ and $\mathbf{b}$ are canonical parameters of Gaussian distribution. By representing the quadratic terms of y in the association and interaction potentials as $y^T Q_1 y$ and $y^T Q_2 y$ respectively, and combining them, we obtain

$$Q = 2(Q_1 + Q_2) \tag{9}$$

$$Q_{1ij} = \begin{cases} \sum_{k=1}^{K} \alpha_k, & i = j \\ 0, & i \neq j \end{cases} \tag{10}$$

$$Q_{2ij} = \begin{cases} \sum_k \sum_{l=1}^{L} \beta_l e_{ik}^l S_{ik}^{(l)}, & i = j \\ -\sum_{l=1}^{L} \beta_l e_{ij}^{(l)} S_{ij}^{(l)}, & i \neq j \end{cases} \tag{11}$$

where S_{ik} represents the similarity between outputs y_i and y_k, and $e_{ik} = 1$ if an edge exists between y_i and y_j, and $e_{ik} = 0$ otherwise.

To get μ, the linear terms in of the Gaussian form are matched with the linear terms in the exponent of the original form to get $\mu = \Sigma b$, where b is a vector with elements

$$b_i = 2\sum_{k=1}^{K} \alpha_k R_k(x) \tag{12}$$

Finally, using this new representation we see that

$$Z(\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{x}) = (2\pi)^{\frac{N}{2}} |\Sigma|^{\frac{1}{2}} exp(const) \tag{13}$$

and hence

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{(2\pi)^{\frac{N}{2}} |\Sigma|^{\frac{1}{2}}} exp(-\frac{1}{2}(\mathbf{y}-\mu)^T \Sigma^{-1}(\mathbf{y}-\mu)) \tag{14}$$

with Σ and μ defined above. The α and β values used in this model are obtained from the association potential defined by the outputs of 2 unstructured predictors, and the interaction potential defined by various similarity measures introduced below, respectively.

3.3.1 Learning and inference

The learning task is to choose α and β to maximize the conditional log-likelihood,

$$(\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}) = \underbrace{\arg\max}_{\boldsymbol{\alpha}, \boldsymbol{\beta}} L(\boldsymbol{\alpha}, \boldsymbol{\beta}), \;\; L(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \log P(\mathbf{y}|X) \tag{15}$$

To have a feasible model with real valued outputs, Z must be integrable, which is ensured by the constraint that all elements of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are greater than 0. In this setting, learning is a constrained optimization problem. To convert it to the unconstrained optimization, we adopt a technique used in [Qin et al., 2008] that applies the exponential transformation of the parameters to guarantee that they are positive. All parameters are learned by the gradient-based optimization. To apply it, we need to find the gradient of the conditional log-likelihood.

$$\begin{aligned} \frac{\partial P}{\partial \alpha_k} = &-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^T \frac{\partial Q}{\partial \alpha_k}(\mathbf{y}-\boldsymbol{\mu}) + \\ &(\frac{\partial b}{\partial \alpha_k} - \boldsymbol{\mu}^T \frac{\partial Q}{\partial \alpha_k})(\mathbf{y}-\boldsymbol{\mu}) + \frac{1}{2} Tr(Q^{-1}\frac{\partial Q}{\partial \alpha_k}) \end{aligned} \tag{16}$$

$$\frac{\partial P}{\partial \beta_k} = -\frac{1}{2}(\mathbf{y}+\boldsymbol{\mu})^T \frac{\partial Q}{\partial \beta_k}(\mathbf{y}-\boldsymbol{\mu}) + \frac{1}{2} Tr(Q^{-1}\frac{\partial Q}{\partial \beta_k}) \tag{17}$$

$$\boldsymbol{\mu} = Q^{-1}\mathbf{b} \tag{18}$$

The inference task is to find the outputs $\mathbf{y}$ for a given inputs X, such that the conditional probability $P(\mathbf{y}|X)$ is maximized. The GCRF model is Gaussian and, therefore, the maximum a posteriori estimate of $\mathbf{y}$ is obtained as the expected value $\boldsymbol{\mu}$ of the GCRF distribution,

$$\mathbf{y}_* = \underbrace{\arg\max}_{y} P(y|X) = \boldsymbol{\mu} \tag{19}$$

Uncertainty for each output can be taken as corresponding element from the diagonal of covariance matrix.

3.4 Variograms

Although the Gaussian Conditional Random Field model is able to both incorporate multiple similarity types, and assign weights to each similarity based on their utility, it can nonetheless become computationally expensive if every conceivable similarity type (as well as combinations of similarity types) is simply thrown into the model, especially if many of these similarities are dense. As such, we first perform an exhaustive analysis of all similarity types, that we constructed for the network, as well as all pairwise combinations of these similarities, by examining them in a variogram-based binary classifier that labels each similarity as either good (potentially useful for predictive purposes) or bad (useless or potentially harmful for predictive purposes).

We accomplish this by computing similarity and variance values for each pair of papers at each time point in our selected subset, and determining whether the similarity-variance relationship is useful over all pairs of nodes in the our network. This decision is based on the intuition that a high similarity which effectively captures a meaningful connection between a pair of nodes will correspond to a lower variance in the node attribute values of the two nodes. In terms of the variogram plot, we want to look for similarities whose variograms exhibit an exponential decay-like behavior. Furthermore, if we take the overall variance of the entire network as a threshold, we can say that a similarity is potentially useful if it shows this decaying behavior, and a significant portion of its variogram is below this threshold. Since many of the variogram plots contain a major spike at the beginning which drops off with higher similarity values, we can also use the intercept of the overall variance line and the variogram plot as a threshold, and set values of the similarity below that threshold to zero. This effectively removes the less helpful portion of the similarity at its lowest values, and has the added benefit of making the similarity more sparse, which improves computation time.

We incorporate both of the ideas described above in a simple classifier that computes the similarity and variance values for all pairs of papers over a pre-established time period for each similarity measure of interest, and labels the similarity as either useful or useless. The decision is made by determining the portion of the plot that is below the overall variance line, and comparing that against a threshold that was determined after inspecting the plots by hand.

Overall, the variogram plots fall into two easily recognizable categories which are consistent with our intuitions of how good similarities should behave: the plot either has a clear decreasing trend that falls below the overall variance line and stays below it, or it does not have a consistent pattern and/or stays primarily above the line. *We computed the percentage of the variogram plot that is below the overall variance line, and discarded any similarity whose corresponding percentage was below a certain threshold. We experimented with different threshold values by incrementing it by 10% and found 50% to be the sweet spot: higher values of the threshold (60% and above) resulted in several of the similarities to be labeled as "bad" when we knew that they were at least potentially useful by inspecting them manually, but they received a "bad" label due to a peak at the first few points which we discarded later in the process anyway. On the other hand, since many of the similarities had completely irregular variogram plots and hence oscillated both above and below the overall variance line (which is exactly the behavior we wanted to avoid), lower values of the threshold (40% and below) resulted in many of these plots to be labeled as "good" due to this oscillating behavior. When we examined the performance of these irregular similarities we found that they offered no benefit to the model despite their "good" label, so we settled on a value of 50% for the threshold since it offered a good balance of keeping potentially useful similarities while weeding out most of the bad ones.* Finally, we applied a simple correlation-based approach, described in more detail in Section 5.2, to further reduce the number of similarities by removing highly-correlated ones.

4 Dataset and Feature Description

4.1 Data

The dataset we used for our experiments was the high energy physics theory bibliographic network which was extracted from arXiv for the 2003 KDD Cup competition. The network consists of 29,955 papers and 352,807 citations spanning over 11 years, and the dataset includes text versions of all papers that can be used to extract additional information about each paper. An XML version of this dataset, from Proximity HEP-Th database, which included most of the metadata available from the full texts of the papers was used to quickly extract the information that was used in our experiments. The Proximity HEP-Th database is based on data from the arXiv archive and the Stanford Linear Accelerator Center SPIRES-HEP database provided for the 2003 KDD Cup competition with additional preparation performed by the Knowledge Discovery Laboratory, University of Massachusetts Amherst. The citation pairs were used to construct a citation history matrix, which represents the number of citations that a particular paper has received at a particular time point.

4.2 Similarity Measures

The various types of information extracted from the XML file were used to compute a number of similarity features for a pair of papers, which were then used as the interaction potential functions within the GCRF framework to augment the outputs of the unstructured predictors. The following ten similarity measures were used in our experiments:

4.2.1 coCiter

Jaccard based similarity measure. Similarity between two papers A and B at time point t is expressed as:

$$Sim_{coCiter}(A,B) = \frac{2 \times \#of\ cocitations\ of\ A\ and\ B\ at\ (t-1)}{\#of\ cit.\ of\ A\ at\ (t-1) + \#of\ cit.\ of B\ at\ (t-1)} \tag{20}$$

4.2.2 history

$$Sim_{history} = \exp^{-\frac{d(A,B)^2}{k}} \tag{21}$$

where

d(A,B) is the Euclidean distance between the citation counts of papers A and B over a history of a particular length (in this case 8), and $k = \sum_t \sum_n \frac{d(A,B)^2}{N \times (N-1) \times T}$ N and T are number of all papers and all timesteps respectively.

4.2.3 term v1

The rest of the similarity measures are all based on the classic TF-IDF term scoring from information retrieval, which were used in [Bethard and Jurafsky, 2010]:

$$tf(t,d) = |\{t' \in terms(d) : t' = t\} \tag{22}$$

$$idf(d) = 1 + log\frac{|D|}{|d \in D : t \in terms(d)| + 1} \tag{23}$$

$$score_{terms}(q,d) = \sum_{t \in q} tf(t,d)^{0.5} idf(d)^2 \tag{24}$$

This score increases when terms (t) are shared between the query (q) and the document (d), but terms that appear in many documents in the collection (D), such as *the*, are heavily discounted.

In order to use this score (which is asymmetric) as a similarity measure, we transform it into a symmetric measure in several ways.

$$Sim_{termv1}(A,B) = \frac{score_{terms}(A,B) + score_{terms}(B,A)}{2} \tag{25}$$

4.2.4 term v2

In this case the same score was calculated, but cosine similarity is used

$$Sim_{termv2}(A,B) = \frac{\mathbf{A}\cdot\mathbf{B}}{\|\mathbf{A}\|\,\|\mathbf{B}\|} = \frac{\sum_{i=1}^{tAB} w_{i,A}w_{i,B}}{\sqrt{\sum_{i=1}^{tA} w_{i,A}^2}\sqrt{\sum_{i=1}^{tB} w_{i,B}^2}} \tag{26}$$

where tAB is the set of terms common to papers A and B, tA is the set of terms in paper A, tB is the set of terms in paper B, and $w_{i,A} = tf(i,A)^{0.5}idf(A)^2$.

4.2.5 author v1

For authors the equations remain the same

$$Sim_{authorv1}(A,B) = \frac{score_{authors}(A,B) + score_{authors}(B,A)}{2} \tag{27}$$

$score_{authors}$ is calculated the same way as $score_{terms}$, except authors are considered as terms.

4.2.6 author v2

In this case the equations are also the same as for terms:

$$Sim_{authorv2}(A,B) = \frac{\mathbf{A}\cdot\mathbf{B}}{\|\mathbf{A}\|\,\|\mathbf{B}\|} = \frac{\sum_{i=1}^{aAB} w_{i,A}w_{i,B}}{\sqrt{\sum_{i=1}^{aA} w_{i,A}^2}\sqrt{\sum_{i=1}^{aB} w_{i,B}^2}} \tag{28}$$

where aAB is the set of authors common to papers A and B, aA is the set of authors of paper A, aB is the set of authors of paper B, $w_{i,A} = tf(i,A)^{0.5}idf(A)^2$, and i is author rather than term.

4.2.7 author v3

In the following two similarity measures tfidf scores of terms/author ids are calculated for the citers of papers that we compare, and averaged over the two papers we compare. The temporal aspect is satisfied because we aggregate all the citers up to the current time point:

$$aggtemp(A,B) = score_{authors}(authors(A), \operatorname*{concat}_{d\in citing(B)}(d)) \tag{29}$$

$$Sim_{authorv3}(A,B) = \frac{aggtemp(A,B) + aggtemp(B,A)}{2} \tag{30}$$

4.2.8 term v3

$$aggtemp(A,B) = score_{terms}(A, \operatorname*{concat}_{d\in citing(B)}(d)) \tag{31}$$

$$Sim_{termv3}(A,B) = \frac{aggtemp(A,B) + aggtemp(B,A)}{2} \tag{32}$$

4.2.9 author v4

This score is exactly the same as the above author temporal score, but we aggregate the citing authors up to the first time point we consider.

4.2.10 term v4

This score is exactly the same as the above term temporal score, but we aggregate the terms used by the citing papers up to the first time point we consider.

4.3 Unstructured Predictors

To represent the association potential among the inputs and outputs of the GCRF model we used two simple unstructured predictors:

4.3.1 k-Nearest-Neighbor (kNN)

A sliding window kNN predictor that forms predictions for a paper's citation counts in a future time point by comparing its citation history to that of citation histories of other papers in the dataset, selecting the k papers that have the most similar history, and averaging their final corresponding citation counts to predict the count for the paper in question. After testing several configurations we selected a predictor using a window of size 8 and a k value of 9.

4.3.2 Multiple Linear Regression (MLR)

MLR predictor whose coefficients were trained on the features of all papers up to the time point we were interested in, and then applying those coefficients on the features at the given time point to form the prediction. For this predictor we used a history of 3 time points as the feature set.

5 Experiments

5.1 Experimental Setup

In order to avoid the problem of sparsity, and allow the model enough known values to learn proper parameter values, we first organized the data so that we were observing only papers that were written before year 2000, and tracking their citation counts starting at year 2000. We then filtered out papers that received less than 25 citations over the resulting 40 month period, leaving us with a matrix of 40 time steps and the citation counts for the 800 most-cited papers at each of those time points. Note that although we focused on the 40

time steps that took place after the year 2000 for training the GCRF model, we still had the citation counts for the previous time steps which we used when training the unstructured predictors.

Because we wanted to avoid training the unstructured predictors concurrently with the GCRF model, we opted to use the first 9 time steps only for unstructured predictors and use the remaining 31 time points for the GCRF model. For the kNN predictor, we used the first 9 time points as the only interval which we could use to find neighbors based on citation history. For the MLR predictor, we only used the first 9 time points of the data to learn the coefficients of the model, and used them to predict in the remaining time points. We experimented with various allotments of the remaining 31 time points into training and testing portions, and found several patterns. Predicting on some of the months was noticeably more difficult for both the unstructured predictors and the GCRF model (overall performance for these months was more than 20% lower than average). The later months also showed to be more difficult for all models, and though the drop in performance wasn't quite as noticeable it was consistent. Increasing the length of the training period did not hinder or improve performance in any meaningful way, but it did increase the time required for the GCRF model to learn its parameters. Therefore, we opted to report our results when training the GCRF model on time points 10-20, and testing them on time points 21-30. Testing on 10 points allows us to somewhat offset the drop in performance that some of the months in this period seem to exhibit with other months that are easier to predict, and we avoid testing on the last 10 months as they seem to be slightly more difficult to predict overall.

5.2 Variograms of Similarity Measures

Once we selected a manageable and sensible subset of papers to be examined, we used the XML file to compute the previously described similarity measures for each pair of papers. After computing all 10 similarities, we applied the simple classifier described in Section 3.4 on all of these similarities, as well as all pairwise combinations of these similarities. Of the 55 similarities that we evaluated with this simple classifier, only 9 received a label of "good", considerably reducing the number of similarities we have to consider for the main model.

After we pruned the pool of similarities to 9, we performed an additional analysis to further reduce the number of similarities and also remove some that may contain mostly redundant information (since we examined the pairwise combinations of all of the similarities, 7 of the 9 good similarities were in fact combinations of similarities that had some overlap). To accomplish this, we first ranked the 9 good similarities according to the portion of their respective variogram which was below the overall variance line, which we computed in the previous step. We then computed the correlation between the highest-ranked similarity and all others by averaging the correlations at each time step for

each pair of similarities. We removed any similarity whose correlation with the highest-ranked similarity was above 0.8 to reduce the effect of having highly related information in some of the similarities. *We varied the threshold in increments of 0.1, and found 0.8 to be the ideal threshold. At higher values the method only kept the two standalone similarities, and we wanted to keep at least several combinations to examine their effectiveness in the model. On the other hand, lower values of the threshold resulted in keeping several similarities in the list which, when tested, were shown to offer no benefit to the model. These similarities received a label of "good" in the previous step due to being a combination of one of the truly useful similarities, but in fact did not offer any useful information. Setting the threshold at 0.8 kept the best standalone similarities as well as two combinations which offered some use, while removing the highly correlated similarities which were of no use.*

Figure 2 shows a visual representation of two bad similarity measures that don't exhibit the two primary intuitions we are looking for (decreasing trend and falling below the overall variance of the data, represented by the horizontal line.) Figure 3 displays two similarity measures that were deemed good and used in the GCRF model, which exhibit the behavior we were looking for. Note that although no similarity measure was consistently below the line of overall variance, we focused on finding the intercept of the two plots and using that as a threshold value to reduce the meaningless portions of the similarity value to zero. This allowed us to find the hidden utility of some of the similarity measures; in particular, the coCiter similarity represented in the figure was useless until we realized that we need to change the lower similarity values to zero, and the threshold to do so was obtained with variograms.

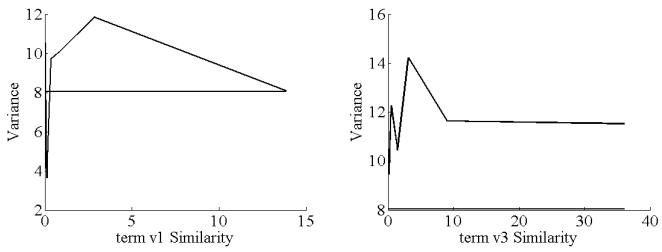

Fig. 2 Examples of variograms of "bad" similarity measures. The horizontal lines represent the overall variance of the dataset, indicating that regardless of the similarity strength, neither of these similarities are useful for identifying a relationship between the true values.

Finally we conducted a series of experiments to explore the performance of the GCRF model using different similarity measures both independently and in combination, as well as comparing those results to those obtained from using GCRF with only one of the two unstructured predictors to help determine how much the similarities were actually helping.

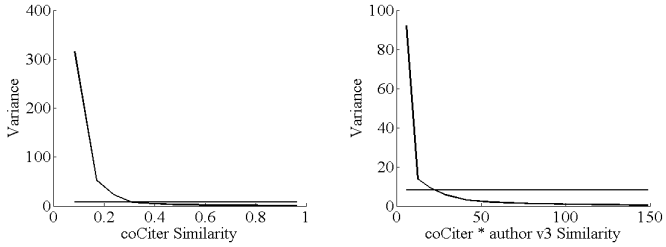

Fig. 3 Examples of variograms of "good" similarity measures. Note that after a certain similarity value, the respective variance drops below the overall variance of the data, indicating that higher similarity values help identify the relationship between outputs. These variograms also help identify the cutoff point for each similarity measure: for example, by setting any value of coCiter * author v3 Similarity below 20 to zero, we ensure that the similarity only identies the meaningful relationships among the data points.

5.3 Effects of Noisy and Missing Data

In order to examine how well the model performs when the network it is being applied to is not very stable (which is very often the case), we conducted another set of experiments by altering the core network we were examining with noise and with missing values. *In order to simulate the real-world effects of noisy data in the context of a citation network, we generated a random negative offset for the number of citations (taken from the original network that we have worked with) for each paper. The offset was generated from a normal distribution with mean 0 and several values of standard deviation, from 0.5 to 3. Such a type of noise might occur in a citation network if, for example, one or multiple journals are not included in the collection of papers being analyzed, due to being a late issue or simply being forgotten for one reason or another, and as a result the perceived number of citations for a paper is lower than the true value. This created a "noisy" version of the network that was less stable than the original network, and hence was a more robust test of the effects of network structure offered by the GCRF model.* We also conducted a set of experiments in which we randomly removed varying portions (from 5% to 40%) of each paper's citation counts. In order to avoid unfairly hampering the performance of the unstructured predictors, instead of treating the new "missing" values as 0, we simply discard any instance containing a missing value when training the unstructured predictors.

5.4 Evaluation Measures

To gauge the performance of the GCRF model as well as the unstructured predictors by themselves, we opted to use a traditional evaluation measure for regression tasks:

5.4.1 R^2 coefficient of determination

– a goodness-of-fit measure that displays how closely the output of the model matches the actual value of the data. A score of 0 indicates a very poor matching, while a score of 1 indicates a perfect match.

$$1 - \frac{\sum_i (y_i - f(x_i))^2}{\sum_i (y_i - y_{average})^2}$$

where $f(x_i)$ is the predicted value, y_i is the true value, and $y_{average}$ is the average of y values. Since we are concerned with the computational complexity of the model and how it is affected by the number of similarities used, we also include the runtime of the model for the main set of experiments to determine whether similarity pruning is necessary at all.

5.5 Results

In order to accurately gauge the effectiveness of the different similarity measures we performed a number of experiments using the parameters defined in the above sections. This includes focusing on a set of 800 papers over the span of 40 months and using months 10-20 to train the model and months 21-30 to test it. Using these parameters we measured the performance of the unstructured predictors by themselves, the performance of the GCRF model using randomly selected sample of bad similarity measures, using each of the good similarity measures individually, as well as some combinations of the good similarity measures. Because the trends of performance with bad similarity measures were consistent over the sample of bad similarity measures, namely, the performance of the GCRF model using bad similarity measures was no better than the MLR unstructured predictor (meaning that the similarity was of no use and hence its labeling as "bad" was accurate), we only show the results of using two of them.

We also observe the performance of GCRF using these similarity measures and only one of the two unstructured predictors to better define the impact of the similarity measures on the individual predictors. The results are organized as follows: in Table 1 the first two rows are the performance results of the individual unstructured predictors; the next two rows are the performance results of our GCRF model using both unstructured predictors and one "bad" similarity measure each. The results in Tables 2 and 3 are trios of GCRF results using the specified similarity measure(s) and both unstructured predictors, just MLR, and just kNN, respectively. Table 2 exhibits the performance of the 4 "good" similarity measures that were left after our filtering approach individually, and Table 3 shows our results with several combinations of similarities. Although the improvements we see appear fairly minor, they are statistically significant, and directly show the utility of adding network structure via similarities. Finally, *figures* 4 *and* 5 *show the results of running the models on data with artificially inserted noise and missing values.*

Table 1 Performance of individual unstructured predictors, as well as GCRFs that used two "bad" similarity measures. Note that there were significantly more "bad" similarity measures, but as the results from using them with GCRF were identical to the results shown here we omit them.

Model	Average R^2
MLR	0.67
kNN	0.58
GCRF term v3	0.67
GCRF author v1	0.67

The results shown here confirm the validity of our variogram-based selection of similarities. Table 1 displays two findings: the GCRF model performs at least as good as the best of its constituent unstructured predictors, which is understandable as it places less weight on the worse predictor, which is our case kNN. The table also confirms the labeling of the termTemporal and author similarities as "bad" by our variogram method, as these similarities offer no improvements in accuracy while taking a considerably longer amount of time (roughly a factor of 10) to run due to being part of the GCRF model.

Table 2 Performance of using a single "good" similarity measure in the GCRF model. Here each column represents using a different unstructured predictor setup with the given similarity measure: "both" means GCRF is trained using both MLR and kNN and the similarity measure, "MLR" means GCRF is trained using only MLR as the unstructured predictor, and "kNN" means only kNN was used as the unstructured predictor.

Similarity Measure	Average R^2 both	Average R^2 MLR	Average R^2 kNN	Runtime both (s)
coCiter	0.71	0.70	0.64	91
author v1 * CoCiter	0.69	0.68	0.59	82
history	0.68	0.67	0.58	89
term v3 * History	0.68	0.67	0.58	94

Table 2 also displays two primary findings: firstly, we can see that using both unstructured predictors is always at least as good as, or better than, using just the best unstructured predictor, suggesting that the model is able to extract at least something useful from an otherwise inferior predictor. We can also see the benefits of the similarity measures that we deemed "good" with the variogram method. Although there are varying degrees of improvement, it is evident that a good similarity measure is able to improve the performance of the model even when we only use a single inferior unstructured predictor.

We can see the effects of combining several similarities in the GCRF model in Table 3. The first evident effect of including multiple similarities in the GCRF model is the drastic increase in runtime. All of the similarities are precomputed prior to using them in the learning step of the GCRF model, so the increase in runtime is solely due to the amount of time it takes to learn the

Table 3 Performance of GCRF using three different combinations of good similarities.

Similarity Combinations	Average R^2 both	Average R^2 MLR	Average R^2 kNN	Runtime both
coCiter, history	0.71	0.69	0.63	343
author v1 * CoCiter, term v3 * History	0.69	0.67	0.60	411
coCiter, history, author v1 * CoCiter, term v3 * History	0.69	0.69	0.61	5229

alpha and beta parameters. Adding one extra similarity to the model causes the runtime to increase by roughly a factor of 4; adding three similarities slows the model by a factor of roughly 50. This drastic increase in complexity is one of the reasons that we opted to pre-emptively filter the pool of potentially useful similarities for use in the GCRF model, even though the model can conceptually handle any number of similarities used concurrently. We can also see that there does not appear to be any benefit to using several similarities of different types in the GCRF model, even though they were shown to improve the performance of the model over the unstructured predictors when used individually. Furthermore, we can see that the performance actually decreases when all 4 similarities are used (when compared to the best-performing individual similarity). We are still exploring the reason behind this effect, but, it is possible that the reason behind this is a conflict between the similarities; that is, the model assigns similar trust to each of these similarities but they lead to different predictions, so the model compromises and as a result the performance of the combination is worse than the performance of the best individual similarity.

Finally, in *figures* 4 *and* 5 we can see the effects of applying noise and missing values to the data to test the effectiveness of the model on a less stable network. The *figures* display the performance of the unstructured predictors by themselves at various levels of noise and missingness, as well as the performance of the GCRF model at the same levels. The kNN predictor is consistently worse than the MLR predictor, which makes sense in the presence of noise and missing values, since the neighbors being found for each paper change drastically as the data becomes less stable. MLR, on the other hand, is able to perform better since it uses the whole data (rather than just a small number of neighbors), and hence the impact of missing values and normal noise is averaged out and ends up being less harmful. Since we are concerned purely with examining how much the network structure incorporated into the GCRF model helps the unstructured predictors, we only show the performance of the GCRF model with the best single similarity measure (coCiter) and both un-

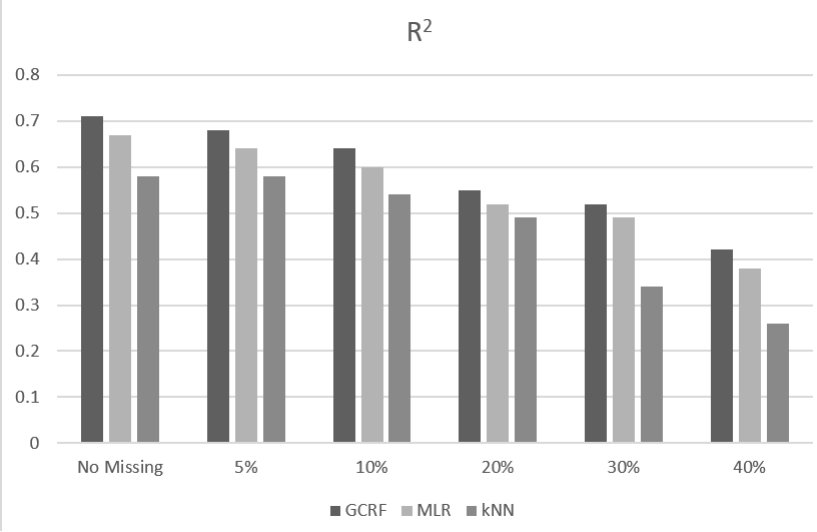

Fig. 4 *Effects of injecting missing values into the data. The percentage indicates the portion of citation counts counted as "missing" for all papers.*

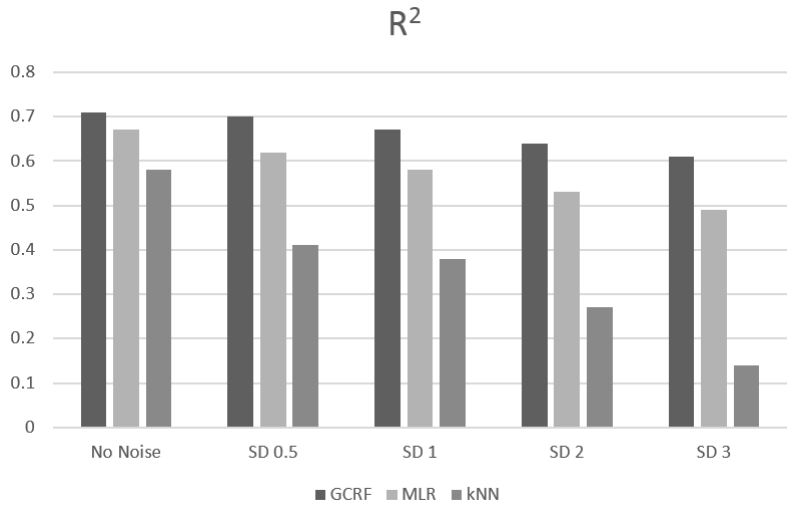

Fig. 5 *Effects of adding a normally distributed amount of noise to the citation counts of each paper. A mean of 0 and several different values of Standard Deviation (SD) were used.*

structured predictors for comparing against the unstructured predictors. The results shown here further exhibit the utility of the model: while the unstructured predictors and GCRF show an understandable decrease in performance as the amount of noise and missing values increases, we can see that GCRF

shows consistent improvements over the unstructured predictors. This effect is most evident at the highest levels of noise and missingness; when the standard deviation of the noise is 3, the drop in performance for GCRF is less than half of that of the unstructured predictors, showing the utility of the network connections.

6 Conclusion and Future Work

In this paper we examined the issue of selecting effective relationships from a large pool of potential connections among nodes in a social network. These relationships were first examined by plotting their respective variograms and observing how the relationship strength relates to the actual values of the data , and then filtered for usefulness and redundancy removal. We then tested the benefits of these relationships when they were implemented in our Gaussian Conditional Random Field model that was adapted for node attribute prediction in temporal social networks with a multigraph structure.

The variogram method we have devised here is entirely network-independent, although we only apply it to a single citation network. The method can be applied to any social network which possesses several types of connections between nodes, and a target node variable that is of interest. It is most useful when the network is temporal, and the connections between nodes vary over time, but it could just as easily be applied to a static network. We defined a number of intuitive and previously studied similarities based on the connections in our citation network for the variogram method, but it could just as easily be applied to any other network with any other connection types, since it is based only on the effects of the connections on the variance in the target variable.

Our experiments confirmed that many of the available similarity measures are in fact harmful, and including them in the model will hurt both the runtime and accuracy. While we were successful in pre-emptively weeding out these bad similarity measures using our variogram approach, we also noticed that some of the good similarity measures we left seemed to impede performance when combined with other good similarities. This suggests that we need a more robust method of analyzing variograms, which is currently being done via several thresholds that are not computed dynamically. We intend to explore this issue in future experiments, and to continue improving the model by adding more complex unstructured predictors and additional similarity measures in other social networks.

Acknowledgement

We are very grateful to the anonymous reviewers whose excellent suggestions really helped us revise the initial submission of this work.

This work is supported in part by DARPA Grant FA9550-12-1-0406 negotiated by AFOSR.

References

[Bethard and Jurafsky, 2010] Bethard, S. and Jurafsky, D. (2010). Who should i cite: learning literature search models from citation behavior. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 609–618, New York, NY, USA. ACM.

[Castillo et al., 2007] Castillo, C., Donato, D., and Gionis, A. (2007). Estimating number of citations using author reputation. In *Proceedings of the 14th international conference on String processing and information retrieval*, SPIRE'07, pages 107–117, Berlin, Heidelberg. Springer-Verlag.

[Lafferty et al., 2001] Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

[Liu et al., 2007] Liu, C., Adelson, E. H., and Freeman, W. T. (2007). Learning gaussian conditional random fields for low-level vision. In *In Proc. of CVPR*, page 7.

[Manjunatha et al., 2003] Manjunatha, J. N., Sivaramakrishnan, K. R., Pandey, R. K., and Murthy, M. N. (2003). Citation prediction using time series approach kdd cup 2003 (task 1). *SIGKDD Explor. Newsl.*, 5(2):152–153.

[Ouzienko et al., 2011] Ouzienko, V., Guo, Y., and Obradovic, Z. (2011). A decoupled exponential random graph model for prediction of structure and attributes in temporal social networks. *Stat. Anal. Data Min.*, 4(5):470–486.

[Qin et al., 2008] Qin, T., Liu, T. Y., Zhang, X. D., Wang, D. S., and Li, H. (2008). Global Ranking Using Continuous Conditional Random Fields. In Koller, D., Schuurmans, D., Bengio, Y., Bottou, L., Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L., editors, *NIPS*, pages 1281–1288. MIT Press.

[Radosavljevic et al., 2010] Radosavljevic, V., Vucetic, S., and Obradovic, Z. (2010). Continuous conditional random fields for regression in remote sensing. In *Proceedings of the 2010 conference on ECAI 2010: 19th European Conference on Artificial Intelligence*, pages 809–814, Amsterdam, The Netherlands, The Netherlands. IOS Press.

[Ristovski et al., 2013] Ristovski, K., Radosavljevic, V., Vucetic, S., and Obradovic, Z. (2013). Continuous conditional random fields for efficient regression in large fully connected graphs. In desJardins, M. and Littman, M. L., editors, *AAAI*, pages 840–846. AAAI Press.

[Shibata et al., 2012] Shibata, N., Kajikawa, Y., and Sakata, I. (2012). Link prediction in citation networks. *J. Am. Soc. Inf. Sci. Technol.*, 63(1):78–85.

[Sutton and Mccallum, 2007] Sutton, C. and Mccallum, A. (2007). An introduction to conditional random fields for relational learning. In Getoor, L. and Taskar, B., editors, *Introduction to Statistical Relational Learning*.

[Uversky et al., 2013] Uversky, A., Ramljak, D., Radosavljevic, V., Ristovski, K., and Obradovic, Z. (2013). Which links should i use?: a variogram-based selection of relationship measures for prediction of node attributes in temporal multigraphs. In Rokne, J. G. and Faloutsos, C., editors, *ASONAM*, pages 676–683. ACM.

[Walter et al., 2008] Walter, F. E., Battiston, S., and Schweitzer, F. (2008). A model of a trust-based recommendation system on a social network. *Autonomous Agents and Multi-Agent Systems*, 16(1):57–74.

[Yan et al., 2012] Yan, R., Huang, C., Tang, J., Zhang, Y., and Li, X. (2012). To better stand on the shoulder of giants. In *JCDL*, pages 51–60.

[Yang et al., 2006] Yang, W.-S., Dia, J.-B., Cheng, H.-C., and Lin, H.-T. (2006). Mining social networks for targeted advertising. In *Proceedings of the 39th Annual Hawaii International Conference on System Sciences - Volume 06*, HICSS '06, pages 137.1–, Washington, DC, USA. IEEE Computer Society.

[Yu et al., 2012] Yu, X., Gu, Q., Zhou, M., and Han, J. (2012). Citation prediction in heterogeneous bibliographic networks. In *SDM'12*, pages 1119–1130.