Predicting Responses and Discovering Social Factors in Scientific Literature

Dani Yogatama, Michael Heilman, Brendan O'Connor, Chris Dyer,
Bryan R. Routledge, and Noah A. Smith

CMU-LTI-11-015

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
www.lti.cs.cmu.edu

Predicting Responses and Discovering Social Factors in Scientific Literature

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 first two models investigate temporal and spatial aspects of scientific community's interests. A third model which jointly summarizes scientific articles when making predictions is also presented. Lastly, we propose a generative approach to explore what social factors influence written scientific articles.

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 report 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).

In the first three models, 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 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. Our approaches substantially outperform text-ignorant baselines on ground-truth predictions. Our last model uses a generative approach to model scientific literature. We show that written scientific communication is influenced by various social factors, and we can uncover these factors by measuring language similarity between articles.

Our major contributions can be summarized as follows:

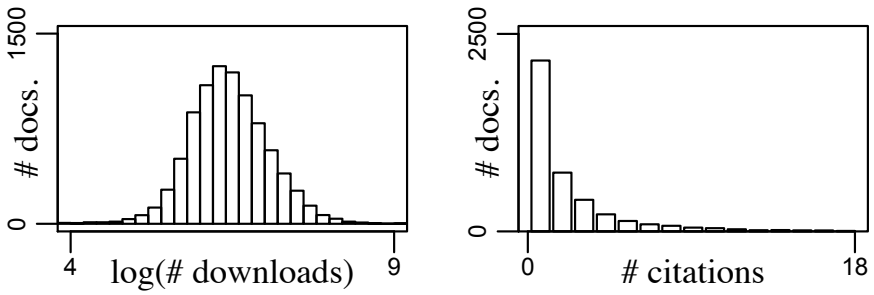


Figure 1: Left: the distribution of log download counts for papers in the NBER dataset one year after posting. Right: the distribution of within-dataset citations of ACL papers within three years of publication (outliers excluded for readability).

- We introduce a new regularization technique that leverages the intuition that the relationship between observable features and response should evolve smoothly over time (§3). 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). It permits flexibility in features and offers a novel and perhaps more interpretable view of the data than summary statistics.

- We show how to use a multiple output regularized linear model to forecast and discover a regional scientific community's interest (§4). The model incorporates global and local features to capture correlation between scientific responses in each region and applies "lasso" regularization to encourage sparsity.

- We propose a latent variable regression model which jointly summarizes an article while making a prediction (§5). In this model, there is a latent variable for each sentence in a document which determines whether a sentence is included or not in the summary, and sentences that are not included in the summary are not considered when making the prediction for a document. We also discuss efficient learning and inference procedures for this model.

- We introduce an influence language model (§6) to test which factors influence authors when writing a scientific paper. Specifically, we investigate spatial, temporal, and social dimensions of written scientific communication.

This report is an extended version of our earlier paper (Yogatama et al., 2011), which corresponds to most of §3.

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. All experiments were conducted on these datasets or a subset of them.

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

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.

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 used text from the abstracts in these experiments.

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, excluding self-citations 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 Temporal Model

First, we explore a class of models which captures first-order temporal effects under the intuition that the model should make use of a long history of training data but rely more strongly on more recent evidence. 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).[3] For 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

[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] Preliminary experiments found no consistent benefit from ℓ_1 ("lasso") models, though we note that ℓ_1-regularization leads to sparse, compact models that may be more interpretable. We also experimented with structured sparsity, i.e., group "lasso" and fused "lasso", but the results were not encouraging and the training procedure was considerably more complex.

"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:

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

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.[4] 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.

[4]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.

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

Scientific text has distinct time series properties that reflect temporal variation in interests and techniques. 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 $t' \neq t$.

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 $\beta_{t+1,j}$). This seems at odds with our intuitive understanding of scientific trends.

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.[5] 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).

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

[5] 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).

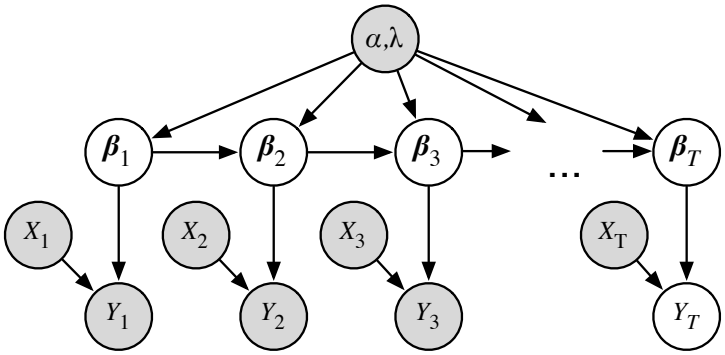


Figure 2: Time series regression as a graphical model; the variables $\boldsymbol{X}_t$ and Y_t are the sets of feature vectors and response variables from documents dated t.

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.[6] 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).

3.4 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. These two tasks correspond to different measures of impact of a scholarly work. 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.

[6] 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.

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).[7] 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.).[8]

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 §3.4).
- Log transformed word counts. We include features for the numbers of words in the title and the abstract (NBER) or the full text (ACL).

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" (see Figure 3 for an illustration). 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 without extrapolation.

[7] 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`

[8] 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.

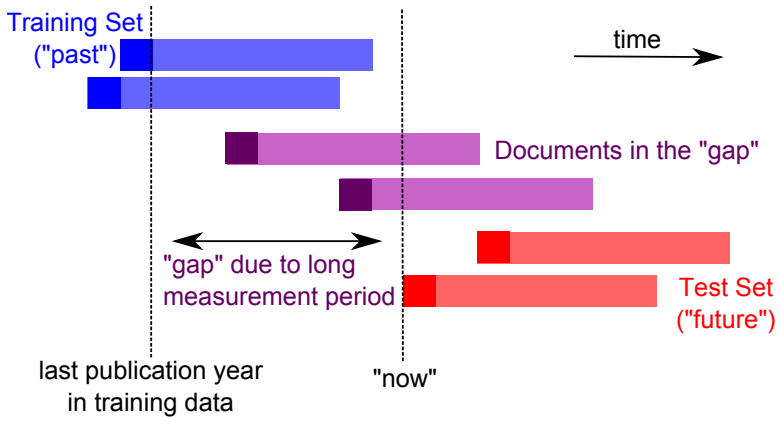


Figure 3: Above, a timeline showing how a measurement follows a paper's date of publication. Below, the emergence of the forecast gap, a period prior to the testing period during which papers may be published, but their influence measurements are not known. Papers during the forecast gap are most highly relevant to test-time trends.

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).

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.[9] 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 4 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

[9] Making predictions using the median leads to performance that is very similar to using the mean.

[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%.

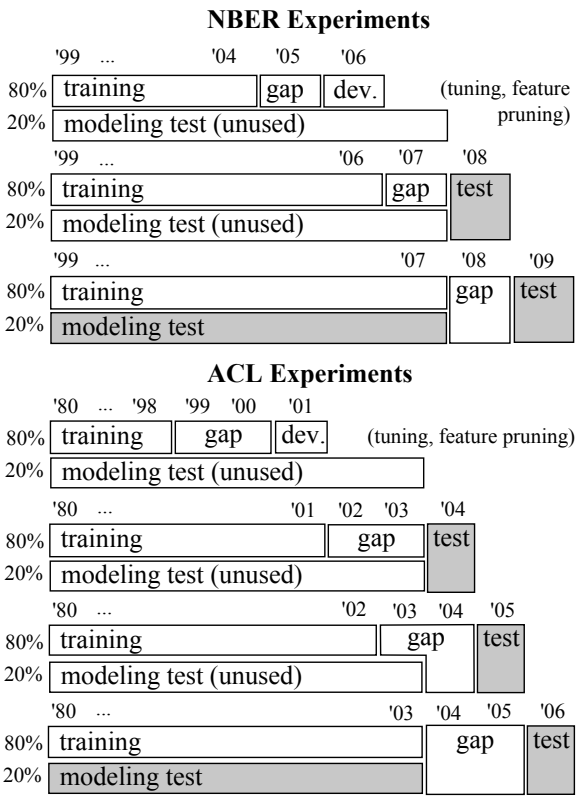


Figure 4: An illustration of how the datasets were segmented for the experiments. Portions of data for which we report results are shaded. Time spans are not to scale.

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 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.

[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	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	2005	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.

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 4) 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

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.

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 significantly 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.

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.

3.5 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.

Trends

First, we illustrate the difference between the time series and the other models in Figure 5, 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 that reflects the underlying trend in paper downloads.

[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).

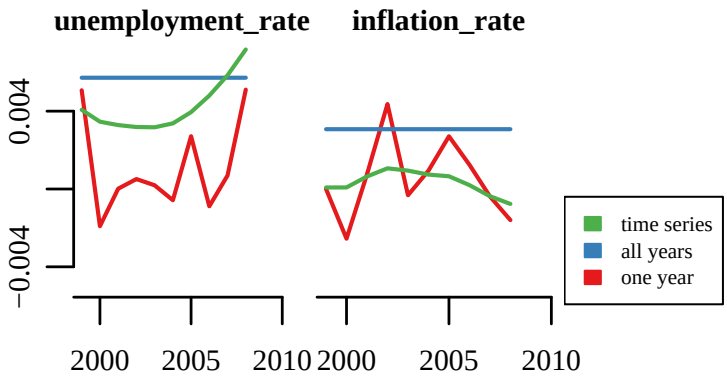


Figure 5: Coefficients for two NBER bigram features.

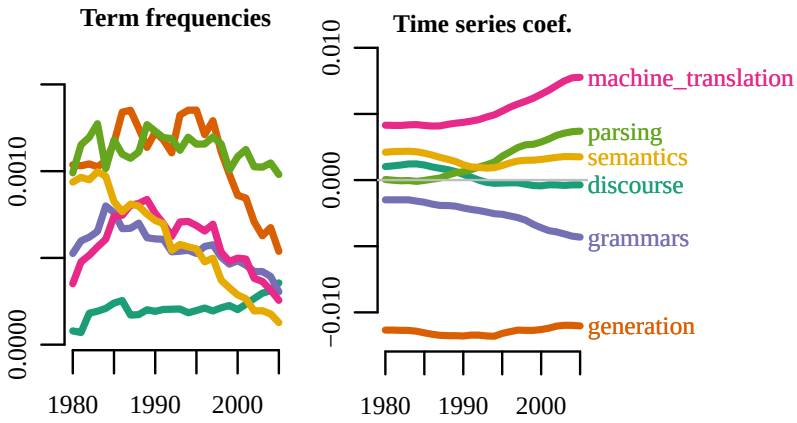


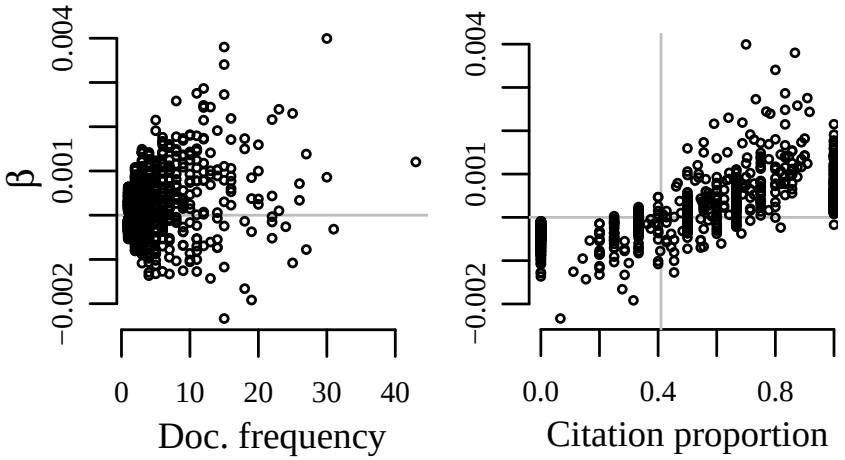
Figure 6: Feature trends: term frequencies vs. model coefficients over time in the ACL corpus. Term freq. is the fraction of tokens (or bigrams for *m.t.*) that year, that are the term, averaged over a centered five-year window.

We leave to future research whether there is a link between this trend in download weights and the measured quantities in the macroeconomy. Over this period, US unemployment did rise and US inflation fell.

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 6 illustrates the $\beta_{t,j}$ trends in the ACL time series model for some selected terms that occur frequently in conference session titles. On the left are term frequencies (with smoothing, since year-to-year frequencies are bumpy). Most terms decline over time. On the right, 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

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



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

Figure 7: 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%.

credence to our analysis; they also broadly agree with Hall et al. (2008).

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 7(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 7(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.

[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?

4 Spatial Model

We now turn our attention to a spatial model which captures regional differences in scientific interests. For example, economists in North America might be interested in different "topics"[15] than economists in Africa. In this section, we introduce a model which predicts scientific responses and identifies trends in different regions. We make use of the NBER dataset, so we will consider downloads as the responses.

4.1 Model

Consider a model that predicts an output $\boldsymbol{y}$ given an input $\boldsymbol{X}$ for an article. When predicting regional downloads, y_r is the number of downloads in region r, and $\sum_r y_r$ is the total number of downloads for a paper in all regions. For the multiple output linear regression model, for each region, given a vector input $\boldsymbol{x} \in \mathbb{R}^{2d}$, the prediction $\hat{y_r} \in \mathbb{R}$ is computed as:

$$\hat{y_r} = \sum_{g=0}^{d} \beta_g x_g + \sum_{l=0}^{d} \beta_{rl} x_l,$$

where $\boldsymbol{\beta}$ is a matrix of coefficients parameterizing the model. Each β_g is shared (global) feature which fires in every region, and each β_{rl} is unshared (local) feature which fires only in region r. Local features are constructed by conjoining β_g with region r for each $r \in R$. Since there are d global features, we have rd local features. When making predictions about y_r, we only consider global features $\boldsymbol{\beta_g}$ and relevant local features $\boldsymbol{\beta_r}$.

We can write the model in matrix notation:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{E},$$

where $\boldsymbol{Y} \in \mathbb{R}^{M \times R}$ is the response matrix, $\boldsymbol{X} \in \mathbb{R}^{M \times 2d}$ is the input matrix, $\boldsymbol{\beta} \in \mathbb{R}^{2d \times R}$ is the parameter matrix, and $\boldsymbol{E} \in \mathbb{R}^{M \times R}$ is the error matrix.

For this model, the regularized paramater estimate becomes:

$$\arg\min_{\boldsymbol{\beta}} R(\boldsymbol{\beta}) + \sum_{i=1}^{M} \sum_{j=1}^{R} (y_{ij} - \hat{y}_{ij})^2$$

ℓ_1 regularization ("lasso") is especially good for this task, since we expand the feature space and many textual features are irrelevant to the regional values being predicted. It yields sparse models with several thousand non-zero weights from hundreds of thousands of features, which can easily be interpreted to reveal scientific community's interests in each region. Specifically, $R(\boldsymbol{\beta})$ is defined as[16]:

$$\begin{aligned} R(\boldsymbol{\beta}) &= \lambda_g \|\boldsymbol{\beta_g}\|_1 + \lambda_l \sum_{r=1}^{R} \|\boldsymbol{\beta_r}\|_1 \\ &= \lambda_g \sum_{g=1}^{d} |\beta_g| + \lambda_l \sum_{r=1}^{R} \sum_{l=1}^{d} |\beta_{rl}|. \end{aligned}$$

We use the orthant-wise limited memory quasi-Newton (OWL-QN) method (Andrew and Gao, 2007), an unconstrained optimization method which is based on the L-BFGS quasi-Newton algorithm, to estimate the parameters of the model.

[15] We define topics very loosely here. In our model, topics are represented as bags of words.

[16] In our experiments, during parameter sweeps, we always set penalties to global features λ_g and local features λ_l to the same value.

Region	Baseline (Mean)	LR	SR	MR
Overall	51.22	48.21	**45.86**	47.36
North America	105.67	120.86	**94.03**	97.01
South America	28.76	**24.41**	29.18	25.72
Western Europe	102.88	**86.39**	86.95	90.93
Eastern Europe	23.26	**19.50**	20.99	20.60
Africa	9.06	**8.06**	8.76	8.94
Asia	79.85	70.81	**73.60**	80.90
Australia	8.50	7.47	7.52	**7.42**

Table 6: Mean absolute error for NBER regional download prediction. Overall results are *macro-averaged* across regions. **LR** is a linear regression model trained on the whole dataset without local features (except a bias feature for each region). **SR** is a linear regression model trained on each region separately. **MR** is the multiple output regression model with global and local features.

4.2 Experiments

Data splits and features

Since 2005, the NBER started to log IP addresses of downloaders. We map these IP addresses to regions using GeoIP.[17] In this experiment, we arbitrarily divided the world into seven regions (North America, South America, Western Europe, Eastern Europe, Africa, Asia, Australia), and predicted the number of downloads in *six months* in each region.[18] We used 3588 papers from 2005–08 to train the model, 508 papers from the first half of 2009 as the development set, and tested the model on 555 papers from the first half of 2010. Since we had to expand the feature space to incorporate global and local features resulting in a large number of features, we did not use the full NBER feature set but only used meta features and unigram text features.

Results

We compared the mean absolute errors of the multiple output linear regression models with global and local features ("MR" model) to $R = 7$ separate ℓ_1-regularized linear regression models (one for each region) with only local features ("SR" model), to a single ℓ_1-regularized regression model trained on the whole dataset with bias features for each region ("LR" model), and to a simple baseline which always uses the mean of the training and development data as the prediction.

Table 6 shows the results of our experiments. The results suggest that introducing globally shared features does not yield consistent performance improvements compared to training a separate linear regression model for each region, nor does it give consistent benefits over a single global model with regional biases. Nonetheless, our model was able to discover distinctive regional features which reveal regional scientific interests, as described in the next section.

4.3 Analysis

Although using both global and local features did not result in performance improvements over other models, it allows us to directly compare community's interests across the world by inspecting highly weighted features in the resulting model. For example, according to the model, most people regardless of regions read

[17] `http://www.maxmind.com/app/city`

[18] Note that this is different than the previous experiments where we predicted the number of downloads one year after a paper's publication date. We chose six months since we have smaller dataset with IP addresses and we wanted to have a reasonable number of papers for training.

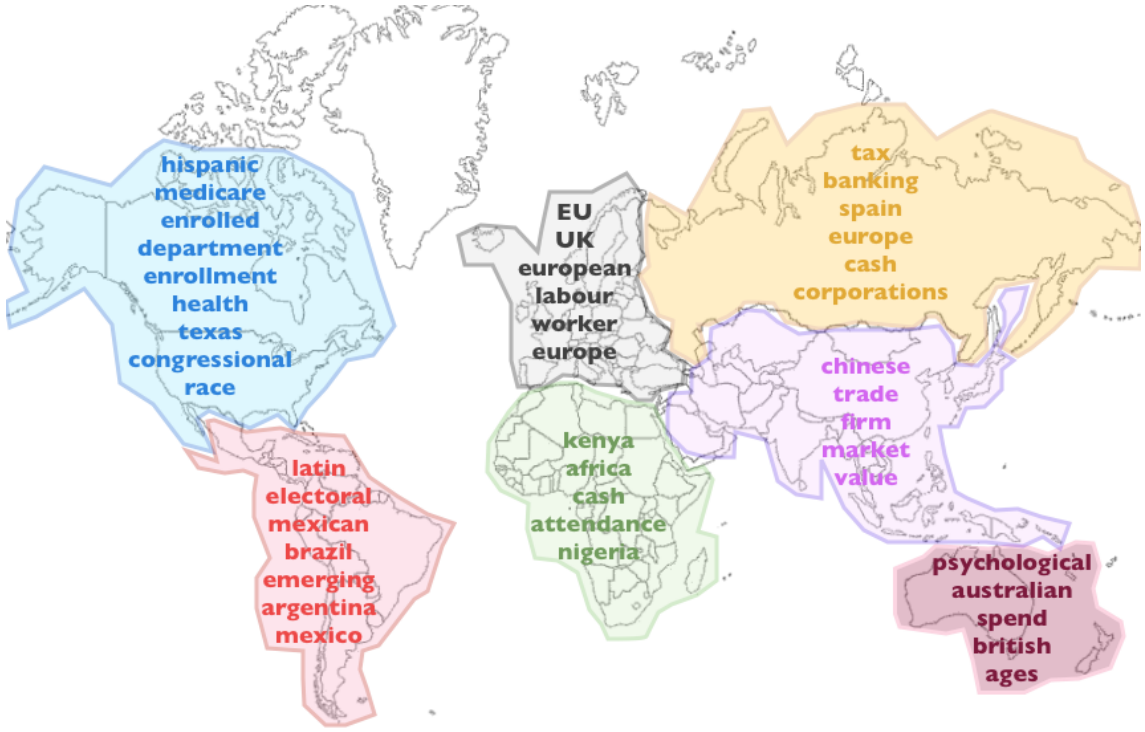


Figure 8: Regionally popular words. Borders are only for illustration purpose and are not accurate.

papers about *productivity*, *bailout*, and *economics*, but are less interested in papers about *gambles*, *safety*, and *protective*.

Figure 8 and Figure 9 show popular and unpopular words in each region respectively. We can see that Asian readers are more interested in *chinese*, *trade*, *market*, *firm*, and *value*, but are less interested in words such as *causal* and *war*. On the other hand, readers in North America are interested in *hispanic*, *medicare*, and *health*, among others. This tells us that readers in different parts of the world do not share similar interests in economics papers, but rather choose to focus on papers about specific "topics" (represented by words in our model) more closely related to their regions. Importantly, similar to our temporal model, the results here also have face validity, indicating that the model learned reasonably suggestive feature coefficients. It further supports our hypothesis that inspecting feature coefficients learned by a linear model allows us to discover social phenomena in a scientific community.

5 Latent Variable Logistic Regression Model

In this section, we consider a model that incorporates latent variables to determine whether a sentence is included or not when making a prediction. Under this model, if we regard the selected sentences as a summary of a document under consideration, the model can be understood as a response-guided summarization model. We describe the model first and show experimental results subsequently.

5.1 Model

Consider a document with S sentences. Our model is a Markov random field with an input variable $\boldsymbol{x}$, a latent variable $\boldsymbol{h} = \langle h_1, h_2, \ldots, h_S \rangle$, and a response variable $y \in \{0, 1\}$. The graphical representation of

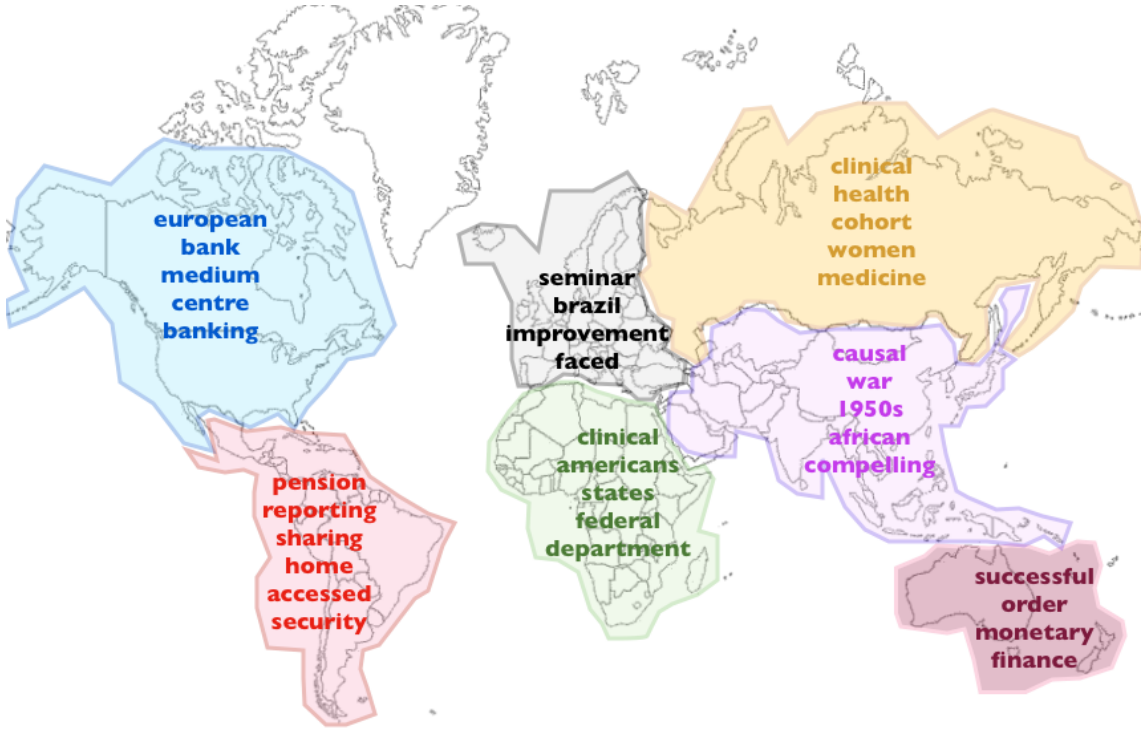


Figure 9: Regionally unpopular words. Borders are only for illustration purpose and are not accurate.

the model is presented in Figure 10. The model has three parameters, $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, and $\boldsymbol{\beta}$ with parametric form:

$$p(y, \boldsymbol{h}|\boldsymbol{x}) = \frac{\exp\left(\boldsymbol{\theta} \cdot \sum_{s=1}^{S} f(\boldsymbol{x}, h_s) + \boldsymbol{\phi} \cdot \sum_{s=2}^{S} f(h_s, h_{s-1}) + \boldsymbol{\beta} \cdot \sum_{s=1}^{S} f(\boldsymbol{x}, h_s, y)\right)}{\sum_{y'} Z(y'|\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\beta})},$$

$$Z(y'|\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\beta}) = \sum_{\boldsymbol{h'}} \exp\left(\boldsymbol{\theta} \cdot \sum_{s=1}^{S} f(\boldsymbol{x}, h'_s) + \boldsymbol{\phi} \cdot \sum_{s=2}^{S} f(h'_s, h'_{s-1}) + \boldsymbol{\beta} \cdot \sum_{s=1}^{S} f(\boldsymbol{x}, h'_s, y')\right)$$

With the independence assumption shown in the figure, the partition function $Z(y'|\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\beta})$ factors as follows:

$$\begin{aligned} Z(y'|\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\beta}) &= \sum_{\boldsymbol{h'}} \exp\left(\boldsymbol{\theta} \cdot \sum_{s=1}^{S} f(\boldsymbol{x}, h'_s) + \boldsymbol{\phi} \cdot \sum_{s=2}^{S} f(h'_s, h'_{s-1}) + \boldsymbol{\beta} \cdot \sum_{s=1}^{S} f(\boldsymbol{x}, h'_s, y')\right) \\ &= \prod_{\boldsymbol{c}} \sum_{\langle h_c^1, h_c^2 \rangle} \exp\left(\boldsymbol{\theta} \cdot f(\boldsymbol{x}, h_c^1) + \boldsymbol{\theta} \cdot f(\boldsymbol{x}, h_c^2) + \boldsymbol{\phi} \cdot f(h_c^1, h_c^2) + \boldsymbol{\beta} \cdot f(\boldsymbol{x}, y', h_c^1) + \boldsymbol{\beta} \cdot f(\boldsymbol{x}, y', h_c^2)\right) \end{aligned}$$

where $\boldsymbol{c} = \langle c_1, c_2, \ldots, c_{S-1} \rangle$ are cliques in the graph, each consisting four variables $\boldsymbol{x}, h_s, h_{s+1}, y$, and $\langle h_c^1, h_c^2 \rangle \in \{\langle 0, 0 \rangle, \langle 0, 1 \rangle, \langle 1, 0 \rangle, \langle 1, 1 \rangle\}$.

The conditional likelihood of the corpus is obtained by marginalizing out the latent variables $\boldsymbol{h}$:

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\beta}) = -\sum_{i=1}^{M} \log \sum_{\boldsymbol{h_i}} P(y_i, \boldsymbol{h_i}|\boldsymbol{x_i}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\beta}).$$

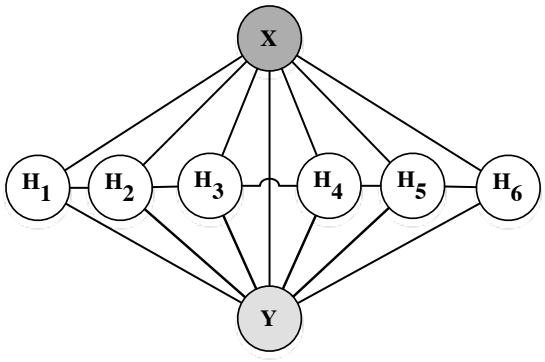


Figure 10: Graphical model representation of the latent variable logistic regression model.

Although we can use any regularization technique, for simplicity, we apply "ridge" penalty to each feature coefficient and so our full objective function becomes:

$$\underset{\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\beta}}{\operatorname{argmin}} \mathcal{L}(\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\beta}) + R(\boldsymbol{\theta}) + R(\boldsymbol{\phi}) + R(\boldsymbol{\beta})$$

We use L-BFGS to estimate the parameters of this model. The first derivatives with respect to the regularization terms are the same as in the previous section, while the first derivatives with respect to the loss function are:

$$\frac{\partial \mathcal{L}}{\partial \theta_k} = \sum_{i=1}^{M} \mathbb{E}_{p_{\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\beta}}(h_s|\boldsymbol{x_i},y_i)}\left[f_k(\boldsymbol{x_i},h_s)\right] - \mathbb{E}_{p_{\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\beta}}(h_s,y'|\boldsymbol{x_i})}\left[f_k(\boldsymbol{x_i},h_s)\right]$$

$$\frac{\partial \mathcal{L}}{\partial \phi_k} = \sum_{i=1}^{M} \mathbb{E}_{p_{\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\beta}}(h_s,h_{s+1}|\boldsymbol{x_i},y_i)}\left[f_k(h_s,h_{s+1})\right] - \mathbb{E}_{p_{\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\beta}}(h_s,h_{s+1},y'|\boldsymbol{x_i})}\left[f_k(h_s,h_{s+1})\right]$$

$$\frac{\partial \mathcal{L}}{\partial \beta_k} = \sum_{i=1}^{M} \mathbb{E}_{p_{\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\beta}}(h_s|\boldsymbol{x_i},y_i)}\left[f_k(\boldsymbol{x_i},h_s,y_i)\right] - \mathbb{E}_{p_{\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\beta}}(h_s,y'|\boldsymbol{x_i})}\left[f_k(\boldsymbol{x_i},h_s,y')\right]$$

Since the graph forms a chain once we observe y, inference in this model can be performed efficiently using belief propagation. To compute the partition function, we can simply run belief propagation twice by fixing y to 0 and 1 respectively. As an alternative to marginalizing the latent variables, we can instead maximize $\boldsymbol{h}$. In this setting, inference can be solved using the well-known Viterbi algorithm. Lastly, we use the following rule to make a prediction:

$$\hat{y} = \underset{y}{\operatorname{argmax}}\, p(y|\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\beta}) = \underset{y}{\operatorname{argmax}} \frac{Z(y|\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\beta})}{\sum_{y'} Z(y'|\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\beta})}$$

We note that this model can be easily generalized to handle multiclass classification problems.

5.2 Experiments

Data splits and features

Since the current model only works for categorical responses, we tested it on the ACL dataset to predict whether a paper will be cited or not within three years after its publication. We used the same data split as in

the previous section, but we segmented every document into sentences using MxTerminator[19] (Ratnaparkhi, 1996), and we removed papers with less than ten sentences (most likely these papers have OCR errors).

Recall that in this model, we have three sets of features $f(\boldsymbol{x}, h_s)$, $f(h_s, h_{s+1})$, and $f(\boldsymbol{x}, h_s, y)$. The first feature set $f(\boldsymbol{x}, h_s)$ is used to predict whether a sentence is included or not in making the prediction. For a sentence s, we have a bias, unigrams, bigrams, and trigrams of the sentence as features. More recently, we have been experimenting with including document position features which indicate the position of the sentence in the document. For example, a sentence can be located in the middle or at the end of the document. Sentences in abstracts and conclusions sections are arguably more important than sentences in the experiments section. For this reason, sentences in different parts of the document should have different probabilities to be included in making predictions. We normalize the position of the sentence relative to the beginning of the document with the document length, so every sentence has a position from 0 to 1. We have three thresholds at $0.25, 0.5, 0.75$, and four binary features for each area between the thresholds. Note that out of these four position features, only one will fire for every sentence. We observed that introducing the document position feature leads to a small improvement in overall document prediction accuracy. However, the results reported here did not include this feature since we have not had full results for these experiments as of the submission of this report.

The second feature set $f(h_s, h_{s+1})$ is parameterized by $\boldsymbol{\phi}$. We only have a single feature which fires when h_s is equal to h_{s+1} (which indicates whether sentence s and sentence $s+1$ are both selected or not selected). To encourage the model to select *block* of sentences for interpretability of the summary, we put a prior on $\boldsymbol{\phi}$ to promote h_s and h_{s+1} to have the same value by incorporating it in the regularization term $R(\boldsymbol{\phi})$.

The last feature set $f(\boldsymbol{x}, h_s, y)$ are binary features for unigrams, bigrams, and trigrams of every selected sentence in the document. We also have a bias feature for every article. In these preliminary experiments, we did not use unigrams, bigrams, and trigrams for the title of the document, and meta features such as publication venue and authors which always fire, although they can easily be incorporated to the model.

Results

Table 7 shows experimental results on the ACL dataset. We tested four different inference strategies for training and testing time, since we can either marginalize or maximize over the latent variables at each case. We compared the method with a vanilla logistic regression model trained on the whole document with ℓ_2-penalty. For the latent variable logistic regression model, results are averaged over five runs. There are a few hyperparameters that need to be tuned for this the model. First, we have λ, the regularization penalty. We tried $\lambda \in \{10^{1,0,\ldots,-7,-8}\}$, selecting the best value using the development data. The prior for $\boldsymbol{\phi}$ was set to 0.1 and 0.01 and the bias feature for sentence inclusion (θ_0) was set to -2.0 and -0.1 when *training* using sum and max inference respectively. These numbers were selected to get a reasonable number of sentences for the summary and were tuned on the development data. Although our preliminary results are not strong, we believe that the model still has a lot of potential. Specifically, with a better initialization technique and introduction of more features for selecting sentences, we believe that the results will eventually be better compare to vanilla ridge logistic regression model.

6 Influence Language Model

We aim to estimate parameters for a log-linear distribution of influence among scientific papers, that takes into account features such as geographical proximity, source institutions, author names, and temporal proximity measures. We make the assumption that influence can be approximated by similarity of language. Specifically, we assume that whether a document x_j has been influenced by a past document x_k can be

[19] `ftp://ftp.cis.upenn.edu/pub/adwait/jmx/`

Feat.	Model	Forecasting		
		2004	2005	2006
–	majority	56	60	50
Text	LR	**70**	**70**	**71**
Text	max-max	63	60	59
Text	max-sum	63	58	57
Text	sum-max	61	62	57
Text	sum-sum	67	67	66

Table 7: Classification accuracy (%) for predicting whether ACL papers will be cited within three years. The models here use only text (no metadata features). "Max" and "sum" refer to the type of inference (hard or soft) performed during learning and during test-time prediction (e.g., "max-sum" means hard inference during learning, soft inference at test time).

modeled using the probability assigned to the words and multi-word phrases in x_j according to a language model estimated from x_k.

The goal of the model is to test what factors affect influence. Suppose we have a parameter vector $\boldsymbol{\beta}$. A parameter in $\boldsymbol{\beta}$ being reliably non-zero provides evidence that the corresponding feature significantly affects influence. For example, we can ask questions about the effects of time and geography: a negative weight on a measure of geographic distance would indicate that papers are more likely to be influenced by papers from nearby institutions (e.g., Harvard and Yale versus Harvard and the University of Chicago).

6.1 Model

Let j be the index for a current document and k be the index for a past document. Let $\mathbf{f}(x_j, x_k)$ be a function that returns a vector of feature values for a specified pair of current and past documents, and let N_j be the number of word tokens in the sequence w_j. The log likelihood function for the model is:

$$\log p(\mathbf{W} \mid \boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{x}) = \sum_j \sum_i^{N_j} \log \sum_k p_{\boldsymbol{\beta}}(z_{ji} = k \mid x_j, x_k) p_{\boldsymbol{\theta}}(w = w_{ji} \mid k, h_{ji}) \tag{2}$$

$p_{\boldsymbol{\theta}}$ is a fixed language model probability for the current word w_{ji} given the previous words h_{ji} (i.e., the history). $p_{\boldsymbol{\beta}}$ is a log-linear distribution over past documents:

$$p_{\boldsymbol{\beta}}(z_{ji} = k \mid x_j, x_k) = \frac{\exp(\boldsymbol{\beta}^\top \mathbf{f}(x_j, x_k))}{\sum_{k'} \exp(\boldsymbol{\beta}^\top \mathbf{f}(x_j, x'_k))} \tag{3}$$

The generative process for each document x_j is:

- For each word w_i in x_j:
  - Draw a past document x_k from a multinomial $\exp(\boldsymbol{\beta}^\top \mathbf{f}(x_j, x_k))/Z$
  - Draw w_i from $p(w_i \mid \psi_k, h_{ij})$, where ψ_k is a fixed language model for document x_k.

We can maximize this function with respect to the pairwise metadata features $\boldsymbol{\beta}$ using Expectation Maximization (EM) algorithm. In the E-step, we fix β and compute, for all j and k, the expected number of words in the present document x_j for which the inuence variable for a word token corresponded to the past

document x_k.

$$q_{jk} \leftarrow \sum_{i}^{N_j} \frac{p_\beta(z_{ji} = k \mid x_j, x_k) p_\theta(w = w_{ji} \mid k, h_{ji})}{p_\beta(z_{ji} = k' \mid x_j, x_k) p_\theta(w = w_{ji} \mid k', h_{ji})} \tag{4}$$

Then, in the M-step, we maximize the following objective:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\beta}) &= \sum_j \sum_k q_{jk} \log p(z_{ji} = k \mid x_j, x_k) \\ &= \sum_j \sum_k q_{jk} \left[\boldsymbol{\beta}^\top \mathbf{f}(x_j, x_k) - \log \sum_{k'} \exp(\boldsymbol{\beta}^\top \mathbf{f}(x_j, x_{k'})) \right]\end{aligned} \tag{5}$$

A standard convex optimizer can be used; we use the L-BFGS quasi-Newton algorithm (Liu and Nocedal, 1989). For computing the gradient, the partial derivative of this objective with respect to a single parameter β_m is the following (which is analogous to Berg-Kirkpatrick et al., 2010):

$$\begin{aligned}\frac{\partial \mathcal{L}(\boldsymbol{\beta})}{\partial \beta_m} &= \sum_j \sum_k q_{jk} \left[\frac{\partial}{\partial \beta_m} \boldsymbol{\beta}^\top \mathbf{f}(x_j, x_k) - \frac{\partial}{\partial \beta_m} \log \sum_{k'} \exp(\boldsymbol{\beta}^\top \mathbf{f}(x_j, x_{k'})) \right] \\ &= \sum_j \sum_k q_{jk} \left[f_m(x_j, x_k) - \sum_{k'} \frac{f_m(x_j, x_{k'}) \exp(\boldsymbol{\beta}^\top \mathbf{f}(x_j, x_{k'}))}{\sum_{k''} \exp(\boldsymbol{\beta}^\top \mathbf{f}(x_j, x_{k''}))} \right] \\ &= \sum_j \sum_k q_{jk} \left[f_m(x_j, x_k) - \sum_{k'} f_m(x_j, x_{k'}) p(z = k' \mid j) \right]\end{aligned} \tag{6}$$

6.2 Experiments

Data splits and features

In this preliminary experiment, we use the NBER dataset to model influences on papers in 2009 from the previous ten years (1999-2008). We pre-computed trigram language models using SRI Language Modeling Toolkit.[20] We have five pairwise features for each pair of document x_j and x_k[21]:

- $\mathbf{1}\{$document x_j and document x_k share co-authors.$\}$
- $\mathbf{1}\{$document x_j and document x_k have contact info addresses (zipcodes) > 0 kilometers apart.$\}$
- $\mathbf{1}\{$document x_j and document x_k have contact info addresses (zipcodes) > 100 kilometers apart.$\}$
- $\mathbf{1}\{$document x_j and document x_k have contact info addresses (zipcodes) > 1000 kilometers apart.$\}$
- Year difference between document x_j and document x_k.

Results

Table 8 shows feature coefficients learned by our model after 46 EM iterations (chosen to limit runtime). The results suggest that scientific papers are influenced by social, geographical, and temporal aspects. First, the highly positive weight for the shared-coauthorship feature indicates that if two papers share at least a co-author, the papers are more likely to be similar. Second, the year difference feature has a high negative weight. It indicates that more recent papers have a higher probability to influence current paper. Last, we can see that geographical distance also affects written scientific articles. Strong negative weights for increasing distances between contact info addresses show that the more distant two papers are, the less likely they are to influence each other.

[20] `http://www.speech.sri.com/projects/srilm/`

[21] Additional features can be incorporated easily to the model.

Feature	Weight
Shared-coauthorship	0.026
Year difference	-0.051
Distance > 0 km	-0.025
Distance > 100 km	-0.026
Distance > 1000 km	-0.015

Table 8: Feature weights learned by the influence language model.

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", and Ramage et al. (2010) rank universities based on scholarly output using topic models. Our models were able to discover interesting trends without utilizing the citation graph, but we can easily extend them to make use of it.

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. We experimented with more advanced regularization techniques, both for individual coefficients and pairs or groups of coefficients. However, we did not get consistent improvements, and the optimization method becomes considerably more complex.

Our latent variable logistic model is inspired by related work in sentiment analysis (Yessenalina et al., 2010). They made use of structural support vector machines to discover hidden explanations and only considered a subset of the sentences which best explain document sentiment when making a prediction. Similar methods based on Markov random fields have also been proposed (McDonald et al., 2007; Tackstrom and McDonald, 2011). Learning and inference in our summarization model are closely related to those of the hidden conditional random fields (Quattoni et al., 2007).

The influence language model is a combination of multiple ideas: featurized log-linear parameterizations in generative models (Berg-Kirkpatrick et al., 2010), Latent Dirichlet Allocation (Blei et al., 2003), and language model interpolation (Hsu, 2007).

8 Conclusions and Future Work

We presented a statistical approach to predicting a scientific community's response to an article and modeling scientific literature, based on its textual content. We showed how temporal and spatial trends can be discovered using methods based on generalized linear models. To improve the interpretability of the linear model, we developed a novel time series regularizer that encourages gradual changes across time steps and introduced a multiple output regression model with overlapping features. Our experiments showed that text

features significantly improve accuracy of predictions over baseline models, and we found that the feature weights reflect important trends in the literature. We also proposed a model which summarizes the document while making predictions, and a model for discovering influential factors in written scientific communication.

There are many possibilities that we leave for future work. For example, we can have a model which jointly explores spatial and temporal trends in a scientific literature or other related corpora. We can also develop a model in which there is a base coefficient for each feature, and other (temporal or regional) coefficients are drawn randomly with small perturbations from it. Extending the latent variable logistic regression model to handle continuous response is also an interesting direction to pursue, although inference in this model will not be straightforward. Additionally, we can make the latent variables represent other structures instead of document summary. For the influence language model, experiments with richer feature sets will reveal more interesting factors which affect written scientific communication.

Acknowledgements

We thank the National Bureau of Economic Research for providing the NBER dataset for this research, Sanjeev Khudanpur, Jason Matheny, Dewey Murdick, and Fallaw Sowell for helpful discussions, and three anonymous reviewers for comments on an earlier draft of this report. 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*.

G. Andrew and J. Gao. 2007. Scalable training of l1-regularized log-linear models. In *Proc. of ICML*.

T. Berg-Kirkpatrick, A. Bouchard-Côté, J. DeNero, and D. Klein. 2010. Painless unsupervised learning with features. In *Proc. of NAACL*.

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*.

D. M. Blei, A. Y. Ng, and M. I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

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.

B.-J. Hsu. 2007. Generalized linear interpolation of language models. In *Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding*.

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.

R. McDonald, K. Hannan, T. Neylon, M. Wells, and J. Reynar. 2007. Structured models for fine-to-coarse sentiment analysis. In *Proc. of ACL*.

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*.

A. Quattoni, S. Wang, L. Morency, M. Collins, and T. Darrell. 2007. Hidden-state conditional random fields. In *Proc. of IEEE PAMI*.

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*.

A. Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In *Proc. of EMNLP*.

O. Tackstrom and R. McDonald. 2011. Semi-supervised latent variable models for sentence-level sentiment analysis. In *Proc. of ACL*.

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*.

A. Yessenalina, Y. Yue, and C. Cardie. 2010. Multi-level structured models for document-level sentiment classification. In *Proc. of EMNLP*.

D. Yogatama, M. Heilman, B. O'Connor, C. Dyer, B. Routledge, and N. Smith. 2011. Predicting a scientific community's response to an article. In *Proc. of EMNLP*.