# A Method of Automated Nonparametric Content Analysis for Social Science

**Daniel J. Hopkins** Georgetown University
**Gary King** Harvard University

*The increasing availability of digitized text presents enormous opportunities for social scientists. Yet hand coding many blogs, speeches, government records, newspapers, or other sources of unstructured text is infeasible. Although computer scientists have methods for automated content analysis, most are optimized to classify individual documents, whereas social scientists instead want generalizations about the population of documents, such as the proportion in a given category. Unfortunately, even a method with a high percent of individual documents correctly classified can be hugely biased when estimating category proportions. By directly optimizing for this social science goal, we develop a method that gives approximately unbiased estimates of category proportions even when the optimal classifier performs poorly. We illustrate with diverse data sets, including the daily expressed opinions of thousands of people about the U.S. presidency. We also make available software that implements our methods and large corpora of text for further analysis.*

Efforts to systematically categorize text documents date to the late 1600s, when the Church tracked the proportion of printed texts which were nonreligious (Krippendorff 2004). Similar techniques were used by earlier generations of social scientists, including Waples, Berelson, and Bradshaw (1940, which apparently includes the first use of the term "content analysis") and Berelson and de Grazia (1947). Content analyses like these have spread to a vast array of fields, with automated methods now joining projects based on hand coding, and have increased at least sixfold from 1980 to 2002 (Neuendorf 2002). The recent explosive increase in web pages, blogs, emails, digitized books and articles, transcripts, and electronic versions of government documents (Lyman and Varian 2003) suggests the potential for many new applications. Given the infeasibility of much larger scale human-based coding, the need for automated methods is growing fast. Indeed, large-scale projects based solely on hand coding have stopped altogether in some fields (King and Lowe 2003, 618).

This article introduces new methods of automated content analysis designed to estimate the primary quantity of interest in many social science applications. These new methods take as data a potentially large set of text documents, of which a small subset is hand coded into an investigator-chosen set of mutually exclusive and

Daniel J. Hopkins is Assistant Professor of Government, Georgetown University, 681 Intercultural Center, Washington, DC 20057 (dhopkins@iq.harvard.edu, http://www.danhopkins.org). Gary King is Albert J. Weatherhead III University Professor, Harvard University, Institute for Quantitative Social Science, 1737 Cambridge St., Cambridge, MA 02138 (king@harvard.edu, http://gking.harvard.edu).

Replication materials are available at Hopkins and King (2009); see http://hdl.handle.net/1902.1/12898. Our special thanks to our indefatigable undergraduate coders Sam Caporal, Katie Colton, Nicholas Hayes, Grace Kim, Matthew Knowles, Katherine McCabe, Andrew Prokop, and Keneshia Washington. Each coded numerous blogs, dealt with the unending changes we made to our coding schemes, and made many important suggestions that greatly improved our work. Matthew Knowles also helped us track down and understand the many scholarly literatures that intersected with our work, and Steven Melendez provided invaluable computer science wizardry; both are coauthors of the open source and free computer program that implements the methods described herein (ReadMe: Software for Automated Content Analysis; see http://gking.harvard.edu/readme). We thank Ying Lu for her wisdom and advice, Stuart Shieber for introducing us to the relevant computer science literature, and http://Blogpulse.com for getting us started with more than a million blog URLs. Thanks to Ken Benoit, Doug Bond, Justin Grimmer, Matt Hindman, Dan Ho, Pranam Kolari, Mark Kantrowitz, Lillian Lee, Will Lowe, Andrew Martin, Burt Monroe, Stephen Purpura, Phil Schrodt, Stuart Shulman, and Kevin Quinn for helpful suggestions or data. Thanks also to the Library of Congress (PA#NDP03-1), the Center for the Study of American Politics at Yale University, the Multidisciplinary Program on Inequality and Social Policy, and the Institute for Quantitative Social Science for research support.

***American Journal of Political Science,*** Vol. 54, No. 1, January 2010, Pp. 229–247

©2010, Midwest Political Science Association ISSN 0092-5853

exhaustive categories.[1] As output, the methods give approximately unbiased and statistically consistent estimates of the proportion of all documents in each category. Accurate estimates of these *document category proportions* have not been a goal of most work in the classification literature, which has focused instead on increasing the accuracy of *classification into individual document categories*. Unfortunately, methods tuned to maximize the percent of documents correctly classified can still produce substantial biases in the aggregate proportion of documents within each category. This poses no problem for the task for which these methods were designed, but it suggests that a new approach may be of use for many social science applications.

When social scientists use formal content analysis, it is typically to make generalizations using document category proportions. Consider examples as far-ranging as Mayhew (1991, chap. 3), Gamson (1992, chaps. 3, 6, 7, and 9), Zaller (1992, chap. 9), Gerring (1998, chaps. 3–7), Mutz (1998, chap. 8), Gilens (1999, chap. 5), Mendelberg (2001, chap. 5), Rudalevige (2002, chap. 4), Kellstedt (2003, chap. 2), Jones and Baumgartner (2005, chaps. 3–10), and Hillygus and Shields (2008, chap. 6). In all these cases and many others, researchers conducted content analyses to learn about the distribution of classifications in a population, not to assert the classification of any particular document (which would be easy to do through a close reading of the document in question). For example, the manager of a congressional office would find useful an automated method of sorting individual constituent letters by policy area so they can be routed to the most informed staffer to draft a response. In contrast, political scientists would be interested primarily in tracking the proportion of mail (and thus constituent concerns) in each policy area. Policy makers or computer scientists may be interested in finding the needle in the haystack (such as a potential terrorist threat or the right web page to display from a search), but social scientists are more commonly interested in characterizing the haystack. Certainly, individual document classifications, when available, provide additional information to social scientists, since they enable one to aggregate in unanticipated ways, serve as variables in regression-type analyses, and help guide deeper qualitative inquiries into the nature of specific documents. But they do not usually (as in Benoit and Laver 2003) constitute the ultimate quantities of interest.

Automated content analysis is a new field and is newer still within political science. We thus begin in the second section with a concrete example to help fix ideas and define key concepts, including an analysis of expressed opinion through blog posts about Senator John Kerry. We next explain how to represent unstructured text as structured variables amenable to statistical analysis. The following section discusses problems with existing methods. We introduce our methods in the fifth section along with empirical verification from several data sets in the sixth section. The last section concludes. The appendix provides intercoder reliability statistics and offers a method for coping with errors in hand-coded documents.

## Measuring Political Opinions in Blogs: A Running Example

Although our methodology works for any unstructured text, we use blogs as our running example. Blogs (or "web logs") are periodic web postings usually listed in reverse chronological order.[2] For present purposes, we define our inferential target as expressed sentiment about each candidate in the 2008 American presidential election. Measuring the national conversation in this way is not the only way to define the population of interest, but it seems to be of considerable public interest and may also be of interest to political scientists studying activists (Verba, Schlozman, and Brady 1995), the media (Drezner and Farrell 2004), public opinion (Gamson 1992), social networks (Adamic and Glance 2005; Huckfeldt and Sprague 1995), or elite influence (Grindle 2005; Hindman, Tsioutsiouliklis, and Johnson 2003; Zaller 1992). We attempted to collect all English-language blog posts from highly political people who blog about politics all the time, as

[1] Although some excellent content analysis methods are able to delegate to the computer both the choice of the categorization scheme and the classification of documents into the chosen categories, our applications require methods where the social scientist chooses the questions and the data provide the answers. The former so-called "unsupervised learning methods" are versions of cluster analysis and have the great advantage of requiring fewer startup costs, since no theoretical choices about categories are necessary ex ante and no hand coding is required (Quinn et al. 2009; Simon and Xeons 2004). In contrast, the latter so-called "supervised learning methods," which require a choice of categories and a sample of hand-coded documents, have the advantage of letting the social scientist, rather than the computer program, determine the most theoretically interesting questions (Kolari, Finin, and Joshi 2006; Laver, Benoit, and Garry 2003; Pang, Lee, and Vaithyanathan 2002). These approaches, and others such as dictionary-based methods (Gerner et al. 1994; King and Lowe 2003), accomplish somewhat different tasks and so can often be productively used together, such as for discovering a relevant set of categories in part from the data.

[2] Eight percent of U.S. Internet users (about 12 million people), claim to have their own blog (Lenhart and Fox 2006). The growth worldwide has been explosive, from essentially none in 2000 to estimates today that range up to 185.62 million worldwide. Blogs are a remarkably democratic technology, with 72.82 million in China and at least 700,000 in Iran (Helmond 2008).

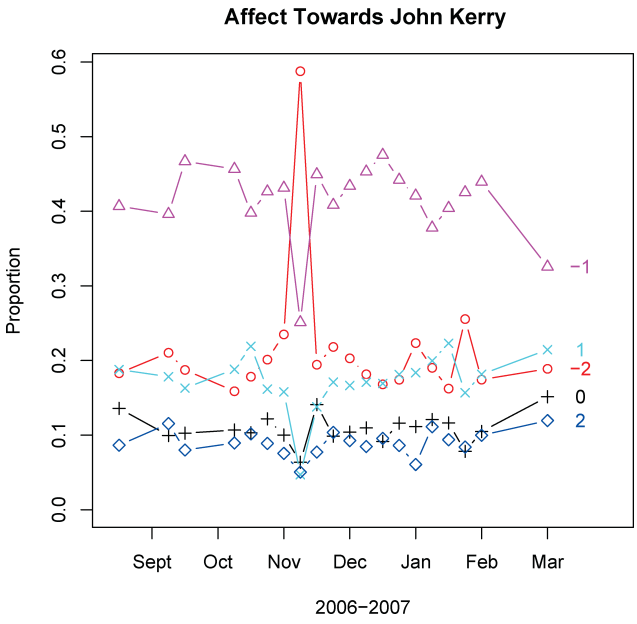
well as others who normally blog about gardening or their love lives, but choose to join the national conversation about the presidency for one or more posts. Bloggers' opinions get counted when they post and not otherwise, just as in earlier centuries when public opinion was synonymous with visible public expressions rather than attitudes and nonattitudes expressed in survey responses (Ginsberg 1986).[3]

Our specific goal is to compute the proportion of blogs each day or week in each of seven categories, including extremely negative (−2), negative (−1), neutral (0), positive (1), extremely positive (2), no opinion (NA), and not a blog (NB).[4] Although the first five categories are logically ordered, the set of all seven categories is not (which rules out innovative approaches like Wordscores, which presently requires a single dimension; Laver, Benoit, and Garry 2003). Bloggers write to express opinions and so category 0 is not common, although it and NA occur commonly if the blogger is writing primarily about something other than our subject of study. Category NB ensures that the category list is exhaustive. This coding scheme represents a difficult test case because of the mixed data types, because "sentiment categorization is more difficult than topic classification" (Pang, Lee, and Vaithyanathan 2002, 83), and because the language used ranges from the Queen's English to "my crunchy gf thinks dubya hid the wmd's, :)!!"[5]

We now preview the type of empirical results we seek. To do this, we apply the nonparametric method described below to blogosphere opinions about John Kerry before, during, and after the botched joke in the 2006 election cycle, which was said to have caused him to not enter the 2008 contest ("You know, education—if you make the most of it . . . you can do well. If you don't, you get stuck in Iraq"). Figure 1 gives a time-series plot of the proportion of blog posts in each of the opinion categories over time. The sharp increase in the extremely negative (−2) category occurred immediately following Kerry's joke. Note also the concomitant drop in other categories occurred primarily from the −1 category, but even the proportion in the positive categories dropped to some degree. Although the media portrayed this joke as his motivation for not entering the race, this figure suggests that his high negatives before and after this event may have been even more relevant.

These results come from an analysis of word patterns in 10,000 blog posts, of which only 442 from five days in early November were actually read and hand coded by the researchers. In other words, the method outlined in this article recovers a highly plausible pattern for several months using word patterns contained in a small, nonrandom subset of just a few days when anti-Kerry sentiment was at its peak. This was one incident in the

**FIGURE 1 Blogosphere Responses to Kerry's Botched Joke**




*Notes:* Each line gives a time series of estimates of the proportion of all English-language blog posts in categories ranging from −2 (extremely negative, colored red) to 2 (extremely positive, colored blue). The spike in the −2 category immediately followed Kerry's joke. Results were estimated with our nonparametric method in Section 5.2.

[3]We obtained our list of blogs by beginning with eight public blog directories and two other sources we obtained privately, including www.globeofblogs.com, http://truthlaidbear.com, www.nycbloggers.com, http://dir.yahoo.com/Computers_and_Internet/Internet/, www.bloghop.com/highrating.htm, http://www.blogrolling.com/top.phtml, a list of blogs provided by blogrolling.com, and 1.3 million additional blogs made available to us by Blogpulse.com. We then continuously crawl out from the links or "blogroll" on each of these blogs, adding seeds along the way from Google and other sources, to identify our target population.

[4]Our specific instructions to coders read as follows: "Below is one entry in our database of blog posts. Please read the entire entry. Then, answer the questions at the bottom of this page: (1) indicate whether this entry is in fact a blog posting that contains an opinion about a national political figure. If an opinion is being expressed (2) use the scale from −2 (extremely negative) to 2 (extremely positive) to summarize the opinion of the blog's author about the figure."

[5]Using hand coding to track opinion change in the blogosphere in real time is infeasible and even after the fact would be an enormously expensive task. Using unsupervised learning methods to answer the questions posed is also usually infeasible. Applied to blogs, these methods often pick up topics rather than sentiment or irrelevant features such as the informality of the text.

run-up to the 2008 campaign, but it gives a sense of the widespread applicability of the methods. Although we do not offer these in this article, one could easily imagine many similar analyses of political or social events where scale or resource constraints make it impossible to continuously read and manually categorize texts. We offer more formal validation of our methods below.

## Representing Text Statistically

We now explain how to represent unstructured text as structured variables amenable to statistical analysis, first by coding variables and then via statistical notation.

### Coding Variables

To analyze text statistically, we represent natural language as numerical variables following standard procedures (Joachims 1998; Kolari, Finin, and Joshi 2006; Manning and Schütze 1999; Pang, Lee, and Vaithyanathan 2002). For example, for our key variable, we summarize a document (a blog post) with its category. Other variables are computed from the text in three additional steps, each of which works without human input, and all of which are designed to reduce the complexity of text.

First, we drop non-English-language blogs (Cavnar and Trenkle 1994), as well as spam blogs (with a technology we do not share publicly; for another, see Kolari, Finin, and Joshi 2006). For the purposes of this article, we focus on blog posts about President George W. Bush (which we define as those that use the terms "Bush," "George W.," "Dubya," or "King George") and similarly for each of the 2008 presidential candidates. We develop specific filters for each person of interest, enabling us to exclude others with similar names, such as to avoid confusing Bill and Hillary Clinton. For our present methodological purposes, we focus on 4,303 blog posts about President Bush collected February 1–5, 2006, and 6,468 posts about Senator Hillary Clinton collected August 26–30, 2006. Our method works without filtering (and in foreign languages), but filters help focus the limited time of human coders on the categories of interest.

Second, we preprocess the text within each document by converting to lowercase, removing all punctuation, and stemming by, for example, reducing "consist," "consisted," "consistency," "consistent," "consistently," "consisting," and "consists" to their stem, which is "consist." Preprocessing text strips out information, in addition to reducing complexity, but experience in this literature is that the trade-off is well worth it (Porter 1980; Quinn et al. 2009).

Finally, we summarize the preprocessed text as dichotomous variables, one type for the presence or absence of each word stem (or "unigram"), a second type for each word pair (or "bigram"), a third type for each word triplet (or "trigram"), and so on to all "n-grams." This definition is not limited to dictionary words. In our application, we measure only the presence or absence of stems rather than counts (the second time the word "awful" appears in a blog post does not provide as much information as the first). Even so, the number of variables remaining is enormous. For example, our sample of 10,771 blog posts about President Bush and Senator Clinton includes 201,676 unique unigrams, 2,392,027 unique bigrams, and 5,761,979 unique trigrams. The usual choice to simplify further is to consider only dichotomous stemmed unigram indicator variables (the presence or absence of each of a list of word stems), which we have found to work well. We also delete stemmed unigrams appearing in fewer than 1% or greater than 99% of all documents, which results in 3,672 variables. These procedures effectively group the infinite range of possible blog posts to "only" $2^{3,672}$ distinct types. This makes the problem feasible but still represents a huge number (larger than the number of elementary particles in the universe).

Researchers interested in similar problems in computer science commonly find that "bag of words" simplifications like this are highly effective (e.g., Pang, Lee, and Vaithyanathan 2002; Sebastiani 2002), and our analysis reinforces that finding. This seems counterintuitive at first, since it is easy to write text whose meaning is lost when word order is discarded (e.g., "I hate Clinton. I love Obama"). But empirically, most text sources make the same point in enough different ways that representing the needed information abstractly is usually sufficient. As an analogy, when channel surfing for something to watch on television, pausing for only a few hundred milliseconds on a channel is typically sufficient; similarly, the negative content of a vitriolic post about President Bush is usually easy to spot after only a sentence or two. When the bag of words approach is not a sufficient representation, many procedures are available: we can code where each word stem appears in a document, tag each word with its part of speech, or include selective bigrams, such as by replacing "white house" with "white_house" (Das and Chen 2001). We can also use counts of variables or code variables to represent meta-data, such as the URL, title, blogroll, or whether the post links to known liberal or conservative sites (Thomas, Pang, and Lee 2006). Many other similar tricks suggested in the computer science

literature may be useful for some problems (Pang and Lee 2008), and all can be included in the methodology described below, but we have not found them necessary for the many applications we have tried to date.

### Notation and Quantities of Interest

Our procedures require two sets of text documents. The first is a small *labeled set*, for which each document $i$ ($i = 1, \ldots, n$) is labeled with one of the given categories, usually by reading and hand coding (we discuss how large $n$ needs to be in the sixth section, and what to do if hand coders are not sufficiently reliable in the appendix). We denote the Document category variable as $D_i$, which in general takes on the value $D_i = j$, for possible categories $j = 1, \ldots, J$.[6] (In our running example, $D_i$ takes on the potential values $\{-2, -1, 1, 0, 1, 2, \text{NA}, \text{NB}\}$.) We denote the second, larger *population set* of documents as the inferential target, and in which each document $\ell$ (for $\ell = 1, \ldots, L$) has an unobserved classification $D_\ell$. Sometimes the labeled set is a sample from the population and so the two overlap; more often it is a nonrandom sample from a different source than the population, such as from earlier in time.

All other information is computed directly from the documents. To define these variables for the labeled set denote $S_{ik}$ as equal to 1 if word Stem $k$ ($k = 1, \ldots, K$) is used at least once in document $i$ (for $i = 1, \ldots, n$) and 0 otherwise (and similarly for the population set, substituting index $i$ with index $\ell$). This makes our abstract summary of the text of document $i$ the set of these variables, $\{S_{i1}, \ldots, S_{iK}\}$, which we summarize as the $K \times 1$ vector of word stem variables $\boldsymbol{S}_i$. We refer to $\boldsymbol{S}_i$ as a *word stem profile* since it provides a summary of all the word stems (or other information) used in a document.

The quantity of interest in most of the supervised learning literature is the set of individual classifications for all documents in the population, $\{D_1, \ldots, D_L\}$. In contrast, the quantity of interest for most content analyses in social science is the aggregate proportion of all (or a subset of all) of these population documents that fall into each category: $P(D) = \{P(D = 1), \ldots, P(D = J)\}'$ where $P(D)$ is a $J \times 1$ vector, each element of which is a proportion computed by direct tabulation:

$$P(D = j) = \frac{1}{L} \sum_{\ell=1}^{L} \mathbf{1}(D_\ell = j), \qquad (1)$$

[6]This notation is from King and Lu (2008), who use related methods applied to unrelated substantive applications that do not involve coding text, and different mnemonic associations.

where $\mathbf{1}(a) = 1$ if $a$ is true and 0 otherwise. Document category $D_i$ is one variable with many possible values, whereas word profile $\boldsymbol{S}_i$ constitutes a set of dichotomous variables. This means that $P(D)$ is a multinomial distribution with $J$ possible values and $P(\boldsymbol{S})$ is a multinomial distribution with $2^K$ possible values, each of which is a word stem profile.

## Issues with Existing Approaches

This section discusses problems with two common methods that arise when they are used to estimate social aggregates rather than individual classifications.

### Existing Approaches

A simple way of estimating $P(D)$ is *direct sampling*: identify a well-defined population of interest, draw a random sample from the population, hand code all the documents in the sample, and count the documents in each category. This method requires basic sampling theory, no abstract numerical summaries of any text, and no classifications of individual documents in the unlabeled population.

The second approach to estimating $P(D)$, the *aggregation of individual document classifications*, is standard in the supervised learning literature. The idea is to first use the labeled sample to estimate a functional relationship between document category $D$ and word features $\boldsymbol{S}$. Typically, $D$ serves as a multicategory dependent variable and is predicted with a set of explanatory variables $\{S_{i1}, \ldots, S_{iK}\}$, using some statistical, machine learning, or rule-based method (such as multinomial logit, regression, discriminant analysis, radial basis functions, CART, random forests, neural networks, support vector machines, maximum entropy, or others). Then the coefficients of the model are estimated, and both the coefficients and the data-generating process are assumed the same in the labeled sample as in the population. The coefficients are then used with the features measured in the population, $\boldsymbol{S}_\ell$, to predict the classification for each population document $D_\ell$. Social scientists then aggregate the individual classifications via equation (1) to estimate their quantity of interest, $P(D)$.

### Problems

Unfortunately, as Hand (2006) points out, the standard supervised learning approach to individual document

classification will fail in two circumstances, both of which appear common in practice. (And even if classification succeeds with high or optimal accuracy, the next subsection shows that estimating population proportions can still be biased.)

First, when the labeled set is not a random sample from the population, both methods fail. Yet "in many, perhaps most real classification problems the data points in the [labeled] design set are not, in fact, randomly drawn from the same distribution as the data points to which the classifier will be applied. . . . It goes without saying that statements about classifier accuracy based on a false assumption about the identity of the [labeled] design set distribution and the distribution of future points may well be inaccurate" (Hand 2006, 2). Deviations from randomness may occur due to "population drift," which occurs when the labeled set is collected at one point and meant to apply to a population collected over time (as with blogs), or for other reasons. The burdens of hand coding become especially apparent when considering the typical analysis within subgroups and the need for a separate random sample within each.

Second, the data-generation process assumed by the standard supervised learning approach predicts $D$ with $\boldsymbol{S}$, modeling $P(D \mid \boldsymbol{S})$, but the world works in reverse. For our running example, bloggers do not start writing and only afterwards discover their affect toward the president: they start with a view, which we abstract as a document category, and then set it out in words. That is, the right data-generation process is the inverse of what is being modeled, where we should be predicting $\boldsymbol{S}$ with $D$, and inferring $P(\boldsymbol{S} \mid D)$. The consequence of using $P(D \mid \boldsymbol{S})$ instead (and without Bayes Theorem, which is not very helpful in this case) is the requirement of two assumptions needed to generalize from the labeled sample to the population. The first is that $\boldsymbol{S}$ "spans the space of all predictors" of $D$ (Hand 2006, 9), which means that once one controls for the measured variables, there exists no other variable that could improve predictive power. In problems involving human language, this assumption is not met, since $\boldsymbol{S}$ is intentionally an abstraction and so by definition does not represent all existing information in the predictors. The other assumption is that the class of models chosen for $P(D \mid \boldsymbol{S})$ includes the "true" model. This is a more familiar assumption to social scientists, but it is no easier to meet. In this case, finding even the best model or a good model, much less the "true one," is a difficult and time-consuming task given the huge number of potential explanatory variables coded from text and potential models to run. As Hand writes, "Of course, it would be a brave person who could confidently assert that these two conditions held" (2006, 9).

### Optimizing for a Different Goal

Here we show that even optimal individual document classification that meets all the assumptions of the last section can lead to biased estimates of the document category proportions. The criterion for success in the classification literature, the percent correctly classified in a test set, is obviously appropriate for individual-level classification, but it can be seriously misleading when characterizing document populations. For example, of the 23 models estimated by Pang, Lee, and Vaithyanathan (2002), the percent correctly predicted ranges from 77% to 83%. This is an excellent classification performance for sentiment analysis, but suppose that all the misclassifications were in a particular direction for one or more categories. In that situation, the statistical *bias* (the average difference between the true and estimated proportion of documents in a category) in using this method to estimate the aggregate quantities of interest could be as high as 17 to 23 percentage points. This does not matter for the authors, since their goal was classification, but it could matter for researchers interested in category proportions.

Unfortunately, except at the extremes, there exists no necessary connection between low misclassification rates and low bias: it is easy to construct examples of learning methods that achieve a high percent of individual documents correctly predicted and large biases for estimating the aggregate document proportions, or other methods that have a low percent correctly predicted but nevertheless produce relatively unbiased estimates of the aggregate quantities. For example, flipping a coin is a bad predictor of which party will win a presidential election, but it does happen to provide an unbiased estimate of the percentage of Democratic presidential victories since 1912. Since the goal of this literature is individual classification, it does not often report the bias in estimating the aggregates. As such, the bulk of the otherwise impressive supervised learning classification literature offers little indication of whether the methods proposed would work well for those with different goals.

## Statistically Consistent Estimates of Social Aggregates

We now introduce a method optimized for estimating document category proportions. To simplify the exposition, we first show how to correct aggregations of any existing classification method and after offer our standalone procedure, not requiring (or producing) a method of individual document classification.

## Corrected Aggregations of Individual Classifications

*Intuition.* Consider multinomial logit or any other method which can generate individual classifications. Fit this model to the labeled set, use it to classify each of the unlabeled documents in the population of interest, and aggregate the classifications to obtain a raw, uncorrected estimate of the proportion of documents in each category. Next, estimate misclassification probabilities by first dividing the labeled set of documents into a training set and a test set (ignoring the unlabeled population set). Then apply the same classification method to the training set alone and make predictions for the test set, $\hat{D}_i$ (ignoring the test set's labels). Then use the test set's labels to calculate the specific misclassification probabilities between each pair of actual classifications given each true value, $P(\hat{D}_i = j \mid D_i = j')$. These misclassification probabilities do not tell us which documents are misclassified, but they can be used to correct the raw estimate of the document category proportions.

For example, suppose we learn, in predicting test set proportions from the training set, that 17% of the documents our method classified as $D = 1$ really should have been classified as $D = 3$. For any one individual classification in the population, this fact is of no help. But for document category proportions, it is easy to use: subtract 17% from the raw estimate of the category 1 proportion in the population, $P(D = 1)$, and add it to category 3, $P(D = 3)$. Even if the raw estimate was badly biased, which can occur despite optimal individual document classification, the resulting corrected estimate would be unbiased so long as the population misclassification errors were estimated well enough from the labeled set (a condition we discuss below). Even if the percent correctly predicted is low, this corrected method can give unbiased estimates of the category frequencies.

*Formalization for Two Categories.* For the special case where $D$ is dichotomous, the misclassification correction above is well known in epidemiology—an area of science directly analogous to the social sciences, where much data are at the individual level, but the quantities of interest are often at the population level. To see this, consider a dichotomous $D$, with values 1 or 2, a raw estimate of the proportion of documents in category 1 from some method of classification, $P(\hat{D} = 1)$, and the true proportion (corrected for misclassification), $P(D = 1)$.[7]

Then define two forms of correct classification as "sensitivity," $\text{sens} \equiv P(\hat{D} = 1 \mid D = 1)$ (sometimes known as "recall"), and "specificity," or $\text{spec} \equiv P(\hat{D} = 2 \mid D = 2)$. For example, sensitivity is the proportion of documents predicted to be in category 1 among those actually in category 1.

Then we note that the proportion of documents estimated to be in category 1 must come from only one of two sources: documents actually in category 1 that were correctly classified and documents actually in category 2 but misclassified into category 1. We represent this accounting identity, known as the Law of Total Probability, as

$$P(\hat{D} = 1) = (\text{sens})\,P(D = 1) + (1 - \text{spec})\,P(D = 2). \tag{2}$$

Since equation (2) is one equation with only one unknown [since $P(D = 1) = 1 - P(D = 2)$], it is easy to solve. As Levy and Kass (1970) first showed, the solution is

$$P(D = 1) = \frac{P(\hat{D} = 1) - (1 - \text{spec})}{\text{sens} - (1 - \text{spec})}. \tag{3}$$

This expression can be used in practice by estimating sensitivity and specificity in the first-stage analysis (separating the labeled set into training and test sets as discussed above or more formally by cross-validation) and using the entire labeled set to predict the (unlabeled) population set to give $P(\hat{D} = 1)$. Plugging these values in the right side of (3) gives a corrected, and statistically consistent, estimate of the true proportion of documents in category 1.

*Generalization to Any Number of Categories.* The applications in epidemiology for which these expressions were developed are completely different than our problems, but the methods developed there are directly relevant. This connection enables us to use for our application the generalizations developed by King and Lu (2008).[8]

[7]The raw estimate $P(\hat{D} = 1)$ can be based on the proportion of individual documents classified into category 1. However, a better estimate for classifiers that give probabilistic classifications is to sum the estimated probability that each document is in the category for all documents. For example, if 100 documents each have a 0.52 probability of being in category 1, then all individual classifications are in this category. However, since we would only expect 52% of documents to actually be in category 1, a better estimate is $P(\hat{D} = 1) = 0.52$.

[8]King and Lu's (2008) article contributed to the field in epidemiology called "verbal autopsies." The goal of this field is to estimate the distribution of the causes of death in populations without medical death certification. This information is crucial for directing international health policy and research efforts. Data come from two sources. One is a sample of deaths from the population, where a relative of each deceased is asked a long (50–100 item) list of usually dichotomous questions about symptoms the deceased may

Thus, we first generalize equation (2) to include any number of categories by substituting $j$ for 1, and summing over all categories instead of just 2:

$$P(\hat{D} = j) = \sum_{j'=1}^{J} P(\hat{D} = j \mid D = j')P(D = j'). \quad (4)$$

Given $P(\hat{D})$ and the misclassification probabilities $P(\hat{D} = j \mid D = j')$, which generalize sensitivity and specificity to multiple categories, this expression represents a set of $J$ equations (i.e., defined for $j = 1, \ldots, J$) that can be solved for the $J$ elements in $P(D)$. This is aided by the fact that the equations include only $J - 1$ unknowns since elements of $P(D)$ must sum to 1.

*Interpretation.* The section entitled "Optimizing for a Different Goal" shows that a method meeting all the assumptions required for optimal classification performance can still give biased estimates of the document category proportions. We therefore offer here statistically consistent estimates of document category proportions, without having to improve individual classification accuracy and with no assumptions beyond those already made by the individual document classifier. In particular, classifiers require that the labeled set be a random sample from the population. Our method only requires a special case of the random selection assumption: that the misclassification probabilities (sensitivity and specificity with 2 categories or $P(\hat{D} = j \mid D = j')$ for all $j$ and $j'$ in equation (4)) estimated with data from the labeled set also hold in the unlabeled population set. This assumption may be wrong, but if it is, then the assumptions necessary for the original classifier to work are also wrong and will not necessarily even give accurate individual classifications. More importantly, our approach will also work with a biased classifier.

## Document Category Proportions Without Individual Classifications

We now offer an approach that requires no parametric statistical modeling, individual document classification, or random sampling from the target population. It also correctly treats $S$ as a consequence rather than cause of $D$.

*The Approach.* This method requires only one additional step beyond that in the previous section: instead of using $\boldsymbol{S}$ and $D$ to estimate $P(\hat{D} = j)$, and then separately correcting via equation (4), we avoid having to compute $\hat{D}$ by using $\boldsymbol{S}$ in place of $\hat{D}$ in that same equation. That is, any observable implication of $D$ can be used in place of $\hat{D}$ in equation (4); because $\hat{D}$ is a function of $\boldsymbol{S}$—since the words chosen are by definition a function of the document category—it is simplest to use it directly. Thus, we have

$$P(\boldsymbol{S} = s) = \sum_{j=1}^{J} P(\boldsymbol{S} = s \mid D = j)P(D = j). \quad (5)$$

To simplify this expression, we rewrite equation (5) as an equivalent matrix expression:

$$\underset{2^K \times 1}{P(\boldsymbol{S})} = \underset{2^K \times J}{P(\boldsymbol{S} \mid D)} \; \underset{J \times 1}{P(D)} \quad (6)$$

where, as indicated, $P(\boldsymbol{S})$ is the probability of each of the $2^K$ possible word stem profiles occurring,[9] $P(\boldsymbol{S} \mid D)$ is the probability of each of the $2^K$ possible word stem profiles occurring within the documents in category $D$ (columns of $P(\boldsymbol{S} \mid D)$ corresponding to values of $D$), and $P(D)$ is our $J$-vector quantity of interest.

*Estimation.* Elements of $P(\boldsymbol{S})$ can be estimated by direct tabulation from the target population, without parametric assumptions: we merely compute the proportion of documents observed with each pattern of word profiles. Since $D$ is not observed in the population, we cannot estimate $P(\boldsymbol{S} \mid D)$ directly. Instead, we make the crucial assumption that its value in the labeled, hand-coded sample, $P^h(\boldsymbol{S} \mid D)$, is the same as that in the population,

$$P^h(\boldsymbol{S} \mid D) = P(\boldsymbol{S} \mid D), \quad (7)$$

and use the labeled sample to estimate this matrix (we discuss this assumption below). We avoid parametric assumptions here too, by using direct tabulation to compute the proportion of documents observed to have each specific word profile among those in each document category.

In principle, we could estimate $P(D)$ in equation (6) assuming only the veracity of equation (7) and the accuracy of our estimates of $P(\boldsymbol{S})$ and $P(\boldsymbol{S} \mid D)$, by solving equation (6) via standard regression algebra. That is, if we

have suffered prior to death ($\boldsymbol{S}_\ell$). The other source of data is deaths in a nearby hospital, where the same data collection of symptoms from relatives is collected ($\boldsymbol{S}_i$) and also where medical death certification is available ($D_i$). Their method produces approximately unbiased and consistent estimates, considerably better than the existing approaches, which included expensive and unreliable physician reviews (where three physicians spend 20 minutes with the answers to the symptom questions from each deceased to decide on the cause of death), reliable but inaccurate expert rule-based algorithms, or model-dependent parametric statistical models.

[9] For example, if we ran the method with only $K = 3$ word stems, $P(\boldsymbol{S})$ would contain the probabilities of each of these $(2^3 = 8)$ patterns occurring in the set of documents: 000 (i.e., none of the three words were used), 001, 010, 011, 100, 101, 110, and 111.

think of $P(D)$ as the unknown "regression coefficients" $\beta$, $P(\boldsymbol{S} \mid D)$ as the "explanatory variables" matrix $X$, and $P(\boldsymbol{S})$ as the "dependent variable" $Y$, then equation (6) becomes $Y = X\beta$ (with no error term). This happens to be a linear expression but not because of any assumption imposed on the problem that could be wrong. The result is that we can solve for $P(D)$ via the usual regression calculation: $\beta = (X'X)^{-1}X'y$ (or via standard constrained least squares to ensure that elements of $P(D)$ are each in [0,1] and collectively sum to 1). A key point is that this calculation does *not* require classifying individual documents into categories and then aggregating; it estimates the aggregate proportions directly.

This simple approach poses two difficulties in our application. First, $K$ is typically very large and so $2^K$ is far larger than any standard computer could handle. Second is a sparseness problem since the number of observations available for estimating $P(\boldsymbol{S})$ and $P(\boldsymbol{S} \mid D)$ is much smaller than the number of potential word profiles ($n << 2^K$). To avoid both of these issues, we adapt results from King and Lu (2008) and randomly choose subsets of between approximately 5 and 25 words. The optimal number of words to use per subset is application-specific, but can be determined empirically through cross-validation within the labeled set. Although the estimator remains approximately unbiased regardless of subset size, in practice we find that setting the number of words per subset too high can lead to inefficiency. The reason is that as the number of words per subset increases, the number of unique subsets increases, reducing the number of common subsets that appear in both the labeled and unlabeled data sets. In addition, in the applications below, the words included in each subset are chosen randomly with equal probabilities, although in some applications, performance may improve by weighting words unequally.

Once we determine the optimal number of subsets through cross-validation, we solve for $P(D)$ in each, and average the results across the subsets. Because $\boldsymbol{S}$ is treated as a consequence of $D$, using subsets of $\boldsymbol{S}$ introduces no new assumptions. This simple subsetting procedure turns out to be equivalent to a version of the standard approach of smoothing sparse matrices via kernel densities, although, unlike the typical use of this procedure, its application here reduces bias. (Standard errors and confidence intervals are computed via standard bootstrapping procedures.)

*Interpretation.* A key advantage of estimating $P(D)$ without the intermediate step of computing the individual classifications is that the required assumptions are much less restrictive. They can still be wrong, and as a result our estimates can be biased, but the dramatic reduction in their restrictiveness means that under the new approach we have a fighting chance to get something close to the right answer in many applications where valid inferences were not previously likely.

Unlike direct sampling or standard supervised learning approaches, our strategy allows the distribution of documents across word-stem profiles, $P(\boldsymbol{S})$, and the distribution of documents across the categories, $P(D)$, to each be completely different in the labeled set and population set of documents. So for example, if a word or pattern of words becomes more popular between the time the labeled set was hand coded and the population documents were collected, no biases would emerge. Similarly, if documents in certain categories are more prevalent in the population than labeled set, no biases would result. In our running example, no bias would be induced if the labeled set includes a majority of conservative Republicans who defend everything President Bush does and the target population has a supermajority of liberal Democrats who want nothing more than to end the Bush presidency. In contrast, changes in either $P(D)$ or $P(\boldsymbol{S})$ between the labeled and population sets would be sufficient to doom existing classification-based approaches. For example, so long as "idiot" remains an insult, our method can make appropriate use of that information, even if the word becomes less common (a change in $P(S)$) or if there are fewer people who think politicians deserve it (a change in $P(D)$).

The key theoretical assumption is equation (7)—that the documents in the hand-coded set contain sufficient good examples of the language used for each document category in the population. To be more specific, among all documents in a given category, the prevalence of particular word profiles in the labeled set should be the same in expectation as in the population set. For example, the language bloggers use to describe an "extremely negative" view of Hillary Clinton in the labeled set must be at least a subset of the way she is described in the target population. They do not need to write literally the same blog posts, but rather need to have the same probabilities of using similar word profiles so that $P^h(\boldsymbol{S} \mid D = -2) = P(\boldsymbol{S} \mid D = -2)$. This assumption can be violated due to population drift or for other reasons, but we can always hand code some additional cases in the population set to verify that it holds sufficiently well. And as discussed above, the proportion of examples of each document category and of each word profile can differ between the two document sets.

The methodology is also considerably easier to use in practice. Applying the standard supervised learning approach is difficult, even if we meet its assumptions. Even if we forget about choosing the "true" model, merely

finding a "good" specification with thousands of explanatory variables to choose from can be extraordinarily time consuming. One needs to fit numerous statistical models, consider many specifications within each model type, run cross-validation tests, and check various fit statistics. Social scientists have a lot of experience with specification searches, but all the explanatory variables mean that even one run would take considerable tuning and many runs would need to be conducted.

The problem is further complicated by the fact that social scientists are accustomed to choosing their statistical specifications on the basis of prior theoretical expectations and results from past research, whereas the overwhelming experience in the information extraction literature is that radically empirical approaches work best for a given amount of effort. For example, we might think we could carefully choose words or phrases to characterize particular document categories (e.g., "awful," "irresponsible," "impeach," etc., to describe negative views about President Bush), and indeed this approach will often work to some degree. Yet, a raw empirical search for the best specification, ignoring these theoretically chosen words, will typically turn up predictive patterns we would not have thought of ex ante. Indeed, methods based on highly detailed parsing of the grammar and sentence structure in each document can also work exceptionally well (e.g., King and Lowe 2003), but the strong impression from the literature is that the extensive, tedious work that goes into adapting these approaches for each application is more productively put into collecting more hand-coded examples and then using an automatic specification search routine.

# The Method in Practice

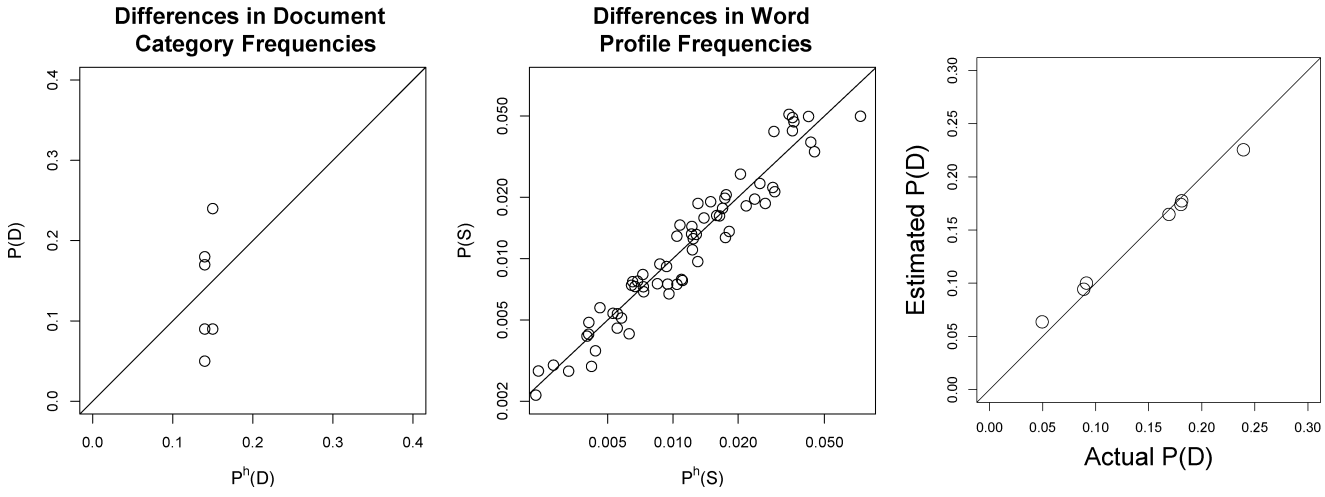
We begin here with a simple simulated example, proceed to real examples of different types of documents, and study how many documents one needs to hand code. We also compare our approach to existing methods and discuss what can go wrong. Readers can replicate and modify any of these analyses using the replication files made available with this article.

## Monte Carlo Simulations

We begin with a simulated data set of 10 words and thus $2^{10} = 1{,}024$ possible word-stem profiles. We set the elements of $P^h(D)$ to be the same across the seven categories, and then set the population document category frequencies, $P(D)$, to very different values. We then draw a value $\tilde{D}$ from $P^h(D)$, insert the simulation into $P^h(\boldsymbol{S} \mid \tilde{D})$, which we set to that from the population, and then draw the simulated matrix $\tilde{\boldsymbol{S}}$ from this density. We repeat the procedure 1,000 times to produce the labeled data set, and analogously for the population.

The left two panels of Figure 2 summarize the sharp differences between the hand-coded and population

**FIGURE 2 Accurate Estimates Despite Differences Between Labeled and Population Sets**



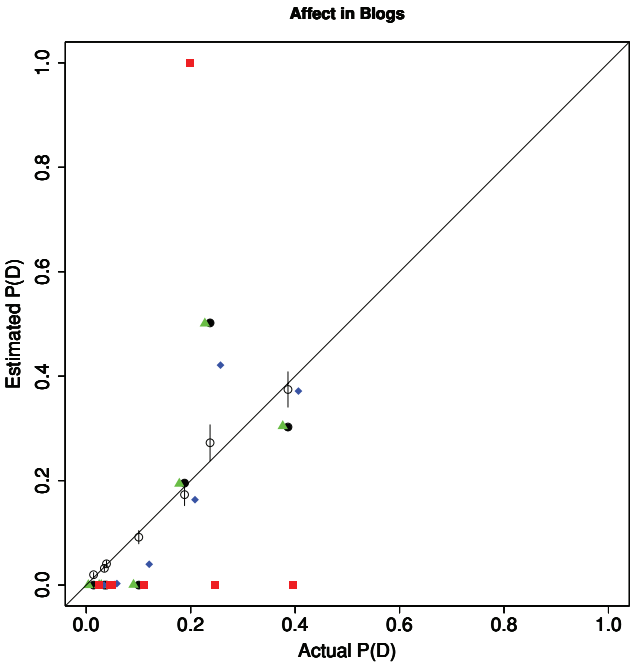

*Notes:* For both $P(D)$ on the left and $P(S)$ in the center, the distributions differ considerably. The direct sampling estimator, $P^h(D)$, is therefore highly biased. Yet, the right panel shows that our nonparametric estimator remains unbiased.

distributions in these data. The left graph plots $P^h(D)$ horizontally by $P(D)$ vertically, where the seven circles represent the category proportions. If the proportions were equal, they would all fall on the 45° line. If one used the labeled, hand-coded sample in this case via direct sampling to estimate the document category frequencies in the population, the result would not even be positively correlated with the truth.

The differences between the two distributions of word frequency profiles appear in the middle graph (where for clarity the axes, but not labels, are on the log scale). Each circle in this graph represents the proportion of documents with a specific word profile. Again, if the two distributions were the same, all the circles would appear on the diagonal line, but again many of the circles fall off the line, indicating differences between the two samples.

Despite the considerable differences between the labeled data set and the population, and the bias in the direct sampling estimator, our approach still produces accurate estimates. The right panel of the figure presents these results. The actual $P(D)$ is on the horizontal axis and the estimated version is on the vertical axis, with each of the seven circles representing one of the document frequency categories. Estimates that are accurate fall on the 45° line. In fact, the points are all huddled close to this equality line, with even the maximum distance from the line for any point being quite small.

**FIGURE 3 Out-of-Sample Validation**




*Notes:* The plot gives the estimated document category frequencies (vertically) by the actual frequencies (horizontally). Our nonparametric approach is represented with black open circles, with 95% confidence intervals as vertical lines. Aggregated optimized SVM analyses also appear for radial basis (black dots), linear (green triangles), polynomial (blue diamonds), and sigmoid kernels (red squares). Estimates closer to the 45° line are more accurate.

## Empirical Evidence

We now offer several out-of-sample tests of our nonparametric approach in different types of real data. Our first test includes the 4,303 blog posts that mention George W. Bush. (For levels of intercoder reliability for this task, see the appendix.) These posts include 47,726 unique words and 3,165 unique word stems. We randomly divide the data set in half between the training set and test set and, to make the task more difficult, then randomly delete half (422) of the posts coded −2 in the test set. Our test set therefore intentionally selects on (what would be considered, in standard supervised learning approaches) the dependent variable. The results from our nonparametric estimator appear in Figure 3 as one open circle for each of the seven categories, with 95% confidence intervals appearing as a vertical line. Clearly the points are close to the 45° line, indicating approximately unbiased estimates, and all are within the 95% confidence intervals.

Also plotted on the same graph are the document category proportions aggregated up from individual classifications given by four separately optimized support vector machine (SVM) classifiers, the most widely used (and arguably the best) of the existing methods. These include SVMs using a radial basis function (black dots), linear kernel (green triangles), polynomial kernel (blue diamonds), and sigmoid kernel (red squares) (Brank et al. 2002; Hastie, Tibshirani, and Friedman 2001; Hsu, Chang, and Lin 2003; Joachims 1998). As can be seen in the graph, these results vary wildly and none do as well as our approach. They are plotted without confidence intervals since SVM is not a statistical method and has no probabilistic foundation. An additional difficulty of using individual classifiers is the highly time-intensive tuning required. Whereas the results from our approach represent only a single run, we followed the advice of the SVM literature and chose the final four SVMs to present in Figure 3 by optimizing over a total of 19,090 separate SVM runs, including cross-validation tests on 10 separate subsets of the labeled set. One run of our nonparametric estimator took 60 seconds of computer time, or a total of five hours for 300 bootstrapped runs. The SVM runs

**TABLE 1 Performance of Our Nonparametric Approach and Four Support Vector Machine Analyses**

| | Percent of Blog Posts Correctly Classified | | | |
|---|---|---|---|---|
| | In-Sample Fit | In-Sample Cross-Validation | Out-of-Sample Prediction | Mean Absolute Proportion Error |
| Nonparametric | — | — | — | 1.2 |
| Linear | 67.6 | 55.2 | 49.3 | 7.7 |
| Radial | 67.6 | 54.2 | 49.1 | 7.7 |
| Polynomial | 99.7 | 48.9 | 47.8 | 5.3 |
| Sigmoid | 15.6 | 15.6 | 18.2 | 23.2 |

*Notes:* Each row is the optimal choice over numerous individual runs given a specific kernel. Leaving aside the sigmoid kernel, individual classification performance in the first three columns does not correlate with mean absolute error in the document category proportions in the last column.

took approximately 8.7 days (running 24 hours/day) on a powerful server and much more in human time.

We give an alternative view of these results in Table 1. The first three numerical columns report individual classification performance whereas the last gives the mean absolute error in the document category proportions. The last column confirms the overall impression from Figure 3 that the nonparametric method has much lower error in estimating the document category proportions. Leaving aside the sigmoid kernel, which did not work well in these data, the SVM results have the familiar patterns for individual classifiers: the models fit best to the in-sample data, followed next by in-sample cross-validation, and lastly by the true out-of-sample predictions. The key result in this analysis is that, even among the SVM analyses, the best individual classifier (the linear kernel) is different from the best choice for minimizing the mean absolute error in the document category proportions (the polynomial kernel). Of course, nothing is wrong with SVM when applied to the individual classification goal for which it was designed.

## Examples from Other Textual Data Sources

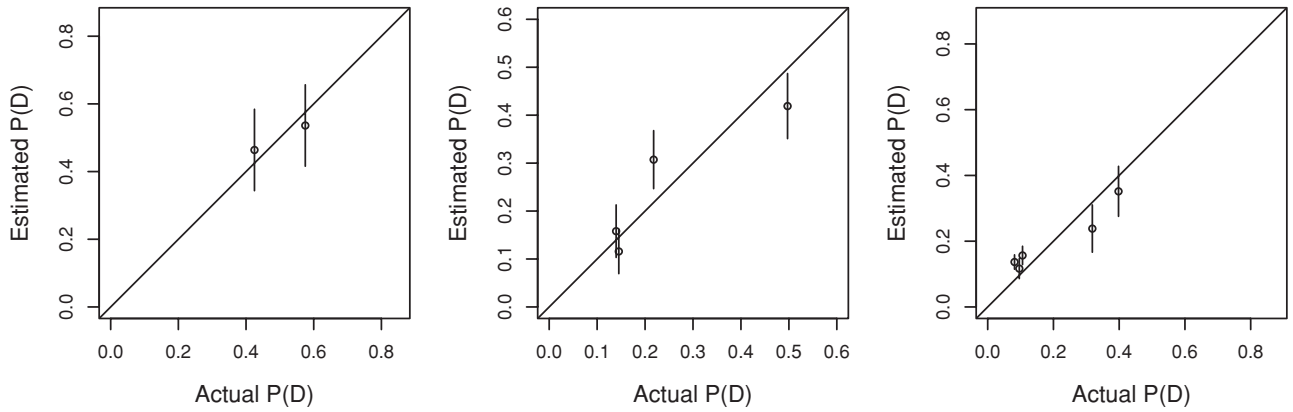
We now give three brief examples applying our method to different sources of unstructured text. The first is from a corpus of congressional speeches used in the computer science literature to evaluate supervised learning methods (Thomas, Pang, and Lee 2006). Researchers selected 3,838 speeches given in the House of Representatives between January 4th and May 12th, 2005, during "contentious" debates, defined as those where more than 20% of the speeches were in opposition. To simulate how a resource-conscious researcher might proceed, we used the 1,887 speeches appearing on even-numbered pages of the congressional record as a training set, and then estimated the distribution of supportive speeches in the test set of 1,951 speeches on odd-numbered pages. The results using the nonparametric estimator appear in the top-left graph in Figure 4 and are again highly accurate.

Another example comes from a data set of 462 immigration editorials that we compiled using Factiva. The editorials appeared in major newspapers between April 1st and July 15th, 2007, and were coded into four nonordered categories indicating editorials supporting the Senate's immigration bill, those opposing it, and two categories that capture letters to the editor and other miscellaneous articles. Here, the training set includes the 283 editorials prior to June 12th, while the test set includes the 179 editorials on or after that date. Deviations from the 45° line are due to slight violations of the assumption in equation (7). This is quite a hard test, since some categories have as few as 40 examples. The small discrepancy can also be fixed easily if this were a real application by adding to the hand-coded set a small number of documents collected over time.

Our final example comes from 1,726 emails sent by Enron employees and classified into five nonordered categories: company business, personal communications, logistic arrangements, employment arrangements, and document editing.[10] To make the task more difficult, we first created a skewed test set of 600 emails that was more uniformly distributed than the training set, with no category accounting for less than 12% or more than 39% of the observations. We then used the remaining 1,126 emails as a mutually exclusive training set where the comparable bounds were 4% and 50%. The results are quite

[10] See http://www.cs.cmu.edu/~enron/.

**FIGURE 4 Additional Out-of-Sample Validation**




*Notes:* The left graph displays the accuracy of the nonparametric method in recovering the distribution of supporting versus opposing speeches in Congress. The center graph shows the same for a categorization of newspaper editorials on immigration, and the right graph shows the distribution across categories of emails sent by enron employees. As before, 95% confidence intervals are represented by vertical lines, and estimates closer to the 45° line are more accurate.

accurate, especially given the paucity of information in many (short) emails, and are displayed in the right panel of Figure 4.

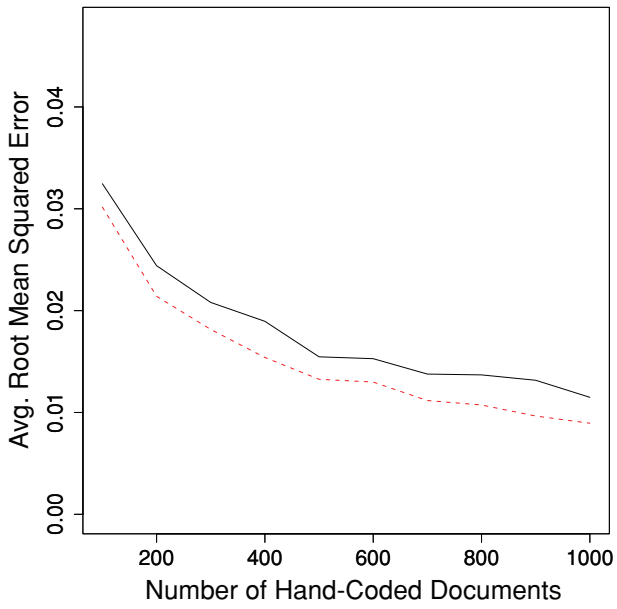
## How Many Documents Need to Be Hand Coded?

Any remaining bias in our estimator is primarily a function of the assumption in equation (7). In contrast, efficiency, as well as confidence intervals and standard errors, are primarily a function of how many documents are hand coded and so are entirely under the control of the investigator. But how many is enough? Hand coding is expensive and time consuming and so we would want to limit its use as much as possible, subject to acceptable uncertainty intervals.

To study this question, we set aside bias by randomly sampling the labeled set directly from the population and plotting in Figure 5 the root mean square error (RMSE) averaged across the categories vertically by the number of hand-coded documents horizontally for our estimator (solid line) and the direct sampling estimator (dashed line). RMSE is lower for the direct estimator, of course, since this sample was drawn directly from the population and little computation is required, although the difference between the two is only about two-tenths of a percentage point.

For our estimator, the RMSE drops quickly as the number of hand-coded documents increases. Even the highest RMSE, with only 100 documents in the labeled set, is only slightly higher than 3 percentage points, which would be acceptable for some applications. (For example, most national surveys have a margin of error of at least 4 percentage points, even when assuming random sampling and excluding all other sources of error.) At about 500 documents, the advantage of more hand coding begins to

**FIGURE 5 Average Root Mean Square Error by Number of Hand-Coded Documents**



suffer diminishing returns. In part this is because there is little more error to eliminate as our estimator then has an average RMSE of only about 1.5 percentage points.

The conclusion here is clear: coding more than about 500 documents to estimate a specific quantity of interest is probably not necessary, unless one is interested in much more narrow confidence intervals than is common or in specific categories that happen to be rare. For some applications, as few as 100 documents may even be sufficient.

## What Can Go Wrong?

We now discuss five problems that can arise with our methods. If they do arise, and steps are not taken to avoid or ameliorate them, they can cause our estimator to be biased or inefficient. We also discuss what to do to ameliorate these problems.

First, and most importantly, our procedure cannot work without reliable information. This requires that the original documents contain the information needed, the hand codings are reliable enough to extract the information from the documents, and the quantitative summary of the document (in $\boldsymbol{S}$) is a sufficiently accurate representation and sufficient to estimate the quantities of interest. Each of these steps requires careful study. Documents that do not contain the information needed cannot be used to estimate quantities of interest. If humans cannot code these documents into well-defined categories with some reasonable level of reliability, then automated procedures are unlikely to succeed at the same task. And many choices are available in producing abstract numerical summaries of written text documents. Although we have found that stemmed unigrams are a sufficient representation to achieve approximately unbiased inferences in our examples, researchers may have to use some of the other tricks discussed in the section entitled "coding variables" for different applications.

Second, a key issue is the assumption in equation (7) that $P(\boldsymbol{S} \mid D)$ is the same in the labeled and population document sets. We thus have much less restrictive assumptions than prior methods, but we still assume a particular type of connection between the two document sets. If we are studying documents over a long time period, where the language used to characterize certain categories is likely to change, it would not be advisable to select the labeled test set only from the start of the period. Checking whether this assumption holds is not difficult and merely requires hand coding some additional documents closer to the quantity presently being estimated and using them as a validation test set. If the data are collected over time, one can either hand code several data sets from different time periods or gradually add hand-coded documents collected over time. In our running example, we are attempting to track opinions over a single presidential campaign. As such, only one hand-coded data set at the start may be sufficient, but we have tested this assumption, and will continue to do so by periodically hand coding small numbers of blogs.[11]

Third, each category of $D$ should be defined so as to be mutually exclusive, exhaustive, and relatively homogeneous. To confront cases where the categories are not mutually exclusive, one can define an additional "both" category. Categories that require many examples to define may be too broad for effective estimation as may occur for residual or catch-all categories. Consider the "NB" category in our data as one example. There are innumerable types of web sites that are not blogs, each with very different language; yet this category was essential since our blog search algorithm was not perfect. In fact, we do find slightly more bias in estimating category NB than the others in our categorization, but not so much as to cause a problem for our applications. Given our experiences, the identification of an effective set of categories in $D$ is an important issue and should involve careful iteration between improving concepts, validation in hand coding tests, and searching for new possibilities in example documents. Intercoder reliability is a crucial metric as well. If human coders cannot agree on a classification, automated approaches are not likely to return sensible results either.

Fourth, our approach requires the choice of the number of word stems to use in each randomly chosen subset. While choosing the number of random subsets is easy (the more the better, and so like any simulation method the number should be chosen based on available computer time and the precision needed), the number of word stems to use in each random subset must be chosen more carefully. Choosing too few or too many will leave $P(\boldsymbol{S})$ and $P(\boldsymbol{S} \mid D)$ too sparse or too short and may result in attenuation bias due to measurement error in $P(\boldsymbol{S} \mid D)$, which serve as the "explanatory variables" in the estimation equation. To make this choice in practice, we use standard automated cross-validation techniques, such as by randomly dividing the labeled set into training and test sets and then checking what works in those data.

[11] To generate a clear example of where this assumption is violated, we divided a test set into subsets based on the sophistication of the language using Flesch-Kincaid scores, which attempt to measure the grade level needed to read a text. We then tried to estimate the document category frequencies from a labeled set that made no such distinctions. Since the language sophistication is computed directly from the document text, equation (7) was violated and our estimates were biased as a result.

In practice, the number of word stems to choose to avoid sparseness bias mainly seems to be a function of the number of unique word stems in the documents. Fixing any problem that may arise via these types of cross-validation tests is not difficult. Given the other recommendations discussed above—stability in $P(\boldsymbol{S} \mid D)$, coding categories that are homogeneous and clearly defined—the choice of the optimal number of subsets can account for many of the performance problems we observe in practice. In some applications, researchers may find it helpful to weight the word stems unevenly, so that words likely to have more information (such as based on their "mutual information") appear more frequently in the subsets, although we have not found this necessary.

Finally, we require a reasonable number of documents in each category of $D$ to be hand coded. Although we studied the efficiency of our procedure as a function of the number of hand-coded documents above, these results would change if by chance some categories had very few hand-coded documents and we cared about small differences in the proportions in these population categories. This makes sense, of course, since the method requires examples from which to generalize. Discovering too few examples for one or more categories can be dealt with in several ways. Most commonly, one can alter the definition of the categories or can change the coding rules.

However, even if examples of some categories are rare, they may be sufficiently well represented in the much larger population set to be of interest to social scientists. To deal with situations like this, we would need to find more examples from these relatively rare categories. Doing so by merely increasing the size of the hand-coded data set would be wasteful given that we would wind up with many more coded documents in the more prevalent categories. Still, it may be possible to use available meta-data to find the needed documents with higher probability. In our blogs data, we could find blog posts of certain types via links from other already hand-coded posts or from popular directories of certain types of blogs. Fortunately, the labeled set is assumed to be generated conditional on the categories, and so no bias is induced if we add extra examples via this "case-control" approach (cf. King and Zeng 2002).

Throughout all these potential problems, the best approach seems to be the radically empirical procedure suggested in the supervised learning literature. If the procedure you choose works, it works; if it doesn't, it doesn't. And so one should verify that the procedures work by subdividing the labeled set into training and (truly out of sample) test sets and then directly testing hypotheses about the success of the procedure. Ideally, this should then be repeated with different types of labeled test sets. The more we make ourselves vulnerable to being wrong, using rigorous scientific procedures, the more we learn. Fortunately, the tools we make available here would seem to make it possible to learn enough to produce a reliable procedure in many applications.

Relatedly, standard errors and confidence intervals take a very different role in this type of research than the typical observational social science work. For most methods, the only way to shrink confidence intervals is to collect more data. For the method introduced here, all a researcher needs to do is to hand code additional documents (selected randomly or randomly conditional on $D$) and rerun the algorithm. As long as no data are discarded along the way, continuing to hand code until one's confidence intervals are small enough induces no bias, since our methodology (like direct sampling) is invariant to sampling plans (Thompson 2002, 286ff). A reasonably general approach is to hand code roughly 200 documents and run the algorithm. If uncertainty is more than desired, then hand code 100 more randomly selected documents, add them to the first set, reestimate, and continue until the uncertainty is small enough.

Given the many possible applications of this method, it is difficult to provide general guidelines about how time-intensive the entire process is likely to be. However, our experience is that identifying clear categories that humans are consistently able to differentiate takes far longer than the automated analyses we propose. Once users have clearly defined categories hand coded for a few hundred documents, they can often estimate the document category proportions for far larger corpora a few hours later.

## Concluding Remarks

Existing supervised methods of analyzing textual data come primarily from the tremendously productive computer science literature. This literature has been focused on optimizing the goals of computer science, which for the most part involve maximizing the percent of documents correctly classified into a given set of categories. We do not offer a way to improve on the computer scientists' goals. Instead of seeking to classify any individual document, most social science literature that has hand- (or computer-) coded text is primarily interested in broad characterizations about the whole set of documents, such as unbiased estimates of the proportion of documents in given categories. Unfortunately, since they are optimized for a different purpose, computer science methods often produce biased estimates of these category proportions.

By developing methods for analyzing textual data that optimize social science goals directly, we are able to considerably outperform standard computer science methods developed for a different purpose. In addition, our approach requires no modeling assumptions, no modeling choices, and no complicated statistical approaches, and lets the social scientist pose the theoretical question to be answered. It also requires far less work than projects based entirely on hand coding, and much less work than most computer science methods of individual classification; it is both fast and can be used in real time. Individual-level classification is not a result of this method, and so it is not useful for all tasks, but numerous quantities of interest, from separate subdivisions of the population or different populations, can be estimated. As with all supervised learning methods, our approach does require careful efforts to properly define categories and to hand code a small sample of texts.

Although we have included only a few applications in this article, the methods offered here would seem applicable to many possible analyses that may not have been feasible previously. With the explosion of numerous types and huge quantities of text available to researchers on the web and elsewhere, we hope social scientists will begin to use these methods, and develop others, to harvest this new information and to improve our knowledge of the political, social, cultural, and economic worlds.

## Appendix Correcting for Lack of Intercoder Reliability

Hand coding is often an error-prone task. Intercoder reliability is measured in many different ways in the literature, but the rates tend to be lower with more categories and more theoretically interesting coding schemes and are almost never perfectly reliable. Unfortunately, "the classical supervised classification paradigm is based on the assumption that there are no errors in the true class labels" (Hand 2006, 9). The problem may be due to "conceptual stretching" (Collier and Mahon 1993) or "concept drift" (Widmer and Kubat 1996) that could in principle be fixed with a more disciplined study of the categories or coder training, but in practice some error is always left. In current practice, scholars typically report some reliability statistics and then use methods that assume no misclassification. Here, we propose to address misclassification via simulation-extrapolation (SIMEX; Cook and Stefanski 1994; Küchenohoff, Mwalili, and Lassaffre 2006).

As an example, before we developed our methods, we had at least two coders categorize each of 4,169 blog posts. In these data, our coders agreed on the classification of 66.5% of the blog posts; they agreed on 71.3% of blog posts among those when both coders agreed the post contained an opinion; and they agreed on 92% of the posts for an aggregated classification of negative, neutral, or positive opinions among posts with opinions. Table 2 gives more detailed information. For any two coders, arbitrarily named 1 and 2, each row gives the probability of coder 2's classification given a particular classification $d$ chosen by coder 1, $P(D_2 \mid D_1 = d)$, with the marginal probability for coder 1 appearing in the last column, $P(D_1)$. The "misclassification" (or "confusion") matrix in this table includes information from all combinations of observed ordered coder pairs.

**Table 2 Intercoder Reliability**

| | −2 | −1 | 0 | 1 | 2 | NA | NB | $P(D_1)$ |
|---|---|---|---|---|---|---|---|---|
| −2 | **.70** | .10 | .01 | .01 | .00 | .02 | .16 | .28 |
| −1 | .33 | **.25** | .04 | .02 | .01 | .01 | .35 | .08 |
| 0 | .13 | .17 | **.13** | .11 | .05 | .02 | .40 | .02 |
| 1 | .07 | .06 | .08 | **.20** | .25 | .01 | .34 | .03 |
| 2 | .03 | .03 | .03 | .22 | **.43** | .01 | .25 | .03 |
| NA | .04 | .01 | .00 | .00 | .00 | **.81** | .14 | .12 |
| NB | .10 | .07 | .02 | .02 | .02 | .04 | **.75** | .45 |

*Notes:* This table presents conditional probabilities for coder 2's classification (in a set of column entries) given a code assigned by coder 1 (corresponding to a particular row), or $\boldsymbol{P(D_2 \mid D_1)}$. For instance, when coder 1 chooses category −2, coder 2 will choose the same category 70% of the time, category −1 10% of the time, and so on across the first row. This matrix is estimated from all 4,169 coding pairs from five coders. The final column denotes the marginal probability that coder 1 placed the blog in each category.

*Intuition.* For intuition, we illustrate our approach by an analogy to what might occur during a highly funded research project as a coding scheme becomes clearer, the coding rules improve, and coder training gets better. For clarity, imagine that through five successive rounds, we have different, more highly trained coders classifying the same set of documents with improved coding rules. If we do well, the results of each round will have higher rates of intercoder reliability than the last. The final round will be best, but still not perfect. If we could continue this process indefinitely, we might imagine that we would remove all misclassification.

Now suppose our estimate of the percent of documents in category 2 is 5% in the first round, 11% in the second, 14% in the third, 19% in the fourth, and 23% in the last round. Following all previously published content analyses, our estimate of the proportion in category 2 would be 23%. This is not unreasonable, but it appears

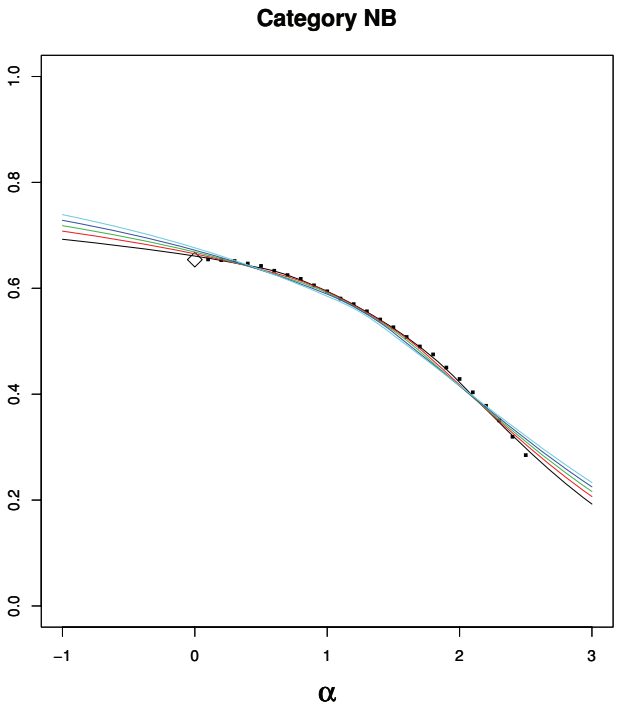
to leave some information on the table. In particular, if the proportion of documents in category 2 is increasing steadily as the level of intercoder reliability at each round improves, then we might reasonably extrapolate this proportion to the point where intercoder agreement is perfect. We might thus conclude that the true proportion in category 2 is actually somewhat larger than 23%. We might even formalize this idea by building some type of regression model to predict the category 2 proportion with the level of intercoder reliability and extrapolate to the unobserved point where reliability is perfect. Since this procedure involves extrapolation, it is inherently model dependent and so uncertainty from its inferences will exceed the nominal levels (King and Zeng 2006). However, a crucial point is that even using the figure from the final round and doing no subsequent processing still involves an extrapolation; it is just that the extrapolation ignores the information from previous rounds of coding. So using 23% as our estimate and ignoring this problem is no safer.

*Formalization.* Following the intuition outlined above, we make use of the misclassifications estimated from a single round of coding with more than one coder, simulate what would have happened to the document category proportions if there were even lower levels of intercoder reliability, and extrapolate back to the point of no misclassification.

To formalize this SIMEX procedure, begin with our estimation method, which would give statistically consistent answers if it were applied to data with no misclassification. The same method applied to error-prone data is presumably biased. However, in this problem, the type of misclassification is easy to characterize, as we do in Table 2. Then we follow five steps: (1) Take each observed data point $D_i$ in the labeled set and simulate $M$ error-inflated pseudo-data points, using the misclassification matrix in Table 2. We do this by drawing $M$ values of $\tilde{D}_i$ from the probability density $P(\tilde{D}_i \mid D_i)$ (given the observed data point $D_i$) which appears in the corresponding row of the table. This step creates $M$ simulated data sets with twice the amount of measurement error, of the same type as in our observed data, to these pseudo-data. We then repeat this procedure starting with these pseudo-data to produce $M$ pseudo-data sets with three times the measurement error as in the original data. Then again with four times the amount of measurement error, etc. (2) We apply our estimator to each of the simulated pseudo-data sets and average over the $M$ results for each level of added error. This leads to a sequence of averaged results from each of the pseudo-estimators, with a different level of intercoder reliability. (3) We transform these data using the multivariate logistic transformation to keep them constrained to the simplex, and then (4) fit a relationship between the transformed average proportion of observations estimated to be in each category from the error-inflated pseudo-data sets and the amount of added error in each. We then (5) extrapolate back to the unobserved point of zero measurement error, and transform the results.

*Illustration.* Figure 6 gives an example of this procedure for one category from our blogs data. The vertical axis of this graph is the proportion of observations in category NB. The horizontal axis, labeled α, gives the number of additional units of misclassification error we have added to the original data, with the observed data at value 0. The estimate of $P(D = \{\text{NB}\})$ from the original data (corresponding to the last round of coding from the earlier example) is denoted with a diamond above the value of zero. A value of α of 1 means that the original data went through the misclassification matrix in Table 2 once; 2 means twice, etc. Some noninteger values are also included. In the application, it seems likely that the

**FIGURE 6 SIMEX Analysis of the Proportion of Documents in Category NB (Not a Blog)**




*Notes:* The estimate from the observed data appears above 0 marked with a diamond; other points are simulated. The goal is to decide on the proportion in category NB at a horizontal axis value of −1.

proportion of documents we would have estimated to be in category NB, if our coders had perfect rates of intercoder reliability, would be higher than the proportion from our actual observed data.

All applications begin with the point estimated from the observed data at zero (marked by a diamond in the figure) and extrapolate it over to the horizontal axis value of $-1$, which denotes the data with no misclassification error. The implicit extrapolation used in prior content analysis research occurs by effectively drawing a flat line from the diamond to the vertical axis on the left. Instead, in Figure 6, estimates from the error-inflated data also appear, as well as several alternative (LOESS-based) models used to form possible extrapolations to the left axis where our estimates appear. In all cases, estimates appear somewhat higher than the nominal (flat line) extrapolation. Differences among the lines indicate uncertainty due to extrapolation-induced model dependence.

## References


Adamic, L. A., and N. Glance. 2005. "The Political Blogosphere and the 2004 U.S. Election: Divided They Blog." *Proceedings of the 3rd International Workshop on Link Discovery*.

Benoit, Kenneth, and Michael Laver. 2003. "Estimating Irish Party Policy Positions Using Computer Wordscoring: The 2002 Election - A Research Note." *Irish Political Studies* 18(1): 97–107.

Berelson, B., and S. de Grazia. 1947. "Detecting Collaboration in Propaganda." *Public Opinion Quarterly* 11(2): 244–53.

Brank, Janez, Marko Grobelnik, Natasa Milic-Frayling, and Dunja Mladenic. 2002. "Feature Selection Using Linear Support Vector Machines." Technical report, Microsoft Research.

Cavnar, W. B., and J. M. Trenkle. 1994. "N-Gram-Based Text Categorization." *Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval*.

Collier, David, and James E. Mahon, Jr. 1993. "Conceptual 'Stretching' Revisited." *American Political Science Review* 87(4): 845–55.

Cook, J., and L. Stefanski. 1994. "Simulation-Extrapolation Estimation in Parametric Measurement Error Models." *Journal of the American Statistical Association* 89: 1314–28.

Das, Sanjiv R., and Mike Y. Chen. 2001. "Yahoo! for Amazon: Opinion Extraction from Small Talk on the Web." Unpublished manuscript, Santa Clara University.

Drezner, Daniel W., and Henry Farrell. 2004. "The Power and Politics of Blogs." Paper presented at the annual meeting of the American Political Science Association.

Gamson, William A. 1992. *Talking Politics*. New York: Cambridge University Press.

Gerner, Deborah J., Philip A. Schrodt, Ronald A. Francisco, and Judith L. Weddle. 1994. "Machine Coding of Event Data Using Regional and International Sources." *International Studies Quarterly* 38(1): 91–119.

Gerring, John. 1998. *Party Ideologies in America, 1828-1996*. New York: Cambridge University Press.

Gilens, Martin. 1999. *Why Americans Hate Welfare*. Chicago: University of Chicago Press.

Ginsberg, Benjamin. 1986. *The Captive Public: How Mass Opinion Promotes State Power*. New York: Basic Books.

Grindle, Merilee S. 2005. "Going Local: Decentralization, Democratization, and the Promise of Good Governance." Cambridge, MA: Kennedy School of Government, Harvard University.

Hand, David J. 2006. "Classifier Technology and the Illusion of Progress." *Statistical Science* 21(1): 1–14.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2001. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer.

Helmond, Anne. 2008. "How Many Blogs Are There? Is Someone Still Counting?" *The Blog Herald* (2/11). http://www.blogherald.com/2008/02/11/how-many-blogs-are-there-is-someone-still-counting/.

Hillygus, Sunshine, and Todd G. Shields. 2008. *The Persuadable Voter: Wedge Issues in Presidential Campaigns*. Princeton, NJ: Princeton University Press.

Hindman, Matthew, Kostas Tsioutsiouliklis, and Judy A. Johnson. 2003. "Googlearchy: How a Few Heavily-Linked Sites Dominate Politics on the Web." Paper presented at the annual meeting of the Midwest Political Science Association.

Hopkins, Daniel, and Gary King. 2009. "Replication Data for: A Method of Automated Nonparametric Content Analysis for Social Science." UNF:3:xlE5stLgKvpeMvxzlLxzEQ== hdl:1902.1/12898 Murray Research Archive [Distributor].

Hsu, C. W., C. C. Chang, and C. J. Lin. 2003. "A Practical Guide to Support Vector Classification." Technical report, National Taiwan University.

Huckfeldt, R. Robert, and John Sprague. 1995. *Citizens, Politics, and Social Communication*. New York: Cambridge University Press.

Joachims, Thorsten. 1998. "Text Categorization with Support Vector Machines: Learning with Many Relevant Features." In *Machine Learning ECML-98*, ed. Claire Nédellec and Céline Rouvierol. Vol. 1398. New York: Springer, 127–42.

Jones, Bryan D., and Frank R. Baumgartner. 2005. *The Politics of Attention: How Government Prioritizes Problems*. Chicago: University of Chicago Press.

Kellstedt, Paul M. 2003. *The Mass Media and the Dynamics of American Racial Attitudes*. New York: Cambridge University Press.

King, Gary, and Will Lowe. 2003. "An Automated Information Extraction Tool for International Conflict Data with Performance as Good as Human Coders: A Rare Events Evaluation Design." *International Organization* 57(3): 617–42. http://gking.harvard.edu/files/abs/infoex-abs.shtml.

King, Gary, and Ying Lu. 2008. "Verbal Autopsy Methods with Multiple Causes of Death." *Statistical Science* 23(1): 78–91. http://gking.harvard.edu/files/abs/vamc-abs.shtml.

King, Gary, and Langche Zeng. 2002. "Estimating Risk and Rate Levels, Ratios, and Differences in Case-Control Studies." *Statistics in Medicine* 21: 1409–27.

King, Gary, and Langche Zeng. 2006. "The Dangers of Extreme Counterfactuals." *Political Analysis* 14(2): 131–59. http://gking.harvard.edu/files/abs/counterft-abs.shtml.

Kolari, Pranam, Tim Finin, and Anupam Joshi. 2006. "SVMs for the Blogosphere: Blog Identification and Splog Detection." American Association for Artificial Intelligence Spring Symposium on Computational Approaches to Analyzing Weblogs.

Krippendorff, D. K. 2004. *Content Analysis: An Introduction to Its Methodology*. Thousand Oaks, CA: Sage.

Küchenohoff, Helmut, Samuel M. Mwalili, and Emmanuel Lassaffre. 2006. "A General Method for Dealing with Misclassification in Regression: The Misclassification SIMEX." *Biometrics* 62 (March): 85–96.

Laver, Michael, Kenneth Benoit, and John Garry. 2003. "Extracting Policy Positions from Political Texts Using Words as Data." *American Political Science Review* 97(2): 311–31.

Lenhart, Amanda, and Susannah Fox. 2006. "Bloggers: A Portrait of the Internet's New Storytellers." Technical report. Pew Internet and American Life Project. http://207.21.232.103/pdfs/PIP%20Bloggers%20Report%20July%2019%202006.pdf.

Levy, P. S., and E. H. Kass. 1970. "A Three Population Model for Sequential Screening for Bacteriuria." *American Journal of Epidemiology* 91: 148–54.

Lyman, Peter, and Hal R. Varian. 2003. "How Much Information 2003." Technical report, University of California. http://www2.sims.berkeley.edu/research/projects/how-much-info-2003/.

Manning, Christopher D., and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. Cambridge, MA: Massachusetts Institute of Technology.

Mayhew, David R. 1991. *Divided We Govern: Party Control, Lawmaking and Investigations*. New Haven, CT: Yale University Press.

Mendelberg, Tali. 2001. *The Race Card: Campaign Strategy, Implicit Messages, and the Norm of Equality*. Princeton, NJ: Princeton University Press.

Mutz, Diana C. 1998. *Impersonal Influence: How Perceptions of Mass Collectives Affect Political Attitudes*. New York: Cambridge University Press.

Neuendorf, K. A. 2002. *The Content Analysis Guidebook*. Thousand Oaks, CA: Sage.

Pang, Bo, and Lillian Lee. 2008. "Opinion Mining and Sentiment Analysis." *Foundations and Trends in Information Retrieval* 2(1): 1–135.

Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan. 2002. "Thumbs Up? Sentiment Classification Using Machine Learning Techniques." *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Porter, M. F. 1980. "An Algorithm for Suffix Stripping." *Program* 14(3): 130–37.

Quinn, K. M., B. L. Monroe, M. Colaresi, M. H. Crespin, and D. R. Radev. 2009. "How to Analyze Political Attention with Minimal Assumptions and Costs." *American Journal of Political Science* 54(1): 209–28.

Rudalevige, Andrew. 2002. *Managing the President's Program*. Princeton, NJ: Princeton University Press.

Sebastiani, Fabrizio. 2002. "Machine Learning in Automated Text Categorization." *ACM Computing Surveys (CSUR)* 34(1): 1–47.

Simon, Adam F., and Michael Xeons. 2004. "Dimensional Reduction of Word-Frequency Data as a Substitute for Intersubjective Content Analysis." *Political Analysis* 12(1): 63–75.

Thomas, Matt, Bo Pang, and Lillian Lee. 2006. "Get Out the Vote: Determining Support or Opposition from Congressional Floor-Debate Transcripts." *Proceedings of EMNLP*. http://www.cs.cornell.edu/home/llee/papers/tpl-convote.home.html.

Thompson, Steven K. 2002. *Sampling*. New York: John Wiley and Sons.

Verba, Sidney, Kay Lehman Schlozman, and Henry E. Brady. 1995. *Voice and Equality: Civic Volunteerism in American Politics*. Cambridge, MA: Harvard University Press.

Waples, D., B. Berelson, and F. R. Bradshaw. 1940. *What Reading Does to People: A Summary of Evidence on the Social Effects of Reading and a Statement of Problems for Research*. Chicago: University of Chicago Press.

Widmer, G., and M. Kubat. 1996. "Learning in the Presence of Concept Drift and Hidden Contexts." *Machine Learning* 23(1): 69–101.

Zaller, John R. 1992. *The Nature and Origins of Mass Opinion*. New York: Cambridge University Press.