iPath: Forecasting the Pathway to Impact

Liangyue Li [*] Hanghang Tong [*] Jie Tang [†] Wei Fan [‡]

Abstract

Forecasting the success of scientific work has been attracting extensive research attention in the recent years. It is often of key importance to foresee the pathway to impact for scholarly entities for (1) tracking research frontier, (2) invoking an early intervention and (3) proactively allocating research resources. Many recent progresses have been seen in modeling the long-term scientific impact for *point prediction*. However, challenges still remain when it comes to *forecasting the impact pathway*. In this paper, we propose a novel predictive model to collectively achieve a set of design objectives to address these challenges, including prediction consistency and parameter smoothness. Extensive empirical evaluations on real scholarly data validate the effectiveness of the proposed model.

1 Introduction

The emerging research area on the "science of science" (e.g., understanding the intrinsic mechanism that drives high-impact scientific work, foreseeing the success of scientific work at an early stage), has been attracting extensive research attention in the recent years, most of which are centered around the citation counts of the scholarly entities (e.g., researchers, venues, papers, institutes) [20, 14, 22]. From the prediction perspective, more often than not, it is of key importance to forecast the pathway to impact for scholarly entities (e.g., how many citations a research paper will attract in each of several consecutive years in the future). The impact pathway often provides a good indicator of the shift of the research frontier. For instance, the rapid citation count increase of the deep learning papers reveals an emerging surge of this topic. The impact pathway can also help trigger an early intervention should the impact trajectory step down in the near future. Research resources could be more judiciously allocated if the impact pathway can be forecast at an early stage. For example, the research management agency could proactively allocate more resources to those rising fields.

The state of the art has mainly focused on modeling the long-term scientific impact for the early prediction. For example, Wang et al. [22] integrate preferential attachment, a temporal citation trend and the underlying "fitness" of the paper into designing a generative model for the citation dynamics of individual papers. Yan et al. [24] focus on designing effective scholarly features (e.g., content features, author features, venue features) for the future citation count prediction. Li et al. [14] propose a joint predictive model to encourage similar research domains to share similar model parameters.

Despite their own success, all the existing work on impact forecasting are essentially for *point prediction*, to predict the number of cumulative citations of a paper in the future. They are not directly applicable to forecasting the impact pathway, e.g., citation counts in each of the next 10 years. One baseline solution is to treat the impacts across different years independently and to train a separate model for each of the impacts. This treatment might ignore the inherent relationship among different impacts across different years, and thus might lead to sub-optimal performance. Having this in mind, a better way could be to apply the existing multi-label learning [29] or multi-task learning [6] methods to exploit the relation among impacts across different years. Nonetheless, these general-purpose multi-label/multi-task learning approaches might overlook some unique characteristics of the impact pathway prediction, which is exactly the focus of this paper.

In this paper, we aim to develop a new predictive model tailored for scholarly entity impact pathway prediction. To be specific, our model will focus on the following two design objectives:

- **D1. Prediction Consistency.** Intuitively, the scholarly impacts at certain years might be correlated with each other, which, if vetted carefully, could boost the prediction performance (i.e., multi-label or multi-task learning). Here, one difficulty for impact pathway prediction is that such a relation structure is often not accurately known a prior. Thus a good predictive model should be capable of simultaneously inferring the impact relation structure from the training data and leveraging such (inferred) relation to improve the prediction performance.

[*] Arizona State University. Email: liangyue@asu.edu; hanghang.tong@asu.edu

[†] Tsinghua University. Email: jietang@tsinghua.edu.cn

[‡] Big Data Labs - Baidu USA. Email: fanwei03@baidu.com

- **D2. Parameter Smoothness.** For a given feature of the predictive model, we do not expect its effect on the impacts of adjacent years would change dramatically. For instance, the effect of "fitness" defined in [22], capturing a scientific work's perceived novelty and importance, is unlikely to change greatly but rather gradually fade away over years. A good predictive model should be able to capture such temporal smoothness.

We propose a new predictive model (*iPath*) to simultaneously fulfill these two design objectives. First, we propose to exploit the prediction consistency (i.e., D1) in the *output* space. Second, to encode the parameter smoothness (i.e., D2) between adjacent time steps, we impose a linear transition process in the *parameter space* from one time step to the next. We formulate it as a regularized optimization problem and propose an effective alternating strategy to solve it. Our method is flexible, being able to handle both linear and non-linear models.

The main contributions of the paper can be summarized as follows:

- **Problem Definitions.** We define a novel scholarly impact pathway prediction problem, to predict the impact of a scholarly entity at several consecutive time steps in the future.
- **Algorithm and Analysis.** We propose and analyze a new predictive model (*iPath*) for the impact pathway forecasting problem.
- **Empirical Evaluations.** We conduct extensive experiments to validate the effectiveness of the proposed algorithm.

The rest of the paper is organized as follows. Section 2 formally defines the pathway to impact forecasting problem. Section 3 introduces the proposed algorithm. Section 4 presents some analysis and comparison with existing work. Section 5 provides the experimental results. Section 6 reviews related work and Section 7 concludes the paper.

2 Problem Definition

In this section, we first present the notations used throughout the paper (summarized in Table 1) and then formally define the pathway to impact forecasting problem.

We use bold upper-case letters for matrices (e.g., $\mathbf{A}$), bold lowercase letters for vectors (e.g., $\mathbf{v}$), and lowercase letters (e.g., α) for scalars. For matrix indexing, we use a convention similar to Matlab's syntax as follows. We use $\mathbf{A}(i, j)$ to denote the entry at the intersection of the i-th row and j-th column of matrix $\mathbf{A}$, $\mathbf{A}(i, :)$ to denote the i-th row of $\mathbf{A}$ and $\mathbf{A}(:, j)$ to denote the j-th column of $\mathbf{A}$. Besides, we use prime for matrix transpose (e.g., $\mathbf{A}'$ is the transpose of $\mathbf{A}$).

Table 1: Symbols

Symbols	Definition
n	number of scholarly entities
d	feature dimension, i.e., number of time steps observed
l	length of the forecasting horizon into the future
$\mathbf{w}_i$	model parameter for predicting the i-th impact
$\mathbf{X}$	feature matrix
$\mathbf{Y}$	impact matrix
$\mathbf{A}$	adjacency matrix of the impact graph
$\mathbf{A}_0$	prior knowledge of the impact graph structure
$\mathbf{B}$	transition matrix
$\mathbf{K}$	kernel matrix
E	energy function
$\Phi_c(\cdot)$	the potential defined on a maximal clique c

For a given scholarly entity (e.g., research papers, researchers, conferences), after observing the impacts in the first few years, we want to forecast its impacts in the next several years (e.g., 10 or 20 years) into the future. Formally, denote $\mathbf{x} \in \mathbb{R}^d$ as the impacts observed in the first d time steps, we want to predict the impact pathway $\mathbf{y} = (y_1, y_2, \dots, y_l)'$ afterwards, where y_i is the citation count in the i-th future time step, and l is the length of the horizon we want to look into the future. Mathematically, the task is to learn a predictive function $f : \mathbf{x} \to \mathbf{y}$ from the training set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)|i = 1, 2, \dots, n\}$, where n is the number of training samples. For convenience, let $\mathbf{X}$ be the feature matrix by stacking all the features (i.e., impact values of the first d time steps) of the n scholarly entities as its rows, that is, $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]'$. Similarly, let $\mathbf{Y}$ be the impact matrix by stacking all the impacts (i.e., values of all the l future time steps) of the n scholarly entities as its rows, that is, $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n]'$.

With the above notations, we formally define the pathway to impact forecasting problem as follows:

Problem 1. *Pathway to Impact Forecasting*

Given: *feature matrix* $\mathbf{X}$ *and impact matrix* $\mathbf{Y}$ *of* n *scholarly entities.*

Predict: *the impacts in each of the continuous future time steps of a new scholarly entity.*

Remarks: At the high-level, this problem setting bears some similarities to the classic multi-label learning [29] or multi-task learning [6] (i.e., predicting each impact is treated as a task). Nonetheless, the impact pathway of a scholarly entity brings several unique characteristics as outlined in the Introduction, which in turn calls for a new method to solve it.

3 Proposed Algorithms

In this section, we present a predictive model to forecast the pathway to impact. We first formulate it as a regularized optimization problem, and then propose an effective alternating optimization algorithm to solve it.

3.1 *iPath* Formulations

Let us first summarize the key ideas behind our proposed formulation. First, we want to leverage the relation across the impacts at different time steps, so that closely related impacts are likely to have consistent predicted outputs. The relation among the impacts at different time steps is encoded in a non-negative matrix $\mathbf{A}$, where the entry $\mathbf{A}_{ij}$ is a large positive value if the i-th impact and j-th impact are closely related. The matrix $\mathbf{A}$ can be regarded as the weight matrix of the impact relationship graph, where vertices are impacts at different time steps and edge exists between two similar impacts. Second, one difficulty is that the impact relation might not be accurately known a prior. We address this by inferring a good relation that can benefit the prediction performance, while not deviating too far from the (noisy) prior knowledge of the relation. Third, as we mentioned in the problem definition, we focus on the impact pathway forecasting, where the effect of features on the impacts at adjacent time steps is expected to transition smoothly. To realize such smoothness, we impose a linear transition process $\mathbf{B}$ between model parameters of adjacent time steps $\mathbf{w}_t$ and $\mathbf{w}_{t+1}$.

Putting all the above aspects together, our model can be formulated as follows:

(3.1)

$$\begin{aligned}\min_{\mathbf{W},\mathbf{B},\mathbf{A}} \quad & \underbrace{\mathcal{L}[f(\mathbf{X},\mathbf{W}),\mathbf{Y}]}_{\text{empirical loss}} + \underbrace{\alpha\sum_{i=1}^{l}\sum_{j=1}^{l}\mathbf{A}_{ij}g(\mathbf{w}_i,\mathbf{w}_j)}_{\text{prediction consistency}} \\ & \underbrace{+\beta\sum_{t=2}^{l}\|\mathbf{w}_t-\mathbf{B}\mathbf{w}_{t-1}\|_2^2}_{\text{parameter smoothness}} \\ & \underbrace{+\gamma\|\mathbf{B}-\mathbf{I}\|_F^2+\delta\sum_{i=1}^{l}\Omega(\mathbf{w}_i)+\epsilon\|\mathbf{A}-\mathbf{A}_0\|_F^2}_{\text{regularizations}}\end{aligned}$$

where $\mathbf{W}$ is the parameter matrix of the prediction parameters for all the impacts as $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_l]$; $f(\mathbf{X},\mathbf{W})$ is the prediction function, which could be linear or non-linear, ; $\mathcal{L}(\cdot)$ is the empirical loss between the predicted impacts and actual impacts; $g(\mathbf{w}_i,\mathbf{w}_j)$ characterizes the prediction consistency between the i-th impact and the j-th impact; $\|\mathbf{w}_t-\mathbf{B}\mathbf{w}_{t-1}\|_2^2$ instantiates the parameter smoothness; the rest terms are regularizations on $\mathbf{B}$, $\mathbf{W}$ and $\mathbf{A}$ respectively; $\mathbf{A}_0$ is the noisy prior knowledge about the impact/label relation; and α, β, γ, δ and ϵ are the trade-off parameters.

Remarks: the second term models the prediction consistency. If the i-th impact and the j-th impact are similar, i.e., $\mathbf{A}_{ij}$ is a large positive number, then the function value $g(\cdot)$ that measures the consistency between the predicted values for the i-th and j-th impacts should be small. In addition, to address the challenge of inferring a good relation, we are learning a relation $\mathbf{A}$ in the model by regularizing it not to deviate too far from our prior knowledge of the impact relation ($\mathbf{A}_0$). The third term models the parameter smoothness by assuming a linear transition process between model parameters at two adjacent time steps. More specifically, the model parameter for time step t, $\mathbf{w}_t$ is close (in the form of Euclidean distance) to the model parameter for the last time step with some linear transition, $\mathbf{B}\mathbf{w}_{t-1}$. When $\mathbf{B}$ is an identity matrix, such smoothness will be a small Euclidean distance between the two parameters themselves. Our model will learn the model parameters $\mathbf{W}$, linear transition process $\mathbf{B}$ and the impacts relation $\mathbf{A}$ jointly.

iPath – linear formulation: in the linear case, the predictions are made by a linear weighted combination of the features, where the offset is absorbed by adding a constant to the feature. The linear model can be formulated as follows:

(3.2)

$$\begin{aligned}\min_{\mathbf{W},\mathbf{B},\mathbf{A}} \quad & \|\mathbf{X}\mathbf{W}-\mathbf{Y}\|_F^2+\alpha\sum_{i=1}^{l}\sum_{j=1}^{l}\mathbf{A}_{ij}\|\mathbf{X}\mathbf{w}_i-\mathbf{X}\mathbf{w}_j\|_2^2 \\ & +\beta\sum_{t=2}^{l}\|\mathbf{w}_t-\mathbf{B}\mathbf{w}_{t-1}\|_2^2+\gamma\|\mathbf{B}-\mathbf{I}\|_F^2 \\ & +\delta\sum_{i=1}^{l}\|\mathbf{w}_i\|_2^2+\epsilon\|\mathbf{A}-\mathbf{A}_0\|_F^2\end{aligned}$$

In this linear formulation, if $\mathbf{A}_{ij}$ is a large positive number, meaning the i-th impact and the j-th impact are similar, then the predicted values for the i-th impact $\mathbf{X}\mathbf{w}_i$ and that for the j-th impact $\mathbf{X}\mathbf{w}_j$ are consistent.

iPath – non-linear formulation: in the non-linear case, the predicted impact is no longer a linear combination of the features, but the linear combination of the *similarities* between the test sample and all the training samples, where the similarities are expressed in the kernel matrix $\mathbf{K}$. The (i, j)-th entry of $\mathbf{K}$ can be computed as $\mathbf{K}(i,j) = \kappa(\mathbf{X}(i,:), \mathbf{X}(j,:))$, where $\kappa(\cdot,\cdot)$ is a kernel function that implicitly computes the inner product in the reproducing kernel Hilbert space (RKHS) [1]. The non-linear model can be formulated as follows:

$$\begin{aligned}\min_{\mathbf{W},\mathbf{B},\mathbf{A}} \quad & \|\mathbf{KW}-\mathbf{Y}\|_F^2 + \alpha\sum_{i=1}^{l}\sum_{j=1}^{l}\mathbf{A}_{ij}\|\mathbf{Kw}_i - \mathbf{Kw}_j\|_2^2 \\ & +\beta\sum_{t=2}^{l}\|\mathbf{w}_t - \mathbf{Bw}_{t-1}\|_2^2 + \gamma\|\mathbf{B}-\mathbf{I}\|_F^2 \\ & +\delta\sum_{i=1}^{l}\mathbf{w}_i'\mathbf{Kw}_i + \epsilon\|\mathbf{A}-\mathbf{A}_0\|_F^2\end{aligned} \tag{3.3}$$

From the objective function, we can see that if $\mathbf{A}_{ij}$ is a large positive number, meaning the i-th impact and the j-th impact are similar, then the predicted values for the i-th impact $\mathbf{Kw}_i$ and that for the j-th impact $\mathbf{Kw}_j$ are consistent.

3.2 *iPath* Optimization Solutions

In this subsection, we introduce an effective alternating optimization strategy to solve *iPath*. Since the optimization for linear and non-linear formulations are very similar, we will focus on the non-linear case and omit the linear case (referred to as *iPath*-lin) due to space limit. In non-linear case, we need to solve Eq. (3.3), which involves the optimization for $\mathbf{W}$, $\mathbf{B}$ and $\mathbf{A}$. We apply an alternating strategy and each time optimize for one group of variables while fixing the others. The details are as follows:

#1. Optimize for W while others are fixed: when others are fixed, the objective function becomes:

$$\begin{aligned}\min_{\mathbf{W}} \quad & \|\mathbf{KW}-\mathbf{Y}\|_F^2 + \alpha\sum_{i=1}^{l}\sum_{j=1}^{l}\mathbf{A}_{ij}\|\mathbf{Kw}_i - \mathbf{Kw}_j\|_2^2 \\ & +\beta\sum_{t=2}^{l}\|\mathbf{w}_t - \mathbf{Bw}_{t-1}\|_2^2 + \delta\sum_{i=1}^{l}\mathbf{w}_i'\mathbf{Kw}_i\end{aligned}$$

As it turns out, it has the following fixed point solution:

$$vec(\mathbf{W}) = \mathbf{S}^{-1}vec(\mathbf{K}'\mathbf{Y}) \tag{3.4}$$

where $vec(\cdot)$ is the vectorization operation on a matrix by stacking the columns of a matrix into one column vector, and $\mathbf{S}$ is a block matrix composed of $l \times l$ blocks. The (i,j)-th block of $\mathbf{S}$, $\mathbf{S}_{ij}$ can be written as follows:

$$\mathbf{S}_{ii} = \begin{cases}(1+\alpha\sum_{j=1}^{l}\mathbf{A}_{ij})\mathbf{K}'\mathbf{K} + \beta\mathbf{B}'\mathbf{B} + \delta\mathbf{K}, & \text{if } i=1\\ (1+\alpha\sum_{j=1}^{l}\mathbf{A}_{ij})\mathbf{K}'\mathbf{K} + \delta\mathbf{K}, & \text{if } i=l\\ (1+\alpha\sum_{j=1}^{l}\mathbf{A}_{ij})\mathbf{K}'\mathbf{K} + \beta(\mathbf{I}+\mathbf{B}'\mathbf{B}) + \delta\mathbf{K}, & \text{otherwise}\end{cases} \tag{3.5}$$

$$\mathbf{S}_{ij} = \begin{cases}-\alpha\mathbf{A}_{ij}\mathbf{K}'\mathbf{K} - \beta\mathbf{B}', & \text{if } i=j-1\\ -\alpha\mathbf{A}_{ij}\mathbf{K}'\mathbf{K} - \beta\mathbf{B}, & \text{if } i=j+1\\ -\alpha\mathbf{A}_{ij}\mathbf{K}'\mathbf{K}, & \text{otherwise}\end{cases} \tag{3.6}$$

#2. Optimize for B while others are fixed: when others are fixed, the objective function becomes:

$$\min_{\mathbf{B}} \quad \beta\sum_{t=2}^{l}\|\mathbf{w}_t - \mathbf{Bw}_{t-1}\|_2^2 + \gamma\|\mathbf{B}-\mathbf{I}\|_F^2$$

It has the following fixed point solution:

$$\mathbf{B} = (\beta\sum_{t=2}^{l}\mathbf{w}_t\mathbf{w}_{t-1}' + \gamma\mathbf{I})(\beta\sum_{t=2}^{l}\mathbf{w}_{t-1}\mathbf{w}_{t-1}' + \gamma\mathbf{I})^{-1} \tag{3.7}$$

#3. Optimize for A while others are fixed: when others are fixed, the objective function becomes:

$$\min_{\mathbf{A}} \quad \alpha\sum_{i=1}^{l}\sum_{j=1}^{l}\mathbf{A}_{ij}\|\mathbf{Kw}_i - \mathbf{Kw}_j\|_2^2 + \epsilon\|\mathbf{A}-\mathbf{A}_0\|_F^2$$

It has the following fixed point solution:

$$\mathbf{A} = \mathbf{A}_0 - \mathbf{D}, \text{where } \mathbf{D}_{ij} = \|\mathbf{Kw}_j - \mathbf{Kw}_i\|_2^2. \tag{3.8}$$

The optimization solution for the non-linear model can be summarized as in Algorithm 1.

Algorithm 1 *iPath*-ker – forecasting the pathway to impact

```
Input: (1)feature matrix $\mathbf{X}$;
       (2)impact matrix $\mathbf{Y}$;
       (3)prior knowledge of the relation $\mathbf{A}_0$;
       (4)balance parameters $\alpha$, $\beta$, $\gamma$, $\delta$ and $\epsilon$;
Output: model parameters $\mathbf{w}_i, i = 1, \dots, l$
Initialize $\mathbf{W}$, $\mathbf{B}$ and $\mathbf{A}$
Construct kernel matrix $\mathbf{K}$ from $\mathbf{X}$
while not converged do
    Update model parameters $\mathbf{W}$ by Eq. (3.4)
    Update linear transition matrix $\mathbf{B}$ by Eq. (3.7)
    Update impact relation $\mathbf{A}$ by Eq. (3.8)
end while
Output model parameters $\mathbf{W}$
```

4 Analysis and Comparisons

In this section, we will first analyze the complexity of the proposed *iPath*, present some variants of it, and then provide a probabilistic interpretation for it, followed up by the comparisons with some existing work.

4.1 Complexity Analysis

We summarize the time complexity of *iPath*-lin and *iPath*-ker in Theorem 4.1.

THEOREM 4.1. *(Time Complexity). iPath-lin takes $O(N \cdot (ndl^2 + d^3l^3))$ time, and iPath-ker (Algorithm 1) takes $O(N \cdot (n^3l^3 + n^2l^2))$ time, where N is the number of iterations.*

Proof. Omitted for brevity.

Remarks: in both *iPath*-lin and *iPath*-ker, the number of iterations is small in practice (typically in 5-10 iterations, see Sec. 5 for details). In *iPath*-lin, each iteration only takes linear time w.r.t. n. In *iPath*-ker, the major computational cost in each iteration is the inverse of a large matrix $\mathbf{S}$ in Eq. (3.4), which is of size nl by nl. One way to speed up is by low-rank approximation on such large matrix [14]. A top-r eigen-decomposition on $\mathbf{S}$ takes $O(n^2l^2r)$, where r is the rank. Then the inverse will become the multiplication of the eigenvector matrices and the inverse of the eigenvalue diagonal matrix, which is very easy to compute. Another way to speed up is to filter out those unpromising training samples. When new training samples arrive, we can first treat them as test samples and make predictions on them using the existing trained model. Those samples whose prediction error is smaller than a specified threshold will be discarded. In this way, the size of matrix $\mathbf{S}$ will also be reduced.

4.2 Variants

The proposed *iPath* model is comprehensive in handling both the *prediction consistency* as well as the *parameter smoothness*. In the case when one or both aspects are not necessary for the prediction in some applications, our model can be naturally adapted to accommodate such special cases. In this subsection, we will discuss two of the variants.

Variant #1: known relation. If the relation among the impacts are accurately known a prior, we can fix the relation in the model instead of learning it. We can do this by setting ϵ to 0 and plug in the known relation matrix $\mathbf{A}$. In the optimization solution, we only need to optimize for $\mathbf{W}$ and $\mathbf{B}$ in this variant.

Variant #2: known relation without parameter smoothness. In some cases, the parameter smoothness might not hold and we do not need to consider the linear transition process between adjacent parameters. We can set β, γ and ϵ to 0. This degenerates to the *iBall* model proposed in [14]. It is a special case of our *iPath* model without considering parameter smoothness and with known relation. Another difference is that *iPath* imposes the prediction consistency in the output space, instead of in the parameter space.

4.3 Probabilistic Interpretation

In this subsection, we will provide a probabilistic interpretation for *iPath*. Our algorithm can be represented by the graphical model shown in Figure 1. The shaded nodes $\mathbf{Y}_i$ are the impacts observed, and in the linear formulation they are linear combination of the features with a multivariate Gaussian noise:

$$\begin{aligned} \mathbf{Y}_i &= \mathbf{X}\mathbf{w}_i + \mathbf{e} \\ \mathbf{e} &\sim \mathcal{N}(\mathbf{0}, \sigma_y^2\mathbf{I}) \\ \mathbf{Y}_i|\mathbf{w}_i &\sim \mathcal{N}(\mathbf{X}\mathbf{w}_i, \sigma_y^2\mathbf{I}) \end{aligned} \tag{4.9}$$

For the model parameters $\mathbf{w}_t$, we assume it is a linear transition of the parameter for the last time step $\mathbf{w}_{t-1}$, with a multivariate Gaussian noise:

$$\begin{aligned} \mathbf{w}_t &= \mathbf{B}\mathbf{w}_{t-1} + \epsilon \\ \epsilon &\sim \mathcal{N}(\mathbf{0}, \sigma_w^2\mathbf{I}) \\ \mathbf{w}_t|\mathbf{w}_{t-1} &\sim \mathcal{N}(\mathbf{B}\mathbf{w}_{t-1}, \sigma_w^2\mathbf{I}) \end{aligned} \tag{4.10}$$

The relation among the impacts is represented as an undirected graph of different impacts $\mathbf{Y}_i$, with $\mathbf{A}$ as the weight matrix. If the i-th impact $\mathbf{Y}_i$ and the j-th impact $\mathbf{Y}_j$ are similar to each other, then the (i,j)-th entry $\mathbf{A}_{ij}$ is a large positive number. To define the distribution over this undirected graph of impacts, we refer to Hammersley-Clifford theorem in Markov Random Field (MRF) [2] and express it in terms of an energy function E and clique potentials defined on maximal cliques of the undirected graph as:

$$p(\mathbf{Y}) = \frac{1}{Z}\exp(-E(\mathbf{Y})), \text{where } E(\mathbf{Y}) = \sum_{c\in\mathcal{C}} \Phi_c(\mathbf{Y}_c). \tag{4.11}$$

Here $\mathcal{C}$ is the set of maximal cliques of the impact graph, Φ_c is a non-negative function defined on the random variables in the clique and Z is the partition function to ensure that the distribution sums to 1. If we only consider the potentials defined on the edge of the graph, as follows:

$$\begin{aligned} \Phi_{e=(\mathbf{Y}_i,\mathbf{Y}_j)} &= \mathbf{A}_{ij}\|\mathbf{Y}_i - \mathbf{Y}_j\|_2^2 \\ &= \mathbf{A}_{ij}\|\mathbf{X}\mathbf{w}_i - \mathbf{X}\mathbf{w}_j\|_2^2 \end{aligned} \tag{4.12}$$

Then, the distribution over the label graph is:

$$p(\mathbf{Y}) = \frac{1}{Z}\exp(-\sum_{i=1}^{l}\sum_{j=1}^{l}\mathbf{A}_{ij}\|\mathbf{X}\mathbf{w}_i - \mathbf{X}\mathbf{w}_j\|_2^2) \tag{4.13}$$

With these distributions defined, we aim to maximize the joint distribution described as follows:

$$\arg\max_{\mathbf{Y},\mathbf{X},\mathbf{W}} \quad = p(\mathbf{w}_1)\prod_{t=2}^{l} p(\mathbf{w}_t|\mathbf{w}_{t-1})\prod_{i=1}^{l} p(\mathbf{Y}_i|\mathbf{w}_i)p(\mathbf{Y}) \tag{4.14}$$

where we assume $p(\mathbf{w}_1) \sim \mathcal{N}(\mathbf{0}, \sigma_1^2\mathbf{I})$. If we maximize the above joint distribution, we can obtain the empirical loss, prediction consistency and parameter smoothness terms in *iPath*.

4.4 Comparison with Existing Work

As we point out in Sec. 4.2, *iBall* [14] is a special case of our *iPath* model. The idea of *iBall* is to leverage the relation among impacts in the parameter space, i.e., if $\mathbf{Y}_i$ and $\mathbf{Y}_j$ are similar, then the parameters $\mathbf{w}_i$ for predicting $\mathbf{Y}_i$ and $\mathbf{w}_j$ for predicting $\mathbf{Y}_j$ are similar. The multi-label learning method *MLRL* [29]

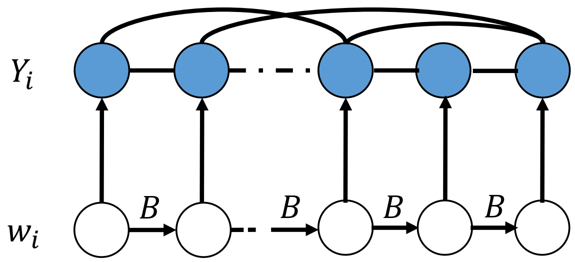


Figure 1: Graphical model representation of *iPath*.

also exploits such relation in the parameter space via maximum a posterior inference by assuming that $\mathbf{W}$ follows a matrix-variate normal distribution, but ignores the parameter smoothness. Our model *iPath* instead applies such relation in the output space and defines a linear transition process between two parameters at adjacent time steps.

5 Empirical Evaluations

In this section, we empirically evaluate the effectiveness of the proposed algorithms for forecasting the pathway to impact.

5.1 Datasets

To evaluate the performance of the proposed *iPath* algorithms, we conduct experiments on the real world citation network dataset provided by AMiner [19] [1], which is a rich dataset for bibliography network analysis and mining. The dataset contains information of 2,243,976 papers, 1,274,360 authors and 8,882 computer science venues. The information about a paper includes its title, authors, references, venue and publication year. The papers date from year 1936 to year 2013. From these, we can extract the number of citations each paper/author obtains in each year ever since its publication year.

5.2 Experiment Setup

Our primary task is to forecast a paper's yearly citations from year 6 to year 15 after its publication, with the first five years' citation history observed. To ensure the papers are at least 15 years old, we only keep papers published between year 1960 and 1998. We process the author data in a similar way and keep those whose research career begins (when they publish the first paper) between year 1960 and 1990. For each scholarly entity (paper and author), we represent it as a five dimensional feature vector, which is the yearly citation counts in the first five years. To evaluate our algorithm, we sort the scholarly entities by their starting year (e.g., publication year), and train the model in the older entities and always test on the latest ones. In the experiment, we incrementally add the training samples by this chronological order, and for the paper impact pathway prediction, we reserve the latest 10% samples as the test set; and for the author impact pathway prediction, we reserve the latest 6% samples as the test set.

Root mean squared error (RMSE) between the actual citations and the predicted ones is used as our accuracy evaluation. All the parameters, including the Gaussian kernel's bandwidth, are chosen through a grid search. All the experiments are run on a Windows machine with four 3.5 GHz Intel Cores and 256 GB RAM.

5.3 Results and Analysis

1. Paper and author impact pathway prediction performance. We compare the prediction accuracy of the following methods:

- *ind-linear:* train a liner ridge regression model for the impact in each year separately.
- *ind-kernel:* train a kernel ridge regression model for the impact in each year separately.
- *MTL-robust:* treat predicting the impact in each year as a task and apply the robust multi-task learning algorithm proposed in [6].
- *MLRL:* the multi-label learning method proposed in [29], where model parameters are assumed to conform matrix-variate normal distribution.
- *iBall-linear:* jointly learn the linear regression models as in [14].
- *iBall-kernel:* jointly learn the kernel ridge regression models as in [14].
- *iPath-lin:* the proposed linear predictive model with prediction consistency and parameter smoothness.
- *iPath-ker:* the proposed non-linear predictive model with prediction consistency and parameter smoothness.

The RMSE results of the above methods for predicting the impact pathway of both research papers and authors are in Figure 2 and 3, respectively. We can make the following observations: (1) the non-linear methods (ind-kernel, *iBall-kernel* and *iPath*-ker) generally perform better than the linear methods (ind-linear, MTL-robust, MLRL, *iBall-linear* and *iPath*-lin), which reflects that the impacts could be over simplified by a linear combination of the features. (2) Among the linear methods, we find that MTL-robust does not help improve the prediction over ind-linear. The possible

[1] https://aminer.org/billboard/citation

reason is that MTL-robust has the assumption that the model parameters admit a low-rank and sparse component, which might not be true in our case. The *iBall-linear* performs better than ind-linear, which shows that the impact relation exploitation can indeed help the forecasting. (3) Furthermore, learning a good relation can further boost the performance, as MLRL has lower RMSE than *iBall-linear*. Our *iPath*-lin performs the best among all the linear models, by integrating prediction consistency and parameter smoothness. It is even comparable with ind-kernel when training size is greater than 30% for the paper impact pathway prediction. (4) We can make the similar observation in the non-linear case, as our *iPath*-ker performs better than *iBall-ker*, which itself is better than ind-kernel.

To evaluate the statistical significance, we perform a t-test between *iPath*-ker and the best competitor *iBall-kernel* with 30% of the training papers in the paper impact pathway prediction, and the p-value is 0.01, which suggests the significance of the improvement.

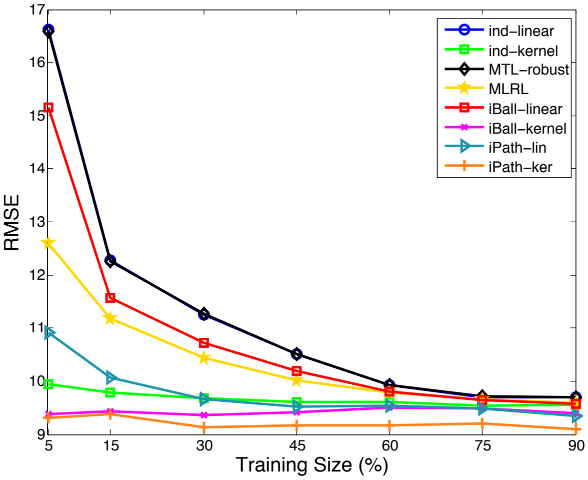


Figure 2: RMSE comparison of all the methods for paper impact pathway prediction.

2. Sensitivity analysis. To investigate parameter sensitivity, we perform parametric studies with the two most important parameters in *iPath*, namely, α that controls the importance of prediction consistency, and β that controls the importance of parameter smoothness. Figure 4 shows that the proposed model is stable in a large range of both parameter spaces.

3. Performance gain analysis. Let us take a closer investigation on where the performance gain of the proposed *iPath* stems from. As we mention above, *iPath* integrates both *prediction consistency* and *parameter smoothness*. We analyze how they contribute to the performance gain. Table 2 shows the results of *iPath*-ker methods on both the paper (60% training) and author (25% training) impact pathway prediction. 'Basic form' sets α, β, γ and ϵ all to zero, essentially ind-kernel method; 'Basic form + relation' incorporates the relations among impacts; 'Basic form + relation + transition' incorporates a known relation and the linear transition in the parameter space; 'Basic form + relation + transition + inferring' considers them all with an inferred relation. From the table, we can see that as we incrementally incorporate the elements, the RMSE decreases gradually, which confirms that all these elements are beneficial in improving the prediction performance.

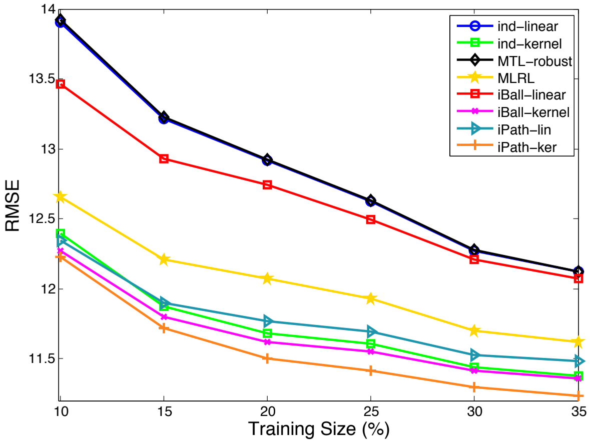
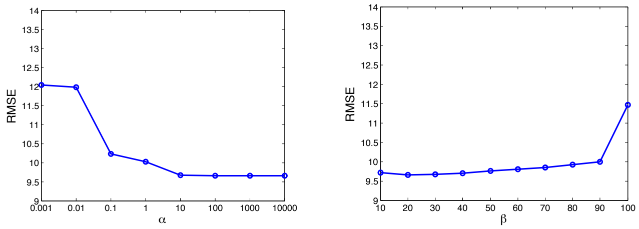


Figure 3: RMSE comparison of all the methods for author impact pathway prediction.



(a) RMSE vs. α

(b) RMSE vs. β

Figure 4: Sensitivity study on *iPath*-lin: study the effect of the parameters α and β in terms of RMSE.

Table 2: Performance gain analysis of *iPath*. Smaller is better.

RMSE	Paper Impact	Author Impact
Basic form	9.602	11.608
Basic form + relation	9.507	11.548
Basic form + relation + transition	9.335	11.489
Basic form + relation + transition + inferring	9.171	11.391

4. Robustness to noise in label graph. As *iPath* can learn a good relation for the prediction from our prior knowledge about the relation, we want to see how robust it is wrt the noise level in our prior knowledge. To this end, we input the same relation matrix with noise to *iBall* (the matrix $\mathbf{A}$) and *iPath* (the matrix $\mathbf{A}_0$). The noise is added to each entry of the label matrix with value $0.1 \times \textsc{NoiseLevel} \times \textsc{rand}$, where $\textsc{rand}$ is a random number from 0 to 1. Figure 5 shows the RMSE results of both *iBall* and *iPath* under different noise levels for paper impact pathway prediction with 30% training samples. We observe a sharp performance drop of *iBall* when noise is added. In contrast, the proposed *iPath* degenerates gradually with the noise level. This shows that *iPath* can learn a relatively good relation even if our prior knowledge is noisy.

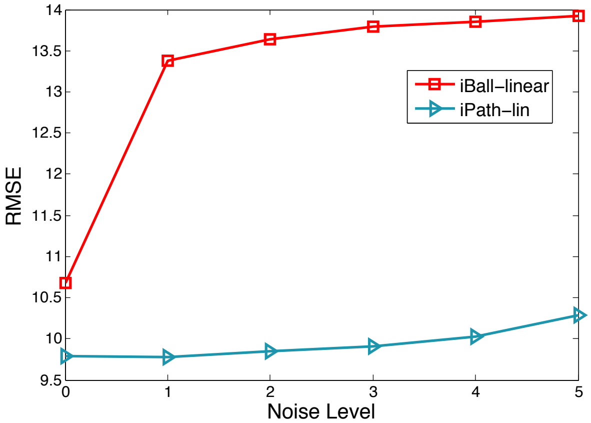
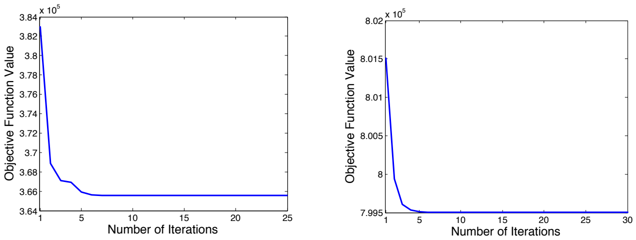


Figure 5: Robustness to noise on the label graph.

5. Convergence analysis. To see how fast the proposed *iPath* converges in practice, we plot the objective function value vs. number of iterations for both paper (15% training samples) and author (10% training samples) impact pathway forecasting as in Figure 6. We observe that *iPath* converges after 5-10 iterations.



(a) Objective function value vs. # iterations on paper impact pathway forecasting.

(b) Objective function value vs. # iterations on author impact pathway forecasting.

Figure 6: Convergence analysis of *iPath*.

6 Related Work

In this section, we review the related work in terms of (a) multi-label learning, (b) time series mining.

Multi-label Learning. Multi-label learning is a machine learning paradigm where each data instance is associated with a set of labels. For example, in image classification, an image could be tagged as nature, ocean and sky; in document categorization, a text might belong to politics and foreign affairs. The algorithms developed for multi-label learning can be roughly categorized into two groups by a recent survey [28]: *problem transformation methods*, to fit data to existing algorithms; and *algorithm adaptation methods*, to adapt existing learning technique to fit the multi-label data. In the first category, binary relevance [3] trains an individual classifier for each of the labels separately, which ignores label correlations and might suffer class imbalance issue. Classifier chains [18] on the other hand incrementally build classifier for each of the labels by augmenting the feature space using preceding predicted labels. The multi-label problems can be also modeled as a label ranking problem through the technique of pairwise comparison [9], essentially binary classifiers trained in one-vs-one fashion. In the second category, multi-label k-nearest neighbor algorithm [27] combines kNN and Bayesian reasoning to make prediction based on labeling information in the neighbors. Decision tree has also been adopted to handle multi-label data by computing the multi-label entropy [7]. Rank-SVM [8] employs maximum margin strategy to define linear models that minimize the ranking loss while having a large margin and enjoying non-linear extension through kernel trick.

Recently, there is a line of work focused on exploiting the relationship among the labels to improve the learning performance. Zhang and Yeung [29] propose a probabilistic model for multi-label learning by assuming that the model parameters follow a matrix-variant normal distribution and the label relationship learning becomes solving for the column covariance matrix in the maximum a posteriori (MAP) solution. Huang and Zhou [11] notice that some label correlations are not shared globally and propose approach that allows correlation sharing in a subset of instances. Ji et al. [12] assume that the model parameters share a low-dimensional subspace and formulate a regularized optimization problem.

Time Series Mining. Time series data are ubiquitous and can be seen in meteorology, finance, medicine, music, etc. Distance measures for time series include the classic Euclidean distance, and more sophisticated dynamic time warping [13] and longest common subsequence [21]. Many algorithms have been developed on time series for clustering [25, 15], classification [26, 10], anomaly detection [23] and motif discovery [17, 16]. Recently, time series has been studied from the perspective of network, e.g., Cai et al. [4, 5] propose the concept of *network of (high-order) time series* to capture the contextual information for better missing value recovery.

7 Conclusions

We focus on the problem of forecasting the impact pathway of scholarly entities, and propose an effective method (*iPath*). The proposed *iPath* can collectively model two important aspects of the impact pathway prediction problem, namely, *prediction consistency* and *parameter smoothness*. It is flexible for handling both linear and non-linear models and empirical evaluations demonstrate its effectiveness for forecasting the pathway to impact. Future work includes the deployment of *iPath* in real scholarly data mining systems, e.g., AMiner[2].

8 Acknowledgment

This work is partially supported by the National Science Foundation under Grant No. IIS1017415, by the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053, by National Institutes of Health under the grant number R01LM011986, Region II University Transportation Center under the project number 49997-33 25 and a Baidu gift. Jie Tang is supported by the National High-tech R&D Program (No. 2014AA015103), National Basic Research Program of China (No. 2014CB340506), and a research fund supported by Huawei Inc.

References

[1] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, pages 337–404, 1950.

[2] A. Blake, P. Kohli, and C. Rother. *Markov Random Fields for Vision and Image Processing*. The MIT Press, 2011. ISBN 0262015773, 9780262015776.

[3] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757 – 1771, 2004.

[4] Y. Cai, H. Tong, W. Fan, and P. Ji. Fast mining of a network of coevolving time series. In *SDM*, 2015.

[5] Y. Cai, H. Tong, W. Fan, P. Ji, and Q. He. Facets: Fast comprehensive mining of coevolving high-order time series. In *KDD*, pages 79–88. ACM, 2015.

[6] J. Chen, J. Zhou, and J. Ye. Integrating low-rank and group-sparse structures for robust multi-task learning. In *KDD*, pages 42–50. ACM, 2011.

[7] A. Clare and R. D. King. Knowledge discovery in multi-label phenotype data. In *Principles of data mining and knowledge discovery*, pages 42–53. Springer, 2001.

[8] A. Elisseeff and J. Weston. A kernel method for multi-labelled classification. In *NIPS*, pages 681–687, 2001.

[9] J. Fürnkranz, E. Hüllermeier, E. L. Mencía, and K. Brinker. Multilabel classification via calibrated label ranking. *Machine learning*, 73(2):133–153, 2008.

[10] B. Hu, Y. Chen, and E. Keogh. Time series classification under more realistic assumptions. In *SDM*, 2013.

[11] S.-J. Huang and Z.-H. Zhou. Multi-label learning by exploiting label correlations locally. In *AAAI*, 2012.

[12] S. Ji, L. Tang, S. Yu, and J. Ye. A shared-subspace learning framework for multi-label classification. *TKDD*, 4(2):8, 2010.

[13] E. Keogh and C. A. Ratanamahatana. Exact indexing of dynamic time warping. *Knowledge and information systems*, 7(3):358–386, 2005.

[14] L. Li and H. Tong. The child is father of the man: Foresee the success at the early stage. In *KDD*, pages 655–664, 2015.

[15] T. W. Liao. Clustering of time series dataa survey. *Pattern recognition*, 38(11):1857–1874, 2005.

[16] A. Mueen and E. Keogh. Online discovery and maintenance of time series motifs. In *KDD*. ACM, 2010.

[17] A. Mueen, E. J. Keogh, Q. Zhu, S. Cash, and M. B. Westover. Exact discovery of time series motifs. In *SDM*, 2009.

[18] J. Read, B. Pfahringer, G. Holmes, and E. Frank. Classifier chains for multi-label classification. In *Machine Learning and Knowledge Discovery in Databases*, volume 5782, pages 254–269. Springer Berlin Heidelberg, 2009.

[19] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su. Arnetminer: extraction and mining of academic social networks. In *KDD*, pages 990–998. ACM, 2008.

[20] B. Uzzi, S. Mukherjee, M. Stringer, and B. Jones. Atypical combinations and scientific impact. *Science*, 342(6157):468–472, 2013.

[21] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, and E. Keogh. Indexing multi-dimensional time-series with support for multiple distance measures. In *KDD*, pages 216–225. ACM, 2003.

[22] D. Wang, C. Song, and A.-L. Barabási. Quantifying long-term scientific impact. *Science*, 342(6154):127–132, 2013.

[23] H. Xiao, J. Gao, D. S. Turaga, L. H. Vu, and A. Biem. Temporal multi-view inconsistency detection for network traffic analysis. In *WWW*, pages 455–465, 2015.

[24] R. Yan, J. Tang, X. Liu, D. Shan, and X. Li. Citation count prediction: Learning to estimate future citations for literature. In *CIKM*, pages 1247–1252. ACM, 2011.

[25] J. Yang and J. Leskovec. Patterns of temporal variation in online media. In *WSDM*, pages 177–186. ACM, 2011.

[26] L. Ye and E. Keogh. Time series shapelets: a new primitive for data mining. In *KDD*. ACM, 2009.

[27] M.-L. Zhang and Z.-H. Zhou. Ml-knn: A lazy learning approach to multi-label learning. *Pattern recognition*, 40(7):2038–2048, 2007.

[28] M.-L. Zhang and Z.-H. Zhou. A review on multi-label learning algorithms. *TKDE*, 26(8):1819–1837, 2014.

[29] Y. Zhang and D.-Y. Yeung. Multilabel relationship learning. *TKDD*, 7(2):7, 2013.

[2] https://aminer.org/