ANNUAL REVIEWS

Annual Review of Statistics and Its Application

Recent Advances in Text Analysis

Zheng Tracy Ke,[1] Pengsheng Ji,[2,*] Jiashun Jin,[3,*] and Wanshan Li[3,*]

[1]Department of Statistics, Harvard University, Cambridge, Massachusetts, USA; email: zke@fas.harvard.edu

[2]Department of Statistics, University of Georgia, Athens, Georgia, USA

[3]Department of Statistics & Data Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA

Annu. Rev. Stat. Appl. 2024. 11:347–72

First published as a Review in Advance on November 29, 2023

The *Annual Review of Statistics and Its Application* is online at statistics.annualreviews.org

https://doi.org/10.1146/annurev-statistics-040522-022138

*These authors contributed equally to this article

Keywords

BERT, journal ranking, knowledge graph, neural network, SCORE, Stigler's model, Topic-SCORE, topic weight

Abstract

Text analysis is an interesting research area in data science and has various applications, such as in artificial intelligence, biomedical research, and engineering. We review popular methods for text analysis, ranging from topic modeling to the recent neural language models. In particular, we review Topic-SCORE, a statistical approach to topic modeling, and discuss how to use it to analyze the Multi-Attribute Data Set on Statisticians (MADStat), a data set on statistical publications that we collected and cleaned. The application of Topic-SCORE and other methods to MADStat leads to interesting findings. For example, we identified 11 representative topics in statistics. For each journal, the evolution of topic weights over time can be visualized, and these results are used to analyze the trends in statistical research. In particular, we propose a new statistical model for ranking the citation impacts of 11 topics, and we also build a cross-topic citation graph to illustrate how research results on different topics spread to one another. The results on MADStat provide a data-driven picture of the statistical research from 1975 to 2015, from a text analysis perspective.

1. INTRODUCTION

Text analysis is an interdisciplinary research area in data science, computer science, and linguistics. It aims to use computers to process a large amount of natural language data and extract information or features. Research in text analysis and natural language processing (NLP) is especially useful for developing autopiloting cars, chatbots (e.g., ChatGPT, where GPT stands for generative pretrained transformer), and artificial intelligence in health care and biomedical engineering. Over the past decades, numerous methods have been proposed for text analysis. Two approaches are especially popular:

- Topic modeling. This approach has a strong statistical flavor. Given a large collection of text documents, this approach assumes that all these documents only discuss a few topics (e.g., finance, politics, sports). Each document discusses the topics with different weights, and given that a particular topic is being discussed, the words in the document are generated from a distribution specific to that topic.
- Neural network modeling. This rapidly developing approach models the generation of text documents via deep neural networks (DNNs) and trains the model with massive text corpora (e.g., English Wikipedia) and domain knowledge. The trained model will be used for different downstream tasks.

The neural network approach has proven effective in many NLP tasks (e.g., text classification and machine translation) and has gained immense popularity, particularly among technology titans such as Google and Meta. However, this approach is internally complex, expensive to train, and resource intensive. These factors substantially restrict the use of the neural network approach, especially for some common types of NLP users, such as social scientists who have only a few hundred text documents from a specific domain of interest. The topic modeling approach provides a valuable alternative and has the following benefits:

- Transparency and interpretability. Many users prefer an approach that (*a*) is not a black box but rather a more transparent step-by-step algorithm, (*b*) is easy to understand and tune (so users can modify it as needed), and (*c*) provides results (e.g., the extracted features) that are easy to interpret (Donoho & Jin 2015, Donoho 2017).
- Analytical accessibility. Topic modeling approaches are relatively simple and allow for delicate theoretical analysis. In particular, some of these methods enjoy statistical optimality. In comparison, neural network approaches are much harder to analyze and often have no theoretical guarantee.

Topic-SCORE (Ke & Wang 2022) is an especially interesting topic modeling method. It is fast and effective, and it enjoys nice theoretical properties. It is also a flexible idea and can adapt to several different settings. These characteristics make Topic-SCORE especially appealing when we analyze the Multi-Attribute Data Set on Statisticians (MADStat) data set (introduced below).

One goal of this article is to review popular topic modeling methods, from the rudimentary topic models of the 1990s to the more recent multigram topic models, with a focus on Topic-SCORE and related problems. In addition, we review neural network approaches. Large neural language modeling is a rapidly developing area, with new research emerging on a weekly basis, making it hard to conduct a comprehensive review. Since the focus of this article is on the topic modeling approach and the MADStat data set, we keep the review of neural network approaches relatively brief.

Another goal of this article is to analyze the MADStat data set using text analysis techniques. MADStat (Ji et al. 2022) is a large-scale, high-quality data set on statistical publications. We

collected and cleaned the data set, with substantial time and effort. It consists of the BibTeX (title, author, abstract, keywords, references) and citation information of 83,331 research papers published in 36 representative journals in statistics and related fields during 1975–2015. The data set contains detailed citation, BibTeX, and author information for each paper (i.e., paper-level data). It can be used to study research problems that cannot be addressed with other data resources that have only journal-level data or include no author information. Using MADStat, for instance, one can easily find the top 30 most-cited papers within our data range, whereas it is unclear how to do so using Google Scholar.

Text analysis on MADStat yields several findings. First, we use Topic-SCORE to identify 11 representative research topics in statistics, and visualize the evolution of the overall weight of statistical publications on each topic. Second, we extend Topic-SCORE to Topic-Ranking-SCORE (TR-SCORE), a method for ranking research topics by their citation exchanges, and we also build a knowledge graph to visualize how the research results on one topic disseminate to others. Third, we rank all 36 journals and suggest that *Annals of Statistics* (*AoS*), *Biometrika*, *Journal of the American Statistical Association* (*JASA*), and *Journal of the Royal Statistical Society Series B* (*JRSSB*) are the four most influential journals in statistics. Last, we find that the (per author) paper counts in statistics are steadily decreasing, suggesting that publishing in statistics has become more and more competitive. Our results provide an evidence-based picture of the whole statistics community and so can be viewed as a data-driven review of statistical research, from a text analysis perspective. The results may help administrators or committees with decision-making (e.g., promotions and awards) and help researchers make research plans and build networks. We use statistics as the object of study, but the same techniques can be used to study other fields (e.g., physics).

Obtaining a large-scale, high-quality data set such as MADStat is a challenging and time-consuming task. Particularly, many public data (e.g., Google Scholar) are quite noisy, and many online resources do not permit large-volume downloads. The data set must also be carefully cleaned; we did so through a combination of manual labor and custom-developed computer algorithms. **Supplemental Appendix A** provides a more detailed discussion on data collection and cleaning.

Supplemental Material >

Below, in Section 2, we review the recent advances on topic modeling. In Section 3, we briefly review neural network language models. In Section 4, we present some preliminary results about MADStat (paper counts, network centrality, journal ranking). In Section 5, we analyze the text data in MADStat using Topic-SCORE as the main tool. In Section 6, we propose TR-SCORE (an extension of Topic-SCORE) for ranking different topics, and we also construct a cross-topic knowledge graph. Section 7 contains a brief discussion.

2. TOPIC MODELS AND THEIR APPLICATIONS

The topic model is one of the most popular models in text analysis. Deerwester et al. (1990) proposed latent semantic indexing (LSI) as an ad hoc approach to word embedding. Later, Hofmann (1999) proposed a probabilistic model for LSI, which is now known as the topic model. Hofmann's topic model can be described as follows. Given n documents written with a vocabulary of p words, let $X \in \mathbb{R}^{p\times n}$ be the word-document-count matrix, where $X(j, i)$ is the count of the jth vocabulary word in document i. Write $X = [x_1, x_2, \ldots, x_n]$ so that $x_i \in \mathbb{R}^p$ is the vector of word counts for document i. Suppose document i has N_i words. For a weight vector (all entries are nonnegative with a unit sum) $\Omega_i \in \mathbb{R}^p$, we assume

$$x_i \sim \text{multinomial}(N_i, \Omega_i), \qquad 1 \le i \le n. \tag{1.}$$

Here, Ω_i is both the probability mass function (PMF) for x_i and the vector of population word frequency; in addition, we implicitly assume the words are drawn independently from the

vocabulary with replacement. Next, while there are a large number of documents, we assume there are only K topics discussed by these documents, and K is a relatively small integer. Fix $1 \le i \le n$ and consider document i. For a weight vector $w_i \in \mathbb{R}^K$ and PMFs $A_1, \ldots, A_K \in \mathbb{R}^p$, we assume ($a$) $w_i(k)$ is document i's weight on topic k, $1 \le k \le K$, and (b) given that the document is (purely) discussing topic k, the population word-frequency vector is A_k. Combining a, b, and Equation 1, it is reasonable to assume $\Omega_i = \sum_{k=1}^K w_i(k) A_k$. Write $\Omega = [\Omega_1, \Omega_2, \ldots, \Omega_n]$, $A = [A_1, \ldots, A_K]$, and $W = [w_1, w_2, \ldots, w_n]$. It follows that

$$\Omega = AW. \qquad 2.$$

We call A and W the topic matrix and the topic weight matrix, respectively.

From time to time, we may normalize X to the word-document-frequency matrix $D = [d_1, \ldots, d_n] \in \mathbb{R}^{p \times n}$, where $D(j, i) = X(j, i)/N_i$ (where N_i is the total number of words in document i, as above). The primary goal of topic modeling is to estimate (A, W) using X or D.

2.1. Anchor Words and Identifiability of the Topic Model

We call a word an anchor word of a given topic if its occurrence almost always indicates that the topic is being discussed. Consider the Associated Press (Harman 1993) data set, for example. A preprocessed version of the data set consists of 2,246 news articles discussing three topics: politics, finance, and crime (Ke & Wang 2022). In this example, we may think of "gunshot" and "Nasdaq" as anchor words for crime and finance, respectively. In the model in Equations 1 and 2, we can make the concept more rigorous: We call word j an anchor word of topic k if $A_k(j) \neq 0$ and $A_\ell(j) = 0$ for all $\ell \neq k$.

The notion of an anchor word is broadly useful. First, it can be used to resolve the identifiability issue of the topic model. Without any extra conditions, the model in Equations 1 and 2 is nonidentifiable [i.e., given an Ω, we may have multiple pairs of (A, W) satisfying $\Omega = AW$]. To make the model identifiable, we may assume $\text{rank}(W) = K$ and impose the anchor-word condition (which requires that each of the K topics has at least one anchor word). The anchor-word condition was first proposed by Arora et al. (2012) for topic models, and in turn was adapted from the separability condition (Donoho & Stodden 2003) for nonnegative matrix factorization (NMF). Second, anchor words are useful in methodological developments: Many topic modeling methods critically depend on the assumption that each topic has one or a few anchor words. For instance, Sections 2.2 and 2.3 provide descriptions of Topic-SCORE and anchor-word-searching methods. Last, but not least, a challenge in real applications is that both the number of topics K and the meanings of each estimated topic are unknown; we can tackle this problem with the (estimated) anchor words. Section 5 includes our analysis of the MADStat data, for example, where we use the estimated anchor words to decide K, interpret each estimated topic, and assign an appropriate label.

2.2. Topic-SCORE: A Spectral Approach to Estimating the Topic Matrix A

In Hofmann's topic model (Equations 1 and 2), we can view $D = AW + (D - AW) = \text{signal} + \text{noise}$, where (typically) $\text{rank}(AW) = K \ll \min\{n, p\}$. To estimate A in such a low-rank signal matrix plus noise scenario, it is preferable to employ a singular value decomposition (SVD) approach, as SVD is effective in both dimension reduction and noise reduction.

Topic-SCORE (Ke & Wang 2022) is an SVD approach to topic modeling, relying on two main ideas: SCORE normalization and use of a low-dimensional simplex structure in the spectral domain. In detail, Ke & Wang (2022) pointed out that a prominent feature of text data is the severe heterogeneity in word frequency: The chance of one word appearing in the documents may be hundreds of times larger than that of another. This heterogeneity poses great challenges for

textbook SVD approaches, so the vanilla SVD must be combined with proper normalizations. Ke & Wang (2022) proposed a pre-SVD approach where, for a diagonal matrix M they constructed, they mapped the data matrix D to $M^{-1/2}D$. Unfortunately, while the pre-SVD normalization may reduce the effects of severe heterogeneity to some extent, many of them persist. To overcome this challenge, Ke & Wang (2022) proposed a post-SVD normalization as follows. Let $\hat{\xi}_k$ be the kth left singular vector of $M^{-1/2}D$. They normalized $\hat{\xi}_2, \ldots, \hat{\xi}_K$ by dividing each of them by $\hat{\xi}_1$, where the division is element-wise division. This gives rises to a matrix $\hat{R} \in \mathbb{R}^{n,K-1}$, where $\hat{R}(i,k) = \hat{\xi}_{k+1}(i)/\hat{\xi}_1(i)$ [by Perron's theorem (Horn & Johnson 2013), all entries of $\hat{\xi}_1$ are positive under a mild condition]. Ke & Wang (2022) argued that, by combining the pre- and post-SVD normalizations, one can satisfactorily alleviate the effects of severe word-frequency heterogeneity. The post-SVD normalization was inspired by the SCORE normalization [proposed by Jin (2015) for analyzing network data with severe degree heterogeneity], hence the name Topic-SCORE.

Ke & Wang (2022) discovered a low-dimensional simplex $\mathcal{S}$ with K vertices as follows. For $1 \le i \le p$, let $\hat{r}_i'$ be the ith row of $\hat{R}$, and view each $\hat{r}_i$ as a point in $\mathbb{R}^{K-1}$. They pointed out that (*a*) when word i is an anchor word, then (up to small noise, with the same true for *b*) $\hat{r}_i$ falls on one of the vertices of $\mathcal{S}$, and (*b*) when word i is a nonanchor word, $\hat{r}_i$ is in the interior of $\mathcal{S}$.

This simplex structure reveals a direct relationship between $\hat{R}$ and quantity of interest A and gives rise to the Topic-SCORE approach as follows. Let $\hat{v}_1, \ldots, \hat{v}_K$ be the estimates of the vertices of $\mathcal{S}$. We can write each $\hat{r}_i$ uniquely as a convex linear combination of $\hat{v}_1, \ldots, \hat{v}_K$, with a barycentric coordinate vector $\hat{\pi}_i \in \mathbb{R}^K$. Topic-SCORE estimates A by $\hat{A} = M^{1/2}\mathrm{diag}(\hat{\xi}_1)[\hat{\pi}_1, \ldots, \hat{\pi}_p]'$ (subject to a column-wise renormalization), where $\mathrm{diag}(\hat{\xi}_1)$ is the diagonal matrix whose diagonal entries are from $\hat{\xi}_1$. In a noiseless case where $D = AW$, Ke & Wang (2022) showed that $\hat{A} = A$, so the approach is valid. An interesting problem here is how to use the rows of $\hat{R}$ to estimate the vertices of $\mathcal{S}$ (i.e., vertex hunting). This problem was studied in hyperspectral unmixing and archetypal analysis, which has many available algorithms. Ke & Wang (2022) recommended the sketched vertex search algorithm (Jin et al. 2023) for its superior numerical performance (for more discussion, see Ke & Jin 2023).

The major computational cost of Topic-SCORE comes from the SVD step, which can be executed relatively fast. For this reason, Topic-SCORE is fast and can easily handle large corpora. For example, it takes only a minute to process the MADStat corpus in Section 5. Topic-SCORE is also theoretically optimal in a wide parameter regime (Ke & Wang 2022).

2.3. The Anchor-Word-Searching Methods for Estimating A

Arora et al. (2012, 2013) proposed an anchor-word-searching approach that estimates A by finding anchor words from the word-word cooccurrence matrix $Q = DD'$. This method first normalizes each row of Q to have unit-ℓ^1-norm, with the resulting matrix denoted by $\bar{Q}$. It then applies a successive projection algorithm to rows of $\bar{Q}$ to get a subset $S \subset \{1, 2, \ldots, p\}$ containing exactly one estimated anchor word per topic. The method then estimates A either by a direct reconstruction or by minimizing some objective function (e.g., Kullback–Leibler divergence). Arora et al. (2012, 2013) were among the first to utilize the anchor-word condition for topic modeling and to provide explicit error rates. A challenge is that the rows of $\bar{Q}$ are in a very high-dimensional space. Similar to Topic-SCORE, this anchor-word-searching approach also relies on a K-vertex simplex, except for a major difference: This simplex is in $\mathbb{R}^p$, while the simplex in Section 2.2 is in $\mathbb{R}^{K-1}$ (e.g., in the abovementioned Associated Press data set, $K = 3$, but p is a few thousand). This gives Topic-SCORE an important edge (in both theory and computation) when it comes to vertex hunting and subsequent steps of estimating A. In particular, Topic-SCORE improves on the error rate of Arora et al. (2012, 2013).

Bing et al. (2020) proposed a different anchor-word-searching approach. Recall that $W \in \mathbb{R}^{K\times n}$ is the topic weight matrix (Equations 1 and 2). Letting $\zeta_k = \|W_k\|_2/\|W_k\|_1$, where W_k is the kth row of W, they assumed $W_k'W_\ell/\|W_k\|\|W_\ell\| < \zeta_k/\zeta_\ell \wedge \zeta_\ell/\zeta_k$, for $1 \le k \ne \ell \le K$. For the same $\bar{Q}$ as above, let S_i be the set of indices j such that $\bar{Q}(i, j)$ attains the maximum value of row i. Bing et al. (2020) proposed an approach and showed that if (*a*) the above assumption holds and (*b*) the model is noiseless (i.e., $D = AW$), then the approach can fully recover the set of anchor words from the index sets $S_1, S_2, \ldots, S_n$. Extending the idea to the real case (where $D = AW +$ noise), they obtained an estimate for the set of anchor words and then a procedure for estimating A.

2.4. Other Approaches for Estimating A: Expectation–Maximization Algorithm and Nonnegative Matrix Factorization Approaches

The expectation–maximization (EM) algorithm is a well-known approach to fitting latent variable models. It has been noted (e.g., Mei & Zhai 2001) that the model in Equations 1 and 2 is equivalent to a latent variable model, so we can estimate A using the EM algorithm. Such an approach is interesting but faces some challenges. First, it does not explicitly use the anchor-word condition, so the model being considered is in fact nonidentifiable (see Section 2.1). Also, since $\min\{n, p\}$ is typically large, the convergence of the EM algorithm remains unclear; even when the EM algorithm converges, the local minimum it converges to is not necessarily the targeted (A, W) (which is uniquely defined under a mild anchor-word condition; see Section 2.1).

Also, note that the model in Equations 1 and 2 implies $D = AW +$ noise, where (D, A, W) are all (entry-wise) nonnegative matrices; hence, the problem of estimating (A, W) can be recast as an NMF problem. There are many NMF algorithms (e.g., Gillis & Vavasis 2013) that have proven successful in applications such as image processing (Lee & Seung 1999), recommender systems, and bioinformatics. However, a direct use of them in topic modeling faces challenges. The noise in most NMF settings is additive and homoskedastic, but the noise matrix $D - \mathrm{E}[D]$ in the topic model is nonadditive and severely heteroskedastic, as indicated by the multinomial distribution. In the model in Equations 1 and 2, the variance of $D(j, i)$ is proportional to word j's frequency in document i. Because of severe word-frequency heterogeneity, the variances of $D(j, i)$ may have different magnitudes; hence, a direct application of NMF algorithms often yields nonoptimal error rates.

2.5. Estimating the Topic Weight Matrix W

In the model in Equations 1 and 2, $D = AW +$ noise, and both A and W are unknown. While most existing works focused on estimating A, W is also of interest (see, e.g., Section 5). To estimate W, a natural approach is to first obtain an estimate $\hat{A}$ for A, and then estimate W by fitting the model $D = \hat{A}W +$ noise. Recall that $W = [w_1, \ldots, w_n]$. Ke & Wang (2022) proposed a weighted least squares approach, where for each $1 \le i \le n$, w_i is estimated by $\hat{w}_i = \mathrm{argmin}_w \|\Theta(d_i - \hat{A}w)\|^2$, where $\Theta \in \mathbb{R}^{p\times p}$ is a diagonal weight matrix (because $w_i \in \mathbb{R}^K$ and K is typically small, this is a low-dimensional regression problem). To handle severe word-frequency heterogeneity, Ke & Wang (2022) suggested $\Theta = M^{-\frac{1}{2}}$, with the same M as in Section 2.2. For our study on the MADStat data in Section 5, we find that taking $\Theta = I_p$ also works fine, if a ridge regularization is added. Noting that the word count vector x_i is distributed as multinomial(N_i, Aw_i), we can also estimate w_i by some classical approaches, such as maximum likelihood estimation, where we replace A by $\hat{A}$ in the likelihood.

The above raises a question: Since $D = AW +$ noise, can we first estimate W and then use $\hat{W}$ to estimate A? There are two concerns. First, in some settings, the optimal rate for estimating A is faster than that for estimating W (see Section 2.6). Therefore, if we first estimate W and

then use $\hat{W}$ to estimate A, then we may achieve the optimal rate in estimating W but likely will not in estimating A. If we first estimate A and then use $\hat{A}$ to estimate W, we have optimal rates in estimating both. Second, many approaches for estimating A rely on the assumption that each topic has some anchor words (see Sections 2.2 and 2.3). If we extend them to estimate W, we need to similarly assume that each topic has some pure documents [document i is pure if $w_i(k) = 1$ and $w_i(\ell) = 0$ for $\ell \neq k$]. However, in many applications, it is more reasonable to assume the existence of anchor words than the existence of pure documents (especially when documents are long). Therefore, though the roles of A and W may appear symmetrical to one other, they are not symmetrical in reality.

2.6. The Optimal Rates for Estimating (A, W)

For simplicity, as is done in many theoretical works on topic modeling, we assume $N_1 = \ldots N_n = N$—that is, documents have the same length. We may have either a long-document (LD) case where $N/p \geq O(1)$ or a short-document (SD) case where $N/p = o(1)$ (where p is the size of the vocabulary).

Consider the rate for estimating A. For any estimate $\hat{A}$, we measure the loss by the ℓ^1-error: $\mathcal{L}(\hat{A}, A) = \sum_{k=1}^{K} \|\hat{A}_k - A_k\|_1$ (subject to a permutation in the K columns of $\hat{A}$). The minimax rate is defined as $R_n = \inf_{\hat{A}} \sup_A \mathrm{E}\mathcal{L}(\hat{A}, A)$. In the LD case, when K is finite, $R_n \asymp \sqrt{p/(Nn)}$ up to a multi-log (p) factor [e.g., $\sqrt{\log(p)}$] (Ke & Wang 2022); when K grows with (n, p), $R_n \asymp K\sqrt{Kp/(Nn)}$, also up to a multi-log (p) factor (Bing et al. 2020). In the SD case, the optimal rate is unclear. Some minimax upper bounds were derived (Arora et al. 2012, Ke & Wang 2022), but they do not yet match the minimax lower bound. The difficulty of the SD case is that the majority of words have a zero count in most documents, which poses challenges in theoretical analysis.

Consider the rate for estimating W. Similarly, for any estimate $\hat{W}$, we measure the loss by $\mathcal{L}(\hat{W}, W) = (1/n)\sum_{i=1}^{n} \|\hat{w}_i - w_i\|_1$ (up to a permutation in the K rows in $\hat{W}$) and define the minimax rate as $R_n = \inf_{\hat{W}} \sup_W \mathrm{E}\mathcal{L}(\hat{W}, W)$. Wu et al. (2023) showed that $R_n \asymp \sqrt{K/N}$. The minimax rate is flat in n; this is not surprising, because the number of free parameters in W is proportional to n.

2.7. Estimating the Number of Topics K

Almost all topic learning algorithms assume K as known a priori, but K is rarely known in real applications. How to estimate K is therefore a fundamental problem.

To estimate K in such a low-rank matrix plus noise situation, a standard approach is to use the scree plot: For a threshold t, we estimate K as the number of singular values of X that exceed t. Ke & Wang (2022) showed that this estimator is consistent, under some regularity conditions. This method does not need topic model fitting and is fast and easy to use, but how to select a data-driven t is an open question. Alternatively, one may select K using the Bayesian information criterion (BIC) or other information criteria: For each candidate of K, we obtain $(\hat{A}, \hat{W})$ by applying a topic learning algorithm, and we estimate K by the candidate that minimizes the BIC. Also, alternatively, one may use cross-validation (CV) approaches by estimating a topic model for each candidate K and each training-validation split. A commonly used validation loss is the perplexity. It measures the predictive power of a trained language model on the held-out test set. To use perplexity, we usually assume w_i are independent and identically distributed, so the approach is more appropriate for the Bayesian version of the topic model (introduced in Section 2.9); we can also use a full Bayesian approach by imposing a prior on K and selecting $\hat{K}$ to minimize the marginal likelihood (Taddy 2012). In both the BIC and CV approaches, we need to fit the topic model many times, so the computational cost is high.

Simulation studies have noted that (*a*) none of these methods is uniformly better than others, and which method is the best depends on the data set, and (*b*) the popular perplexity approach often overestimates K. For these reasons, in real applications, whenever some inside information is available, we hope to use it to help determine K. For example, in the study of MADStat (see Section 5), we investigate the estimated anchor words by Topic-SCORE for different Ks and use our knowledge of the statistical community to choose the K with the most reasonable results. In some applications, the best K depends on the perspectives of the users, and even experts may differ in their opinions. In such a case, we may want to consider several different Ks. Such flexibility may be helpful.

2.8. Global Testing Associated with Topic Models

The problem of global testing is closely related to the problem of estimating K. The goal is to test $H_0 : K = 1$ versus $H_1 : K > 1$. Global testing is a fundamental problem: If no method can reliably differentiate between $K = 1$ and $K > 1$, it is impossible to estimate K or estimate the matrices (A, W) in the model in Equations 1 and 2.

Recall that $x_i \sim \text{multinomial}(N_i, Aw_i)$, $1 \le i \le n$, in the model in Equations 1 and 2. Cai et al. (2023) proposed a test statistic ψ_n called DELVE (debiased and length-assisted variability estimator). They showed that when $K = 1$, although the model has many unknown parameters, $\psi_n \to N(0, 1)$, and the limiting distribution does not depend on unknown parameters. This result is practically useful. For example, we can use it to compute an approximate p-value and use the p-value to measure the research diversity of different authors in the MADStat data set; Ji et al. (2022, section 3.3) show a similar use of global testing in the network setting (Jin et al. 2018, 2021).

Denote by λ_2 the second largest (in magnitude) eigenvalue of $\Sigma_A = A'[\text{diag}(A\mathbf{1}_K)]^{-1}A$. Similar to Section 2.6, we assume $N_i = N$ for $1 \le i \le N$. Consider the DELVE test that rejects H_0 if $|\psi_n| \ge t$, for a threshold $t > 0$. Cai et al. (2023) showed that this test achieves a sharp phase transition as follows. If $|\lambda_2|/\sqrt{p/(N^2 n)} \to \infty$, for an appropriate t, the sum of the type I and type II errors of the DELVE test converges to 0 as $p \to \infty$. If $|\lambda_2|/\sqrt{p/(N^2 n)} \to 0$, for any test, the sum of the type I and type II errors converges to 1. Compared with earlier works (e.g., Bing et al. 2020, Ke & Wang 2022), such a result is more satisfying. In earlier works, we usually assume all eigenvalues of Σ_A are on the order of $O(1)$. Here, we may have $\lambda_2 = o(1)$, especially when $p \ll N^2 n$.

2.9. The Latent Dirichlet Topic Model and Its Estimation

The latent Dirichlet allocation (LDA) model by Blei et al. (2003) is one of the most popular topic models, and it can be viewed as a Bayesian version of the Hofmann topic model. In the LDA model, we start with the model in Equations 1 and 2 and further assume that the topic weight vectors $w_1, w_2, \ldots, w_n$ are drawn, independent and identically distributed, from a Dirichlet distribution with parameters $\alpha = (\alpha_1, \ldots, \alpha_K)$, where $\alpha_k \ge 0$ and $\sum_{k=1}^{K} \alpha_k = 1$. The LDA model has parameters (A, α) and treats w_is as latent variables. In such a setting, (A, α) are estimated by a variational EM algorithm, and the posterior of w_is can be obtained using Markov chain Monte Carlo (MCMC). This is essentially the approach proposed by Blei et al. (2003). Compared with the model in Equations 1 and 2, LDA does not assume any structure on the topic matrix A. If our goal is to estimate A, and if the anchor-word condition is satisfied, all of the methods in Sections 2.2 and 2.3 are still applicable. In particular, compared with the variational EM approach of Blei et al. (2003), Topic-SCORE in Section 2.2 not only is faster but also provides desired theoretical guarantees (Ke & Wang 2022). On the other hand, LDA puts a Dirichlet prior on the topic weights w_i. This allows us to learn the posterior distribution of w and may provide additional insights. Recall that in Section 2.5, we proposed a regression approach to estimating W (without any priors on W). The regression approach is still useful for the LDA model (e.g., we can use this

method to estimate the parameter α in the LDA model and plug the estimated value in to the variational EM algorithm).

2.10. The *m*-Gram Topic Models

The Hofmann topic model and the LDA are so-called bag-of-word or unigram models, as they model only the counts of single words, neglecting word orders and word context. There are several ideas about extending these models to incorporate word orders and word context.

One idea is simply to expand the vocabulary to include phrases. For example, we may include all possible m-grams in the vocabulary (an m-gram is a sequence of m words). Unfortunately, even for a small m, the size of this vocabulary is too large, making topic estimation practically infeasible. To address the issue, we may include only a subset of carefully selected m-grams. For example, we may exclude low-frequency phrases or apply a phrase retrieval algorithm (Fagan 1988). Once the vocabulary is determined, we treat each item in the vocabulary as a word and model the words by Equations 1 and 2 as above; the resulting model is still a unigram model in flavor.

Another idea is the bigram topic model (Wallach 2006). For each $1 \le i \le n$, document i is modeled as an ordered sequence of words satisfying a Markov chain with a transition matrix $M_i \in \mathbb{R}^{p\times p}$ (p is the vocabulary size), where $M_i(j, \ell)$ is the probability of drawing word ℓ when the word immediately preceding it is word j. For transition matrices $A_1, A_2, \ldots, A_K \in \mathbb{R}^{p\times p}$, we have $M_i = \sum_{k=1}^{K} w_i(k)A_k$, where each A_k is treated as a topic and $w_i \in \mathbb{R}^K$ is the topic weight vector as above. Wallach (2006) proposed a Gibbs EM algorithm for estimating the parameters and showed that, compared with the unigram topic model, this bigram model led to better predictive performance and more meaningful topics on two real-world data sets.

2.11. Supervised Topic Models

In many applications, we observe not only text documents but also some response variables associated with documents. For example, many online customer reviews contain numeric ratings; we treat a review as a text document and the corresponding rating as the response. We would like to build a joint model for text and response to help predict future ratings.

The model by Ke et al. (2019) is a supervised topic model of this kind. The authors studied the problem of how to use news articles to improve financial models. They focused on the news articles in *Dow Jones Newswires*. These articles are tagged with the identifier of a firm (the study excluded articles tagged with multiple firms). They model the news article with the model in Equations 1 and 2 and $K = 2$ (so there are only two topics), where the two topics are positive sentiment and negative sentiment, respectively. In such a simple case, for any $1 \le i \le n$, let $w_i = (a_i, 1 - a_i)'$ be the topic weight of document i as above (w_i captures the sentiment level of article i). Meanwhile, let y_i be the stock return of the firm being tagged with document i. They assume that $\mathrm{P}(y_i > 0) = f(a_i)$ for an (unknown) function f that is monotone increasing. This model jointly models text and return data, allowing for a better estimation of w_i (which, in turn, may lead to a better prediction of stock returns). Compared with other approaches that also estimate news sentiment and use it to predict returns, this approach improves substantially on real-data performance. Moreover, McAuliffe & Blei (2007) discuss other supervised topic models with a similar flavor.

3. DEEP NEURAL NETWORK APPROACHES TO NATURAL LANGUAGE PROCESSING

The DNN approaches to natural language processing (DNN-NLP) have become very popular recently, with successes observed in a variety of NLP tasks, such as text classification, question answering, and machine translation, among others (Otter et al. 2020).

In statistics, a "model" is a generative model with some unknown parameters we need to estimate. In DNN-NLP, researchers use the term slightly differently: A neural language model usually refers to a pretrained neural network equipped with estimated parameters. A neural language model usually consists of three components:

- A neural network architecture. This is the core of a neural language model. It specifies how an input text is processed to generate the desirable output. The encoder–decoder structure is commonly used: The encoder is a neural network that maps the input text into a numeric vector (a.k.a. the encoder state), and the decoder converts the encoder state to the targeted output (e.g., a variable-length sequence of tokens). Many neural network models were inspired by new architectures proposed in the literature.
- The NLP tasks used to train the neural networks. A neural language model usually targets one specific task (e.g., machine translation) or several specific NLP tasks [e.g., the BERT (bidirectional encoder representations from transformers) model (Devlin et al. 2018) outputs document embeddings, which can be used in various downstream tasks]. In either case, pretraining the neural networks (i.e., estimating the parameters) must use specific NLP tasks to define the objective function. Hence, the same architecture may lead to different neural language models if they are pretrained using different NLP tasks.
- The text corpora and domain knowledge used in training. Even with the same architecture and the same NLP tasks in training, the resulting neural language model still varies with the training corpora. One strategy is selecting training corpora to obtain a domain-specific language model. For example, BERT has variants such as BioBERT (Lee et al. 2020), trained using publications in biomedicine. Besides domain-specific corpora, other knowledge such as a domain-specific vocabulary can be employed.

The research on DNN-NLP has multiple goals, including but not limited to (*a*) prediction of the next word given the previous words in a sentence [e.g., the GPT family (Radford et al. 2018)], (*b*) extraction of numeric features from text [e.g., the BERT family (Devlin et al. 2018)], and (*c*) modeling the (syntactic and semantic) relationships of words [e.g., word2vec (Mikolov et al. 2013)]. DNN-NLP is a fast-developing area, which is hard to review comprehensively (especially as our focus is on the topic modeling approaches and the MADStat data set). For these reasons, we select a few interesting topics in DNN-NLP to review, focusing on (*a*) popular DNN architectures for NLP and (*b*) BERT, a powerful feature extraction tool developed by Google, Inc. We also discuss word embedding and how to apply a neural language model (e.g., BERT) to a text corpus in our own research (see Remarks 1 and 2).

3.1. Commonly Used Neural Network Architectures

Some well-known network architectures for NLP include convolutional neural networks (CNNs), recursive neural networks (RNNs), and transformers. CNNs and RNNs are more traditional, and transformers have become very popular in recent years.

CNNs use structural layers (e.g., convolutional layers and pooling layers) to capture the spatial patterns in the input and are extensively used in signal (speech, image, video) processing. In processing a text document, sometimes it is not important whether certain words appear, but rather whether or not they appear in particular localities. Hence, CNNs are also useful for NLP tasks such as sentence modeling (Kalchbrenner et al. 2014) and sentiment analysis (Dos Santos & Gatti 2014).

RNNs are especially useful for sequence data with variable lengths, making them suitable for text analysis. Long short-term memory networks (LSTMs) (Hochreiter & Schmidhuber 1997) are the most popular variant of RNNs. In vanilla RNNs, information may be diluted with

successive iterations, preventing the model from remembering important information from the distant past. LSTMs add neurons (called gates) to retain, forget, or expose specific information, so they can better capture the dependence between two far-apart words in the sequence. The standard LSTMs are unidirectional (i.e., text is processed from left to right). It is preferable to process text bidirectionally, as a word may depend on the words behind it. The bidirectional LSTMs combine outputs from left-to-right layers and right-to-left layers.

Transformers (Vaswani et al. 2017) are a type of architecture based on the attention mechanism (Bahdanau et al. 2014). In a traditional encoder–decoder pair, the encoder maps the input sequence into a fixed-length vector, and the decoder has access to this vector only. The attention mechanism allows the encoder to pass all the hidden states (not just the final encoded vector) to the decoder, along with annotation vectors and attention weights to tell the decoder which part of information to pay attention to. The attention mechanism was shown to be much more effective than RNNs in processing long documents. Vaswani et al. (2017) proposed a special architecture called a transformer that uses self-attention within the encoder and decoder separately and cross-attention between them. The transformer has become the most popular architecture in NLP. For example, the encoder part of the transformer is the building block of models like BERT (see below), and the decoder part of the transformer is the building block of models like GPT (Radford et al. 2018) for text generation.

3.2. BERT

BERT is a state-of-the-art language model developed by Google AI Language (Devlin et al. 2018), which provides a numerical representation for each sentence. As mentioned above, a neural language model consists of three components: architecture, pretraining tasks, and training corpora. For architecture, BERT uses the transformer encoder with bidirectional self-attention. For training corpora, BERT uses BooksCorpus (800 million words) (Zhu et al. 2015) and English Wikipedia (2.5 billion words). The main innovation of BERT is in the pretraining tasks it used: BERT was pretrained using two tasks, masked language modeling and next sentence prediction. In masked language modeling, some tokens of the input sequence are randomly masked, and the objective is to predict those masked tokens from their left and right contexts. In next sentence prediction, the inputs are two sentences A and B from a corpus, and the objective is to determine whether B is the next sentence of A. These tasks do not require manual labeling of text.

BERT has been applied to different downstream NLP tasks, with superior performance. Numerous language models based on BERT have been created, such as modifications of the architecture (e.g., ALBERT and DistillBERT) and pretraining tasks (e.g., RoBERTa and ELECTRA), adaptation to other languages (e.g., XLM and ERNIE), and inclusion of domain-specific corpora (e.g., BioBERT and UmlsBERT). A comprehensive survey is provided by Rahali & Akhloufi (2023).

Remark 1. Another major goal of NLP is to learn the syntactic and semantic relationships between words. To do so, a standard approach is word embedding (i.e., finding vector representations of words). Despite the fact that word embedding is frequently used in neural language models (often as the first layer), its primary purpose is to understand or mimic various syntactic and semantic regularities in natural languages. A frequently mentioned example is that vector("king") − vector("man") + vector("woman") ≈ vector("queen"). Word2vec (Mikolov et al. 2013) is a popular word embedding model. It was trained using a Google News corpus, and its performance was tested on a semantic–syntactic relationship question set manually created by the authors.

Remark 2. Many modern DNN-NLP tools (such as BERT) are owned by high-tech companies. They were trained with a huge amount of data and effort, and many parts of them are not publicly available. A typical NLP user has his/her own (domain-specific) text corpus (1,000 to 10,000 documents), which

is not large enough to retrain BERT (say). To help these users to apply modern DNN-NLP tools, there are two approaches: transfer learning and fine-tuning. In the first approach, the user inputs his/her own documents to BERT (say) and obtains an embedded vector for each document. The embedded vectors can then be used as features for downstream analysis. In the second approach, a user may alter the parameters of the pretrained model. By adding additional layers to the neural networks, one can convert the output of a pretrained neural language model to the targeted output of a downstream task (e.g., document classification). Next, all the parameters—those in the pretrained model and those for the added layers—are updated together (this can be done by running stochastic gradient descents starting from parameters of the pretrained model).

4. MADSTAT BASICS: PAPER COUNTS, JOURNAL RANKING, AND NETWORK CENTRALITY

Supplemental Material >

MADStat contains the BibTeX (e.g., author, title, abstract, journal, year, references) and citation information of 83,331 papers from 47,311 authors, spanning 41 years (1975–2015). We collected and cleaned the data with substantial time and effort and have made them publicly available (the links to download the data can be found in Ji et al. 2022). In the **Supplemental Appendix**, we present (*a*) details on data collection and cleaning, (*b*) the list of the 36 journals and their abbreviations, and (*c*) supplementary results of the text analysis conducted in this article (such as selection of K for Topic-SCORE). In this section, we discuss some basic findings from the data set, including paper counts, network centrality, and journal ranking.

4.1. Paper Counts

The paper counts provide valuable information for studying how the productivity of statisticians evolves over time. **Figure 1*a*** shows two curves with the number of papers per year and the number of active authors per year, respectively (an author is active in a given year if he/she publishes at least one paper in that year). In both curves, we notice a sharp increase near 2005–2006, possibly because several new journals [*Annals of Applied Statistics*, *Bayesian Analysis*, and the *Electronic Journal of Statistics* (*EJS*)] were launched between 2006 and 2008 (see **Supplemental Table 1**). **Figure 1*b*** presents the yearly paper counts, defined as the average number of papers per active author. We consider both standard count and fractional count, where, for an m-author paper, each author is counted as having published 1 and $1/m$ papers, respectively. In the standard count, the yearly paper counts increase between 1975 and 2009, from about 1.2 papers per author to about 1.4 papers per author, and decrease after 2009, to about 1.3 papers per author in 2015. In the fractional count,

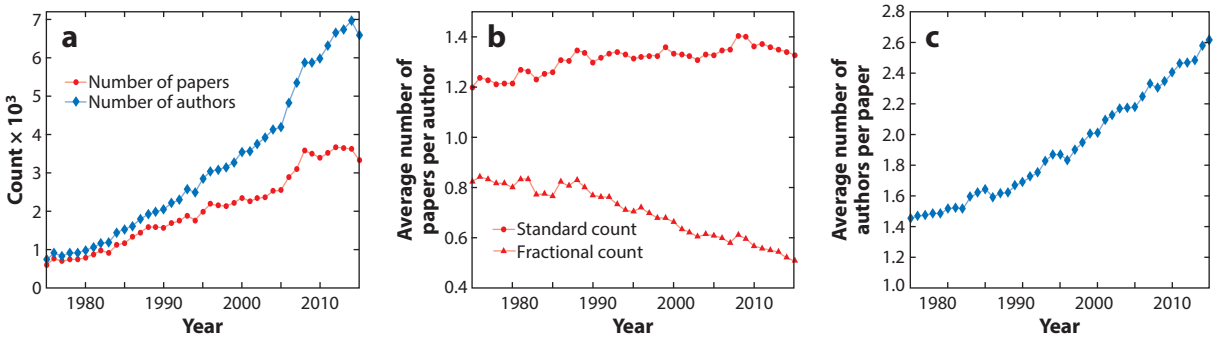


Figure 1

Paper counts and author counts in MADStat. (*a*) Number of papers and number of active authors per year. (*b*) Average number of papers per active author per year. (*c*) Average number of authors per paper. In counting the average number of papers per author, each coauthor of an m-author paper is counted as publishing 1 paper in the standard count and $1/m$ paper in the fractional count.

Table 1 The top 10 authors ordered by the number of coauthors, citers, and citations[a]

Author name	Coauthors	Author name	Citers	Author name	Citations
Raymond Carroll	234	Donald B. Rubin	5,337	Peter Hall	6,847
Peter Hall	222	Nan Laird	5,079	Donald B. Rubin	6,825
N. Balakrishnan	186	Bradley Efron	4,500	Jianqing Fan	5,726
Jeremy Taylor	159	Robert Tibshirani	4,076	Robert Tibshirani	5,074
Joseph Ibrahim	158	Peter Hall	3,789	Nan Laird	5,040
Geert Molenberghs	146	Arthur P. Dempster	3,406	Bradley Efron	4,589
James S. Marron	130	Scott Zeger	3,311	Raymond Carroll	4,415
Malay Ghosh	119	Kung Yee Liang	3,231	Scott Zeger	3,802
Emmanuel Lesaffre	119	Trevor Hastie	3,174	Trevor Hastie	3,582
Xiaohua Zhou	119	Raymond Carroll	3,110	Kung Yee Liang	3,366

[a]We count only coauthors and citations within the range of MADStat (the Multi-Attribute Data Set on Statisticians).

the yearly paper counts always decrease, from about 0.85 papers per author in 1975 to about 0.5 papers per author in 2015. The explanation is that the average number of authors per paper has been steadily increasing over the years. **Figure 1*c*** presents the average number of authors per paper; the curve is steadily increasing.

Supplemental Material >

The above counts are further explained in **Supplemental Figure 1**, in which (*a*) the paper count each year is partitioned into the counts of m-author papers for different m and (*b*) the author count each year is partitioned into the counts of k–year–senior author for different k. The results show some interesting patterns, and we refer the reader to **Supplemental Appendix D** for details.

4.2. Network Centrality

Network centrality metrics (e.g., a measure of the most-collaborative authors) provides information on leadership and trends in statistical research. **Table 1** presents the top 10 authors who have the most coauthors, the most citers (a citer for any given author is any other author who has cited this author), and the most citations. **Supplemental Table 2** (see **Supplemental Appendix E**) presents the top 10 most-cited papers. Note that the numbers of coauthors, citers, and citations here are all counted using only the papers in our data range, so there may be some biases in our ranking. For example, in **Supplemental Table 2**, if we instead use the citation counts by Google Scholar on December 31, 2022, then the papers by Benjamini & Hochberg (1995) on false discovery rates, Donoho & Johnstone (1994) on wavelets, and Efron et al. (2004) on least angle regression will receive better rankings, as these papers have many citations from papers outside our data range. Despite this, our approach is still valuable. For example, using our data, we can provide the ranking (e.g., by number of citations) for any author or any paper in our data set, but how to do this using Google Scholar is unclear: We need to build a large database for the citation relationships between many authors and papers and spend substantial time cleaning such citation data. Compared with Google Scholar, our citation data are of higher quality, so our results on network centrality shed new light that Google Scholar cannot provide.

4.3. Citation Patterns and the Sleeping Beauties

Identification of representative citation patterns is an interesting problem, as it helps distinguish short-term citation effects from long-lasting citation effects. By a careful study of the yearly citation curves of individual papers, we identify four representative citation patterns: sleeping beauty, transient, steadily increasing, and sudden fame. Sleeping beauty refers to the papers that receive low citations within a few years after publication but become frequently cited after a certain point

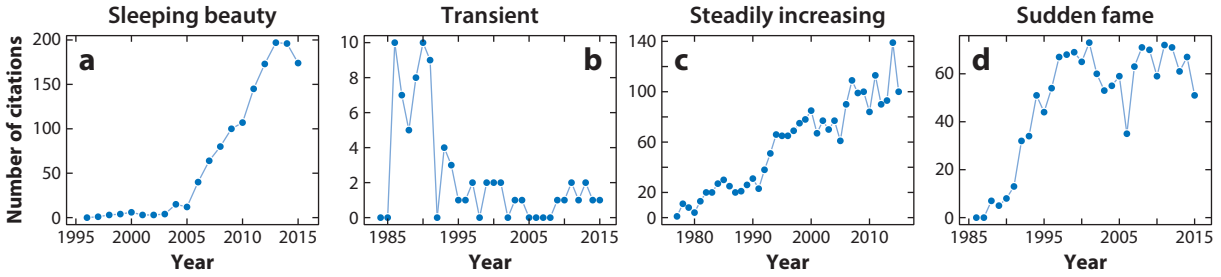


Figure 2

Yearly citation curves for four representative papers: (*a*) Tibshirani (1996) on the lasso (least absolute shrinkage and selection operator), (*b*) a representative paper with the transient pattern, (*c*) Dempster et al. (1977) on the expectation–maximization algorithm, and (*d*) Liang & Zeger (1986) on the generalized linear model.

(a.k.a. waking up). Representative papers include those by Tibshirani (1996) on the lasso (least absolute shrinkage and selection operator) and Benjamini & Hochberg (1995). Transient refers to the papers that receive a good number of citations for a few years shortly after publication, but then their citations drop sharply and remain low for years. Steadily increasing refers to those papers whose citations have been increasing at a modest rate for many years, with a large number of citations over a relatively long time period. A representative paper is that by Dempster et al. (1977) on the EM algorithm. Sudden fame refers to papers that receive a large number of citations shortly after publication and the citations remain high for many years. Representative papers include those by Liang & Zeger (1986) on longitudinal data, Gelfand & Smith (1990) on marginal densities, and Efron et al. (2004) (**Figure 2**).

Supplemental Material >

The sleeping beauty pattern is especially interesting. To identify the sleeping beauties in our data range, we use the metric suggested by Ke et al. (2015). It outputs a measure B_i for each paper i (the details are in the **Supplemental Appendix**); the larger B_i is, the more likely that a given paper is a sleeping beauty. We select the 300 papers with the largest maximum number of yearly citations and arrange them in descending order of B_i. **Supplemental Table 3** and **Supplemental Figure 2** show the papers with largest B_i (e.g., Azzalini 1985, Hubert & Arabie 1985, Tibshirani 1996).

4.4. Journal Ranking

Journal ranking has been widely used in appointing to academic positions, awarding research grants, and ranking universities and departments. A common approach is the Impact Factor (IF), but IF is known to have some issues (Varin et al. 2016). We instead use Stigler's model (Stigler 1994) for journal ranking: Given N journals, let $\mu_1, \ldots, \mu_N \in \mathbb{R}$ be their export scores, and for two papers i and j published in journals ℓ and m, respectively, let C_{ij} be the indicator of a citation from i to j. We assume $\mathrm{P}(C_{ij} = 1 | C_{ij} + C_{ji} = 1) = \exp(\mu_m - \mu_\ell)/[1 + \exp(\mu_m - \mu_\ell)]$. We fit this model using the quasi-likelihood approach of Varin et al. (2016). For comparison, we also consider the PageRank approach, with the same tuning parameter α as suggested by Varin et al. (2016). Among the 36 journals (see **Supplemental Table 1** for a full list), there are relatively few citation exchanges between the 3 journals focusing on probability and the other 33 journals, so we exclude these 3 probability journals. For each journal pair, we count the citations between them using a 10-year window. For instance, if 2014 is the current year, then we count one citation from journal i to journal j if and only if a paper published in journal i in 2014 has cited a paper published in journal j between 2005 and 2014. This gives rise to a 33×33 between-journal citation matrix for 2014. Last, we take the sum of the two matrices for 2014 and 2015 to improve the stability and

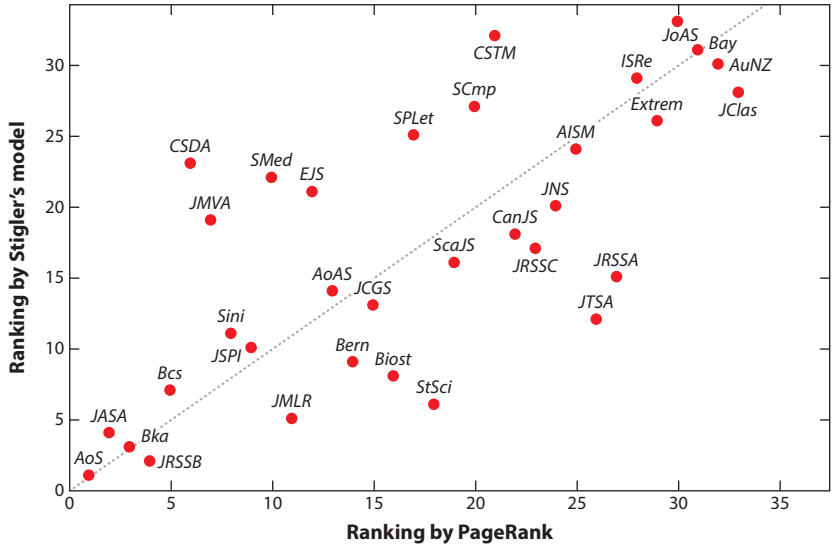


Figure 3

Journal ranking by PageRank and Stigler's model. Each point is a journal, with its abbreviation as the text label (for the full journal names, see **Supplemental Table 1**).

Supplemental Material >

reliability of results. This is the final data matrix fed into journal ranking. The results are shown in **Figure 3**.

Both approaches rank *AoS*, *Biometrika*, *JASA*, and *JRSSB* as the top four. In particular, both approaches rank *AoS* as number one and *Biometrika* as number three; PageRank ranks *JASA* as number two, and the Stigler approach ranks *JRSSB* as number two. The rankings of the two methods are quite consistent. A few exceptions are *Computational Statistics & Data Analysis* (*CSDA*), *EJS*, *Journal of Multivariate Analysis* (*JMVA*), *Journal of the Royal Statistical Society Series A* (*JRSSA*), *Journal of Time Series Analysis* (*JTSA*), and *Statistics in Medicine* (*SMed*). We notice that PageRank weighs each citation equally, while the Stigler model gives citations from higher-ranked journals greater weight than those from lower-ranked journals (Varin et al. 2016). The results of PageRank are fairly close to those of ranking by citation numbers, but the results from the Stigler approach may be significantly different. A closer look at the citation counts reveals that a large proportion of citations of *SMed*, *CSDA*, *JMVA*, and *EJS* are self-citations, and after these self-citations are excluded, most citations to these journals are from journals with relatively low rankings. That explains why these journals are ranked relatively high by PageRank but relatively low by Stigler's model. Also, while neither *JTSA* nor *JRSSA* has a large number of citations, most of their citations come from journals with high rankings; consequently, the two journals are ranked much higher by Stigler's model than by PageRank.

5. APPLICATION OF TOPIC-SCORE TO THE MADSTAT DATA SET

In this section, we apply Topic-SCORE (see Section 2.2) to analyze the abstracts in MADStat. We use all paper abstracts for the time period 1990–2015 in 33 journals, excluding the 3 probability journals (for the full journal list, see **Supplemental Table 1**), since the topics in these journals are very different from those in the other 33 journals. This gives a total of 63,187 abstracts. We then perform a word screening by removing stop words and infrequent words, which gives rise to a vocabulary of 2,106 words. Finally, we compute the length of each abstract by the number of

Supplemental Material >

words (a word not in the abovementioned vocabulary is not counted) and remove approximately the shortest 10% of abstracts. We have 56,500 remaining abstracts. The preprocessing details are presented in **Supplemental Appendix G**. The final data matrix is $X = [x_1, \ldots, x_n] \in \mathbb{R}^{p\times n}$, with $(p, n) = (2, 106, 56, 500)$; as in Section 2, $x_i \in \mathbb{R}^p$ contains the word counts of the ith paper abstract.

5.1. Anchor Words and the 11 Identified Topics

To apply Topic-SCORE, we need to decide the number of topics. This is a hard problem (see Section 2.7), and we tackle it by combining the scree plot, substantial manual effort, and our knowledge of the statistical community (see **Supplemental Appendix H**). We find that $K = 11$ is the most reasonable choice.

Since $K = 11$, there are 11 research topics identified by Topic-SCORE. To interpret and label these topics, we introduce a rule for selecting representative words and papers for each topic. The anchor words (see Section 2.1) appear only in one topic. For example, "lasso" and "prior" may be anchor words for the topics of "variable selection" and "Bayes," respectively. Given $\hat{A}$, define the topic loading vector $a_j \in \mathbb{R}^K$ for each word j by $a_j(k) = \hat{A}_k(j)/[\sum_{\ell=1}^K \hat{A}_\ell(j)]$, $1 \le k \le K$. Note that $0 \le a_j(k) \le 1$, and, in theory, $a_j(k) = 1$ if and only if word j is an anchor word of topic k. Fix $1 \le k \le K$. The most frequent anchor word in topic k is the word $\hat{j}$, where $\hat{j} = \text{argmax}_j\{a_j(k) : 1 \le j \le p\}$. Similarly, we can define the mth most frequent anchor word for any $m \ge 1$. **Figure 4** shows the 20 most frequent anchor words for each of the 11 identified topics. Based on these words, we suggest a name for each topic (see **Table 2**, second column). To check whether the proposed labels are reasonable and to get more insight into each topic, we also use $\hat{W}$ to identify representative papers. For each $1 \le k \le 11$, we pull out the top 300 papers with the largest $\hat{w}_i(k)$ (the titles of the top three within each topic are given in **Supplemental Table 6**). We manually examine the titles of these papers and suggest a list of major research topics covered by each brief label (see **Table 2**, third column).

Our topic learning results are based on abstract similarity (i.e., the research areas covered by the same topic have similar word counts in their abstracts). Such similarity does not necessarily imply similarity in the intellectual content of the papers. Also, our goal here is to use statistical methods to identify a few interpretable topics, and it is possible that some research topics in the data set are not well represented here.

5.2. Topic Weights for Representative Authors

How to estimate the research interests of an author is an interesting problem. It helps us understand an author's research profile and may be useful in decision making (e.g., awards, funding, promotions); it may also help the author plan for future research. We estimate the research interest of an author as follows. For an author a, let $\mathcal{N}_a \subset \{1, 2, \ldots, n\}$ be the collection of papers he/she published in our data range. Each paper i has an estimated topic weight vector $\hat{w}_i$ for its abstract. A reasonable metric of author a's interest on topic k is $\bar{w}_a(k) = (1/|\mathcal{N}_a|)\sum_{i\in\mathcal{N}_a} \hat{w}_i(k)$, $1 \le k \le 11$. Let $\bar{w}(k)$ be the average of $\hat{w}_i(k)$ over all 56,500 abstracts. We define the centered topic interest vector of author a by $z_a = \bar{w}_a - \bar{w} \in \mathbb{R}^{11}$. The entries of z_a sum to 0, so it has both positive and negative entries. We are interested in its positive entries, since $z_a(k) > 0$ indicates a greater-than-average weight on topic k.

We can compute the vector z_a for almost every author in our data range. **Supplemental Table 7** contains the results of 80 selected authors. **Figure 5** presents z_a for 12 representative authors. We have some interesting findings:

- James Berger has a prominently high weight on Bayesian statistics, Raymond Carroll and Jianqing Fan have prominently high weights on regression analysis, and Michael Jordan

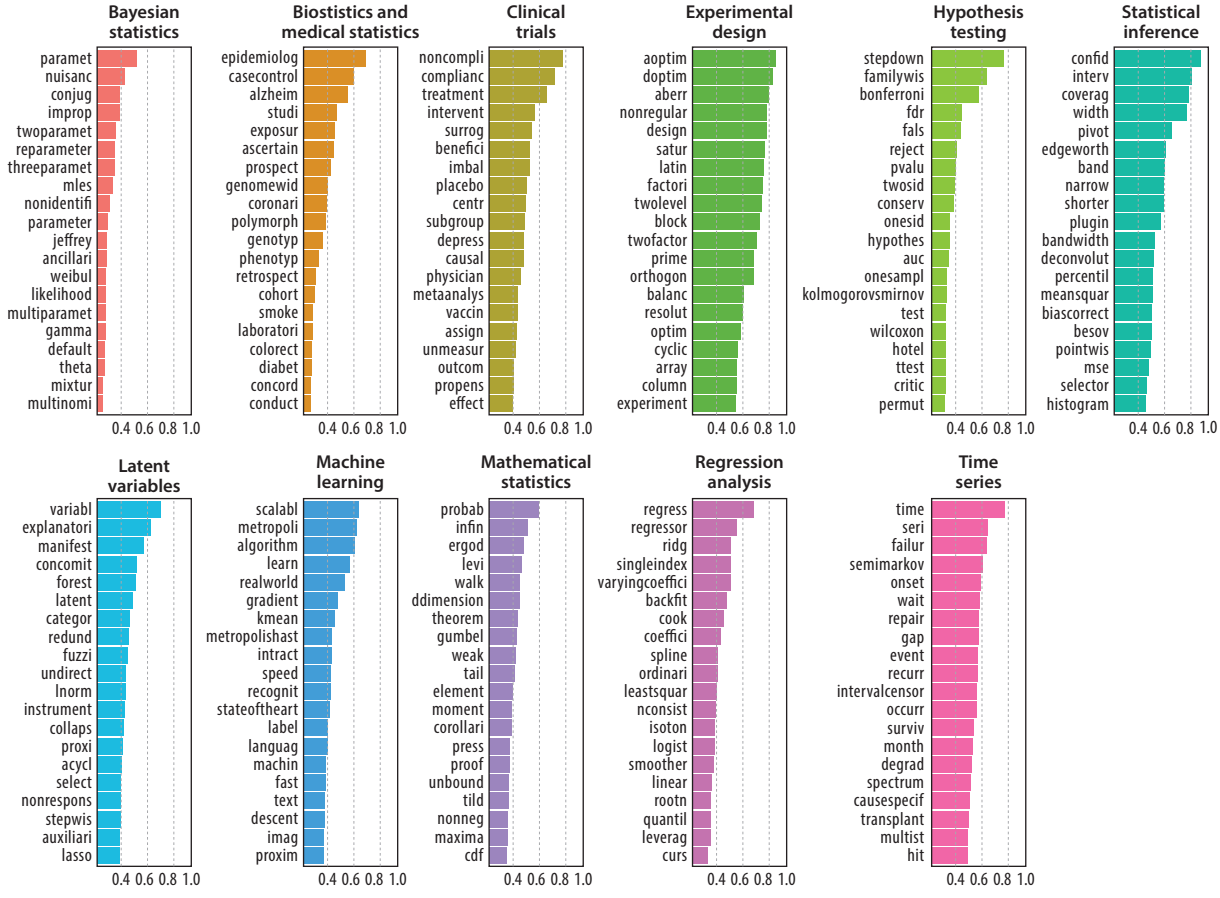


Figure 4

The 11 identified topics. Each panel contains the bar plot of the 20 most frequent anchor words of an estimated topic, where the bar length of a word is equal to the corresponding topic loading value. All words have been stemmed in data preprocessing.

and Jun Liu have prominently high weights on machine learning. These results are reasonable: Berger has many works in Bayesian statistics and decision theory; Carroll has many works in semiparametric models; Fan has many works in nonparametric regression and high-dimensional variable selection; Jordan has many works in machine learning, nonparametric Bayes, and Bayesian computation; and Liu has many works in Bayesian computation and MCMC.

- Peter Hall has notably high weights on statistical inference, machine learning, and regression analysis; Xihong Lin has notably high weights on clinical trials, regression analysis, and biostatistics and medical statistics; Larry Wassermann has notably high weights on statistical inference, machine learning, and Bayesian statistics; and Cun-Hui Zhang has notably high weights on statistical inference, regression analysis, and mathematical statistics.
- **Figure 5** suggests that the research interests of Peter Bickel, David Donoho, and Kathryn Roeder are relatively diverse, covering many topics; these are consistent with our impression of these authors and the information on the 11 topics in **Table 2**.

Table 2 Interpretation of the 11 estimated topics

	Topic label	Abbreviation	Corresponding research topic(s)
1	Bayesian statistics	Bayes	Bayesian methods
2	Biostatistics and medical statistics	Bio./Med.	Observational studies, genetics, genomics
3	Clinical trials	Clinic.	Clinical trials, causal inference
4	Experimental design	Exp.Design	Experimental design
5	Hypothesis testing	Hypo.Test	Hypothesis testing, goodness of fit
6	Statistical inference	Inference	Confidence intervals, bootstrapping, empirical likelihood
7	Latent variables	Latent.Var.	Latent variable model, incomplete data, mixtures, clustering, factor model, graphical model, variable selection, categorial data analysis, dimension reduction
8	Machine learning	Mach.Learn.	Machine learning, computation, expectation–maximization algorithm, Monte Carlo methods, clustering
9	Mathematical statistics	Math.Stats.	Asymptotics, mathematical statistics, probability, stochastic process
10	Regression analysis	Regression	Linear models, nonparametric regression, quantile regression, semiparametric models
11	Time series	Time Series	Time series, longitudinal data, stochastic processes, survival analysis

5.3. Topic Trends

How to characterize the evolution of statistical research over time is an intriguing problem (Kolar & Taddy 2016). We tackle it by combining the estimated topic weights and the time and journal information of each paper.

First, we study how the yearly average topic weights change over time. Recall that $\hat{w}_i$ is the estimated topic weight vector for paper i by Topic-SCORE. For each year, we compute the average topic weight for all papers published in this year, smoothed by a weighted moving average in a 3-year window (weights: 0.25, 0.50, and 0.25) (**Figure 6**). We observe that five topics, mathematical statistics, regression analysis, biostatistics and medical statistics, Bayesian statistics, and hypothesis testing, have higher-than-average weights, suggesting that they have attracted more attention; from 1990 to 2015, the weight of biostatistics and medical statistics increases relatively fast, the weights of mathematical statistics and hypothesis testing gradually decrease, and the weights of regression analysis and Bayesian statistics are relatively flat. Among the remaining six topics, machine learning increases quickly; its weight passes the overall average starting in 2014. Latent variables is another topic where the weight is steadily increasing.

Second, we select a few journals and study the evolution of the yearly average topic weights for each journal. In Section 4.4, we ranked the 33 journals (excluding the 3 probability journals) by Stigler's model and PageRank. We select the 7 journals with highest average ranks: *AoS*, *Biometrika*, *JASA*, *JRSSB*, *Biometrics*, *Journal of Machine Learning Research*, and *Statistica Sinica*. For each journal, we obtain the yearly average topic weight (i.e., the average of $\hat{w}_i$ among papers published in this journal each year) and smooth the curves as above. The results are in **Supplemental Figure 4**. A partial result is shown in **Figure 7*a***, where each subpanel corresponds to a topic. Fixing a topic k, for each journal, we plot the kth entry (subject to smoothing over time) in the yearly average of $\hat{w}_i$s among papers published in this journal. These curves of different journals for the same topic can be used to study journal friendliness to this topic.

Supplemental Material >

We observe that in some time periods, some journals are clearly in favor of some topics. When this happens, we say that the journal is friendly to this topic. **Figure 7*b*** lists the friendliest journals for 11 topics. Note that the short label of a topic may not be accurate for all research topics it

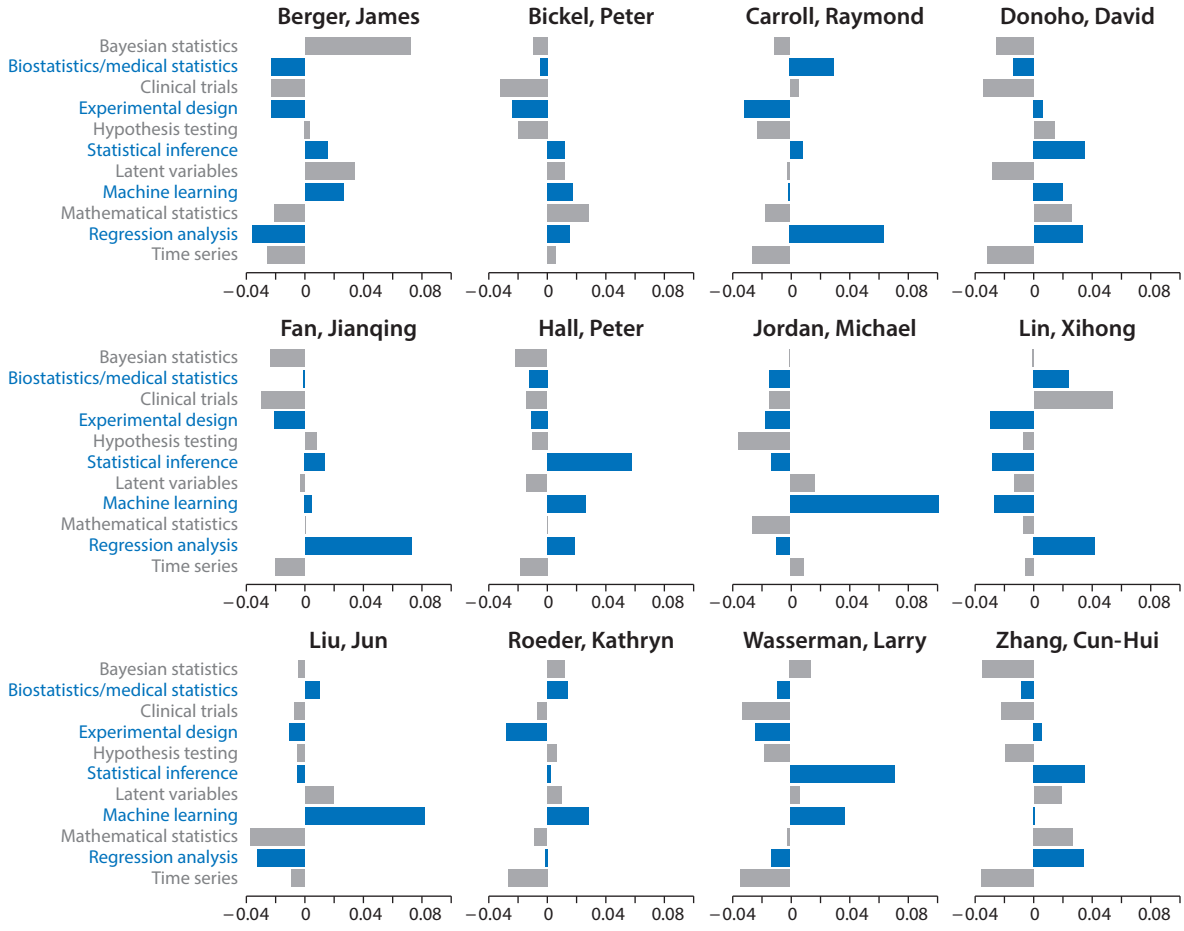


Figure 5

The overall topic interests of some authors. For interpretation purposes, we select some authors we are familiar with, but similar figures can be generated for other authors. Fixing an author a, we show the bar plots of the 11 entries of the vector z_a, where the kth entry of z_a is this author's weight on topic k minus the overall average weight on topic k among all authors.

covers, and **Table 2** contains more complete information (e.g., time series includes longitudinal data and survival analysis, and which is why this topic has a high weight in the journal *Biometrics*). Among the seven journals, *Journal of Machine Learning Research* has a significantly higher weight on machine learning than on the other topics, *Biometrics* has a significantly higher weight on biostatistics and medical statistics and clinical trials, and *AoS* has a considerably higher weight on mathematical statistics. Furthermore, four journals, *AoS*, *Biometrika*, *JASA* and *JRSSB*, are traditionally considered the leading journals in statistical methods and theory. Among these four journals, *AoS* is friendlier to mathematical statistics, statistical inference, hypothesis testing, regression analysis, and experimental design; *JASA* is friendlier to machine learning, biostatistics and medical statistics, clinical trials, and time series; *JRSSB* is friendlier to machine learning, Bayesian statistics, and latent variables; and *Biometrika* is friendlier to Bayesian statistics and regression analysis (*JASA* publishes more on clinical trials and on biostatistics and medical statistics than *Biometrika*, possibly because *JASA* has an applications and case-study track).

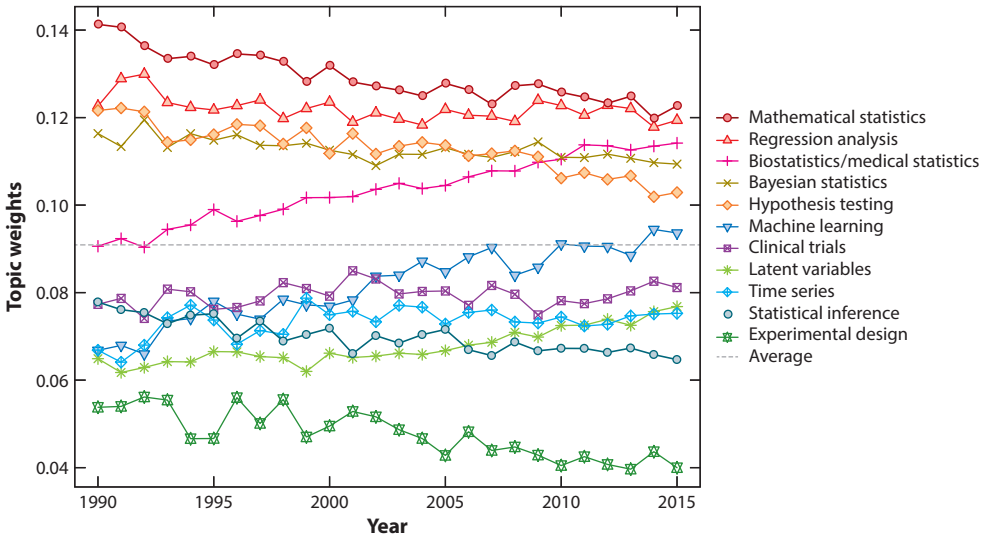


Figure 6

The yearly average topic weights (averaged for all 33 journals), 1990–2015.

6. TR-SCORE: AN EXTENSION OF TOPIC-SCORE FOR TOPIC RANKING

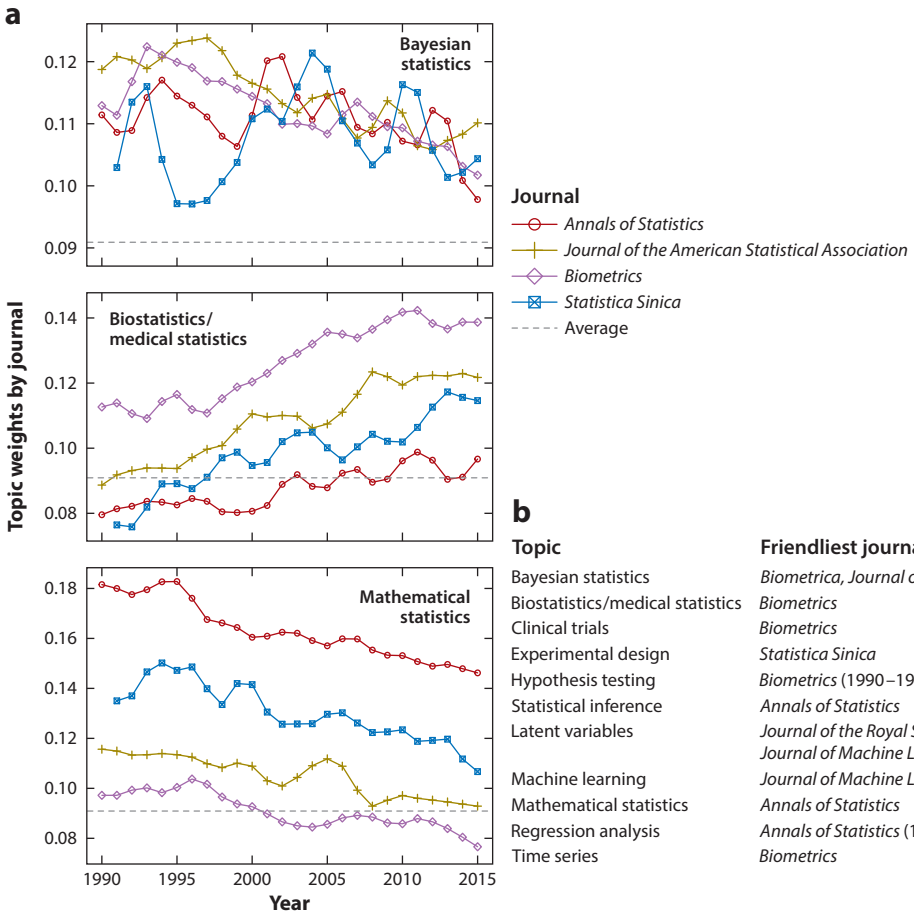
Topic-SCORE is a flexible idea and can be extended in many directions. In this section, we extend Topic-SCORE by proposing Topic-Ranking-SCORE (TR-SCORE) as a new approach to ranking the citation impacts of different topics. Since TR-SCORE is directly motivated by the analysis of MADStat, we focus our discussion in this section on MADStat, but keep in mind that the idea is useful in other applications.

In Section 4, we discuss how to use citation exchanges to rank different journals. We can extend the idea to topic ranking, but there is a major challenge: Citation exchanges between papers or journals are well defined and directly observable, but citation exchanges between research topics are not. We tackle this problem by combining the abstracts and the citation data: We first propose a model that jointly models text abstracts and citations, including an idea to measure the (unobserved) citation exchanges between research topics. We then introduce TR-SCORE and use it to rank different topics and to construct a knowledge graph visualizing the cross-topic citation exchanges.

6.1. The Hofmann–Stigler Model for Abstract and Citation Data

Consider n papers in MADStat, where the abstract data are summarized in a $p \times n$ word-document-count matrix $X = [x_1, x_2, \ldots, x_n]$ as in Section 2 (p is the vocabulary size), and citation data are summarized in an adjacency matrix $C \in \mathbb{R}^{n\times n}$, where $C_{ij} = 1$ if there is a citation from paper i to paper j and $C_{ij} = 0$ otherwise ($1 \le i, j \le n$).

We propose the Hofmann–Stigler model to jointly model the data matrices X and C: It combines the Hofmann topic model in Section 2 and Stigler's model in Section 4.4. We assume that all the paper abstracts focus on K different research topics $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K$. Inspired by Stigler's model, we introduce $\mu = (\mu_1, \mu_2, \ldots, \mu_K)'$, where μ_k is the export score associated with topic k ($1 \le k \le K$). Intuitively, a topic with a larger export score means that it has larger impacts. Now, fix $1 \le i \le n$ and consider paper i. Similar to in Section 2, let $w_i \in \mathbb{R}^K$ be the weight vector of



b

Topic	Friendliest journals
Bayesian statistics	*Biometrica, Journal of the Royal Statistical Society Series B*
Biostatistics/medical statistics	*Biometrics*
Clinical trials	*Biometrics*
Experimental design	*Statistica Sinica*
Hypothesis testing	*Biometrics* (1990–1998), *Annals of Statistics* (2002–2015)
Statistical inference	*Annals of Statistics*
Latent variables	*Journal of the Royal Statistical Society Series B* (2008–2012), *Journal of Machine Learning Research* (2004–2007)
Machine learning	*Journal of Machine Learning Research*
Mathematical statistics	*Annals of Statistics*
Regression analysis	*Annals of Statistics* (1990–2002)
Time series	*Biometrics*

Figure 7

(*a*) The yearly average topic weights in selected journals (owing to space limits, we plot only the curves for three topics in four selected journals; the complete result of 11 topics in seven selected journals can be found in **Supplemental Figure 4**). (*b*) The friendliest journal (out of seven selected journals) for each topic.

document i [i.e., $w_i(k)$ is the weight that abstract i puts on topic k]. When paper i is cited by another paper j, we have two different ways to attribute this particular citation count:

- Orthodox citation attribution (OCA). We simply attribute the citation to paper i.
- Topic weight citation attribution (TWCA). We attribute the citation to each of the K topics, with weights $w_i(1), \ldots, w_i(K)$ [note that $\sum_{k=1}^{K} w_i(k) = 1$].

Supplemental Material >

In Section 4.4, we discuss journal ranking, in which OCA is a good choice. For topic ranking, TWCA is more appropriate. Under TWCA, we view $\mu' w_i = \sum_{k=1}^{K} \mu_k w_i(k)$ as the export score of paper i and assume that the Bernoulli variables C_{ij} and C_{ji} satisfy

$$\mathrm{P}(C_{ij} = 1 | C_{ij} + C_{ji} \geq 1) = \frac{\exp(\mu' w_j - \mu' w_i)}{1 + \exp(\mu' w_j - \mu' w_i)}. \qquad 3.$$

This gives the model of the citation exchange matrix C. To model the word-document-count matrix X, we use the same model as in Section 2:

$$x_i \sim \text{multinomial}(N_i, Aw_i), \qquad A \in \mathbb{R}^{p\times K},\ w_i \in \mathbb{R}^K, \tag{4}$$

where A is the topic matrix as in Section 2 and N_i is the size (total word count) of document i. For identifiability, we assume $\text{median}(\mu_1, \ldots, \mu_K) = 0$. Also, for simplicity, we assume X and C are independent (but their distributions are related by w_is), and this can be relaxed. We call Equations 3 and 4 the Hofmann–Stigler model.

6.2. TR-SCORE

We propose using TR-SCORE for topic ranking. The inputs are X, C, and the number of topics K, and the output is an estimated export score vector $\hat{\mu}$. TR-SCORE has three steps:

1. Topic matrix estimation. Apply Topic-SCORE (e.g., Section 2.2) to get $\hat{A} \in \mathbb{R}^{p\times K}$.
2. Topic weight estimation. For $1 \le i \le n$, estimate w_i by $\hat{w}_i = (\hat{A}'\hat{A} + \lambda I_K)^{-1}\hat{A}'d_i$, where $\lambda > 0$ is a regularization parameter that we usually fix at $\lambda = 0.3$.
3. Topic ranking. Plug $\hat{w}_1, \ldots, \hat{w}_n$ into Equation 3 and obtain an estimate $\hat{\mu}$ for the export score vector μ. Rank topics according to the descending order of $\hat{\mu}_1, \hat{\mu}_2, \ldots, \hat{\mu}_K$.

We discuss Step 3 in detail. We use a quasi-likelihood method with overdispersion to obtain $\hat{\mu}$. Recall that C is the adjacency matrix of between-paper citations. Write $\bar{C} = C + C'$ (i.e., $\bar{C}_{ij} = C_{ij} + C_{ji}$). Recall that $W = [w_1, w_2, \ldots, w_n] \in \mathbb{R}^{K,n}$ is the topic weight matrix. Let $\tau(x) = e^x/(1 + e^x)$ denote the logistic function. We now slightly modify Equation 3 to assume

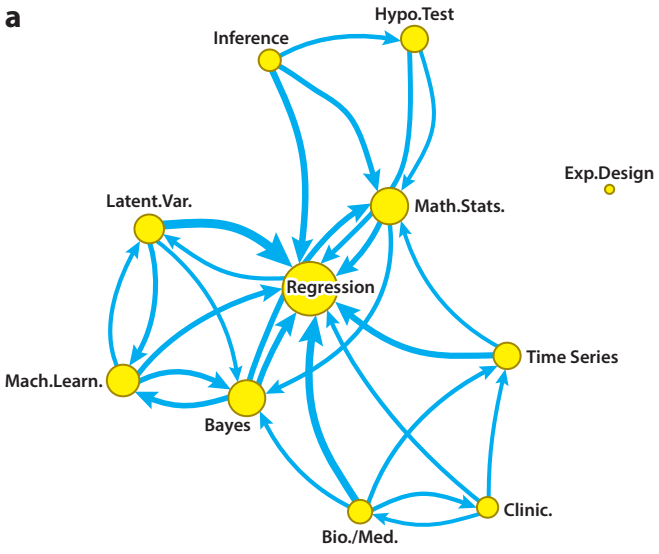
$$\text{E}[C|\bar{C}] = \bar{C} \circ \Omega, \quad \text{Var}(C|\bar{C}) = \phi[\Omega \circ (1 - \Omega)], \quad \text{with } \Omega = \tau(\mathbf{1}_n\mu'W - W'\mu\mathbf{1}_n'), \tag{5}$$

where $\circ$ is the Hadamard product, both $\text{var}(C|\bar{C})$ and $(1 - \Omega)$ are element-wise operations, and $\phi > 0$ is the dispersion parameter. The model in Equation 3 corresponds to fixing $\phi = 1$, but a better strategy is to estimate ϕ from data, as commonly used in fitting count data [for a similar strategy for fitting Stigler's model, see, e.g., Varin et al. (2016)]. When W is known, we estimate $\mu_1, \mu_2, \ldots, \mu_K$ by maximizing the quasi-likelihood, which is equivalent to maximizing the likelihood of the model in Equation 3. This is done by first fixing $\mu_1 = 0$ and treating Equation 3 as a generalized linear model with $(K - 1)$ predictors and $N := \sum_{i,j} 1\{\bar{C}_{ij} = 1\}$ samples, so that it can be solved by a standard package. We then recenter $\hat{\mu}_1, \hat{\mu}_2, \ldots, \hat{\mu}_K$ so that their median is zero. The dispersion parameter is estimated by $\hat{\phi} = (1/N - K + 1)\sum_{(i,j):i<j,\bar{C}_{ij}\ge 1}(C_{ij} - \bar{C}_{ij}\hat{\Omega}_{ij})^2/[\bar{C}_{ij}\hat{\Omega}_{ij}(1 - \hat{\Omega}_{ij})]$, where $\hat{\Omega}_{ij} = \tau(\hat{\mu}'w_j - \hat{\mu}'w_i)$. So far, W is assumed known. For unknown W, we use the same procedure, except that W is replaced by $\hat{W}$ from step 2.

6.3. Topic Ranking and a Cross-Citation Graph

In Section 5, we apply Topic-SCORE to a set of 56,500 (preprocessed) abstracts and identified 11 representative research topics in statistics. We now use TR-SCORE on the same set of abstracts and rank all 11 topics. We also build a cross-topic citation graph (as a type of knowledge graph) to visualize the dissemination of knowledge across areas, an important research topic in the area of modern knowledge discovery (Shi et al. 2015).

We first build a cross-topic citation graph. This is a weighted and directed graph with 11 nodes, each of which is a discovered topic. We propose two definitions of edge weights. In the first one, let $N_{k\ell} = \sum_{i,j=1}^n \hat{w}_i(k)\hat{w}_j(\ell)C_{ij}$ and $P_{k\ell} = N_{k\ell}/(\sum_{m=1}^{11} N_{km})$, for $1 \le k, \ell \le 11$, where C is the between-paper citation adjacency matrix and $\hat{w}_i$ is the topic weight vector of abstract i. Here $N_{k\ell}$ is the (allocated) citation counts from topic k to topic ℓ, and $P_{k\ell}$ is the proportion of citations to topic



b

Topic	Export score
Mathematical statistics	3.474
Regression analysis	1.688
Machine learning	1.671
Statistical inference	1.131
Bayesian statistics	0.159
Experimental design	0
Time series	−0.142
Latent variables	−0.240
Hypothesis testing	−1.279
Clinical trials	−1.611
Biostatistics/medical statistics	−5.427

Figure 8

(*a*) The weighted directed graph for cross-topic citations. The diameter of a node (topic) is proportional to the total citations the topic has received from other topics, and the width of a directed edge (*arrows*) is proportional to the weight defined in the text. For better visualization, all arrows corresponding to a weight of less than 0.09 are omitted. (*b*) The estimated export scores of 11 topics, subject to median$(\hat{\mu}_1, \ldots, \hat{\mu}_{11}) = 0$. Abbreviations: Bayes, Bayesian statistics; Bio./Med., biostatistics/medical statistics; Clinic., clinical trials; Exp.Design, experimental design; Hypo.Test, hypothesis testing; Inference, statistical inference; Latent.Var., latent variables; Mach.Learn., machine learning; Math.Stats., mathematical statistics; Regression, regression analysis.

ℓ among all citations from topic k. We use $P \in \mathbb{R}^{11\times 11}$ as the weighted adjacency matrix of this graph. In the second definition, we group all papers based on the dominant topic—the topic with the largest weight in $\hat{w}_i$ (if there is a tie, we pick the smaller k). Let $w_i^* \in \{e_1, e_2, \ldots, e_K\}$ denote the group label of abstract i. Define $N_{k\ell}^* = \sum_{i,j=1}^n \hat{w}_i^*(k)\hat{w}_j^*(\ell)C_{ij}$ and $P_{k\ell}^* = N_{k\ell}^*/(\sum_{m=1}^{11} N_{km}^*)$. We then use $P^* \in \mathbb{R}^{11\times 11}$ as the weighted adjacency matrix. This definition uses "winner takes all" to allocate each citation to a single pair of topics. The two matrices P and P^* are shown in **Supplemental Tables 8 and 9**. Both definitions make sense, but the second one leads to a sparser graph, which is presented in **Figure 8** (the first one is shown in **Supplemental Figure 5**).

Supplemental Material >

In **Figure 8*a***, the width of the edge from node k to node ℓ is proportional to $P_{k\ell}^*$, and the edge is presented only when $P_{k\ell}^* \geq 0.09$. We have interesting observations. First, experimental design has relatively few citation exchanges with other topics, and the majority of the citations it receives are from within the topic itself. Since a one-way edge from node k to node ℓ is presented when $P_{k\ell}^* \geq 0.09$, no edge from or to the experimental design topic is shown in **Figure 8**. Second, regression analysis and mathematical statistics are the two topics that have attracted the most citations from other topics, and biostatistics and medical statistics and statistical inference are the two that have cited other topics most often. Third, Bayesian statistics, latent variables, and machine learning all have considerably many outgoing and incoming citations. Last, hypothesis testing and statistical inference form a close pair, and most in-between citations are from statistical inference to hypothesis testing; clinical trials and biostatistics and medical statistics form a close pair, and the citation exchanges are relatively balanced between them.

We then consider topic ranking. **Figure 8*b*** shows the export scores of 11 topics by TR-SCORE. Mathematical statistics is the highest-ranked topic. This is reasonable, as the focus of

mathematical statistics is mathematical analysis and probability, which may have a long-lasting impact on other topics in statistics. Regression analysis and machine learning are also highly ranked. This is also understandable, as the two topics cover many popular research topics (**Table 2**). The rankings of biostatistics and medical statistics and clinical trials are relatively low; one reason is that a significant fraction of their impacts are over research areas outside our data range.

7. CONCLUSION

Text analysis is a rapidly developing research area in data science. In this article, we have surveyed recent methods for text analysis, ranging from topic modeling to neural language models. For topic modeling, we have discussed the anchor word condition, several different algorithms, optimal rates, and extensions to bigram and supervised models. In particular, we focus on Topic-SCORE, a fast algorithm that enjoys appealing theoretical properties. For neural language models, we have provided a brief introduction to its key components, reviewed the popular BERT and word embedding models, and discussed how to apply them to solve downstream NLP tasks.

We have also presented a data set, MADStat, about academic publications in statistics. It was collected and cleaned by ourselves with substantial effort. In this article, we analyzed text abstracts of the papers in MADStat, using the Topic-SCORE algorithm. We identified 11 representative topics and visualized the trends and patterns in statistical research. We also proposed the Hofmann–Stigler model to jointly model text abstracts and citation data and the TR-SCORE algorithm for ranking the citation impacts of the 11 topics. These results not only are applications of text analysis but also can be viewed as a data-driven review of the academic statistical community.

Nowadays, a vast amount of text data is generated on a daily basis. Recent advancements in NLP have revolutionized our everyday lives and have also provided a big opportunity for statistics. On the one hand, the statistical approaches to NLP are typically transparent, sample efficient, fast to compute, and theoretically tractable, making them a suitable choice for many ordinary NLP users (who may have a moderate-size domain-specific corpus but cannot access the data and resources owned by the tech giants). On the other hand, statistical text analysis is still quite underdeveloped. Even for topic modeling, there are still many unresolved problems, such as how to estimate the number of topics and how to improve the accuracy when the documents are extremely short. We hope that this review article provides useful information to researchers interested in this area. We also hope that MADStat, which we collected and shared with the public, serves as a good platform for testing existing methods and inspiring new research in text analysis.

Supplemental Material >

The MADStat data set and the code for text analysis conducted in this article can be found online (**https://github.com/ZhengTracyKe/MADStat-Text** and in the **Supplemental Data and Code**).

DISCLOSURE STATEMENT

The authors are not aware of any affiliations, memberships, funding, or financial holdings that might be perceived as affecting the objectivity of this review.

ACKNOWLEDGMENTS

Z.T.K. was partially supported by National Science Foundation (NSF) CAREER grant DMS-1943902. J.J. was partially supported by NSF grant DMS-2015469. Z.T.K. thanks Russell Kunes and Xiao-Li Meng for helpful discussions.

LITERATURE CITED

Arora S, Ge R, Halpern Y, Mimno D, Moitra A, et al. 2013. A practical algorithm for topic modeling with provable guarantees. *Proc. Mach. Learn. Res.* 28(2):280–88

Arora S, Ge R, Moitra A. 2012. Learning topic models—going beyond SVD. In *IEEE 53rd Annual Symposium on Foundations of Computer Science*, pp. 1–10. Piscataway, NJ: IEEE

Azzalini A. 1985. A class of distributions which includes the normal ones. *Scand. J. Stat.* 12(2):171–78

Bahdanau D, Cho K, Bengio Y. 2014. Neural machine translation by jointly learning to align and translate. arXiv:1409.0473 [cs.CL]

Benjamini Y, Hochberg Y. 1995. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. Ser. B* 57(1):289–300

Bing X, Bunea F, Wegkamp M. 2020. A fast algorithm with minimax optimal guarantees for topic models with an unknown number of topics. *Bernoulli* 26(3):1765–96

Blei DM, Ng AY, Jordan MI. 2003. Latent Dirichlet allocation. *J. Mach. Learn. Res.* 3:993–1022

Cai TT, Ke ZT, Turner P. 2023. Testing high-dimensional multinomials with applications to text analysis. *J. R. Stat. Soc. Ser. B*. In press

Deerwester S, Dumais ST, Furnas GW, Landauer TK, Harshman R. 1990. Indexing by latent semantic analysis. *J. Am. Soc. Inf. Sci.* 41(6):391–407

Dempster A, Laird N, Rubin D. 1977. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. Ser. B* 39(1):1–22

Devlin J, Chang MW, Lee K, Toutanova K. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805 [cs.CL]

Donoho D. 2017. 50 years of data science. *J. Comput. Graph. Stat.* 26(4):745–66

Donoho D, Jin J. 2015. Higher criticism for large-scale inference, especially for rare and weak effects. *Stat. Sci.* 30(1):1–25

Donoho D, Stodden V. 2003. When does non-negative matrix factorization give a correct decomposition into parts? In *Advances in Neural Information Processing Systems 16* (*NeurIPS 2003*), ed. S Thrun, L Saul, B Schölkopf, pp. 1141–48. Red Hook, NY: Curran

Donoho DL, Johnstone JM. 1994. Ideal spatial adaptation by wavelet shrinkage. *Biometrika* 81(3):425–55

Dos Santos C, Gatti M. 2014. Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, ed. J Tsujii, J Hajic, pp. 69–78. Stroudsburg, PA: Assoc. Comput. Linguist.

Efron B, Hastie T, Johnstone I, Tibshirani R. 2004. Least angle regression. *Ann. Stat.* 32(2):407–99

Fagan JL. 1988. *Experiments in automatic phrase indexing for document retrieval: a comparison of syntactic and nonsyntactic methods*. Tech. Rep., Cornell Univ., Ithaca, NY

Gelfand AE, Smith AFM. 1990. Sampling-based approaches to calculating marginal densities. *J. Am. Stat. Assoc.* 85(410):398–409

Gillis N, Vavasis SA. 2013. Fast and robust recursive algorithms for separable nonnegative matrix factorization. *IEEE Trans. Pattern Anal. Mach. Intell.* 36(4):698–714

Harman DK, ed. 1993. *The First Text Retrieval Conference* (*TREC-1*). Washington, DC: US Dep. Commer.

Hochreiter S, Schmidhuber J. 1997. Long short-term memory. *Neural Comput.* 9(8):1735–80

Hofmann T. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 50–57. New York: ACM

Horn RA, Johnson CR. 2013. *Matrix Analysis*. Cambridge, UK: Cambridge Univ. Press. 2nd ed.

Hubert L, Arabie P. 1985. Comparing partitions. *J. Classif.* 2:193–218

Ji P, Jin J, Ke ZT, Li W. 2022. Co-citation and co-authorship networks of statisticians (with discussion). *J. Bus. Econ. Stat.* 40(2):469–504

Jin J. 2015. Fast community detection by SCORE. *Ann. Stat.* 43(1):57–89

Jin J, Ke ZT, Luo S. 2018. Network global testing by counting graphlets. *Proc. Mach. Learn. Res.* 80:2333–41

Jin J, Ke ZT, Luo S. 2021. Optimal adaptivity of signed-polygon statistics for network testing. *Ann. Stat.* 49(6):3408–33

Jin J, Ke ZT, Luo S. 2023. Mixed membership estimation for social networks. *J. Econom.* In press

Kalchbrenner N, Grefenstette E, Blunsom P. 2014. A convolutional neural network for modelling sentences. arXiv:1404.2188 [cs.CL]

Ke Q, Ferrara E, Radicchi F, Flammini A. 2015. Defining and identifying sleeping beauties in science. *PNAS* 112(24):7426–31

Ke ZT, Jin J. 2023. The SCORE normalization, especially for heterogeneous network and text data. *Stat* 12(1):e545

Ke ZT, Kelly BT, Xiu D. 2019. *Predicting returns with text data*. NBER Work. Pap. 26186

Ke ZT, Wang M. 2022. Using SVD for topic modeling. *J. Am. Stat. Assoc.* **https://doi.org/10.1080/01621459.2022.2123813**

Kolar M, Taddy M. 2016. Discussion of "Coauthorship and citation networks for statisticians." *Ann. Appl. Stat.* 10(4):1835–41

Lee DD, Seung HS. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature* 401(6755):788–91

Lee J, Yoon W, Kim S, Kim D, Kim S, et al. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* 36(4):1234–40

Liang KY, Zeger SL. 1986. Longitudinal data analysis using generalized linear models. *Biometrika* 73(1):13–22

McAuliffe J, Blei D. 2007. Supervised topic models. In *Advances in Neural Information Processing Systems 20* (*NeurIPS'07*), ed. JC Platt, D Koller, Y Singer, ST Roweis, pp. 121–28. Red Hook, NY: Curran

Mei Q, Zhai C. 2001. A note on EM algorithm for probabilistic latent semantic analysis. In *CIKM '20: Proceedings of the 29th ACM International Conference on Information and Knowledge Management*. New York: ACM

Mikolov T, Chen K, Corrado G, Dean J. 2013. Efficient estimation of word representations in vector space. arXiv:1301.3781 [cs.CL]

Otter DW, Medina JR, Kalita JK. 2020. A survey of the usages of deep learning for natural language processing. *IEEE Trans. Neural Netw. Learn. Syst.* 32(2):604–24

Radford A, Narasimhan K, Salimans T, Sutskever I. 2018. *Improving language understanding by generative pre-training*. Work. Pap., OpenAI, San Francisco, CA. **https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf**

Rahali A, Akhloufi MA. 2023. End-to-end transformer-based models in textual-based NLP. *AI* 4(1):54–110

Shi F, Foster JG, Evans JA. 2015. Weaving the fabric of science: dynamic network models of science's unfolding structure. *Soc. Netw.* 43:73–85

Stigler SM. 1994. Citation patterns in the journals of statistics and probability. *Stat. Sci.* 9:94–108

Taddy M. 2012. On estimation and selection for topic models. *Proc. Mach. Learn. Res.* 20:1184–93

Tibshirani R. 1996. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B* 58(1):267–88

Varin C, Cattelan M, Firth D. 2016. Statistical modeling of citation exchange between statistics journals. *J. R. Stat. Soc. A* 179(1):1–63

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, et al. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30* (*NeurIPS'17*), ed. U von Luxburg, I Guyon, S Bengio, H Wallach, R Fergus, pp. 6000–10. Red Hook, NY: Curran

Wallach HM. 2006. Topic modeling: beyond bag-of-words. In *ICML '06: Proceedings of the 23rd International Conference on Machine Learning*, pp. 977–84. New York: ACM

Wu R, Zhang L, Cai TT. 2023. Sparse topic modeling: computational efficiency, near-optimal algorithms, and statistical inference. *J. Am. Stat. Assoc.* 118:1849–61

Zhu Y, Kiros R, Zemel R, Salakhutdinov R, Urtasun R, et al. 2015. Aligning books and movies: towards story-like visual explanations by watching movies and reading books. In *Proceedings of the 2015 IEEE International Conference on Computer Vision* (*ICCV*), pp. 19–27. Piscataway, NJ: IEEE