Predicting a Scientific Community's Response to an Article

Dani Yogatama Michael Heilman Brendan O'Connor Chris Dyer
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
{dyogatama,mheilman,brenocon,cdyer}@cs.cmu.edu

Bryan R. Routledge
Tepper School of Business
Carnegie Mellon University
Pittsburgh, PA 15213, USA
routledge@cmu.edu

Noah A. Smith
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
nasmith@cs.cmu.edu

Abstract

We consider the problem of predicting measurable responses to scientific articles based primarily on their text content. Specifically, we consider papers in two fields (economics and computational linguistics) and make predictions about downloads and within-community citations. Our approach is based on generalized linear models, allowing interpretability; a novel extension that captures first-order temporal effects is also presented. We demonstrate that text features significantly improve accuracy of predictions over metadata features like authors, topical categories, and publication venues.

1 Introduction

Written communication is an essential component of the complex social phenomenon of science. As such, natural language processing is well-positioned to provide tools for understanding the scientific process, by analyzing the textual artifacts (papers, proceedings, etc.) that it produces. This paper is about modeling collections of scientific documents to understand how their *textual content* relates to how a scientific community responds to them. While past work has often focused on citation structure (Borner et al., 2003; Qazvinian and Radev, 2008), our emphasis is on the text content, following Ramage et al. (2010) and Gerrish and Blei (2010).

Instead of task-independent exploratory data analysis (e.g., topic modeling) or multi-document summarization, we consider supervised models of the collective *response* of a scientific community to a published article. There are many measures of impact of a scientific paper; ours come from direct measurements of the number of downloads (from an established website where prominent economists post papers before formal publication) and citations (within a fixed scientific community). We adopt a discriminative approach based on generalized linear models that can make use of any text or metadata features, and show that simple lexical features offer substantial power in modeling out-of-sample response and in *forecasting* response for future articles. Realistic forecasting evaluations require methodological care beyond the usual best practices of train/test separation, and we elucidate these issues.

In addition, we introduce a new regularization technique that leverages the intuition that the relationship between observable features and response should evolve smoothly over time. This regularizer allows the learner to rely more strongly on more recent evidence, while taking into account a long history of training data. Our time series-inspired regularizer is computationally efficient in learning and is a significant advance over earlier text-driven forecasting models that ignore the time variable altogether (Kogan et al., 2009; Joshi et al., 2010).

We evaluate our approaches in two novel experimental settings: predicting downloads of economics articles and predicting citation of papers at ACL conferences. Our approaches substantially outper-

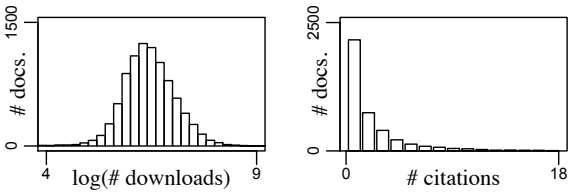


Figure 1: Left: the distribution of log download counts for papers in the NBER dataset one year after posting. Right: the distribution of within-dataset citations of ACL papers within three years of publication (outliers excluded for readability).

form text-ignorant baselines on ground-truth predictions. Our time series models permit flexibility in features and offer a novel and perhaps more interpretable view of the data than summary statistics.

2 Data

We make use of two collections of scientific literature, one from the economics domain, and the other from computational linguistics and natural language processing. Statistics are summarized in Table 1.

2.1 NBER

Our first dataset consists of research papers in economics from the National Bureau of Economic Research (NBER) from 1999 to 2009 (`http://www.nber.org`). Approximately 1,000 research economists are affiliated with the NBER. New NBER working papers are posted to the website weekly. The papers are not yet peer-reviewed, but given the prominence of many economists affiliated with the NBER, many of the papers are widely read. Text from the abstracts of the papers and related metadata are publicly available. Full text is available to subscribers (universities typically have access).

The NBER provided us with download statistics for these papers. For each paper, we computed the total number of downloads in the first year after each paper's posting.[1] The download counts are log-normally distributed, as shown in Figure 1, and so our regression models (§3) minimize squared errors in the log space. Our download logs begin in 1999. We use the 8,814 papers from 1999–2009 period (there are 16,334 papers in the full dataset dating back to 1985). We only use text from the abstracts, since we were able to obtain full texts for just a portion of the papers, and since the OCR of the full texts we do have is very noisy.

2.2 ACL

Our second dataset consists of research papers from the Association for Computational Linguistics (ACL) from 1980 to 2006 (Radev et al., 2009a; Radev et al., 2009b). We have the full texts for papers (OCR output) as well as structured citation data. There are 15,689 papers in the whole dataset. For the citation prediction task, we include conference papers from ACL, EACL, HLT, and NAACL.[2] We remove journal papers, since they are characteristically different from conference papers, as well as workshop papers. We do include short papers, interactive demo session papers, and student research papers that are included in the companion volumes for these conferences (such papers are cited less than full papers, but many are still cited). The resulting dataset contains 4,026 papers. The number of papers in each year varies because not all conferences are annual.

We look at citations in the three-year window following publication, including self-citations[3] and only considering citations from papers within these conferences. Figure 1 shows a histogram; note that many papers (54%) are not cited at all, and the distribution of citations per paper is neither normal nor log-normal. We organize the papers into two classes: those with zero citations and those with non-zero citations in the three-year window.

3 Model

Our forecasting approach is based on generalized linear models for regression and classification. The models are trained with an ℓ_2-penalty, often called a "ridge" model (Hoerl and Kennard, 1970).[4] For

[1] For the vast majority of papers, most of the downloads occur soon after the paper's posting. We explored different measures with different download windows (two years, for example) with broadly similar results. We leave a more detailed analysis of the time series patterns of downloads to future work.

[2] EMNLP is a relatively recent conference, and, in this collection, complete data for its papers postdate the end of the last training period, so we chose to exclude it from our dataset.

[3] The original version of this paper, published in the proceedings of EMNLP 2011, said that we exclude self citations. We have corrected it here.

[4] Preliminary experiments found no consistent benefit from ℓ_1 ("lasso") models, though we note that ℓ_1-regularization leads

Dataset	# Docs.	Avg. # Words	Response
NBER	8,814	155	# downloads in first year (mean 761)
ACL	4,026	3,966	at least 1 citation in first 3 years? (54% no)

Table 1: Descriptive statistics about the datasets.

the NBER data, where (log) number of downloads is nearly a continuous measure, we use linear regression. For the ACL data, where response is the binary cited-or-not variable we use logistic regression, often referred to as a "maximum entropy" model (Berger et al., 1996) or a log-linear model. We briefly review the class of models. Then, we describe a time series model appropriate for time series data.

3.1 Generalized Linear Models

Consider a model that predicts a response y given a vector input $\boldsymbol{x} = \langle x_1, \ldots, x_d \rangle \in \mathbb{R}^d$. Our models are linear functions of $\boldsymbol{x}$ and parameterized by the vector $\boldsymbol{\beta}$. Given a corpus of M document features, $\boldsymbol{X}$, and responses Y, we estimate:

$$\hat{\boldsymbol{\beta}} = \operatorname{argmin}_{\boldsymbol{\beta}} R(\boldsymbol{\beta}) + \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{X}, Y) \qquad (1)$$

where $\mathcal{L}$ is a model-dependent loss function and R is a regularization penalty to encourage models with small weight vectors. We describe models and loss functions first and then turn to regularization.

For the NBER data, the (log) number of downloads is continuous, and so we use least-squares linear regression model. The loss function is the sum of the squared errors for the M documents in our training data: $\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{X}, Y) = \sum_{i=1}^{M} (y_i - \hat{y}_i)^2$, where the prediction rule for new documents is: $\hat{y} = \sum_{j=0}^{d} \beta_j x_j$. Probabilistically, this equates to an assumption that $\boldsymbol{\beta}^\top \boldsymbol{x}$ is the mean of a normal (i.e., Gaussian) distribution from which random variable y is drawn.

For the ACL data, we predict y from a discrete set C (specifically, the binary set of zero citations or more than zero citations), and we use logistic regression. This model assumes that for the ith training input $\boldsymbol{x}_i$, the output y_i is drawn according to:

$$p(y_i \mid \boldsymbol{x}_i) = \left(\exp \boldsymbol{\beta}_c^\top \boldsymbol{x}_i\right) / \left(\textstyle\sum_{c' \in C} \exp \boldsymbol{\beta}_{c'}^\top \boldsymbol{x}_i\right)$$

where there is a feature vector $\boldsymbol{\beta}_c$ for each class $c \in C$. Under this interpretation, parameter estimation is maximum *a posteriori* inference for $\boldsymbol{\beta}$, and $R(\boldsymbol{\beta})$ is a log-prior for the weights. The loss function is the negative log likelihood for the M documents: $\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{X}, Y) = -\sum_{i=1}^{M} \log p(y_i \mid \boldsymbol{x}_i)$. The prediction rule for a new document is: $\hat{y} = \operatorname{argmax}_{c \in C} \sum_{j=0}^{d} \beta_{c,j} x_j$. Generalized linear models and penalized regression are well-studied with an extensive literature (Mccullagh and Nelder, 1989; Hastie et al., 2009). We leave other types of models, such as Poisson (Cameron and Trivedi, 1998) or ordinal (McCullagh, 1980) regression models, to future work.

3.2 Ridge Regression

With large numbers of features, regularization is crucial to avoid overfitting. In ridge regression (Hoerl and Kennard, 1970), a standard method to which we compare the time series regularization discussed in §3.3, the penalty $R(\boldsymbol{\beta})$ is proportional to the ℓ_2-norm of $\boldsymbol{\beta}$:

$$R(\beta) = \lambda \|\boldsymbol{\beta}\|_2 = \lambda \textstyle\sum_j \beta_j^2$$

where λ is a regularization hyperparameter that is tuned on development data or by cross-validation.[5] This penalty pushes many β_j close (but not completely) to zero. In practice, we multiply the penalty by the number of examples M to facilitate tuning of λ.

The ridge linear regression model can be interpreted probabilistically as each coefficient β_j is drawn i.i.d. from a normal distribution with mean 0 and variance $2\lambda^{-1}$.

3.3 Time Series Regularization

A simple way to capture temporal variation is to conjoin traditional features with a time variable. Here, we divide the dataset into T time steps (years). In the new representation, the feature space expands from $\mathbb{R}^d$ to $\mathbb{R}^{T \times d}$. For a document published at year t, the elements of $\boldsymbol{x}$ are non-zero only for those features that correspond to year-t; that is $x_{t',j} = 0$ for all $t' \neq t$.

to sparse, compact models that may be more interpretable.

[5] The linear regression has a bias β_0 that is always active. The logistic regression also has an unpenalized bias $\beta_{c,0}$ for each class c. This weight is not regularized.

Estimating this model with the new features using the ℓ_2-penalty would be effectively estimating separate models for each year under the assumption that each $\beta_{t,j}$ is independent; even for features that differed only temporally (e.g., $\beta_{t,j}$ and $\beta_{t+1,j}$).

In this work, we apply time series regularization to GLMs, enabling models that have coefficients that change over time but prefer gradual changes across time steps. Boyd and Vandenberghe (2004, §6.3) describe a general version of this sort of regularizer. To our knowledge, such regularizers have not previously been applied to temporal modeling of text.

The time series regularization penalty becomes:

$$R(\boldsymbol{\beta}) = \lambda \sum_{t=1}^{T} \sum_{j=1}^{d} \beta_{t,j}^2 + \lambda\alpha \sum_{t=2}^{T} \sum_{j=0}^{d} (\beta_{t,j} - \beta_{t-1,j})^2$$

It includes a standard ℓ_2-penalty on the coefficients, and a penalty for differences between coefficients for adjacent time steps to induce smooth changes.[6] Similar to the previous model, in practice, we multiply the regularization constant λ by $\frac{M}{T}$ to facilitate tuning of λ for datasets with different numbers of examples M and numbers of time steps T. The new parameter, α, controls the smoothness of the estimated coefficients. Setting α to zero imposes no penalty for time-variation in the coefficients and results in independent ridge regressions at each time step. Also, when the number of examples is constant across time steps, setting a large α parameter ($\alpha \to \infty$) results in a single ridge regression over all years since it imposes $\beta_{t,j} = \beta_{t+1,j}$ for all $t \in T$.

The partial derivative is:

$$\begin{aligned} \partial R / \partial \beta_{t,j} = \; & 2\lambda\beta_{t,j} \\ & + \; \mathbf{1}\{t > 1\} 2\lambda\alpha(\beta_{t,j} - \beta_{t-1,j}) \\ & + \; \mathbf{1}\{t < T\} 2\lambda\alpha(\beta_{t,j} - \beta_{t+1,j}) \end{aligned}$$

This time series regularization can be applied more generally, not just to linear and logistic regression.

With either ridge regularization or this time series regularization scheme, Eq. 1 is an unconstrained convex optimization problem for the linear models we describe here. There exist a number of optimization procedures for it; we use the L-BFGS quasi-Newton algorithm (Liu and Nocedal, 1989).

[6] Our implementation of the time series regularizer does not penalize the magnitude of the weight for the bias feature (as in ridge regression). It does, however, penalize the difference in the bias weight between time steps (as with other features).

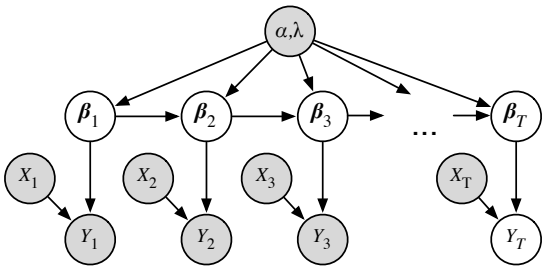


Figure 2: Time series regression as a graphical model; the variables $\boldsymbol{X}_t$ and Y_t are the sets of feature vectors and response variables from documents dated t.

Probabilistic Interpretation

We can interpret the time series regularization probabilistically as follows. Let the coefficient for the jth feature over time be $\boldsymbol{\beta}_j = \langle \beta_{1,j}, \beta_{2,j}, ..., \beta_{T,j} \rangle$. $\boldsymbol{\beta}_j$ are draws from a multivariate normal distribution with a tridiagonal precision matrix $\Sigma^{-1} = \Lambda \in \mathbb{R}^{T \times T}$:

$$\Lambda = \lambda \begin{bmatrix} 1+\alpha & -\alpha & 0 & 0 & \dots \\ -\alpha & 1+2\alpha & -\alpha & 0 & \dots \\ 0 & -\alpha & 1+2\alpha & -\alpha & \dots \\ 0 & 0 & -\alpha & 1+2\alpha & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

The form of $R(\boldsymbol{\beta})$ follows from noting:

$$-2 \log p(\boldsymbol{\beta}_j; \alpha, \lambda) = \boldsymbol{\beta}_j^\top \Lambda \boldsymbol{\beta}_j + constant$$

The squared difference between adjacent time steps comes from the off-diagonal entries in the precision matrix.[7] Figure 2 shows a graphical representation of the time series regularization in our model. Its Markov chain structure corresponds to the off-diagonals.

There is a rich literature on time series analysis (Box et al., 2008; Hamilton, 1994). The prior distribution over the sequence $\langle \beta_{1,j}, \dots, \beta_{T,j} \rangle$ that our regularizer posits is closely linked to a first-order autoregressive process, AR(1).

[7] Consistent with the previous section, we assume that parameters for different features, $\boldsymbol{\beta}_j$ and $\boldsymbol{\beta}_k$, are independent.

	NBER	ACL
Response	log(#downloads+1)	$\mathbf{1}\{\#\text{citations} > 0\}$
GLM type	normal / squared-loss	logistic / log-loss
Metric 1	mean absolute error	accuracy
Metric 2	Kendall's τ	Kendall's τ

Table 2: Summary of the setup for the NBER download and ACL citation prediction experiments.

4 Features

NBER metadata features

- Authors' last names. We treat each name as a binary feature. If a paper has multiple authors, all authors are used and they have equal weights regardless of their ordering.
- NBER program(s).[8] There are 19 major research programs at the NBER (e.g., Monetary Economics, Health Economics, etc.).

ACL metadata features

- Authors' last names as binary features.
- Conference venues. We use first letter of the ACL anthology paper ID, which correlates with its conference venue (e.g., *P* for the ACL main conference, *H* for the HLT conference, etc.).[9]

Text features

- Binary indicator features for the presence of each unigram, bigram, and trigram. For the NBER data, we have separate features for titles and abstracts. For the ACL data, we have separate features for titles and full texts. We pruned text features by document frequency (details in §5).
- Log transformed word counts. We include features for the numbers of words in the title and the abstract (NBER) or the full text (ACL).

[8] Almost all NBER papers are tagged with one or more programs (we assign untagged papers a "null" tag). The complete list of NBER programs can be found at `http://www.nber.org/programs`

[9] Papers in the ACL dataset have a tag which shows which workshop, conference, or journal they appeared in. However, sometimes a conference is jointly held with another conference, such that meta information in the dataset is different even though the conference is the same. For this reason, we simply use the first letter of the paper ID.

5 Experiments

For each of the datasets in §2, we test our models for two tasks: **forecasting** about future papers (i.e., making predictions about papers that appeared after a training dataset) and **modeling** held-out papers from the past (i.e., making predictions within the same time period as the training dataset, on held-out examples).

For the NBER dataset, the task is to predict the number of downloads a paper will receive in its first year after publication. For the ACL dataset, the task is to predict whether a paper will be cited at all (by another ACL paper in our dataset) within the first three years after its publication. To our knowledge, clean, reliable citation counts are not available for the NBER dataset; nor are download statistics available for the ACL dataset. Table 2 summarizes the variables of interest, model types, and evaluation metrics for the tasks.

5.1 Extrapolation

The lag between a paper's publication and when its outcome (download or citation count) can be observed poses a unique methodological challenge. Consider predicting the number of downloads over g future time steps. If t is the time of forecasting, we can observe the texts of all articles published before t. However, any article published in the interval $[t-g, t]$ is too recent for the outcome measurement of y to be taken. We refer to the interval $[t-g, t]$ as the "forecast gap". Since recent articles are sometimes the most relevant predictions at t, we do not want to ignore them. Consider a paper at time step $t', t-g < t' < t$. To extrapolate its number of downloads, we consider the observed number in $[t', t]$, and then estimate the ratio r of downloads that occur in the first $t-t'$ time steps, against the first g time steps, using the fully observed portion of the training data. We then scale the observed downloads during $[t', t]$ by r^{-1} to extrapolate. The same method is used to extrapolate citation counts.

In preliminary experiments, we observed that extrapolating responses for papers in the forecast gap led to better performance in general. For example, for the ridge regressions trained on all past years with the full feature set, the error dropped from 262 to 259 when using extrapolation compared to with-

out extrapolation. Also, the extrapolated download counts were quite close to the true values (which we have but do not use because of the forecast gap): for example, the mean absolute error of the extrapolated responses was 99 when extrapolated based on the median of the fully observed portion of the training data (measured monthly).

5.2 Forecasting NBER Downloads

In our first set of experiments, we predict the number of downloads of an NBER paper within one year of its publication.

We compare four approaches for predicting downloads. The first is a baseline that simply uses the median of the log of the training and development data as the prediction. The second and third use GLMs with ridge regression-style regularization (§3.2), trained on all past years ("all years") and on the single most recent past year ("one year"), respectively. The last model ("time series") is a GLM with time series regularization (§3.3).

We divided papers by year. Figure 3 illustrates the experimental setup. We held out a random 20% of papers for each year from 1999–2007 as a test set for the task of modeling the past. To define the feature set and tune hyperparameters, we used the remaining 80% of papers from 1999–2005 as our training data and the remaining papers in 2006 as our development data. After pruning,[10] we have 37,251 total features, of which 2,549 are metadata features. When tuning hyperparameters, we *simulated* the existence of a forecast gap by using extrapolated responses for papers in the last year of the training data instead of their true responses. We considered $\lambda \in 5^{\{2,1,\ldots,-5,-6\}}$, and $\alpha \in 5^{\{3,2,\ldots,-1,-2\}}$ and selected those that led to the best performance on the development set.

We then used the selected feature set and hyperparameters to test the forecasting and modeling capabilities of each model. For forecasting, we predicted numbers of downloads of papers in 2008 and 2009. We used the baseline median, ridge regression, and time series regularization models trained on papers in 1999–2007 and 1999–2008, respectively. We treated the last year of the training data (2007 and 2008, respectively) as a forecast gap, since we would not have observed complete responses of papers in these years when forecasting. For the "one year" models, we trained ridge regressions only on the most recent past year, using papers in 2007 and 2008, respectively, as training data.[11] To test the additive benefit of text features, we trained models with just metadata features (NBER programs and authors, denoted "Meta") and with both metadata

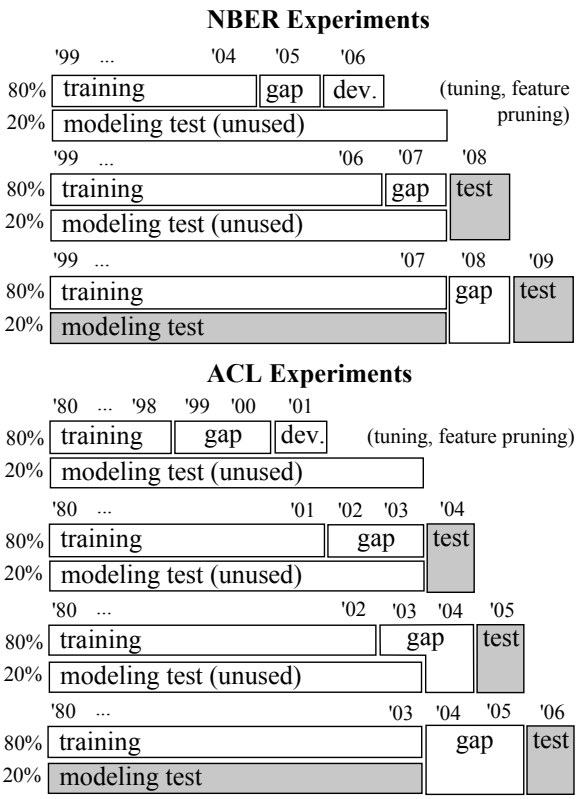


Figure 3: An illustration of how the datasets were segmented for the experiments. Portions of data for which we report results are shaded. Time spans are not to scale.

[10] For NBER, text features appearing in less than 0.1% or more than 99.9% of the training documents were removed. For ACL, the thresholds were 2% and 98%.

[11] Papers from the most recent past year in a training set have incomplete responses, so the models were trained on extrapolated responses for that year. For the NBER development set from 2005, a ridge regression on just 2004 papers (for which extrapolation is needed) outperformed a regression on just 2003 (for which extrapolation is not needed), 278 to 367 mean absolute error. For the ACL development set from 2001, a regression on just 2000 (for which extrapolation is needed) led to slightly lower performance (59% versus 61%) than a regression on just 1998 (for which extrapolation is not needed), probably due to the relatively small number of conferences and papers in 2000. For consistency with the other models and with the NBER experiments, we evaluated regressions on the most recent (extrapolated) year in our ACL experiments.

Features	Model	Modeling 1999–07	Forecasting 2008	Forecasting 2009
–	median	333	371	397
Meta	one year	279	354	375
Meta	all years	303	334	378
Meta	time series	279	353	375
Full	one year	271	346	351
Full	all years	265	†**300**	339
Full	time series	*†**245**	*321	***332**

Table 3: Mean absolute errors for the NBER download predictions. "$*$" indicates statistical significance between time series models using metadata features and the full feature set. "†" indicates statistical significance between the time series and ridge regression models using the full feature set (Wilcoxon signed-rank test, $p < 0.01$).

Feat.	Model	Modeling 1980–03	Forecasting 2004	Forecasting 2005	Forecasting 2006
–	majority	55	56	60	50
Meta	one year	61	56	54	62
Meta	all years	65	58	53	60
Meta	time series	66	56	53	56
Full	one year	69	**70**	64	67
Full	all years	67	69	**70**	70
Full	time series	**70**	*69	***70**	***72**

Table 4: Classification accuracy (%) for predicting whether ACL papers will be cited within three years. "$*$" indicates statistical significance between time series models using metadata features and the full feature set (binomial sign test, $p < 0.01$). With the full feature set, differences between the time series and ridge (all years) models are not statistically significant at the 0.01 level, but for the modeling task p is estimated at 0.026, and for the 2006 forecasting task, p is estimated at 0.050.

and text features (denoted "Full").

To evaluate the modeling capabilities, we trained the ridge regression and time series regularization models on papers from 1999–2008 and predicted the numbers of downloads of held-out papers in 1999–2007. For comparison, we also trained ridge regression models on each individual year ("one year") and predicted the numbers of downloads of the held-out papers in the corresponding year.

Table 3 shows mean absolute errors for each method on both forecasting test splits, and mean absolute errors averaged across papers over nine modeling test splits. For interpretability, we report predictions in terms of download counts, though the models were trained with log counts (§2.1). The results show that even a simple n-gram representation of text contains a valuable, learnable signal that is predictive of future downloads. While the time series model did not significantly outperform ridge regression at predicting future downloads, it did result in significantly better performance for *modeling* papers in the past.

5.3 Forecasting ACL Citations

We now turn to the problem of predicting citation levels. Recall that here we aim to predict whether an ACL paper will be cited within our dataset within three years. Our experimental setup (Figure 3) is similar to the setup for the NBER dataset, except that we use logistic regression to model the discrete cited-or-not response variable. We also make the simplifying assumption that all citations occur at the end of each year. Therefore, the forecast gap is only two years (we have observed complete citations in the test year).

After feature pruning, there were 30,760 total features, of which 1,694 are metadata features. We considered $\lambda \in 5^{\{2,1,\cdots,-8,-9\}}$ ("Full") and $\lambda \in 5^{\{2,1,\cdots,-11,-12\}}$ ("Meta"); and $\alpha \in 5^{\{6,5,\cdots,0,-1\}}$ (both "Full" and "Meta"), selecting the best values using the development data.

Again, we compare four methods: a baseline of always predicting the most frequent class in the training data, "all years" and "one year" logistic regression models, and a logistic regression with the time series regularizer.

For the forecasting task, we used papers in 2004, 2005, and 2006 as test sets. As the training sets for the "all years" and time series models, we used papers from 1980 up to the last year before each test set, with the last two years extrapolated. As the training sets for the "one year" models, we used papers from the year immediately before the test set, with extrapolated responses.

To evaluate modeling capabilities, we predicted citation levels of held-out papers in 1980–2003. We used the "all years" and time series models trained on 1980–2005. We trained "one year" models separately for each year and predicted downloads for the held-out papers in that year.

Table 4 shows classification accuracy for each model on the test data for both the forecasting and modeling tasks. It is again clear that adding text sig-

nificantly improved the performance of the model. Also, the time series regression model shows a small, though not statistically significant, gain for modeling whether past papers will be cited—as well as similarly small gains on two of the three forecasting test years.

5.4 Ranking

We can also use the models for ranking to help decide which papers are expected to have the greatest impact. With rankings, we can use the same metric both for download and citation predictions. For the NBER data, we ranked test-set papers based on the predicted numbers of downloads and computed the correlation to the actual numbers of downloads. For the ACL data, we ranked papers based on the *probability* of being cited (within the next three years) and computed the correlation to the actual numbers of citations.[12]

To measure ranking models' ranking quality, we used Kendall's τ, a nonparametric statistic that measures the similarity of two different orderings over the same set of items. Here, the items are scientific papers and the two metrics are the gold standard numbers of downloads (or citations) and model predictions for the numbers of downloads, or citation probabilities. If q is the chance that a randomly drawn pair of items will be ranked in the same way by the two metrics, then $\tau = 2(q - 0.5)$.

Table 5 shows Kendall's τ for each model for the forecasting tasks (i.e., prediction of future citations or downloads) in both datasets. As in the previous experiments, we see small benefits for the time series regression model on most held-out data splits—and larger benefits for including text features along with metadata features.

6 Analysis

An advantage of the time series regularized regression model is its interpretability. Inspecting feature coefficients in the model allows us to identify trends and changes of interests over time within a scientific community.

[12]Here, we use models of responses to individual papers for ranking (i.e., in a pointwise ranking scheme). Time series regularization could also be applied to ranking models that model pairwise preferences to optimize metrics like Kendall's τ directly, as discussed by Joachims (2002).

Feat.	Model	NBER		ACL		
		'08	'09	'04	'05	'06
Meta	one year	.29	.22	.17	.08	.16
Meta	all years	.31	.22	.15	.12	.21
Meta	time series	.29	.22	.14	.10	.17
Full	one year	.35	.31	.44	.39	.33
Full	all years	**.43**	.37	.42	.43	.40
Full	time series	**.43**	**.38**	**.47**	**.44**	**.43**

Table 5: Kendall's τ rank correlation for future prediction models on both datasets.

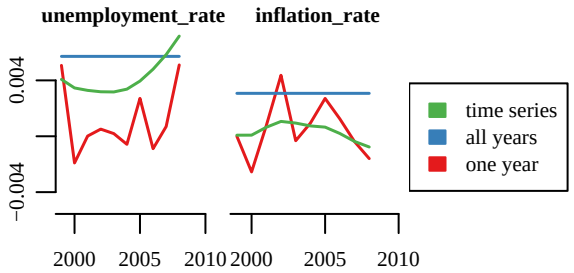


Figure 4: Coefficients for two NBER bigram features.

First, we illustrate the difference between the time series and the other models in Figure 4, for NBER models' weights for *unemployment rate* and *inflation rate* appearing in a paper's abstract. The year-to-year weights of "one year" models fluctuate substantially, and the "all years" model is necessarily constant, but the time series regularizer gives a smooth trajectory.

6.1 Trends

Previous work has examined the flow of ideas as trends in word and phrase frequencies, as in the Google Books Ngram Viewer (Michel et al., 2011).[13] Topic models have been used extensively to explore trends in low-dimensional spaces (Blei and Lafferty, 2006; Wang et al., 2008; Wang and McCallum, 2006; Ahmed and Xing, 2010). By contrast, our approach allows us to examine trends in the *impact* of text related to specific observation variables: the coefficient trendline for a feature illustrates its association with measurements of scholarly impact (citation and download frequency).

Text frequencies can be quite different from the discriminative weights our model assigns to features. Figure 5 illustrates the $\beta_{t,j}$ trends in the ACL time series model for some selected terms that oc-

[13]`http://ngrams.googlelabs.com`

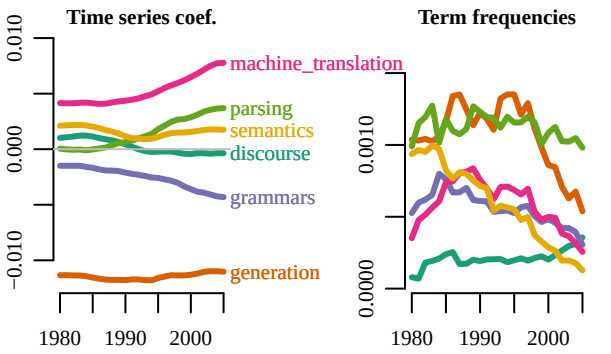
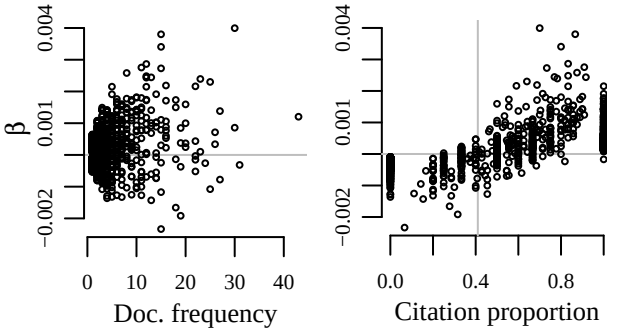


Figure 5: Feature trends: model coefficients vs. term frequencies over time in the ACL corpus. Term freq. is the fraction of tokens (or bigrams for *m.t.*) that year, that are the term, averaged over a centered five-year window.



(a) Doc. freq. vs. coef. (b) Citation prop. vs. coef.

Figure 6: Analysis of author citation coefficients. Every point is one ACL author, and the vertical axis shows the citation coefficient, compared to (a) the number of documents co-authored by the author; and (b) the proportion of an author's papers that are cited within three years. The vertical bar is the *macro*-averaged citation proportion across authors, 41%.

cur frequently in conference session titles. On the right are term frequencies (with smoothing, since year-to-year frequencies are bumpy). Most terms decline over time. On the left, by contrast, are the weights learned by our time series model. They tell a very different story: for example, parsing has shown a definite increase in interest, while interest in grammars (e.g., formalisms) has declined somewhat. These trends have face validity, giving credence to our analysis; they also broadly agree with Hall et al. (2008).

6.2 Authors

The regression method also allows analysis of author influence, since we fit a coefficient for each of the authors in the ACL dataset. Figure 6(a) addresses the following question: do prolific authors get cited more often, even after accounting for the content of their papers?[14] The effect is present but relatively small according to our model: the total number of papers co-authored by an author has a weak correlation to the author's citation prediction coefficient ($\tau = 0.16$).

Next, does the model provide more information than the simple citation probability of an author? Figure 6(b) compares coefficients to an author's papers' probability of being cited. Since we did not prune author features, there are many authors with only a few papers, resulting in unsmoothed probabilities of 0, 0.5, 1, etc. (these correspond to the vertical "bands" in the plot). By contrast, the ℓ_2-penalty of the model naturally assigned coefficients close to zero for such authors if it is justified.

In general, the simple probability agrees with the coefficient, but there are differences. The semantics of the regression imply we are measuring the relative citation probability of an author, *controlling* for text and venue effects. If an author has a high citation prediction coefficient but a low citation probability, that implies the author has better-cited work than would be expected according to the n-grams in his or her papers. We have omitted names of authors from the figure for clarity and confidentiality, but high outlier authors tend to be well-known researchers in the ACL community. Obviously, since the prediction model is not perfect, it is not possible to completely verify this hypothesis, but we feel this analysis is reasonably suggestive.

7 Related Work

Previous work on modeling scientific literature mostly focused on citation graphs (Borner et al., 2003; Qazvinian and Radev, 2008). Some researchers, e.g., Erosheva et al. (2004), have used text content. Most of these are based on topic models: Gerrish and Blei (2010) measure scholarly impact, Hall et al. (2008) study the "history of ideas",

[14] More precisely: if a prolific author and a non-prolific author write a paper, does the prolific author's paper have a higher probability of being cited than the non-prolific author's, all other things being equal?

and Ramage et al. (2010) rank universities based on scholarly output using topic models.

Download rates and citation prediction were two of the main tasks in the KDD Cup 2003 (McGovern et al., 2003; Brank and Leskovec, 2003). Bethard and Jurafsky (2010) considered the problem slightly differently and proposed an information retrieval approach to citation prediction. Our approach is novel in that we formulate the problem as a forecasting task and we seek to predict *future* impact of articles.

Linear regression with text features has been used to predict financial risk (Kogan et al., 2009) and movie revenues (Joshi et al., 2010). While the forecasts in those papers are similar to ours, those authors did not consider a forecast gap or allowing the parameters of the model to vary over time.

Our time series regularization is closely related to the fused lasso (Tibshirani et al., 2005). It penalizes a loss function by the ℓ_1-norm of the coefficients and their differences. The ℓ_1-penalty for differences between coefficients encourages *sparsity* in the differences. We use the ℓ_2-norm to induce *smooth* changes across time steps.

8 Conclusions

We presented a statistical approach to predicting a scientific community's response to an article, based on its textual content. To improve the interpretability of the linear model, we developed a novel time series regularizer that encourages gradual changes across time steps. Our experiments showed that text features significantly improve accuracy of predictions over baseline models, and we found that the feature weights learned with the time series regularizer reflect important trends in the literature.

Acknowledgements

We thank the National Bureau of Economic Research for providing the NBER dataset for this research, Fallaw Sowell for helpful discussions, and three anonymous reviewers for comments on an earlier draft of this paper. This research was supported by the Intelligence Advanced Research Projects Activity under grant number N10PC20222 and TeraGrid resources provided by the Pittsburgh Supercomputing Center under grant number TG-DBS110003.

References

A. Ahmed and E. P. Xing. 2010. Timeline: A dynamic hierarchical Dirichlet process model for recovering birth/death and evolution of topics in text stream. In *Proc. of UAI*.

A. L. Berger, V. J. Della Pietra, and S. A. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

S. Bethard and D. Jurafsky. 2010. Who should I cite? Learning literature search models from citation behavior. In *Proc. of CIKM*.

D. Blei and J. Lafferty. 2006. Dynamic topic models. In *Proc. of ICML*.

K. Borner, C. Chen, and K. Boyack. 2003. Visualizing knowledge domains. In B. Cronin, editor, *Annual Review of Information Science and Technology*, volume 37, pages 179–255. Information Today, Inc.

G. Box, G. M. Jenkins, and G. Reinsel. 2008. *Time Series Analysis: Forecasting and Control*. Wiley Series in Probability and Statistics.

S. Boyd and L. Vandenberghe. 2004. *Convex Optimization*. Cambridge University Press.

J. Brank and J. Leskovec. 2003. The download estimation task on KDD Cup 2003. *SIGKDD Explorations*, 5(2):160–162.

A. Cameron and P. Trivedi. 1998. *Regression Analysis of Count Data*. Cambridge University Press.

E. Erosheva, S. Fienberg, and J. Lafferty. 2004. Mixed membership models of scientific publications. In *Proc. of PNAS*.

S. Gerrish and D. M. Blei. 2010. A language-based approach to measuring scholarly impact. In *Proc. of ICML*.

D. Hall, D. Jurafsky, and C. D. Manning. 2008. Studying the history of ideas using topic models. In *Proc. of EMNLP*.

J. D. Hamilton. 1994. *Time Series Analysis*. Princeton University Press.

T. Hastie, R. Tibshirani, and J. Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

A. E. Hoerl and R. W. Kennard. 1970. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

T. Joachims. 2002. Optimizing search engines using clickthrough data. In *Proc. of KDD*.

M. Joshi, D. Das, K. Gimpel, and N. A. Smith. 2010. Movie reviews and revenues: An experiment in text regression. In *Proc. of HLT-NAACL*.

S. Kogan, D. Levin, B. R. Routledge, J. S. Sagi, and N. A. Smith. 2009. Predicting risk from financial reports with regression. In *Proc. of HLT-NAACL*.

D. C. Liu and J. Nocedal. 1989. On the limited memory BFGS method for large scale optimization. *Mathematical Programming B*, 45(3):503–528.

P. Mccullagh and A. J. Nelder. 1989. *Generalized Linear Models*. London: Chapman & Hall.

P. McCullagh. 1980. Regression models for ordinal data. *Journal of the Royal Statistical Society B*, 42(2):109–142.

A. McGovern, L. Friedland, M. Hay, B. Gallagher, A. Fast, J. Neville, and D. Jensen. 2003. Exploiting relational structure to understand publication patterns in high-energy physics. *SIGKDD Explorations*, 5(2):165–172.

J. Michel, Y. Shen, A. Aiden, A. Veres, M. Gray, The Google Books Team, J. Pickett, D. Hoiberg, D. Clancy, P. Norvig, J. Orwant, S. Pinker, M. Nowak, and E. Aiden. 2011. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182.

V. Qazvinian and D. R. Radev. 2008. Scientific paper summarization using citation summary networks. In *Proc. of COLING*.

D. R. Radev, M. T. Joseph, B. Gibson, and P. Muthukrishnan. 2009a. A bibliometric and network analysis of the field of computational linguistics. *Journal of the American Society for Information Science and Technology*.

D. R. Radev, P. Muthukrishnan, and V. Qazvinian. 2009b. The ACL anthology network corpus. In *Proc. of ACL Workshop on Natural Language Processing and Information Retrieval for Digital Libraries*.

D. Ramage, C. D. Manning, and D. A. McFarland. 2010. Which universities lead and lag? Toward university rankings based on scholarly output. In *Proc. of NIPS Workshop on Computational Social Science and the Wisdom of the Crowds*.

R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight. 2005. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society B*, 67(1):91–108.

X. Wang and A. McCallum. 2006. Topics over time: A non-Markov continuous-time model of topical trends. In *Proc. of KDD*.

C. Wang, D. Blei, and D. Heckerman. 2008. Continuous time dynamic topic models. In *Proc. of UAI*.