Text Analysis: Estimating Policy Preferences From Written and Spoken Words*

Kenneth Benoit
London School of Economics
and Trinity College Dublin

Alexander Herzog
Clemson University

February 17, 2015

Abstract

This chapter provides an introduction into the emerging field of quantitative text analysis. Almost every aspect of the policy-making process involves some form of verbal or written communication. This communication is increasingly made available in electronic format, which requires new tools and methods to analyze large amounts of textual data. We begin with a general discussion of the method and its place in public policy analysis, including a brief review of existing applications in political science. We then discuss typical challenges that readers encounter when working with political texts. This includes differences in file formats, the definition of "documents" for analytical purposes, word and feature selection, and the transformation of unstructured data into a document-feature matrix. We will also discuss typical pre-processing steps that are made when working with text. Finally, in the third section of the chapter, we demonstrate the application of text analysis to measure individual legislators' policy preferences from annual budget debates in Ireland.

*Prepared for *Analytics, Policy and Governance*, eds. Jennifer Bachner, Kathyrn Wagner Hill, and Benjamin Ginsberg. This research was supported by the European Research Council grant ERC-2011-StG 283794-QUANTESS.

1 Text Analysis as a Tool for Analyzing Public Policy

Public policy in democratic systems is produced by numerous political actors with different preferences. These differences are especially pronounced between parties in governments and those in opposition, but also exist between parties. We know from a great deal of research in political science, furthermore, that political parties are not unitary actors, but rather collections of individuals with often very divergent preferences influenced by different, sometimes conflicting pressures. Governing coalitions in parliamentary systems often expend a great deal of energy managing these differences, at the risk of coming apart should they fail to do so.

Accurately measuring the policy preferences of individual political actors has therefore long formed a key part of efforts to model intra-party politics and the public policy outcomes that result. The bulk of this work has traditionally relied on measuring differences through scaling roll call votes (Poole and Rosenthal, 1997; Clinton et al., 2004) or using roll call votes to measure voting agreement (e.g. Hix et al., 2005). Yet roll call votes in parliamentary systems suffer from a number of problems that prevent them from forming a reliable basis for estimating legislators' preferences for policy. In most settings, a large share of legislative votes are not recorded as roll calls, and the votes that are selected for roll calls may be so chosen for strategic political reasons (Hug, 2010). Measures of policy preferences based on these selective votes produces selection bias in the resulting measures (VanDoren, 1990; Carrubba et al., 2006, 2008). Another problem with measuring intra-party differences on policy from roll call votes is that in most parliamentary systems, voting is tightly controlled through party discipline. This means that legislators vote with their party possibly not because of their policy preferences, but rather in spite of them (Laver et al., 2003; Proksch and Slapin, 2010).

These problems with roll call votes has led to the rapid growth in recent years in political science and policy analysis of using text as data for measuring policy preferences. Researchers have developed and applied a variety of scaling methods for measuring policy

preferences from the speeches and writings of political parties and their members. The conventional wisdom is that while party discipline may strongly constrain what legislators *do* (in terms of voting behavior), these constraints do apply less to what legislators *say*, as recorded in floor debates, committee hearings, campaign speeches, web sites, social media, or press releases. To make use of this information, a growing subfield within political science has developed to extract policy preferences using text as data (e.g. Laver and Garry, 2000; Proksch and Slapin, 2010; Monroe and Maeda, 2004; Laver and Benoit, 2002; Lauderdale and Herzog, 2014).

Grimmer and Stewart (2013) provide an excellent review of current approaches, which they divide roughly into classification approaches and scaling approaches. Scaling approaches include the methods we have discussed, for measuring preferences on policy, and may be divided into supervised and unsupervised methods. The most common supervised method in political science is the "Wordscores" method developed by Laver et al. (2003). With roots in both Naive Bayes machine learning approaches as well as regression approaches, Wordscores involves a training step on documents of "known" positions to produce scores for each word. These scores can then be used to estimate the position on the input dimension of any out of sample texts. This method has been used successfully in dozens of applications (e.g. Laver and Benoit, 2002; Giannetti and Laver, 2005; Laver et al., 2006; Benoit et al., 2005; Hakhverdian, 2009; Klemmensen et al., 2007; Baek et al., 2011; Warwick, 2015).

The most commonly used unsupervised method for scaling policy preferences is the latent variable model dubbed "Wordfish" by Slapin and Proksch (2008), which models word generation in a document as a Poisson process from which a latent variable representing the document position can be estimated. This approach has been successfully applied to measure party preferences in German elections (Slapin and Proksch, 2008), European interest group statements (Klüver, 2009), the European Parliament (Proksch and Slapin, 2010), and Irish budget speeches (Lowe and Benoit, 2013).

Classification approaches use mainly unsupervised methods adapted from computer sci-

ence for topic discovery and for estimating the content and fluctuations in the discussions over policy. Since the publication of a seminal paper by Blei et al. (2003) describing a Latent Dirichlet Allocation (LDA) model for estimating topics based on collections of unlabeled documents, topic modeling has seen many extensions to political science. These have included methodological innovations by political scientists, including the dynamic multitopic model (Quinn et al., 2010) and the expressed agenda model (Grimmer, 2010). Other innovations developed in political science include revising the standard LDA model to allow for the incorporation of additional information to better estimate topic distributions (Stewart et al., 2011).

Estimating preferences or topics from text as data, of course, requires that texts have been prepared and that a researcher is familiar with the tools for carrying out this preparation and for estimating the models we have described. It is to this topic that we turn attention in the next section.

2 Practical Issues in Working with Text

One of the overwhelming advantages of working with text as data is its easy and nearly universal availability: there are almost no aspects of the policy-making process that do not involve the verbal or written use of language that is recorded and published. This also poses a challenge, however, since there are almost as many formats for disseminating these texts as there are sources of texts. To be converted into useful data, texts must be processed, sometimes heavily. In this section, we outline some of these challenges and discuss several common solutions.

2.1 Wrestling with Text File Formats

Sources of textual data are found almost everywhere, but typically require a good deal of preparation in order to be ready for analysis as data. Unlike most quantitative data,

Format	Filename extension	Where found
Single document formats		
Plain text	`.txt`	Various
Microsoft Word	`.doc, .docx`	Web sites, file archives
Hypertext Markup Language	`.htm, .html`	Web sites
Extensible Markup Language	`.xml`	Various structured software
Portable Document Format (text)	`.pdf`	File archives, web sites
Documents as images	`.pdf, .png, .jpg`	Scanned documents, photographs of documents
Multiple document formats		
Comma-separated value	`.csv`	File archives, some distribution outlets
Tab-separated value	`.csv`	File archives, some distribution outlets
JSON (Javascript Open Notation)	`.json`	Text provider APIs
Microsoft Excel	`.xls, .xlsx`	File archives, some distribution outlets
SPSS	`.sav`	SPSS statistical software
Stata	`.dta`	Stata (13) statistical software
R	`.RData, .rda`	R statistical software format

Table 1: Common data formats for text

which usually comes in structured formats where rows constitute units of observation and columns represent variables, textual data is usually unstructured or minimally structured when published. In addition, text may be embedded in a wide variety of different publication formats, such as HTML (if published to a web site), a relational database, a set of many files, or a single large file. Databases and files containing texts, furthermore, come in many possible formats, which require conversion.

Table 1 lists a number of common file formats in which text files are often distributed, as well as the most likely filename extensions associated with each type and where they are mostly like to be found. The types are distinguished by whether they tend to contain one document per file, or whether a single file might contain numerous documents. Together, the documents will form a *corpus*, a collection of multiple documents to be treated as data. Some file formats usually contain one document per file, such as a collection of political party manifestos from a single national election, which might be distributed as a set of Portable

Document Format (pdf) files, one per political party. In such a case, the document-level meta-data such as the political party name might be known only through the file name, and need to be added later. A set of texts distributed in a "dataset" format such as `.csv` format or Stata format – which as of version 13 allows text variables of practically unlimited size – will already have one document per row and possibly additional variables about the texts embedded as additional columns.

Commonly known as *meta-data*, extra-textual information associated with each text provides additional information about the texts, their sources, and the circumstances of their production, beyond their textual content. A set of parliamentary speeches, for instance, may also have the name and party of the speaker as document-level meta-data, in addition to a date and time stamp, and possibly the title of the debate in which the member of parliament spoke. Managing this meta-data and ensuring that it is associated with the text files through each stage of processing is one of the challenges of working with textual data, and one of the key reasons why researchers tend to use specialist software for this purpose.

When files may contain multiple documents, they may also be associated with additional document-level information, as additional "variables". Texts in the form of single files, on the other hand, only contain additional, non-textual information if this is somehow embedded in the original texts as tags or document meta-data. Nearly all documents contain some form of meta-data, although the standards for meta-data differ widely among different formats. In HTML or XML formats, for instance there are `DOCTYPE` declarations, as well as tags for the HTML version, and additional metadata declared by `<meta key="value">` tags. In Figure 1, for instance, the author, content type, source, and character encoding are recorded as meta-data. While nothing prevents individual documents from using a key-value tag convention for extra-textual data, this must not only be added to the text by the researcher or data source, but also the text processing tools used for analysis must be capable of separating this information from textual data and recording it separate from the content of the texts.

For large collections of documents that have been maintained by official sources, such as

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<meta charset="utf-8">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta name="generator" content="pandoc" />
<meta name="author" content="Ken Benoit" />
```

Figure 1: Document meta-data example for HTML format.

parliamentary records, it is common for texts to be stored in relational databases. Web sites and DVDs that publish large collections of textual data, for instance, often use a database back-end to store texts, and use web front-ends to query that database as per the user request. Either way this information must be extracted or "scraped" from the front-end, in order to be stored as a corpus that can be analyzed.

For example, "DPSI: Database of Parliamentary Speeches in Ireland" (Herzog and Mikhaylov, 2013) contains the complete record of parliamentary speeches from Dáil Éireann, the House of Representatives of the Irish parliament, from 1919 to the current session. This dataset had to be extracted using automated tools (in this case, text scraping programs written in the Python programming language), tagged with identifying information, and stored into another database. To extract texts for analysis, this database was queried with the results saved as plain texts documents, and read into an R package for subsequent processing.

Other, more direct alternatives are "Application Programming interfaces" (APIs) defined by a textual data source, such as Twitter or Facebook, that can be called using a syntax specified by the creator of the data source. This usually requires authentication with an API key that must be obtained from the data provider. A query calling the API generally results in a JSON formatted file returned as a set of key-value pairs, with one key representing the textual data, and the other keys defining document variables and meta-data. JSON is itself a text file, but its tags defining the key-value pairs need to be read by additional software capable of keeping track of the keys and values. To give an example, Figure 2 shows the

```
{
    "results": [
        {
            "count": 664,
            "percentage": 0.045457968494341715,
            "total": 1460690,
            "month": "201401"
        },
        {
            "count": 590,
            "percentage": 0.04291370998478382,
            "total": 1374852,
            "month": "201402"
        },

...

        {
            "count": 221,
            "percentage": 0.06546770901528863,
            "total": 337571,
            "month": "201412"
        }
    ]
}
```


Figure 2: Example of JSON data output collected through Capitol Words' API for the frequency count of the word "tax" during 2014.

output of a query to Capitol Words' API (http://capitolwords.org/api/) that retrieves frequency counts for the word "tax" in congressional speeches during each month of 2014. Here, they key sets in each record are identical, but additional processing is needed to convert the keys into variables (columns) in a traditional, rectangular dataset format.

Sometimes, documents are scanned in an image format, meaning the document exists only as a binary image in a pdf or photo file format (such as JPEG). This poses a challenge for extracting the text, because instead of having text encoded digitally in a way that a computer recognizes, image format files contain only the encoding of the image of the text. To be usable as textual data, the image must first be converted into text encoding using

optical character recognition (OCR) software. The quality of this conversion will depend a lot on the quality of the image, and of the software. Such conversion frequently introduces errors, especially with low-resolution or poor-quality images, but also caused by common typographic features such as ligatures in variable-width fonts.

To be useful for text processing, files containing text are invariably converted into "plain text" format, stripped of mark-up tags, application-specific formatting codes, and any additional binary information that might be part of a file format. Unfortunately, there are numerous, differing conventions for what constitutes "plain" text. Computers represent text digitally by mapping into numbers the glyphs that human readers recognize as letters, numbers, and additional symbols. Unfortunately, this processing of *text encoding* did not develop according to a single standard, and has led to a proliferation of "code pages" mapping characters into numeric representations, a problem mainly affecting non-English languages (with accented characters or non-Roman character sets), but also affecting typographic symbols such as dashes and quotation marks. Depending on the country, platform, and year when the text was digitally recorded, the same numbers might map to different characters, causing headaches for novice and expert users alike. This explains why accented characters that look fine on one computer appear as diamond-shaped question marks or small alien head symbols on another computer.

Tools are widely available for converting between encodings, but not all encodings are detected automatically. The best recommendation we can offer is to make sure that all texts are coded as UTF-8, an eight-bit variable-length encoding of the Unicode standard, a modern mapping of almost every character in existence to a unique code point. Most modern operating systems operate in UTF-8, and most new software tools also use this text encoding as standard.

Once texts have been converted into a usable format, two key issues remain before the data can be processed. Because the "document" will form the aggregate unit of analysis, this unit first needs to be defined from the corpus. Second, once documents are defined,

decisions also need to be made as to what textual features to extract.

2.2 Defining Documents

A corpus is a collection of documents, but the manner in which that corpus is segmented into documents is up to the user. Segmentation is the process of separating the texts found in the corpus into units that make sense for a particular analysis. This segmentation may be quite natural and simple, but may also involve reorganizing the units depending on the research purpose.

Often, source texts are organized in ways that naturally correspond to document units. The corpus of presidential inaugural addresses, for instance, consists of 57 separate speeches with a median length of about 2,100 words. For most purposes, each speech would define a "document". The same may be true for party manifestos, parliamentary bills, campaign speeches, or judicial opinions.

In other cases, however, we may need to combine different pieces of text to form a single "document". In the legislative speeches we analyze below, for instance, the Irish parliamentary archive recorded each non-interrupted speech act separately. Because interruptions are very frequent in legislative settings – and certainly in the Irish case – we concatenated all of a single speaker's contributions in the debate into a single "document". Because this resulted in some speech "documents" for members of parliament who did not really make speeches, but were nonetheless recorded as having delivered a short "speech" consisting only of "Hear, hear!", we also defined a threshold of words below which we simply discarded the text.

Concatenating texts into larger "documents" is very common when dealing with social media data, especially the micro-blogging site Twitter. The 140-character limit imposed on any single "Tweet" means that these may be too short to analyze separately. Many researchers therefore define a document as the concatenation of all Tweets of a user over a fixed period.

On the other hand, some longer texts may require segmentation to break them up into

smaller units, the opposite of combining (concatenating) them. If the sentence or paragraph will form our unit of analysis, then we could use automatic methods to segment the texts based on the characters that define paragraph or sentence units.[1] In many other cases, we will use human-assigned breaks to denote meaningfully different sections of text that will define documents, possibly for later selection and analysis in groups. Slapin and Proksch (2008), for example, analyzed German manifestos on different dimensions of policy (economic, social, and foreign policy), for instance, by first segmenting manually based on which topic was being discussed.

2.3 Defining and Selecting Features

There are many ways to define features in text analysis. The most common feature is the word, although here we have used the more general term "features" to emphasize that features can be both more or less than words as they are used in the source texts. In addition, various strategies for *selecting* features may lead to some being discarded. This is a complex field, and here we describe it only in the broadest terms.

Defining features is the first step. Most approaches define words as features, through a process known as *tokenizing* the texts. This language comes from linguistics, where word occurrences are known as "tokens", and unique words as word "types". Tokenization can be performed automatically and reliably by software that uses whitespace and few additional padding characters as delimiters.[2] Not every word type may become a feature, however, due to selection. Selection involves, very broadly speaking, two strategies, one based on combining feature types, and the other based on excluding them.

Combining feature types is the strategy of treating different words as equivalent, on syntactic or semantic grounds. When we perform "stemming", we are using rules to convert

[1] Surprisingly, methods for sentence segmentation are far from perfect. Different languages use different end of sentence delimiters, and many delimiters — especially the "." character — are also used for other purposes such as to separate the decimal places in a printed number or in abbreviations (e.g. "e.g."). Detecting paragraph delimiters can be even harder.

[2] Although some languages, such as Chinese, do not use inter-word delimiters and tokenization therefore relies on rule- and statistical-based detection.

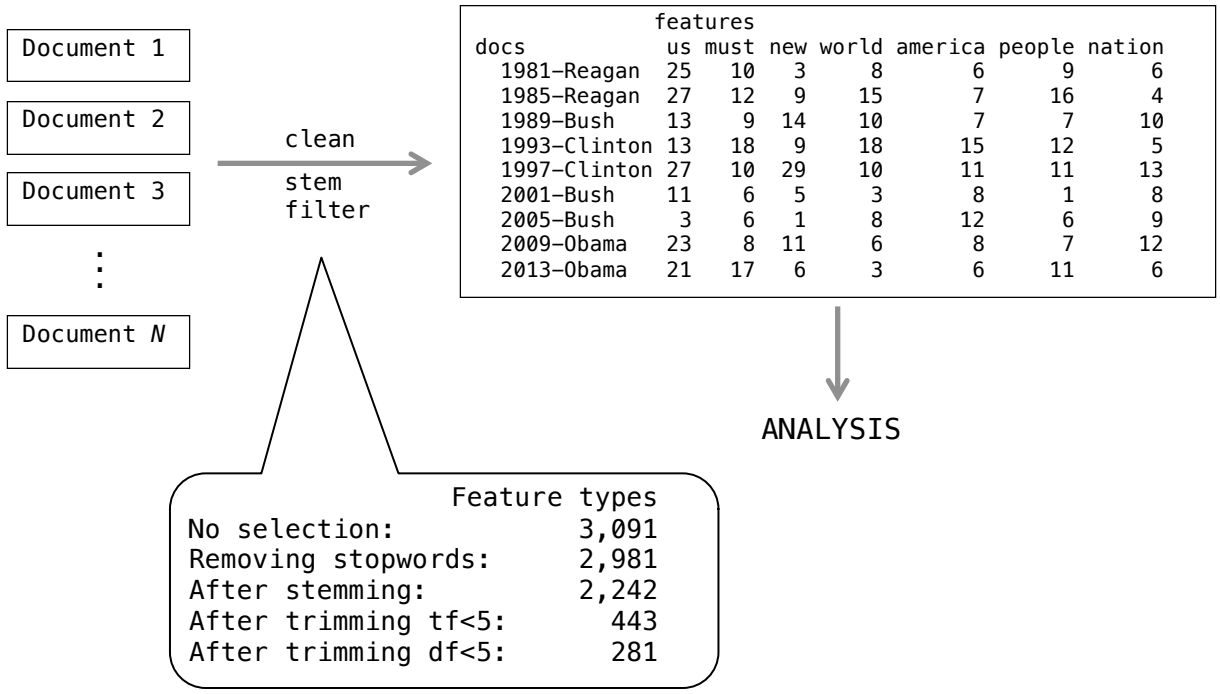

Figure 3: Illustration of converting Presidential inaugural speeches into quantitative data, using feature selection.

morphological variations of words into their canonical or "dictionary" forms. For fiscal policy, for instance, stemmed features would treat the types "tax," "taxes", "taxing", and "taxed" as equivalent occurrences of the lemma "tax". Similar methods might be used to regularize the spellings across dialects, for instance to treat "colour" and "color" as the same feature type. Equivalence classes may also be defined according to semantic categories. This is exactly the strategy taken by dictionary approaches, such as the Linguistic Inquiry and Word Count, a dictionary of language and psychological processes by Pennebaker et al. (2007), which treats "hate", "dislike", and "fear" as members of an equivalent class labeled "Negative emotion".

Exclusion approaches for feature selection are based on dropping feature types that are deemed uninteresting for research purposes. Feature exclusion is typically done on an *ex ante* basis or on the basis of patterns of occurrence. *Ex ante*, it is quite common to exclude word features found in lists of "stop words", usually a catalog of conjunctions, articles, prepositions, and pronouns. Few analysts expect the cross-document frequencies of the

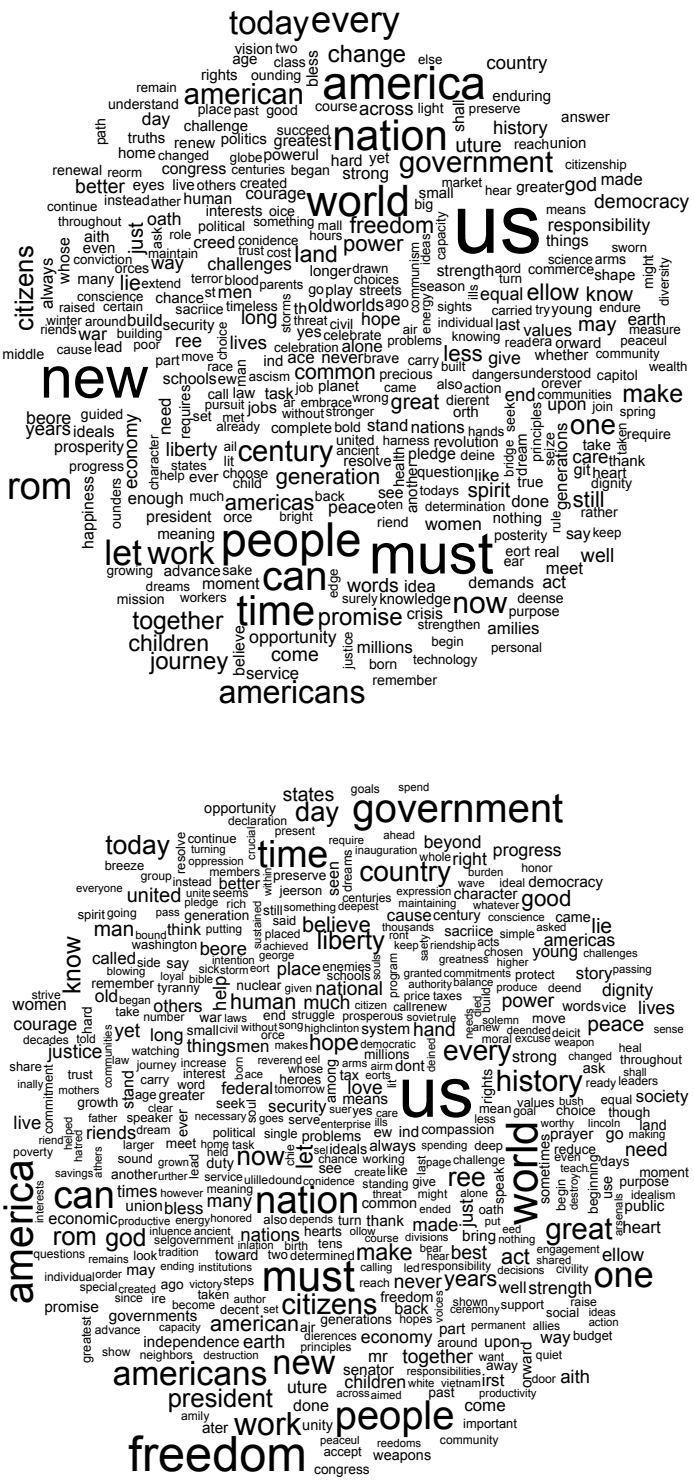

Figure 4: Word cloud plots for Presidential inaugural speeches since 1981, by (a) Democrat and (b) Republican presidents. Stop words have been excluded.

word "the" (the most commonly occurring word in English) to be substantively meaningful, for instance. Stop word lists should be used with caution, however, since there is no one-size fits-all list, although many text analysis packages provide stop words lists that are accepted as "standard" by users who seldom bother even to inspect the contents of these lists.

The other exclusion strategy is to drop features below a threshold of term or document minimum frequency: some absolute threshold of how many times a feature must occur in a document or in how many documents it must occur. In processing the corpus of inaugural speeches since Reagan illustrated in Figure 3, for instance, selection strategies narrowed down an unfiltered number of features from 3,091 to 2,242 once stemming had been been applied, and a further reduction to 443 and 281 once features with feature ("term") and document frequencies less than five had been removed. The final feature set of 281 feature types is plotted in Figure 4 as "word clouds" according to the political party of the president. Word clouds are common visualization tools for observing the most common word features in a text, plotting the size of the word feature proportional to its relative frequency.

A quotidian form of feature selection is known as *cleaning*, the process of removing unwanted characters or symbols from text prior to tokenization into features. Almost universally, words are transformed to lower case and punctuation symbols are discarded. It is also very common to remove numerals, although some numbers (especially years) may form substantively meaningful features depending on the research topic.

2.4 Converting Documents and Features Into Quantitative Information

To analyze text as data, documents and features must be converted into a structured, numerical format. The starting point for the analysis stage of quantitative text research is the creation of a *document-feature matrix* that contains information on the number of occurrences of each feature in each document. Usually, this matrix represents documents as rows and features as columns, as we depict in Figure 3. From this matrix, any number of analyses

can be performed.

The conversion into a document-feature matrix is done efficiently by software tools designed for this purpose, in essence a large-scale cross-tabulation. When working with natural language texts with more than a trivial number of documents, many features will occur zero times in a number of documents. As a consequence, creating a simple cross-tabulation of features by documents as a *dense* matrix, or one in which zero-frequency are recorded, becomes very inefficient. In practice, therefore, many software implementations record document-feature matrices as *sparse* representations, in essence storing only the features that occur at least once, and along with their frequency indexed by document. This results in significant savings of both storage space and computational power.

Document-feature matrices are commonly transformed by either weighting features, smoothing them, or both. What we have described so far, recording a feature count, is the simplest form of weighting. Because documents differ in length, however, this will result in larger feature counts for longer documents, all other things being equal. A common remedy to this is to convert feature frequencies into relative feature frequencies, replacing the counts by the proportion of times each feature occurs in a document. Known as *normalizing*[3] the text, this process makes feature values comparable across documents by dividing each feature count by the total features per document. Many other possibilities exist (see Manning et al., 2008, Ch. 2), such as *tf-idf* a weighting, in which each feature is divided by a measure of the proportion of documents in which a term occurs, to down-weight the features that occur commonly across documents.[4] Tf-idf weighting is most commonly used in information retrieval and machine learning as a way of making the results of search queries more relevant or for improving the accuracy of predictive tools.

[3] In computer science, this is known as *L1 normalization*.

[4] If N is the total number of documents and df_t (*document frequency*) is the number of documents that contain a term t, then *inverse document frequency* is defined for term t as $\text{idf}_t = \log \frac{N}{\text{df}_t}$. The *tf-idf* weight for term t in document d is then given by $\text{tf-idf}_{t,d} = \text{tf}_{t,d} \times \text{idf}_t$, where $\text{tf}_{t,d}$ (*term frequency*) is the number of occurrences of term t in document d (Manning et al., 2008, Ch. 6).

3 Application to Fiscal Policy

We demonstrate the quantitative analysis of textual data with an analysis of budget speeches from Ireland. Government stability in parliamentary systems depends crucially on one overriding characteristic of legislative behavior: unity. Without party discipline in voting, especially on critical legislation, governments quickly come apart, formally or informally, leading to a new government or new elections. However, we know that there is a large degree of intra-party heterogeneity in policy preferences. Legislators have different preferences, and often vote in spite of these, instead of because of them. Moreover, legislators often answer to more than one type of principal, and this may cause tensions when constituency representation clashes with party demands (e.g. Strøm and Müller, 2009; McElroy and Benoit, 2010). The more acute the tension between the personal interests of the legislator and the group interests of his or her party, the more we would expect the legislator's preferences to diverge.

Because of party unity, voting records tells us little about intra-party politics in legislatures where party discipline is strong. What legislators *say*, however, is typically less constrained. Legislative speeches are seldom, if ever, subject to formal sanction for those who speak out of turn. Indeed, party leaders may view floor debates as an opportunity for reluctantly faithful members to send messages to their constituents, as long as they follow party instructions when it comes to voting. For these reasons, the text analysis of parliamentary speeches has formed an important leg of the empirical study of intra-party preferences (e.g. Proksch and Slapin, 2010; Laver and Benoit, 2002; Lauderdale and Herzog, 2014). The words that legislators use can be scaled into positions providing a much more valid indicator of their preferences than the votes they cast.

In this section, we exploit this feature of parliamentary texts to measure the strain placed on party unity by austerity budgets: those splitting not only government and opposition, but also governing parties and coalitions by virtue of requiring deep and deeply painful clawbacks of services, tax raises, and spending cuts. Austerity budgets are an unfortunately

familiar feature of European politics, since the onset of the euro zone crisis in banking and sovereign debt servicing. The challenge of passing these severe budgets, often necessitated by externally imposed conditions of emergency funding packages, has split and sometimes brought down governments. Yet even in the face of such conflict, it is seldom manifest in legislative voting, even when voting on unpopular austerity budgets. To observe the stain on governing parties, we must look at what legislators say.

3.1 Budgets and Politics of Economic Crisis in Ireland

Our case study in austerity budgets comes from Ireland, one of the first European states to experience a deep banking crisis and receive a multi-billion euro bailout with austerity conditions attached. Since 2008, the country experienced a steep decline in economic output and a sharp rise in unemployment, and a massive debt problem caused by the financial load of recapitalizing a failing banking system. This forced the government to implement a number of severe austerity measures against growing public resentment, ultimately leading to a record low in the popularity ratings for the government parties and a breakdown in January 2011 of the coalition led by Fianna Fáil (FF), a party that had led Ireland continuously since 1997. Addressing the crisis required a €85 billion rescue package from the European Union and the International Monetary Fund, a bailout that led to tax cutbacks in social spending equivalent to €20 billion, or 13 per cent of GDP (Bergin et al., 2011, 51), including highly controversial changes to taxes and wage agreements, while leaving the public perception that the bankers who had caused the crisis were getting rescued.

We include in our analysis all annual budget debates from 1987 to 2013. During these debates, legislators are free to discuss the budget, with governing party members and ministers expressing support, and opposition parties invariably criticizing the government and its budget. Given the strong party discipline in Ireland (Gallagher, 2009), votes tend to follow strict party lines. Voting against the government's financial bill or resigning from the party are extreme measures that only a few legislators are willing to face. Party discipline in

Ireland, indeed, makes the two equivalent, since voting against the party on a budget would result in expulsion from the party. In parliamentary systems like Ireland, where budgets are written entirely by the party in government, votes on these national fiscal plans are very much votes for or against the government itself, and indeed were the government to lose such a vote, it would fall and a new coalition would have to be formed (Gallagher et al., 2011).

To scale the budget speeches, we used the "Wordscores" method of Laver et al. (2003). We scaled each debate separately, training each scaling using the speech of the finance minister to represent the "pro-budget" position, and the speech of the financial spokesperson of the opposition (equivalent to the "shadow finance minister") to represent the opposition position. Representing these positions as +1 and -1 respectively, we transformed the "virgin" text scores using the rescaling proposed by Martin and Vanberg (2007), a procedure that ensures that the scaled positions of the reference texts are set to the scores used to train the system (+1 and -1). While not always recommended, this transformation ensures that all other documents' scaled values are positioned relative to the reference documents (Benoit and Laver, 2007), an outcome we explicitly desired in our budget-by-budget comparison. Each position is then "fixed" relative to the positions of the government and opposition finance spokespersons, making the scores comparable across budgets according to a common benchmark.

3.2 Results

The results of the scaling procedure is an estimated position on the latent dimension for each speaker. Figure 5 displays these speaker positions for the 1999 budget debate. The figure shows that our analysis is able to re-cover the division between government and opposition parties: almost all members of the Fianna Fáil-Progressive Democrat coalition have positive Wordscores position, while the majority of opposition members have negative scores. Moreover, we also find interesting intra-party differences. Within the government parties, we find that on average, cabinet members have more pro-government positions while government

backbenchers are closer to the opposition.

Figure 6 shows estimated positions for the 2009 budget debate, which was the first austerity budget implemented by the government at the beginning of the financial and economic crisis. Compared to the budget debate during the boom years (Figure 5), we find much more overlap between the government and opposition – a first indication that the crisis has increased tensions with members of the government parties.

In an attempt to explain this pattern more systematically, we plot the difference between the average positions of cabinet members and the average positions of backbenchers against government debt as a percentage of GDP, in Figure 7. The distance between the two groups appears to be a function of the economy: the gap is relatively small in good economic times, but increases as the economy worsens.

To further test the relationship between intra-government divisions and economic performance, we estimate a regression model with the distance between cabinet members and government backbenchers as the dependent variable. As controls, we include government debt, a variable that indicates years in which an election occurred, and three dummy variables that indicate an alternation in government, a change of the prime minister, and a change of the finance minister, respectively. Figure 7 summarizes the results of this regression. We find a significant effect for government debt that confirms the relationship displayed in Figure 7: the gap between cabinet members and backbenchers widens through economic bad times. The anticipation of an election, in contrast, decreases the gap, possibly because government members need to demonstrate a unified front when competing in the election.

4 Concluding Remarks

In this chapter, we have provided readers with the basic ideas of quantitative text analysis and outlined typical challenges that readers will encounter when they apply this method to the study of public policy. This included the various file formats in which text is stored, the

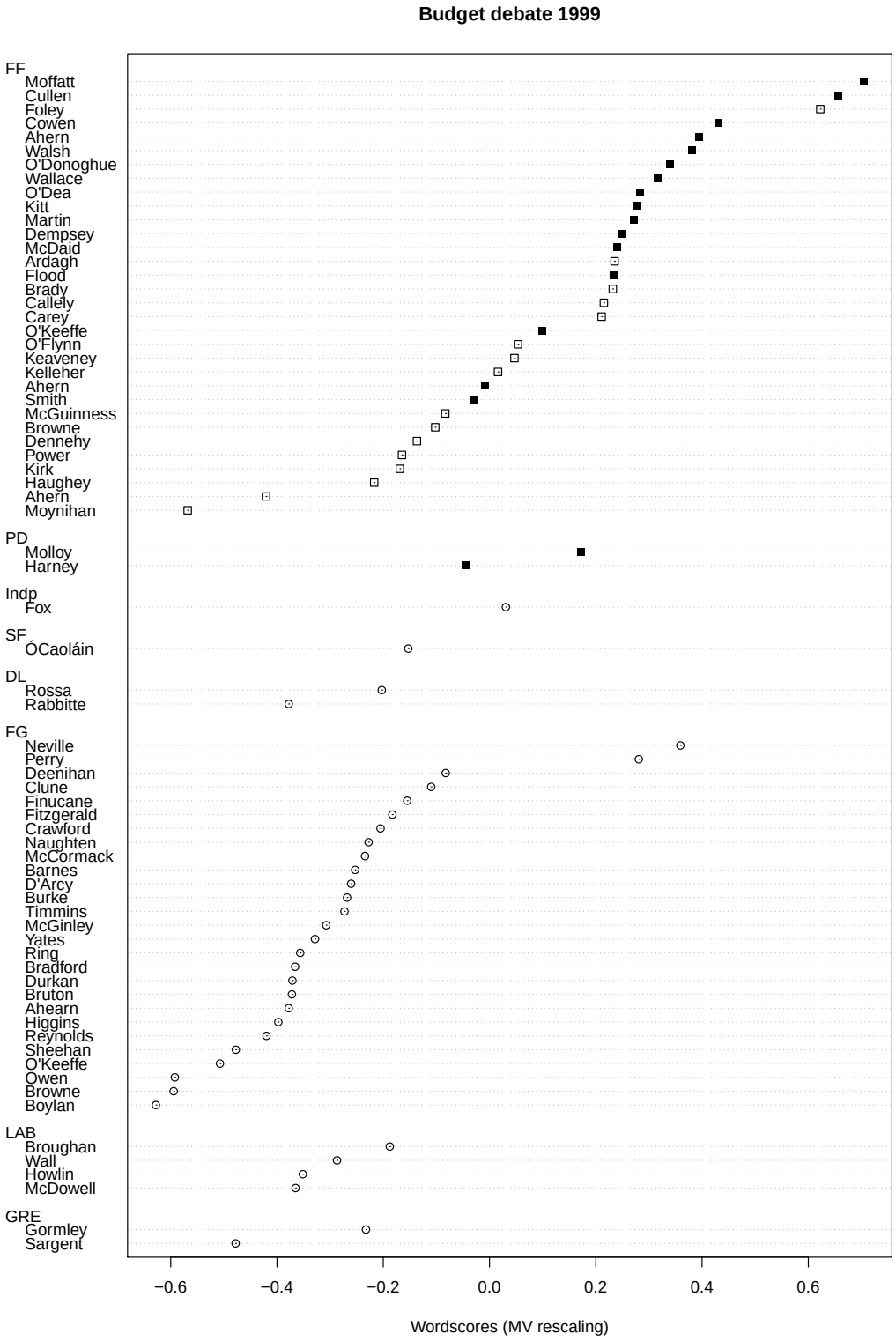

Figure 5: Estimated positions for 1999 budget debate (boom years). *Note:* Symbols: Squares–government members, solid squares–cabinet ministers and junior ministers, circles–opposition members. Party abbreviations: FF–Fianna Fáil, PD–Progressive Democrats, Indp–Independent, SF–Sinn Féin, DL–Democratic Left, FG–Fine Gael, LAB–Labour, GRE–Greens.

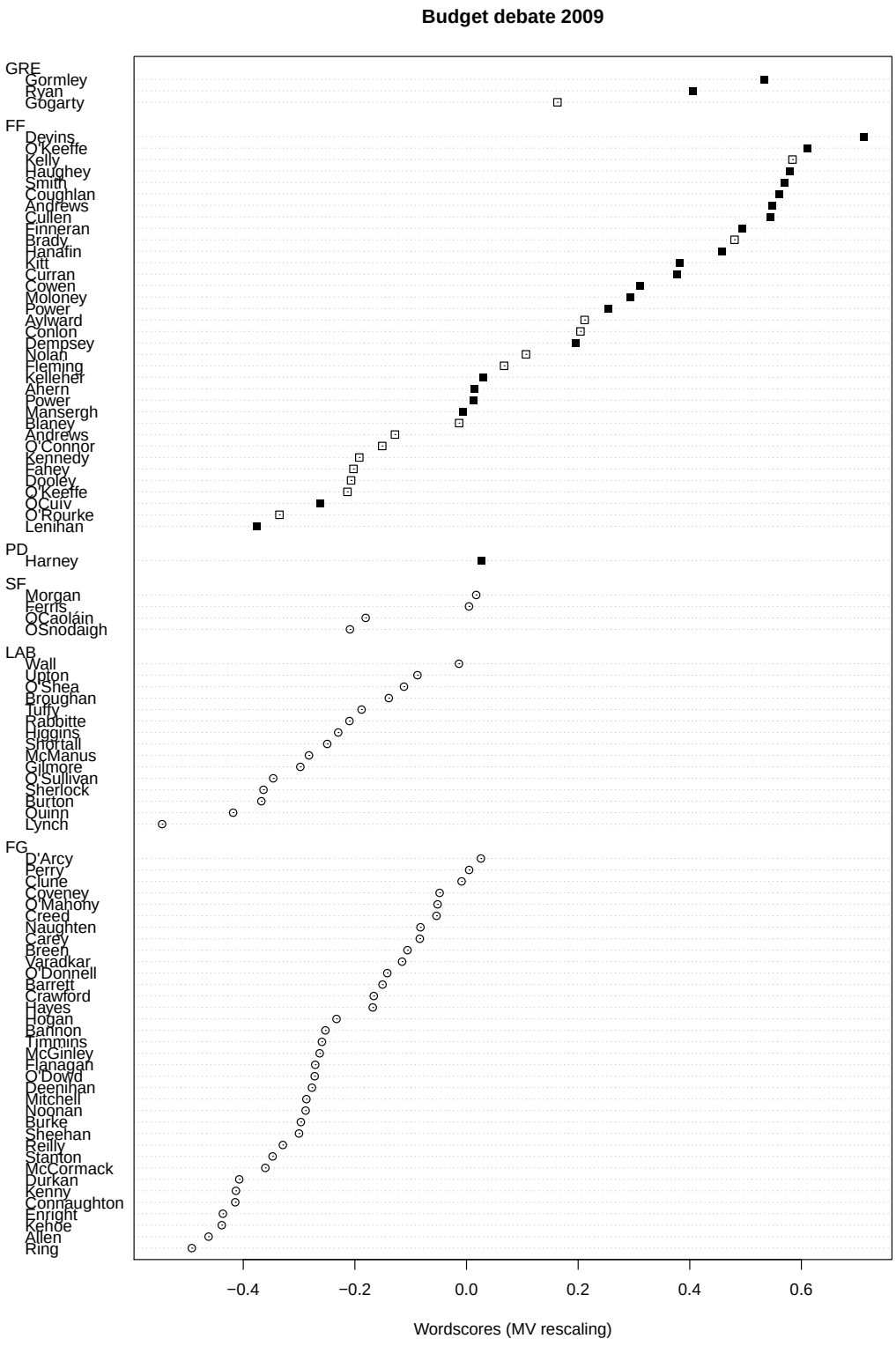

Figure 6: Estimated positions for 2009 budget debate (crisis years).

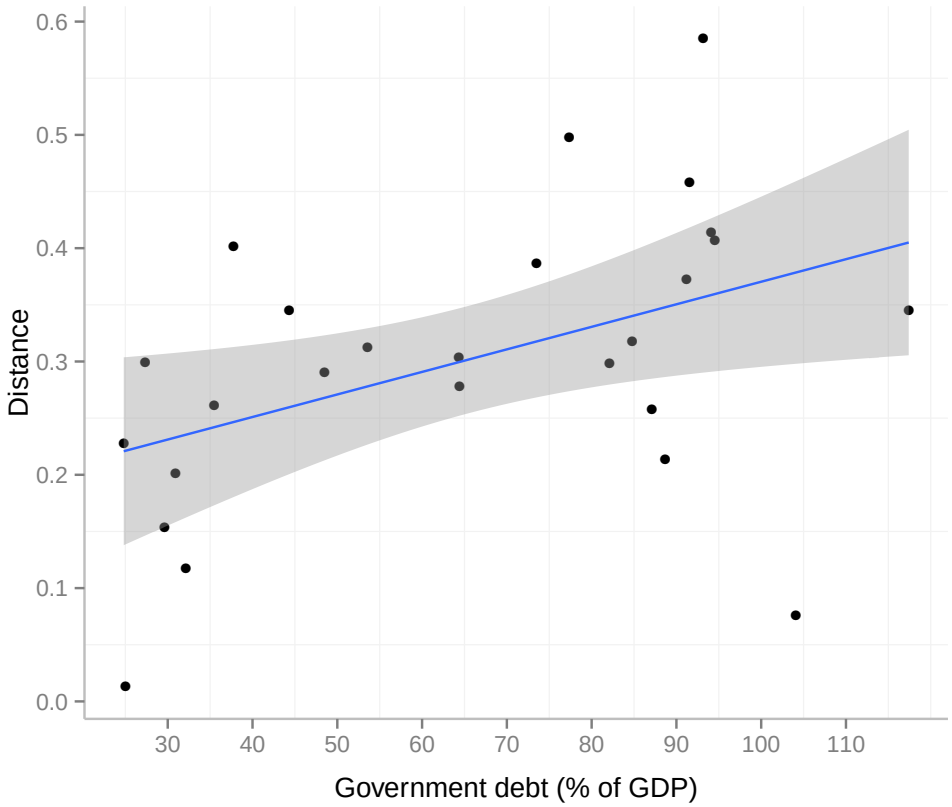

Figure 7: Distance between average Wordscores estimates for cabinet members and government backbenchers against government debt (% GDP) with OLS regression line and 95% confidence interval.

construction of a corpus (i.e., collection of documents), the selection of words and features on which the analysis will be based, and the transformation of unstructured data into a document-feature matrix as the starting point for the analysis of textual data. We have illustrated the application of quantitative text analysis with data from annual Irish budget debates during the period from 1987 to 2013. Our investigation of legislative positioning over austerity debates has explored the differences in preferences for austerity expressed by legislators whose votes on the budget fail to reveal any differences in their preferences due to strict party discipline.

Our results demonstrated how positional information about the relative policy preferences of individual speakers, and specifically members of a governing coalition responsible

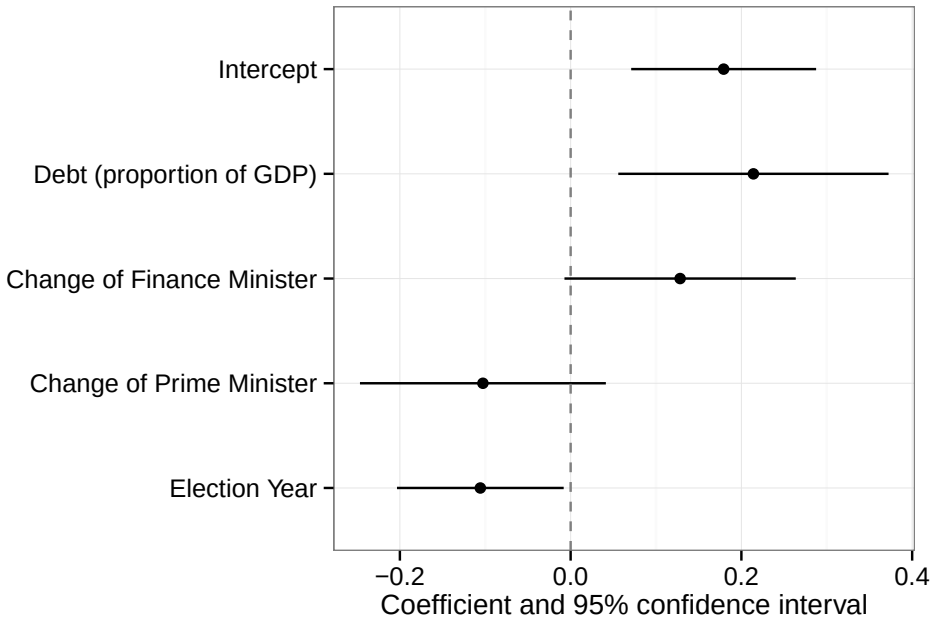

Figure 8: Results of OLS regression of distance between cabinet members and government backbenchers on constituency and economic variables.

for implementing painful budget cuts, can be measured from speeches on the legislative floor using text as data. Our results have a high degree of face validity when compared to known legislative positions, with government ministers being most supportive of the budgets, opposition speakers most opposed, and government backbenchers in between. Text scaling as used here provides a valid method for measuring intra-party differences as expressed in speeches made during debates over annual budgets.

Quantitative text analysis is a new and exciting research tool that allows social scientists to generate quantities of interest from textual data. Like any method in the social sciences, the estimation of these quantities requires a theory guided research design and careful validation of the results. One particular problem in the analysis of political texts is potential selection bias. The decision to speak or to publish a document is often the result of strategic considerations, which, if not accounted for, can bias results. In the legislative context, for example, party leaders may strategically decide who (and who is not) allowed to speak (Proksch and Slapin, 2012). We furthermore must be careful when interpreting the

results and not equate the measures we estimate with the "true" preferences of politicians, which, one may argue, are inherently unobservable. Texts produced in the policy-making process are, again, the result of strategic considerations of single politicians or groups of actors. But this is true for any method that attempts to measure policy preferences and reinforces the need to validate the results with external measures.

References

Baek, Y. M., J. N. Cappella, and A. Bindman (2011). Automating content analysis of open-ended responses: Wordscores and affective intonation. *Communication methods and measures 5*(4), 275–296.

Benoit, K. and M. Laver (2007). Compared to what? A comment on 'A robust transformation procedure for interpreting political text' by Martin and Vanberg. *Political Analysis 16*(1), 101–111.

Benoit, K., M. Laver, C. Arnold, P. Pennings, and M. O. Hosli (2005). Measuring national delegate positions at the convention on the future of Europe using computerized word scoring. *European Union Politics 6*(3), 291–313.

Bergin, A., J. F. Gerald, I. Kearney, and C. O'Sullivan (2011). The Irish fiscal crisis. *National Institute Economic Review 217*, 47–59.

Blei, D., A. Ng, and M. Jordan (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research 3*, 993–1022.

Carrubba, C., M. Gabel, and S. Hug (2008). Legislative voting behavior, seen and unseen: A theory of roll-call vote selection. *Legislative Studies Quarterly 33*(4), 543–572.

Carrubba, C. J., M. Gabel, L. Murrah, R. Clough, E. Montgomery, and R. Schambach (2006). Off the record: Unrecorded legislative votes, selection bias and roll-call vote analysis. *British Journal of Political Science 36*(04), 691–704.

Clinton, J., S. Jackman, and D. Rivers (2004). The statistical analysis of roll call data. *American Journal of Political Science 98*(2), 355–370.

Gallagher, M. (2009). Parliament. In J. Coakley and M. Gallagher (Eds.), *Politics in the Republic of Ireland* (5th ed.). London: Routledge.

Gallagher, M., M. Laver, and P. Mair (2011). *Representative Government in Modern Europe* (5th ed.). McGraw-Hill.

Giannetti, D. and M. Laver (2005). Policy positions and jobs in the government. *European Journal of Political Research 44*(1), 91–120.

Grimmer, J. (2010). A bayesian hierarchical topic model for political texts: Measuring expressed agendas in Senate press releases. *Political Analysis 18*(1), 1–35.

Grimmer, J. and B. M. Stewart (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis 21*(3), 267–297.

Hakhverdian, A. (2009). Capturing government policy on the left–right scale: Evidence from the United Kingdom, 1956–2006. *Political Studies 57*(4), 720–745.

Herzog, A. and S. Mikhaylov (2013). DPSI: Database of Parliamentary Speeches in Ireland. Working Paper.

Hix, S., A. Noury, and G. Roland (2005). Power to the parties: cohesion and competition in the European Parliament, 1979–2001. *British Journal of Political Science 35*(02), 209–234.

Hug, S. (2010). Selection effects in roll call votes. *British Journal of Political Science 40*(1), 225–235.

Klemmensen, R., S. B. Hobolt, and M. E. Hansen (2007). Estimating policy positions using political texts: An evaluation of the Wordscores approach. *Electoral Studies 26*(4), 746–755.

Klüver, H. (2009). Measuring interest group influence using quantitative text analysis. *European Union Politics 10*(4), 535–549.

Lauderdale, B. E. and A. Herzog (2014). Measuring political positions from legislative speech. Text as Data Conference, October 10–11, 2014, Chicago.

Laver, M. and K. Benoit (2002). Locating TDs in policy spaces: the computational text analysis of Dáil speeches. *Irish Political Studies 17*(1), 59–73.

Laver, M., K. Benoit, and J. Garry (2003, May). Extracting policy positions from political texts using words as data. *American Political Science Review 97*(2), 311–331.

Laver, M., K. Benoit, and N. Sauger (2006). Policy competition in the 2002 French legislative and presidential elections. *European Journal of Political Research 45*(4), 667–697.

Laver, M. and J. Garry (2000). Estimating policy positions from political texts. *American Journal of Political Science 44*(3), 619–634.

Lowe, W. and K. Benoit (2013). Validating estimates of latent traits from textual data using human judgment as a benchmark. *Political Analysis 21*(3), 298–313.

Manning, C. D., P. Raghavan, and H. Schütze (2008). *Introduction to Information Retrieval*. Cambridge University Press.

Martin, L. W. and G. Vanberg (2007). A robust transformation procedure for interpreting political text. *Political Analysis 16*(1), 93–100.

McElroy, G. and K. Benoit (2010). Party policy and group affiliation in the European Parliament. *British Journal of Political Science 40*(2), 377–398.

Monroe, B. and K. Maeda (2004). Talk's cheap: Text-based estimation of rhetorical ideal-points. POLMETH Working Paper.

Pennebaker, J. W., R. J. Booth, and M. E. Francis (2007). Linguistic inquiry and word count (LIWC2007). http://LIWC.net.

Poole, K. T. and H. Rosenthal (1997). *Congress: A Political-Economic History of Roll Call Voting*. Oxford/New York: Oxford University Press.

Proksch, S.-O. and J. B. Slapin (2010). Position taking in European Parliament speeches. *British Journal of Political Science 40*, 587–611.

Proksch, S.-O. and J. B. Slapin (2012). Institutional foundations of legislative speech. *American Journal of Political Science 56*(3), 520–37.

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

Slapin, J. B. and S.-O. Proksch (2008). A scaling model for estimating time-series party positions from texts. *American Journal of Political Science 52*(3), 705–722.

Stewart, B., E. Airoldi, and M. Roberts (2011). Topic models and structure. American Political Science Association (APSA) Annual Meeting, 2011.

Strøm, K. and W. C. Müller (2009). Parliamentary democracy, agency problems, and party politics. In D. Giannetti and K. Benoit (Eds.), *Intra-Party Politics and Ccoalition Governments in Parliamentary Democracies*. London: Routledge.

VanDoren, P. M. (1990). Can we learn the causes of congressional decisions from roll-call data? *Legislative Studies Quarterly 15*(3), 311–340.

Warwick, P. V. (2015). Public opinion and government policy in britain: A case of congruence, amplification or dampening? *European Journal of Political Research 54*(1), 61–80.