Measuring issue salience: Using supervised machine learning to generate data from free responses to the "most important problem" question

Solomon Messing

February 15, 2011

Abstract

Future political communication research will depend on measuring issue-salience in massive data sets largely consisting of unstructured text. We examine a set of supervised machine-learning approaches that can categorize short segments of unstructured text data based on human judgments of content, with an attractive cost structure. We achieve levels of accuracy that are generally acceptable in human-coder content analysis, and which has perfect replicability once human coders produce appropriate training data. We use this approach to classify free responses to the "most important problem" question from the 2000 NAES rolling cross section, and show that our results generally correspond to exogenous real-world events that are widely thought to have shifted the attention of the public.

1 Measuring salience

Of critical importance to many questions in political communication is the ebb and flow of public issue salience leading up to electoral contests. Perhaps the best way to gauge public opinion on current issues is a well-designed survey. However, the most widely used type of survey question—the structured, close-ended response—tends to inject a wide range of biases in measurement. Most notably, closed responses constrain the set of responses. However, the problem of constraining responses is not mitigated by merely adding the option to name another problem outside the set of possible choices: Schuman (2008, 32) split 349 participants into open and closed with an "other" response to the most important problem question, and found that 60% of respondents in the

closed condition selected issues from the set of closed responses, while only 2% of respondents in the open condition chose those issues.

The usual problems with fixed responses are also in play. For example, the order in which fixed responses to a question are presented to a respondent can inject primacy bias in written surveys, or recency bias in the case of verbally administered surveys, a problem that is especially pronounced for respondents with low cognitive sophistication (Krosnick & Alwin, 1987; Holbrook et al., 2007). Additionally, the wording of fixed responses can bias results, and can do so in a way that interacts with the intensity of the respondent's attitude being measured (Krosnick & Schuman, 1988). Differences in question form, for example asking respondents to choose among statements that describe their views on an issue versus asking them to agree or disagree with a series of statements, also produces very different results (Krosnick & Alwin, 1987).

Another way to gather data on public opinion from surveys is to code open-ended responses. Since 1939, when survey research pioneer George Gallup asked a representative sample of American adults: "What do you think is the most important problem facing this country today?" survey researchers have put this question to the American people many times per year, usually in an open-ended format. The debate between administering open-ended versus closed surveys is old. It began in the 1940s, when a rivalry developed within the U.S. government's two survey organizations, the U.S. Polling Division led by commercial pollsters Elmo Roper, Elmo Wilson, and George Gallup, which advocated closed questions which they claimed were cheaper and no less effective, and the Surveys Division, headed by academic social psychologist, Rensis Likert, who advocated open-responses,[1] which he argued allowed respondents to express their opinions more fully (Krosnick & Fabrigar, 1999, 2). The dispute was mediated by sociologist Paul Lazarsfeld, who saw merit in both approaches and eventually suggested collaborative research efforts to take advantage of the relative strengths of each approach. This collaboration failed to materialize, and the closed response approach

[1]This in spite of his work on scaled responses.

has come to dominate the field.

Advocates of open-ended responses have long claimed that such questions have substantial "face validity" (Stouffer, 1955; Schuman, 2008, 30) and bring to light concerns that respondents use in political calculations (Campbell, 1980; Kelley, 1983; Knight, 1985; RePass, 1971). Others point out that because respondents much choose between many options when answering closed questions, they can easily satisfice and choose an answer without expanding the effort to retrieve each issue from memory and weigh the relative importance of each possible response (Krosnick & Fabrigar, 1999, 10). Though respondents may attempt to satisfice when answering an open-ended question as well, offering a "don't know" response is not explicitly legitimized by the question, and doing so violates an important "rule of the game," which respondents are generally hesitant to do (Schuman & Presser, 1996, 299).

Detractors contend that respondents do not probe their memories in sufficient depth to remember all of the information that generated their overall judgment of candidates and issues (Smith, 1989) so results may be biased toward accessibility Higgins et al. (1985), and point out that closed responses can help define a question's frame of reference by making explicit the set of intended possible responses (Krosnick & Fabrigar, 1999, 7). Regardless, the modern dominance of fixed response (close-ended) questions stems from the relative ease of asking, coding and analyzing such responses—and *not* because of any empirical evaluation of the superiority of one over the other in terms of measurement (Geer, 1991; Schuman, 2008). It is, for the most part, a simple cost compromise.

The cost of using free responses stems primarily from the labor associated with human-coding unstructured text in order to generate meaningful data. Since the 1960s, social scientists have proposed utilizing natural language processing (NLP) approaches to code open-ended survey responses (Frisbie & Sudman, 1968), in large part to alleviate these cost burdens. Early approaches were based on counting the occurrence of words associated with particular fixed categories, such as the General Inquirer system

(Stone et al., 1966). However, dictionary methods have high startup costs. To build a dictionary requires a human analyst with subject-matter knowledge to determine which words best represent which constructs of interest, often in a cycle of deductive and inductive approaches, and fine tuning the categories often requires much trial and error (Quinn et al., 2010, 212). This approach also assumes that the dictionary categories capture exactly how respondents will express a particular concept of interest to researchers, and that the words used by respondents will fit exactly in one category—both faulty assumptions.

This problem of textual complexity is not unique to survey responses or even text categorization, and computer scientists have devoted considerable effort to the question of whether machines can "learn" to identify unspecified patterns based on exposure to categorized data. These techniques have been applied to text categorization problems with results as good as human coders (King, 2003), but these approaches have generally been developed and are suitable for documents with many more words than free responses typically contain. Techniques to analyze short units of text, typically short message system (SMS) text messages, generally seek to "detect" phenomena, such as disease outbreaks (Ginsberg et al., 2009; Lampos & Cristianini, 2010) or spam email (Cormack et al., 2007; Hidalgo et al., 2006). Only recently have approaches surfaced in the literature that apply multiple categories to short queries, with results that approach accuracy rates that would be acceptable in human-coder content analyses (Munro & Manning, 2010, categorizing medical questions). However, attempts to take advantage of these machine-learning techniques for categorizing relatively short free responses have mostly not been able to achieve accuracy on par with human coders (for example, Giorgetti & Sebastiani, 2003).

Below, we examine the suitability of a series of pre-processing techniques and supervised machine learning algorithms in categorizing free responses to the "most important problem" question, in an attempt to shed light on this ebb and flow of issue salience.

2 Design

We first investigate which machine learning techniques best categorize free responses to the "most important issue" question, then categorize free responses from the 2000 National Annenberg Election Study (NAES), and then examine examine the proportion of Democrats, Republicans and other citizens who name owned issues as most important. We also make use of the rolling cross sectional design of the NAES to examine the proportion of citizens naming owned issues as most important over time. This lays the groundwork for more fine-grained examinations of the relationship between campaign communications, media appearances, and issue salience, and the way in which these phenomena impact the extent to which opposing campaigns converge and diverge in terms of issue coverage.

2.1 Data

Our examination of issue salience relies on categorizing free-responses from the 2000 National Annenberg Election Study (NAES) rolling cross sectional questionnaires. The data covers the entire presidential campaign period, with data collected over a period from December 14, 1999 to January 11, 2001. The NAES selected respondents based on a random digit dialing (RDD) design in which telephone numbers were selected at random and respondents completed the survey via telephone. The NAES interviewed a total of 58,370 respondents for the data under consideration here. During the period in question, there were "an average of 50 to 300 interviews conducted each day" according to the NAES codebook.

The wording of the "important issue" question follows: "In your opinion, what is the most important problem facing our country today?" Responses to this question were transcribed by human analysts and tagged for a variety of peculiarities, including providing more than one problem per response, elaborating on the scope an issue previously raised in the response, and/or providing specific detail about a problem (among

others). A random check of 500 responses revealed three responses that contained spelling errors. There were a total of 23,069 unique responses (out of 58,370), of which 4863 were blank. A (distinct) sample of 500 random responses was hand-coded by the authors for the purpose of training supervised machine learning classifiers, using a codebook derived from Gallup poll issue categories (Jones, 2010; Newport, 2010). The codebook appears in Table 1.

Table 1: Codebook for training classifiers

Category	Explanation
ECON	Economy (general)
CRIM	Crime
DRUG	Drugs
ENV	Environment
JOBS	Jobs, unemployment
GOVT	Dissatisfaction w/ government
IMMG	Immigration
HLTH	Healthcare
NDR	Natural disaster response, relief
BUD	Budget, taxes
GAS	Fuel, gas prices
POV	Poverty, lack of money
EDU	Education
ETH	Ethics, morals, lack of religious faith
WAR	War (general)
IRAQ	Situation in Iraq
AFGH	Situation in Afghanistan
OTH	Other
NA	Blank

2.2 Categorizing Free Responses

In order to convert these free responses into categorical data, we first compare the accuracy of a variety of supervised machine learning techniques. The first step in any such analysis is to convert the raw text into some kind of quantitative data. The standard approach is to disregard word order and simply compile word counts over

some unit of text such as a sentence, paragraph, or single free-response, which we will refer to as document from here on. This is commonly referred to as the "bag of words" approach and assumes that word order adds little information about the topic of the document (Manning et al., 2008). We use the R package `tm` to create this frequency data for each document (Meyer et al., 2008), which represents each free-response as a 1 x N matrix, where N is the total number of unique words in the entire corpus of free-responses. Prior to creating this frequency data, we employ a variety of techniques that serve to "clean" the text, represent words with the same or similar meaning as single features, and remove potential noise from the resulting data. The first is to ignore punctuation and capitalization. We also extract all of the tags and set them aside as features (variables) to be used later. We also transform certain key multi-word symbols such as "U.S.," "u.s.," and "usa" to some single token such as "usa."

Next, we compare the impact and utility of a variety of slightly more aggressive textual pre-processing techniques. The first transformation is to remove common stop-words that carry little meaning such as "the," "and," "for," etc. Second, we examine the impact of stemming, which collapses sets of words that differ slightly in spelling but represent the same general concept into a single token. For example, "calculate," "calculated," "calculating," and "calculates" would be transformed so that they are all represented by the token "calculat." Perhaps the most common algorithm to accomplish stemming is Porter-stemmer algorithm (Porter, 1980), which is what we employ here. Of course, terms like "calculating" can mean quite different things depending on context, which can serve to confound any machine classifier (hence we evaluate the impact of this technique before using it in the analysis). We also examine the impact of including unigrams, bigrams, and trigrams, or counting the frequency of not only single words but also groups of two and three words.

We also examine the impact of applying term-frequency, inverse-document-frequency (tf-idf) weighting. The tf-idf weighting scheme serves to emphasize the relative weight of words that are frequent in the document under consideration but that are rare in

other documents in the corpus. Such words often contain the most information about the topic of the document. The formula for weighting each term, i, in each document, j, follows:

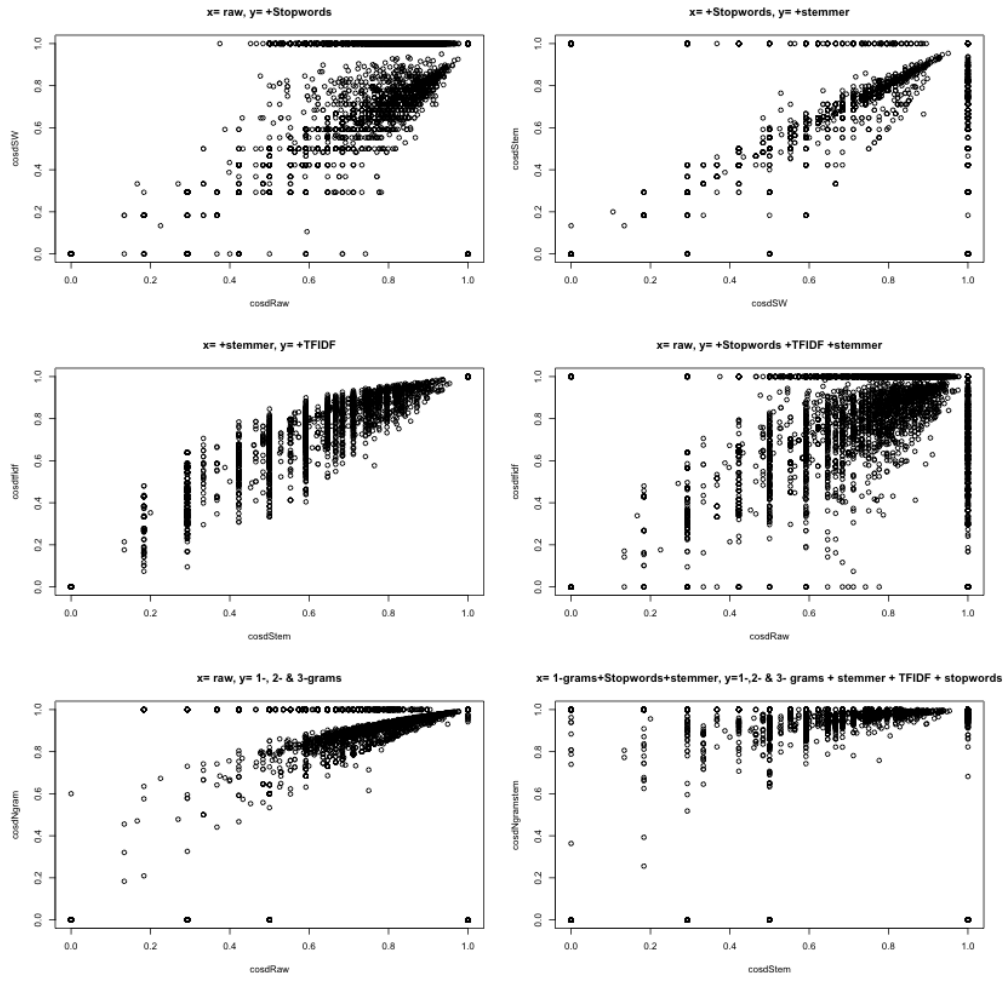
$$TFIDF_{ij} = \frac{n_{ij}}{N_j} * log[(\frac{d_j : t_i \in d_j}{D})^{-1}] \quad (1)$$

where n_{ij} is the frequency of term i in document j, which is normalized by N_j, the sum of all terms in document d_j (normalized term-frequency); and $d_j : t_i \in d_j$ is the number of documents in which term i occurs, which is log-normalized by the total number of documents, D (log-normalized inverse document frequency).

In order to illustrate the impact of these pre-processing techniques on actual free-response survey data, below we compare how the distribution of words differs after the application of each technique, and later examine the differences in accuracy between each technique. After computing each of the transformations described above, we compute the cosine distance of each document in the corpus to every other document. Then, we simply plot the distance matrices against each other as shown in Figure 1. The further a dot appears from the center, the more different the documents are across the transformation in question. As Figure 1 demonstrates, the most radical changes occur when including bigrams and trigrams, which appears to increase the distance between every document. Stemming appears to decrease the distance between (some) documents, which is as expected. The td-idf weighting transformation impacts cross-document distances quite a bit, but not apparently in favor of one direction or another. Removing stopwords also appears alter the similarity of a wide range of documents, but not necessarily in favor of one direction.

In addition to looking at how the distribution of words changes cosine distances between document vectors, we would like to get a sense of how each of these transformations affects the accuracy of supervised machine learning classifiers. Supervised learning classifiers fit a statistical model to a set of training data, and then use that

Figure 1: Cosine distances between documents



model to predict/classify previously unseen data. In our case, the features in the training data consist of counts of each word and the response variable is our topical category.[2] We proceed by predicting each human-coded category as the response variable in a model that features every word in the corpus of responses. In our bag-of-words approaches, this comes out to about 300-600 features (words) depending on the type of pre-processing. In our unigram, bigram and trigram models, this comes out to 2500-3000 features, again, depending on preprocessing. We also include the tags extracted

[2]We are concerned here with supervised approaches that classify text based on pre-defined categories, because the existing literature analyzes the most important issues question by constructing researcher-defined categories, and we would like our analysis to be comparable.

earlier and include them in the analysis as features.[3]

Let us now turn to the question of which pre-processing method best predicts the hand-coded categories. We start by fitting a model to our sample of 500 hand-coded documents, predicting document category based on all of the features (word and tag frequencies) using a support vector machine (SVM), a commonly used machine learning algorithm used to predict categories. The SVM provides an estimate of the probability that each response belongs to a particular category and chooses the response with the highest probability.[4] Because we use so many features, there is a strong risk that our model will overfit the the data. In fact, a quick glance at model fit statistics suggests that the application of an SVM to our entire raw data matrix correctly classifies 99% of our training cases, yielding a kappa statistic of .99, which in light of the existing computer science literature on machine learning tells us that overfitting is quite likely. In order to provide a more conservative indication of how well the model classifies unseen text, we utilize 10-fold cross-validation, in which we fit the model to 90% of the training data and evaluate how well it predicts the remaining 10% of the training data. We do this one time for each set of 10% of our data and average the accuracy (and any other cross-validation statistics) over the set of 10 results. By evaluating the classifier's performance on "unseen" data, 10-fold cross validation helps avoid over-fitting.

We assess each method with standard machine-learning statistics. First, we provide accuracy, which is simply the percent of out-of-sample cases that the model predicts correctly. We provide the false positive rate, or the fraction of out-of-category examples that the model predicts as in-category, and the false negative rate, or the fraction of of in-category cases predicted as out-of-category. We provide precision, the fraction of cases that actually turn out to be in-class that the model predicts to be in-class, and recall, the fraction of in-category examples correctly predicted by the classifier. We

[3]These features added about 2% to the accuracy of the SVM classifier when classifying stemmed documents.

[4]An additional analytical possibility is to only accept responses that are above a given cutoff, which would likely serve to increase accuracy, but would risk yielding categories with cases not missing at random.

also provide the F-measure, which summarizes precision and recall:

$$F = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \quad (2)$$

Table 2 shows the 10-fold cross validated accuracy (percent correctly predicted) and kappa statistic for each pre-processing method.[5] From this table it is easy to see that we get the best performance using unigrams combined with stemming. In contrast to the majority of the computer science literature on document categorization, removing stopwords hinders accuracy here, which may be due to the relative brevity of free responses compared to the news-article-length documents that are usually the topic of computer science studies.[6] Furthermore, while the SMS spam-prediction literature usually shows a large performance gain when including bigrams and trigrams (Cormack et al., 2007; Hidalgo et al., 2006), we see a dramatic decline in performance. We tried to filter out the noise that bigrams and trigrams apparently introduce by eliminating all features occurring in more than 95% of the data and less than the bottom quartile, though other procedures may result in better performance. It may be that there are a few key bigrams and trigrams that strongly predict a spam SMS message, but which are useless or worse when attempting to predict multiple topical categories.

Table 2: Comparing pre-processing procedures

	Accuracy	Precision	Recall	F	ROC	FPR	FNR
raw	0.80	0.85	0.72	0.77	0.89	0.02	0.28
stopwords	0.79	0.86	0.71	0.77	0.88	0.02	0.29
stem	0.82	0.87	0.77	0.81	0.92	0.01	0.23
tfidf	0.72	0.80	0.64	0.70	0.90	0.02	0.36
ngram	0.31	0.30	0.15	0.16	0.55	0.06	0.85
ngramstem	0.28	0.28	0.11	0.10	0.52	0.07	0.89

Having examined various pre-processing strategies, we move on to examine the

[5]Note that the "other" category has been removed from this analysis to focus only on the relevant categories (10-fold cross-validated accuracy decrease by about 4% if this ambiguous category is included in the analysis).

[6]Of course, it is possible that other algorithms might perform differently using different pre-processing procedures, but preliminary testing using the models explored below indicates that this is not the case.

suitability of various supervised machine learning algorithms for categorizing free responses. We evaluate the following algorithms:

Support vector machines (SVMs) estimate the maximal margin hyperplane that best separates data features into a set of binary categories (in the case of linear SVM). Usually an SVM model is estimated via standard Lagrange multiplier methods, but here we utilize sequential minimal optimization (SMO), which optimizes a series of small quadratic programming problems, which is suitable for sparse feature sets such as word frequency counts (Platt, 1998). The procedure can be made more flexible by enlarging the feature space using basis expansions such as polynomials (after which the separation is no longer linear) (Hastie et al., 2009, 423), though below we only include results from the linear SVM, which exhibited the best performance. We extend SVM to our multi-category problem by building a classifier for each pair of categories and choose the one that most often dominates (Friedman, 1996; Hastie & Tibshirani, 1998; Hastie et al., 2009, 438).[7] SVM works very well with high-dimensional data and has been empirically shown to perform well on a variety of real-world problems (Tan et al., 2005; Hastie et al., 2009, 457).

AdaBoost iteratively runs a base classifier over weighted distributions of the training data, then combines the results into a single composite classifier. It emphasizes hard-to-classify cases by down-weighting correctly classified cases at each round. We use SVM as the base classifier here (Freund & Schapire, 1996; Tan et al., 2005).

Naïve Bayes, which estimates the class-conditional probability by assuming that all features are conditionally independent given the category. This can be formalized as follows:

$$P(\mathbf{X}|Y) = \frac{P(Y)\prod_{i=1}^{d} P(X_i|Y)}{P(\mathbf{X})} \tag{3}$$

[7]Another way to extend SVM to multi-category problems is to estimate this hyperplane separately for each category versus the rest, determine the probability that a given case belongs to each category, then select the category with the highest probability (Tan et al., 2005).

where $\mathbf{X}$ consists of all d features. Estimation for qualitative (categorical) features is straightforward, and can be easily calculated for quantitative (continuous) features if we are willing to make a distributional assumption (usually Gaussian). The classifier is robust to noise points, missing values, and irrelevant features (because $P(X_i|Y)$ is approximately normally distributed). However, correlated features can degrade the performance of naïve Bayes classifiers due to the violation of conditional independence (Tan et al., 2005), which is certainly a concern for text data.

MaxEnt, short for maximum entropy, uses multinomial logistic regression to predict categories based on a feature set, utilizing a maximum likelihood approach for estimation. Unlike naïve Bayes classifiers, MaxEnt does not assume that features are conditionally independent from each other. However, logistic regression models are mainly used to understand the role of a parsimonious set of the features in explaining the outcome (Hastie et al., 2009, 121), in this case a category label. As implemented here, the specification of our MaxEnt model is built based on ridge estimation (Cessie & Houwelingen, 1992).

Table 3 shows 10-fold cross validation results for various machine learning algorithms. As predicted in the medical outbreak (Lampos & Cristianini, 2010) and spam email (Cormack et al., 2007; Hidalgo et al., 2006) literature, SVM performs best in categorizing these short units of text. The SVM results here utilize a simple linear support vector. Basis expansions including second through ninth degree polynomials only served to degrade the results. The meta-algorithm, AdaBoost, performs no better than its SVM base algorithm, and requires significantly more resources to fit. Our naïve Bayes and MaxEnt classifiers did not perform as well as SVM.[8] It is worthwhile to point out that our MaxEnt classifier, which uses ridge regression for feature selection, did only slightly better than our naïve Bayes classifier, which requires significantly less

[8]We also examined the Hillard-Purpura classifier, which "votes" between the results of SVM, boosting, and MaxEnt results, but results were slightly worse than SVM alone. Stephen Purpura recommended instead an approach wherein we build a cost function that is optimized for the classification task and the SVM algorithm, which he noted is similar to re-parameterizing a model.

resources to fit.

Table 3: Comparing machine learning algorithms for categorization

	Accuracy	Precision	Recall	F	ROC	FPR	FNR
svm	0.82	0.87	0.77	0.81	0.92	0.01	0.23
adaboostsvm	0.82	0.87	0.75	0.80	0.87	0.02	0.25
naivebayes	0.74	0.75	0.66	0.69	0.93	0.02	0.34
maxentlog	0.74	0.75	0.72	0.73	0.90	0.02	0.28

Having examined a variety of preprocessing techniques and classifiers, we now classify the free responses from the NAES "most important issues" question using stemmed unigrams and an SVM classifier. It is worthwhile to note that by necessity we limit the features of the NAES data to those which occur in our random sample of 500 hand-coded responses. These features consist of counts of tags and single words, of which there are 498 after applying stemming.

Table 4 anecdotally suggests that the classifier is performing reasonably well. Out of a random sample of 20 responses, only responses 5 and 17 are clearly misclassified, which is roughly consistent with our cross-validated accuracy measure (blank rows correspond to 'NA" values in our data). The proportion of responses categorized in each category are presented in Table 5.

3 Results

With our free-responses classified according to Gallup Poll categories, we turn to our actual substantive questions. First we address the extent to which partisan citizens care about issues that their parties own. Figure 2 plots the proportion of respondents stating that each problem was most important, by five point partisan identification, and shows that a large plurality of respondents were concerned with problems related to ethics and morals, with government, crime, education and healthcare relatively close behind. Furthermore, the figure confirms that Republicans and Democrats do indeed

Table 4: Random sample of 20 responses, machine-categorized

	Category	Transcribed Text
1	CRIM	crime
2	CRIM	crime
3	GOVT	politicians
4	ETH	pills senior citizens
5	ETH	employment
6	ETH	lack of leadership
7	EDU	high schools
8		
9	CRIM	drug addiction it leads to other crime no
10	EDU	education
11	ETH	giving kids values
12	GOVT	social security
13	GOVT	the presidential election
14	EDU	school
15	ETH	troubled children
16	CRIM	violence
17	GOVT	computers I wish that they would do away with the computers because of the automated systems you could never get through to talk to a live person
18	HLTH	health care
19	ETH	turning away from god
20	HLTH	health care

disproportionately care about issues owned by their party, but also suggests that the difference in proportions is quite narrow. Figure 3 plots the actual difference in the proportion of Democrats and Republicans (pooled over strong and weak partisans) who report each problem to be most important. Again, though the pattern does show the expected pattern in which partisans care about issues that their parties own, the differences by party are quite narrow. The same pattern can be seen in Figure 4, which plots each issue as a function of the proportion of Democrats and Republicans reporting it as most important.

Consider the context of these results: the nation was still recovering from the Monica Lewinskey scandal, the economy, driven by the technology bubble, was roaring, and

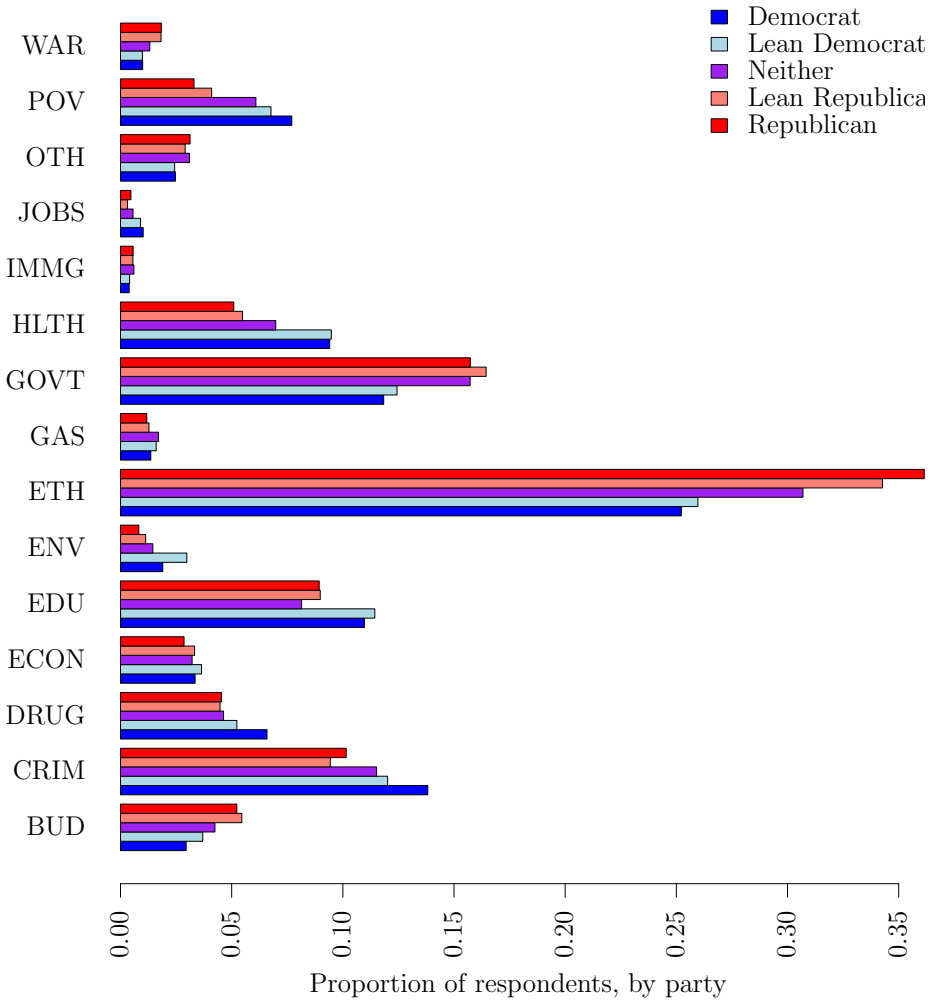
Table 5: Proportion of responses per category

Category	Proportion
ETH	0.30
GOVT	0.14
CRIM	0.12
EDU	0.10
HLTH	0.07
POV	0.06
DRUG	0.05
BUD	0.04
ECON	0.03
OTH	0.03
ENV	0.01
WAR	0.01
GAS	0.01
JOBS	0.01
IMMG	0.01

despite relatively minor conflicts in the former Yugoslavia in 1991-1995, the bombing of Serbia in 1999, American security concerns were minimal. Though of course there was a rise in security concerns in the wake of the escalating Middle East violence and U.S.S. Cole bombing in October 2000.

To examine how issue salience change over time, we take issues that Republicans and Democrats are widely thought to own and examine the dynamics of how public concern with these issues change over the course of the campaign. For Republicans, we examine ETH (ethics and moral issues), BUD (spending and taxes), and WAR (military issues); for Democrats, we examine POV (concern for the poor), HLTH (healthcare), EDU (education), and JOBS (unemployment and job creation). Figure 5 shows a daily time series of the proportion of respondents who stated that these problems were the most important, smoothed with a 30-day moving average (the 30 first and last days are removed from the data). Healthcare and education both see a dramatic rise leading up to the election, which coincides with presidential campaigns that emphasized these issues. Likewise, we see a dramatic decline in concern over budgetary and tax issues

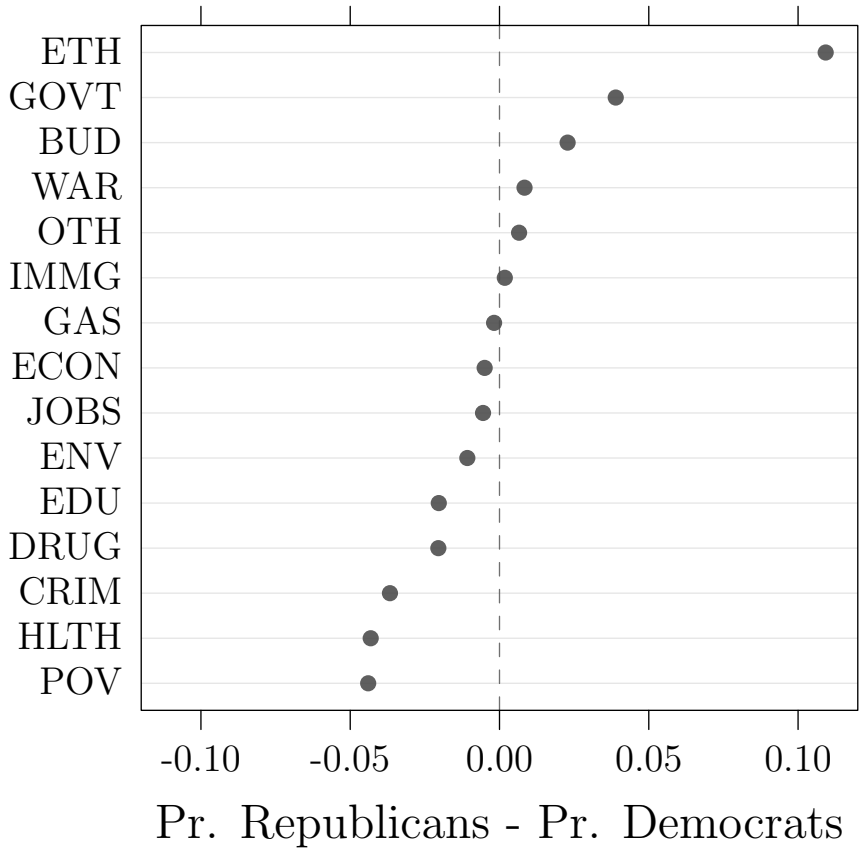
Figure 2: Most important problem, by party



immediately following the election, which coincides with the victory of presidential candidates George W. Bush, who campaigned on tax relief. We also see a dramatic increase in concern for the economy in December 2000, which coincides with a decline in stock prices.

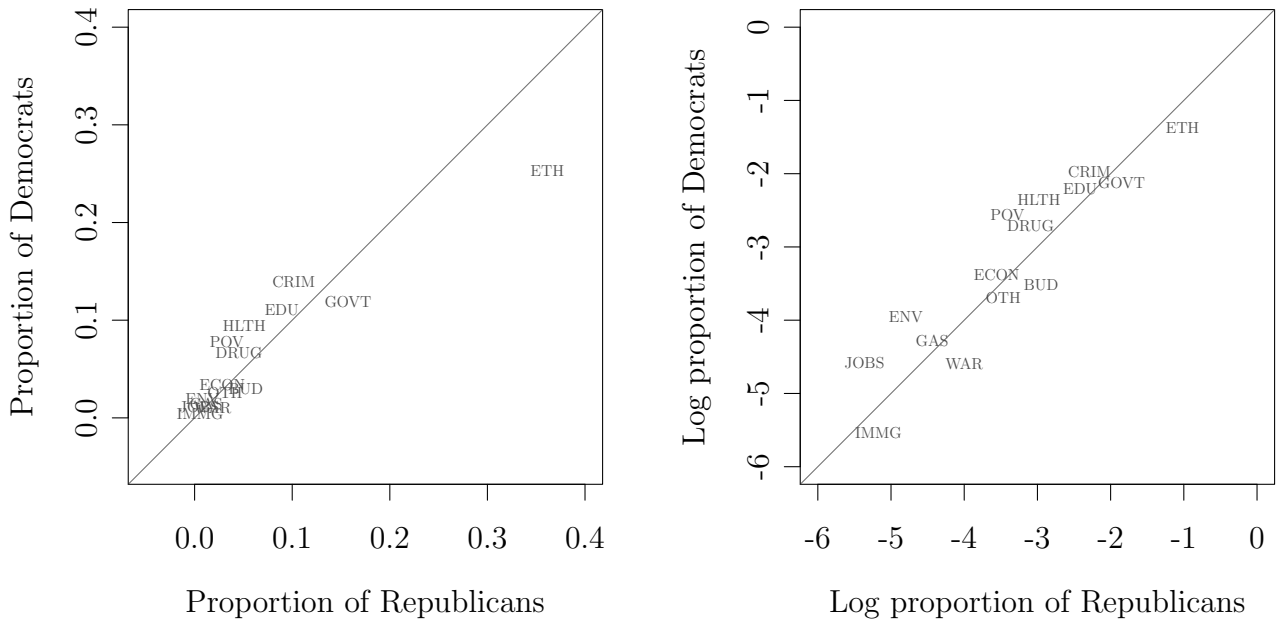
Figure 6 shows the proportion of Democrats and Republicans reporting each problem as most important over time, again smoothed using a 30-day moving average. As we might expect, the proportion of Democrats stating that healthcare is the most im-

Figure 3: Differences in issue importance (Republican on right)



portant issue facing the nation increases more than for Republicans during the height of the presidential campaign. Also, we see a faster drop-off for Republican concerns about education after the campaign. Also, as we saw in Figure 5, Figure 6 shows a sharp rise concerns about international conflict leading up to the Middle East conflict that occured in November 2000, with Republicans reacting slightly more sharply than Democrats.

Figure 4: Most important problem scaled by proportion



4 Discussion

We have examined a series of text pre-processing techniques and supervised machine learning algorithms for the purpose of categorizing free responses to the "most important problem" question. We found that the application of a linear SVM to a stemmed feature set best categorizes survey responses according to the Gallup poll categories. Our accuracy measure for this approach, .82, approaches levels generally deemed acceptable in human-coded data, though ideally we would like to present results comparing agreement between multiple human analysts to that of our classifier.[9] We used this supervised machine learning approach to classify a sample of free responses to the "most important problem" question from the 2000 NAES rolling cross section, and

[9]One way to proceed here is to have three human analysts code training documents and either utilize a "majority rules" voting system and/or arbitrate categories.

Figure 5: Salience of issues over time

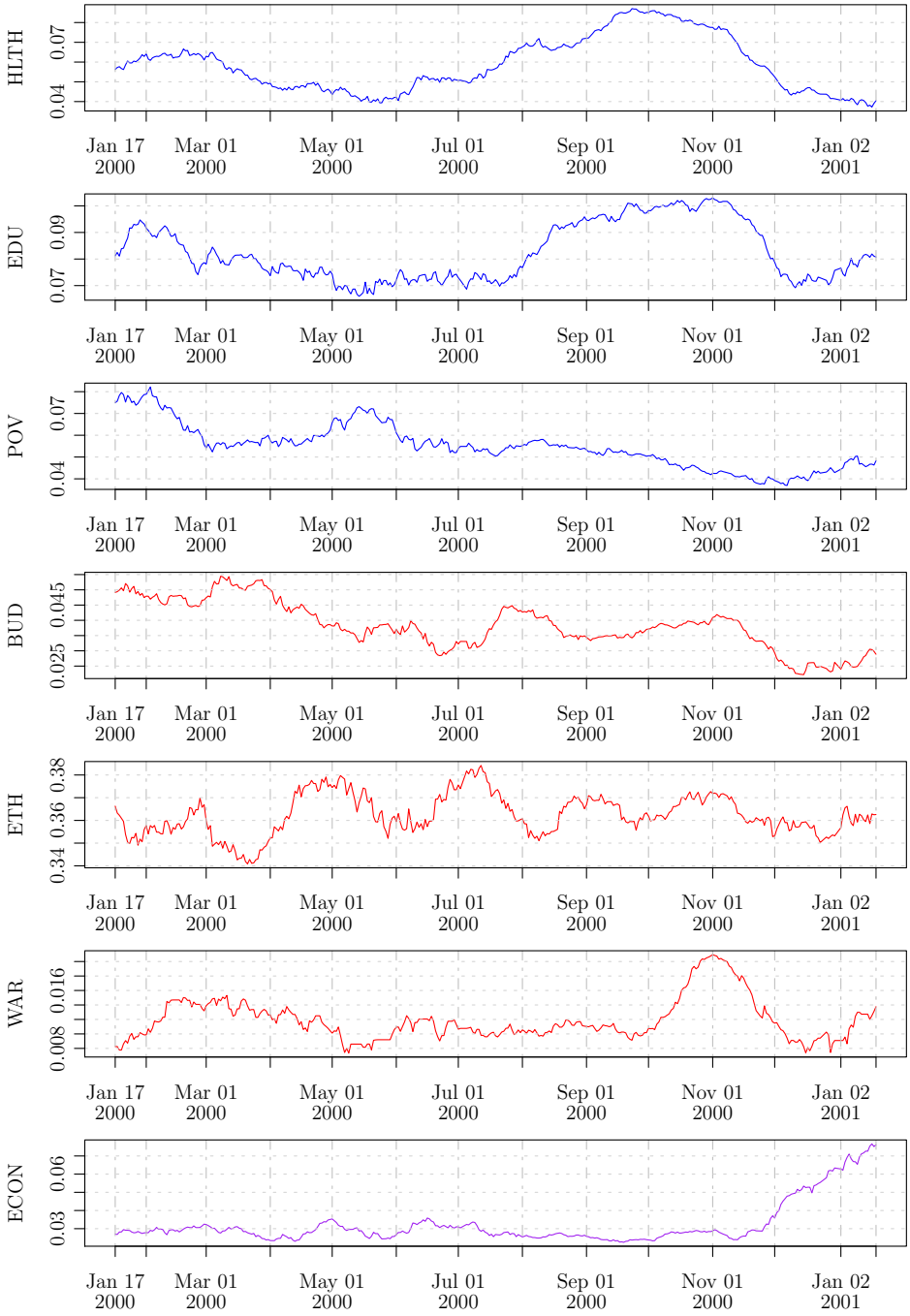
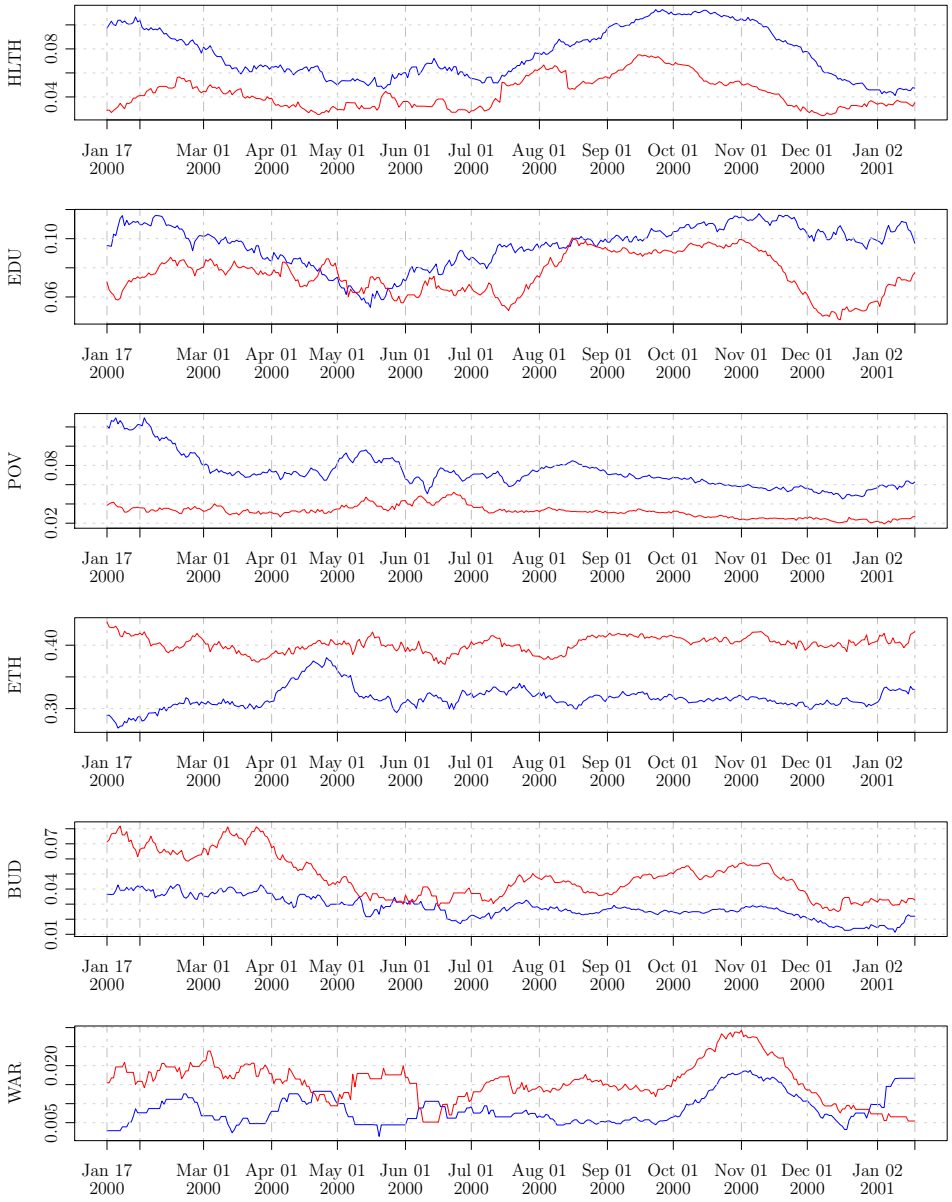


Figure 6: Salience of issues, Democrats blue, Republicans red



showed that our results generally correspond to exogenous real-world events that are widely thought to have shifted the attention of the public.

Our examination of machine approaches to classifying free text responses is by no means complete, and a variety of techniques could be employed to potentially increase the accuracy of our SVM classifier. First of all, we have a limited sample of hand-coded training documents, and larger set would undoubtedly provide the classifier with a richer vocabulary set and result in better classification for unseen documents. We also did not attempt to dichotomize our data features, instead using counts, and it may be that the mere presence or absence of a word is more meaningful in short responses than a full word frequency count. We also did not examine various feature selection techniques, which have been shown to increase accuracy in the literature, such as Lampos & Cristianini (2010) which use Bolasso (bootstrapped LASSO) to extract a consistent set of features from the set of n-grams in the training data. Other approaches use techniques to build a richer set of features, including incorporating morphological and phonological variation (Munro & Manning, 2010), using wordnets to expand the initial set of features from the training data, or using clustering techniques to expand the initial set of data. We also did not try to optimize the SVM cost function (though we did try using non-linear kernels without much effect). Such techniques should be explored in future research on supervised learning approaches to categorizing survey free responses. Another alternative approach is to forgo the SVM approach and model supervised topics explicitly (Blei & McAuliffe, 2008).

Lastly, we did not exmaine fully automated methods, such as topic models, that do not require making any assumptions about categories and have much lower human-coder costs (Quinn et al., 2010). Of course, the trade-off when applying such methods to more than one type of unstructured text corpus is that the categories may not match up, and thus be incomparable. This makes it difficult for example, to examine the relative frequence of a particular issue in terms of free responses versus media content. Nonetheless, it may be possible to create topic models that generate consistent

categories across multiple corpora.

References

Blei, D. M. & McAuliffe, J. (2008). Supervised topic models. *Advances in Neural Information Processing Systems*, *20*, 121–128.

Campbell, A. (1980). *The American voter*. Chicago: University of Chicago Press.

Cessie, S. L. & Houwelingen, J. C. V. (1992). Ridge estimators in logistic regression. *Journal of the Royal Statistical Society - Applied Statistics*, *41*(1), 191–201.

Cormack, G. V., Hidalgo, J. M. G., & Sanz, E. P. (2007). Feature engineering for mobile (SMS) spam filtering. In *Proceedings of the 30th annual international ACM SIGIR conference on research and development in information retrieval*. Amsterdam, The Netherlands, 871.

Freund, Y. & Schapire, R. (1996). Experiments with a new boosting algorithm. In *Machine learning: proceedings of the Thirteenth International Conference (ICML'96)*. 148–156.

Friedman, J. H. (1996). Another approach to polychotomous classification. Technical report, Stanford University.

Frisbie, B. & Sudman, S. (1968). The use of computers in coding free responses. *Public Opinion Quarterly*, *32*(2), 216–232.

Geer, J. G. (1991). Do open-ended questions measure "salient" issues? *Public Opinion Quarterly*, *55*(3), 360.

Ginsberg, J., Mohebbi, M. H., Patel, R. S., Brammer, L., Smolinski, M. S., & Brilliant, L. (2009). Detecting influenza epidemics using search engine query data. *Nature*, *457*(7232), 1012–1014.

Giorgetti, D. & Sebastiani, F. (2003). Automating survey coding by multiclass text categorization techniques. *Journal of the American Society for Information Science and Technology*, *54*(14), 1269–1277.

Hastie, T. & Tibshirani, R. (1998). Classification by pairwise coupling. *Annals of Statistics*, *26*(2), 451–471.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*. New York: Springer, 2nd edition.

Hidalgo, J. M. G., Bringas, G. C., Sanz, E. P., & Garcia, F. C. (2006). Content based SMS spam filtering. In *Proceedings of the 2006 ACM symposium on Document engineering - DocEng '06*. Amsterdam, The Netherlands, 107.

Higgins, E. T., Bargh, J. A., & Lombardi, W. (1985). Nature of priming effects on categorization. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *11*(1), 59–69.

Holbrook, A. L., Krosnick, J. A., Moore, D., & Tourangeau, R. (2007). Response Order Effects in Dichotomous Categorical Questions Presented Orally. *Public Opinion Quarterly*, *71*(3), 325 –348.

Jones, J. M. (2010). Economy, Jobs Easily Top Problems in Americans' Minds. http://www.gallup.com/poll/143135/Economy-Jobs-Easily-Top-Problems-Americans-Minds.aspx.

Kelley, S. (1983). *Interpreting elections*. Princeton, NJ: Princeton University Press.

King, G. (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*, 617–642.

Knight, K. (1985). Ideology in the 1980 election: Ideological sophistication does matter. *The Journal of Politics*, *47*(3), 828–853.

Krosnick, J. A. & Alwin, D. F. (1987). An evaluation of a cognitive theory of response-order effects in survey measurement. *Public Opinion Quarterly*, *51*(2), 201–219.

Krosnick, J. A. & Fabrigar, L. R. (1999). Open and closed questions. In *The Handbook of Questionnaire Design*. New York: Oxford University Press.

Krosnick, J. A. & Schuman, H. (1988). Attitude intensity, importance, and certainty and susceptibility to response effects. *Journal of Personality and Social Psychology*, *54*(6), 940–952.

Lampos, V. & Cristianini, N. (2010). Tracking the flu pandemic by monitoring the social web. In *Cognitive Information Processing (CIP), 2010 2nd International Workshop on*. 411–416.

Manning, C. D., Raghavan, P., & Schutze, H. (2008). *Introduction to Information Retrieval*. Cambridge, UK: Cambridge University Press, 1st edition.

Meyer, D., Hornik, K., & Feinerer, I. (2008). Text Mining Infrastructure in R. *Journal of Statistical Software*, *25*(05).

Munro, R. & Manning, C. D. (2010). Subword variation in text message classification. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. 510–518.

Newport, F. (2010). Economy Dominates as Nation's Most Important Problem. http://www.gallup.com/poll/141275/economy-dominates-nation-important-problem.aspx.

Platt, J. C. (1998). Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods - Support Vector Learning*. Cambridge, MA: MIT Press, 185–208.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program: Electronic Library & Information Systems*, *40*(3), 211–218.

Quinn, K. M., Monroe, B. L., Colaresi, M., Crespin, M. H., & Radev, D. R. (2010). How to analyze political attention with minimal assumptions and costs. *American Journal of Political Science*, *54*(1), 209–228.

RePass, D. E. (1971). Issue salience and party choice. *The American Political Science Review*, *65*(2), 389–400.

Schuman, H. (2008). *Method and meaning in polls and surveys*. Cambridge, MA: Harvard University Press, 1st edition.

Schuman, H. & Presser, S. (1996). *Questions and answers in attitude surveys: Experiments on question form, wording, and context*. Thousand Oaks, CA: Sage Publications, Inc.

Smith, E. R. A. N. (1989). *The unchanging American voter*. Berkeley, CA: University of California Press.

Stone, P. J., Dunphy, D. C., Smith, M. S., & Ogilvie, D. M. (1966). *General Inquirer: A computer approach to content analysis*. Cambridge, MA: MIT Press.

Stouffer, S. A. (1955). *Communism, conformity, and civil liberties: a cross-section of the nation speaks its mind*. New York: Transaction Publishers.

Tan, P., Steinbach, M., & Kumar, V. (2005). *Introduction to Data Mining*. Boston: Addison Wesley.

5 Appendix I: Sequence of R scripts and dependent .Rda files to reproduce categorization

1: Run `Words2DataSample.R`, depends on `trainingSet-SOL500.csv`, writes `samptext.Rda, samptagfeatures.Rda, TextData.Rda`.

2: Determine optimal pre-processing strategy—run `SampPredCV.R`, depends on `TextData.Rda, samptagfeatures.Rda, trainingSet-SOL500.csv`, writes `detailsproc.Rda, detailsmod.Rda`. TODO: Integrate sLDA testing here.

3: Run `Words2DataFull.R`, depends on `Words2DataFull.R` writes `tagfeatures.Rda, fulltext.Rda, dtmstemfull.Rda` - the last item will change if the optimal pre-processing data set changes.