Predicting the Impact of Legislative Texts: An Application of Supervised Machine Learning to Statutory Instruments in the United Kingdom, 2005-2015

Radoslaw Zubek, University of Oxford, radoslaw.zubek@politics.ox.ac.uk

Abhishek Dasgupta, University of Oxford, abhishek.dasgupta@cs.ox.ac.uk

David Doyle, University of Oxford, david.doyle@politics.ox.ac.uk

Abstract

Political scientists are increasingly interested in prediction. In this paper, we offer a logistic regression machine-learning model to predict the importance of UK statutory instruments (SIs), a common form of delegated executive legislation. We use data on Google search hits to determine an exogenous measure of legislative significance. We develop a computational algorithm on a set of textual and non-textual features based on a training set of more than 10,000 SIs from 2005-2014. We then demonstrate how our model can, with some success, predict the importance of SIs from 2015. The results reported here provide some early indication of the validity of our approach to predicting legislative impacts.

1 Introduction

Successfully predicting outcomes, most notably election results, has become something of a Holy Grail for political science. Early efforts focused on polling and public approval data (Brody & Sigelman, 1983), or the trend in GDP per capita (Hibbs Jr, 1982), or some combination of polling data and economic conditions (Lewis-Beck & Rice, 1984; Abramowitz, 1988), to predict electoral outcomes, with often limited success. Today, popular blogs like FiveThirtyEight (Silver, 2010) have become synonymous with electoral prediction through their statistical aggregation of multiple polls (for an overview, see Blumenthal, 2014). But it has really been the evolution of social media, which has provided political science with its main engine for electoral prediction. Probably the most prominent early example of this now flourishing industry used mentions of political parties on Twitter to predict the vote share of German parties (Tumasjan *et al.* , 2010). More recently, political scientists have turned to machine learning in order to train algorithms on existing polls or electoral outcomes as a function of public sentiment on Twitter to predict election results (for example, see Bermingham & Smeaton, 2011; Huberty, 2013; Livne *et al.* , 2011; Sang & Bos, 2012; Beauchamp, 2016). To some extent, the prediction of electoral outcomes is well facilitated by the vast array of public sentiment and existing polling data readily available to researchers.

In this paper, we are also interested in prediction, although what we wish to predict is slightly more complicated than electoral outcomes. A branch of political science has become increasingly

interested in identifying and classifying 'important' or 'significant' legislation. Beginning with the work of Chamberlain (1946) mid way through the twentieth century, a number of different coding schema have now been proposed in order to classify 'significant' or 'important' legislation. These range from the seminal work of Mayhew (1991), which established the template for classifying legislative importance and which has spawned numerous variations (e.g. Coleman, 1999; Edwards III *et al.* , 1997; Binder, 1999), to work on the classification of important legislation in Europe (Blondel, 1970) inspired by a coding scheme to measure the significance of local legislation developed by Polsby (1963), to work on important labor legislation in Western Europe (Scholtz & Trantas, 1995; Döring *et al.* , 1995; Tsebelis, 1999) to work employing more sophisticated Item Response Theory (IRT) models to classify important US legislation between 1877 and 1994 (Clinton & Lapinski, 2006, 2007).

We combine both the desire to predict and the impetus to identify important legislation. We propose a supervised machine learning approach for the classification of government legislation; specifically, classifying this legislation according to whether it exerts an impact or not. With this machine learning approach to the classification of legislation, we can then use our model to predict, based on textual and non-textual features, the probability that new legislation will *become* important and exert a discernible impact in the future. As far as we are aware, this is the first effort to predict the importance of government legislation.

We also propose a novel method for classification that does not require any human coding whatsoever. By defining impact as an exogenously observed variable, we fully automate our classification and prediction process. Here, we measure impact as the number of Google hits or searches that each piece of government legislation has received since its publication. In this paper, we develop a logistic regression machine learning model to identify the textual and non-textual features that best identify the impact of government legislation, specifically UK statutory instruments, and their accompanying explanatory memoranda, between 2005 and 2014. We then use the textual and non-textual features identified in this classification task to *predict* the impact of UK statutory instruments from 2015 (our test set). Our results indicate support for our strategy.

We think our method will have a number of important implications. It will be relevant for work concerned with the classification of government legislation into different topics (e.g. The Comparative Agendas Project, Baumgartner *et al.* , 2013; John *et al.* , 2013; Hillard *et al.* , 2008), but more specifically for work focused on the classification of important government legislation (Mayhew, 1991; Clinton & Lapinski, 2006, 2007). In addition, we think our proposed method will be of interest to those working on political texts more generally (e.g. Monroe & Maeda, 2004; Slapin & Proksch, 2008; Benoit & Däubler, 2014; Lowe, 2016) and those concerned with classification (Grimmer & King, 2011) and supervised learning (Hillard *et al.* , 2008; Grimmer & Stewart, 2013) in political science.

The paper proceeds as follows. In the next section we briefly oultine work on the classification of important legislation. We then present our text corpora and describe1 our goal: classifying the

impact of this text according to our exogenous measure of impact, and describe the models and features we select to do this, before presenting the results of these models in terms of various metrics, including accuracy of predicting impact of UK statutory instruments from 2015. Our results suggest the viability of our strategy. The final section concludes.

2 Classifying Important Legislation

Most analogous to our purpose here are studies that focus on the classification of 'important', 'significant' or 'notable' legislation. This work is based on the intuitive premise that not all legislation is equal: some government laws are more important than others. Of course, defining what we mean by 'important' or 'significant' legislation is no easy task. For example, in his seminal study, Mayhew (1991) defined important legislation as that which is "innovative and consequential." Unsurprisingly, this definition has received criticism for its vagueness and openness to interpretation (e.g., see Clinton & Lapinski, 2007). Here, we follow the definition of Clinton & Lapinski (2006), who define important legislation as that which "has been identified as noteworthy by a reputable chronicler-rater," and the definition of Blondel (1970), who suggests that the importance of legislation should be "measured by the extent to which they appear to be designed to affect the community." We are interested in legislation that exerts an impact (particularly as described in our explanatory memoranda) and that has been identified as noteworthy by a rater, in our case, Google hits.

Efforts to classify and code important legislation have largely been inspired by scholars seeking to understand exactly how lawmaking functions and who believe that in order do this, we need a comparable measure of 'legislative output' rather than measures of legislator behavior, such as roll-call voting (e.g., see Lapinski, 2008, p. 236). A major theoretical impetus for much of this work has been to understand the effect of unified vs. divided government on the production of important legislation (e.g. Mayhew, 1991; Edwards III *et al.* , 1997). One of the first studies to classify legislation according to its significance was Chamberlain (1946), who selected the most important US laws across a number of different policy areas, although the means by which bills were deemed important primarily rested on the post-hoc rationalization of Chamberlain (1946) himself. Blondel (1970) provided an early coding scheme for the identification of important legislation in four European countries, and India. He developed an index of legislative importance with a coding scheme inspired by Polsby (1963), and assigned values to each piece of legislation based on the anticipated impact of legislation on the wider community, and its desired result, together with the length of the bills. For all Western European countries, Scholtz & Trantas (1995), Döring *et al.* (1995) and Tsebelis (1999) used the International Labor Organization database, NATLEX, to identify all labor legislation, and from this list, by cross-referencing NATLEX with the *Encyclopaedia of Labor Law* developed by Blanpain (1977), they were able to identify which labor legislation could be deemed significant, a remarkably time-intensive task conducted manually by one the researchers themselves (Scholtz & Trantas, 1995, p. 639).

The seminal study however, in the classification of notable or significant legislative output is that of Mayhew (1991) in his book, *Divided We Govern*. Motivated by the desire to explore the effect

of divided government, relative to unified government, on the importance of legislative output (his study actually finds no major effect), Mayhew (1991) used a series of contemporaneous evaluations of important legislation, including the *New York Times* and *Washington Post* (his Sweep One), in addition to retrospective evaluations of whether legislation has proven to be important based on the assessments of public policy scholars (his Sweep Two), in order to classify important bills. This data has remained the go-to source for work on legislative output in the US Congress and this classification scheme has remained the template for further classification exercises on US legislation (e.g. Coleman, 1999; Edwards III *et al.* , 1997; Binder, 1999).

There have been some notable additional efforts. The *Policy Agendas Project* classifies the importance of legislation according to the number of column lines about each piece of legislation in the *Congressional Quarterly Almanac* (Baumgartner & Jones, 2002). In a series of studies Clinton & Lapinski (2007, 2006) and Lapinski (2008) employ a range of retrospective and contemporaneous ratings of legislative importance from the wider literature, including those of Mayhew (1991) and Baumgartner & Jones (2002) etc., and use IRT models to determine the underlying latent dimension of importance among the various different measures of legislative importance produced by 'chronicler-raters.' It also allows for the generation of concomitant standard errors for each measure of importance for each individual statute.

While the statistically sophisticated work of Clinton & Lapinski (2007, 2006) is a major advance in the classification of important legislation, the original rating of legislation, both contemporaneously and retrospectively, is still all conducted by human coders. This is the same for the work of Mayhew (1991), Blondel (1970), Baumgartner & Jones (2002), Scholtz & Trantas (1995), Döring *et al.* (1995), Tsebelis (1999) and Edwards III *et al.* (1997). As such, the classification of legislation by importance remains a remarkably time and labor-intensive endeavor. Furthermore, the vast majority of this literature has been developed for the US Congress and many of the contemporaneous sources employed in this work are solely US based, for example the *Congressional Quarterly Almanac*, thus undermining the applicability of these coding schemes for other legislatures, thereby limiting the potential for comparative studies on the production of important legislation. An exception in this regard is the work of Blondel (1970) and Scholtz & Trantas (1995), Döring *et al.* (1995) and Tsebelis (1999). Finally, given how difficult classification has proven to be, unlike the large literature on electoral forecasting, no one has yet attempted to *predict* the importance or impact of new legislation.

We believe our work can help advance this literature in three main ways. Firstly, we propose a fully automated solution for classifying the impact or importance of legislation. Our proposed use of an exogenous measure of impact, i.e. Google hits, allows us to train our features in a fully automated manner, without the need for intensive hand-coding. Some work has gone a long way towards classifying government legislation by topic or issue dimension, most notably long-running projects such as the Congressional Bills Project (Adler & Wilkerson, 2008) and The Comparative Agendas Project (Baumgartner *et al.* , 2013; John *et al.* , 2013; Hillard *et al.* , 2008), although in both projects the mammoth undertaking of classifying bills and legislation by topic has largely been undertaken by time-intensive hand-coding (e.g. Adler & Wilkerson,

2008). There are some studies that have used supervised learning methods to classify political text, including efforts to classify the public statements of Russian political elites into restrained, activist or neutral categories (Grimmer & Stewart, 2013) and machine learning models that classified US Congressional Bills into their main topics, as defined by the Congressional Bills Project Congressional Bills (Hillard *et al.* , 2008). Even in these studies however, human coding has remained an essential feature to classify the test set. Although these methods are much faster and just as effective, they still do not completely obviate the necessity for some form of hand-coding.

Secondly, we derive a set of features from our machine learning model that will enable us to predict whether future legislation is likely to *become* important and exert an impact. We can generate such a prediction within minutes after a piece of legislation is produced. Thirdly, by using Google hits to classify legislation, we develop a method that is easily replicable and applicable to other legislative environments, and which lends itself to comparative work. We present this method in the next section.

3 Methods

We shall use a supervised machine learning algorithm to learn an output variable from input features. This section is organized as follows: section 3.1 describes the data source, section 3.2 describes the output feature used in the problem, section 3.3 describes the input features, section 3.5 describes the chosen model, section 3.6 describes preparation of the dataset and training, with metrics and evaluation in section 3.7.

3.1 Data

In this paper, we analyze statutory instruments (SIs), which are the dominant form of secondary legislation in the United Kingdom and are used to provide detail that would be too complex to include in the primary legislation. There are many different types of statutory instruments including regulations, directions, orders and licences. SIs are adopted by Whitehall departments under UK laws, but they can also be adopted by devolved executives as laws passed by regional legislatures. Whitehall-adopted SIs may have regional or national coverage, while instruments passed by regional executives, by definition, have local coverage. In this paper we are only concerned with Whitehall-adopted SIs with both national and regional coverage.

Each statutory instrument (SI) is obtained as an XML file from the `www.legislation.gov.uk` website. We choose the period 2005–2015 to study secondary legislation due to the obligatory presence of explanatory memoranda for most SIs since 2004. An explanatory memorandum (EM) is a document associated with each SI, which contains detailed descriptions of proposed policies, but crucially for our purposes, they also contain (since June 2004 in the UK) an assessment of whether a given piece of legislation will have an impact on business (including small and medium sized enterprises) and the voluntary or public sector and whether this effect will be positive or negative in cost terms. Such assessments reflect the government's best estimate of the expected impact of its legislation. This assessment is very much in line with the understanding

of legislative importance developed by Blondel (1970), based on the assumption that important legislation should be "measured by the extent to which they appear to be designed to affect the community." Using the EMs allows us to obtain textual features for the model, which we explore in section 4.2.

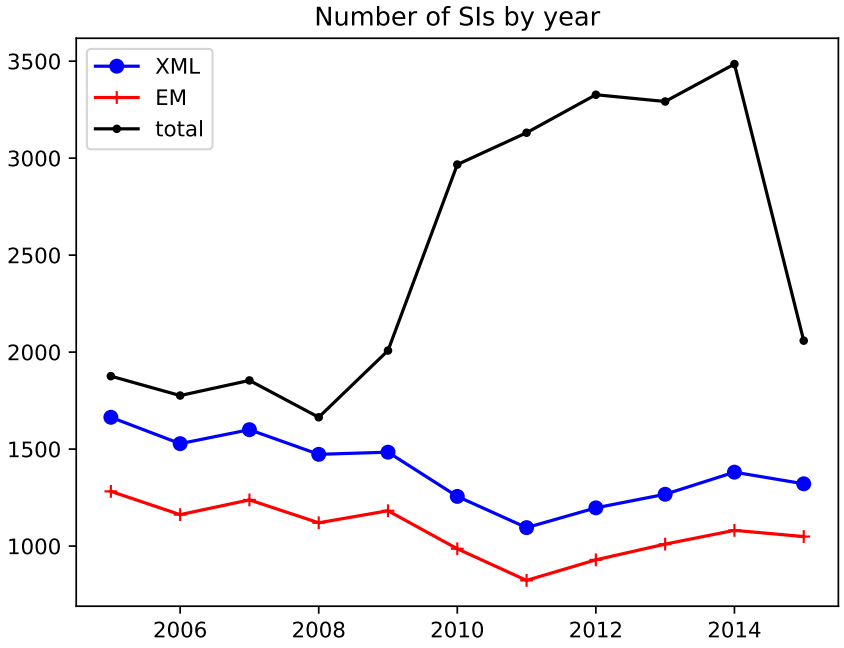

Figure 1 *Number of SIs by year*

For the period of 2005–2015, there are a total of 27439 SIs. Out of these, we only consider the subset of SIs which have been converted to XMLs (15266 SIs) and have an explanatory memorandum attached (not all SIs have explanatory memoranda). This brings the number of actual SIs for our dataset to 11862. The number of SIs per year is shown graphically in fig. 1. The discrepancy in the actual number of SIs and the number of SIs available as XML is due to the fact that not all SIs are required to be published as XML; thus unimportant SIs are only available in PDF format.

3.2 Output variable

As a proxy for the impact of legislation, we use the number of Google hits for each SI. We see this measure as akin to earlier studies that attempted to classify the importance of legislation by some count of mentions, or according to subjective assessments, in various news sources, documents and policy histories. For example, Baumgartner & Jones (2002) in their *Policy Agendas Project* use the number of column lines about each piece of legislation in the *Congressional Quarterly Almanac*, while in 'Sweep One', Mayhew (1991) uses the *New York Times* and the *Washington Post*; similarly, Clinton & Lapinski (2007) use a method based on the mentions, and classifications, of US legislation in various chronicler-raters. We see Google hit data as analogous

to the chronicler-raters of Clinton & Lapinski (2006). We obtained Google hit data for each SI published from 2005–2015 in December 2016. While Google hit counts are noisy and subject to fluctuation (e.g. Mayr & Tosques, 2006), we mitigate this issue by (i) using a single snapshot of the Google hit counts (ii) predicting impact or no-impact as a binary variable based on Google hits being above a certain threshold, which reduces the effect of noisy data. We use four query variants while obtaining the Google hits:

- quoted: quoted SI title;
- brackets-out: SI title in quotes, but with brackets moved into separate terms, as well as separating the type of SI and year into its own search term. For example, the brackets-out query corresponding to the SI
  `The Hinkley Point C (Nuclear Generating Station) (Amendment) Order 2015`; would be
  `"Hinkley Point C" "Nuclear Generating Station" "Amendment" "Order 2015"`;
- subset-quoted: same as quoted, with sites restricted to .com and .co.uk;
- subset-brackets-out: same as brackets-out with sites restricted to .com and .co.uk.

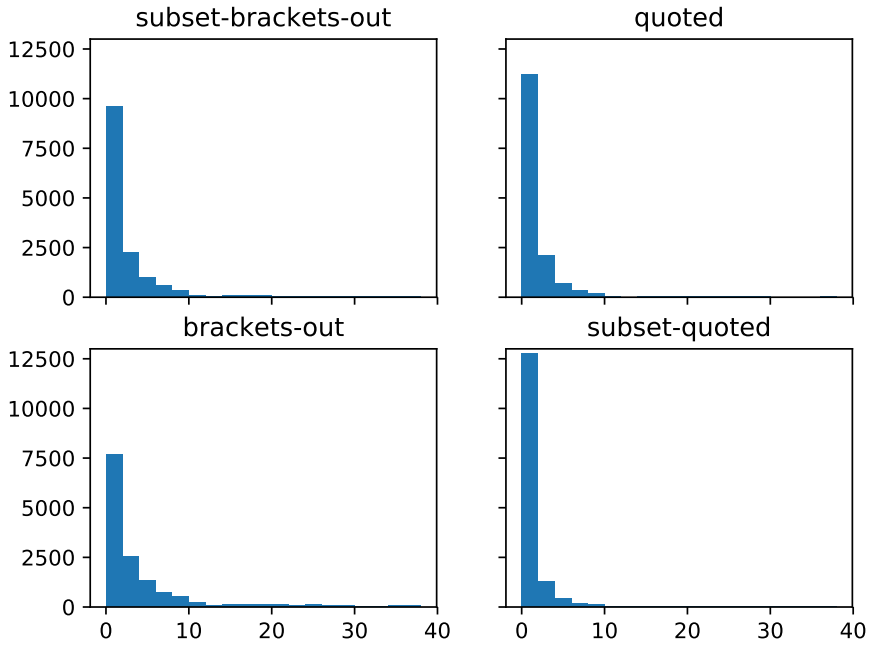

Figure 2 *Frequency distribution of Google hit data.*

Using different Google search types allows us to observe their effect and obtain the best dataset – the one that consistently outperforms the others with respect to our chosen metrics. We also drop the leading 'The' and always exclude search results from the gov.uk sites as these are usually the canonical references to the secondary legislation which are always present. Figure 2 shows the frequency distribution of Google hit data for the four different datasets (quoted, brackets-out, subset-quoted, subset-brackets-out).

3.3 Input features

For the input features, we chose the following features obtained from the XML files:[1]

- `t`: age of the SI, measured in months from the time when it was published to December 2016, when the searches were performed – this was scaled by a tanh transformation to limit its values to [0, 1].
- `laid-before-parliament`: whether the SI was laid before parliament
- `department (d)`: SI department
- `subject (s)`: SI subject (like "income tax" or "highways")
- `location (l)`: extent of application of the SI (England, Wales, Scotland, Northern Ireland or whole of UK)
- `amendment`: whether the SI is an amendment
- `type (t)`: type of SI (rules, order or regulations)

3.4 Textual Features

As explained above, one of the key motivations for using SIs from 2005–2015 is access to EMs. Each EM has an impact section, which describes the impact of the statutory instrument on businesses, charities, voluntary bodies and the public sector. After filtering out EMs without an impact section and EMs which do not follow the prescribed format, we are left with 11862 EMs. The impact section is usually a combination of some texts derived from a template and some free text. We lemmatise[2] the impact text to reduce variations. We also remove stopwords such as "the", "has", etc. giving us a total of 402,576 words.

We construct the bigram frequency matrix from this text corpus which leaves us with 104,520 bigrams. Out of these we choose the top 10000 bigrams and compute the tf-idf matrix. The tf-idf matrix is obtained from the term frequency (tf) matrix by multiplying the inverse of the document frequency (idf) or the number of documents in which the bigram appears, and N is the total number of documents. This leads to common bigrams being assigned lower weights. We also normalise the tf-idf row vectors (eq. 2) to control for the effect of varying text lengths, giving us textual features $\mathbf{K}$ with rows (d) in EMs and columns (t) as bigrams:

$$\mathbf{K'}_{dt} = \mathbf{T}_{dt}\left(\log\frac{N}{\mathbf{D}_t}+1\right) \tag{1}$$

$$\mathbf{K}_i = \frac{\mathbf{K'}_i}{\sqrt{\mathbf{K'}_{i1}+\mathbf{K'}_{i2}+\cdots+\mathbf{K'}_{in}}} \tag{2}$$

[1] Here the features with bracketed single-letter abbreviations are categorical features; the rest (except `t`) are binary features. For the categorical features, the single letter abbreviation is used to indicate feature importances for labels in that category (as an example: `d/home-office` refers to the department label for the Home Office.) Later we shall consider textual features (bigrams) – these are indicated in feature importances tables by the prefix `w`. Feature interactions are indicated by the presence of multiple features in the same label, like `l/england amendments` representing the interaction between the location:England and the amendments feature.

[2] Lemmatisation is reducing a word to its *lemma* or root, changing "walking" to "walk", "better" to "good".

Here, $\mathbf{T}$ is the term-frequency matrix and $\mathbf{D}$ is the document vector, with has the number of documents in which term t appears. The tf-idf matrix $\mathbf{K}'$ differs from the standard definition by adding 1 to the idf (second term in eq. 1) instead of the idf's denominator; this avoids zero division errors.

We shall look at the results from the model incorporating textual features in section 4.2.

3.5 Choice of Model

As this is a supervised classification problem, we can use any of the standard supervised learning algorithms such as random forests, naive Bayes, support vector machines or logistic regression (Bishop, 2006, section 4.3.2) to develop our model.

Out of these, we chose logistic regression due to the following reasons: (i) being a linear classifier, it is more easily interpretable compared to random forests, (ii) it is less time-intensive to train when compared to support vector machines and (iii), while naive Bayes is faster to train and scales well with a number of features, logistic regression outperforms naive Bayes given enough data (Ng & Jordan, 2002).

Logistic Regression. Logistic regression is a supervised learning classifier which predicts a categorical (usually binary) variable – in our case impact or no impact. It models the probability of an input feature $\mathbf{x}$, given particular weights $\mathbf{w}$ as $\sigma(\mathbf{w}^{\mathrm{T}}\mathbf{x})$ where $\sigma(x) = e^x/(1+e^x)$ is the sigmoid function. A key feature of the sigmoid function is that its range is [0,1] with $\sigma(0) = 0.5$ allowing its output to be interpreted as a probability. From this, and a matrix of input features $\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_n]^{\mathrm{T}}$ we can derive the probability or *likelihood* of the entire dataset as a function of the feature weights $\mathbf{w}$. The optimization algorithm (for example, stochastic gradient descent) then tries to maximize the likelihood – or as is usually the case, equivalently minimize the negative log likelihood. We also use a L2 regularization term on the feature weights to penalize higher weights, leaving the final equation to be optimized as in eq. (3). Regularization helps to prevent over-fitting and allows the classifier to generalize to unseen data.

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} + C\sum_{i=1}^{n} \log(\exp(-y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i)) + 1) \tag{3}$$

Here C is the inverse of the regularisation coefficient, and is usually estimated by cross validation by splitting the dataset into test and training. In our case we use $C = 0.9$.

Feature transformations. The output feature we use is Google hits. As Google hits are discrete integers, we need to transform them into binary variables. This is done by considering hits above a certain threshold k to have impact ($y = 1$), while those below this threshold to have no impact ($y = 0$). The choice of k is arbitrary, so we pick four different thresholds 4, 9, 14 and 19 and see the effect on the metrics. Formally, given a SI i and threshold k, with Google hits $\mathbf{H}_i$, our output feature is a function of k:

$$\mathbf{y}_i(k) = [\mathbf{H}_i > k] \tag{4}$$

where $[P]$ is the Iverson bracket which is 1 if P is true and 0 otherwise.

As logistic regression is a linear classifier, it can only properly classify if the two categories are linearly separable in feature space. To address this issue, we interact all features with each other (except t and subject) to introduce nonlinearity. We do not interact time with the other nontextual features as they are binary features and we leave out the SI subject because it made computation of the input feature matrix computationally more expensive. We also note that categorical features have to be one-hot encoded to use them in logistic regression.[3]

3.6 Training

Supervised algorithms have two stages: training and test. We train on the SIs from years 2005–2014 and test on the SIs from 2015. Since the number of 1s (impact) in the output depends on k (eq. (4)), we first look at the impact fraction in the output for various thresholds and across the four datasets in fig. 3. We see that the fraction of important SIs has generally gone up - particularly if we choose a threshold of 4. Classifiers also need balanced datasets to reduce the bias of the classifier towards a particular category (impact or no impact). As an example, a classifier trained on a dataset with 10% positive samples and 90% negative samples (or vice-versa) can predict the entire dataset to be one class and still get a high accuracy.

Thus, we chose a scheme of yearly balanced dataset for training where we pick an equal number of impact and no impact cases for each year from 2005–2014. This avoids classifier bias towards any particular year as well as towards any particular category. We also chose an equal number of impact and no impact cases for the development test set. The development test set is used for cross validation and choosing the best model (for each threshold and Google dataset), which we use to report metrics on the test set, that is, SIs from 2015.

3.7 Evaluation and Metrics

The classifier is trained and evaluated on the development test set for 15 iterations. We reshuffle the training and development test set on every iteration and use the best model to predict the test set. We use the AUC (area under curve) metric, a standard metric for binary classifiers, in addition to accuracy, specificity and the true positive rate. The latter three metrics can be derived from the confusion matrix C_{ij}, which represents the number of cases where a sample with true label i was classified as j. Then we have:

Accuracy	True positive rate (TPR)	Specificity (SPC)
$(C_{00}+C_{11})/\sum C_{ij}$	$C_{11}/(C_{11}+C_{10})$	$C_{00}/(C_{00}+C_{01})$

Once we have obtained models for each of the Google hit datasets and threshold, we choose the model with the best TPR. We choose TPR as our discriminatory metric as we would rather mark an unimportant SI as important rather than vice-versa. With the best model we obtain, we then use our features to predict the impact of the test set of UK SIs from 2015.

[3]One-hot encoding is usually used to represent categorical features as input to linear classifiers like logistic regression. For a category variable C with possible values from $\{c_1, c_2, \ldots, c_n\}$, the representation of c_k is a vector of size n (where n is the number of possible values of the categorical variable), with the kth position being 1, and the rest 0. Formally, the representation of c_k is thus $\{\delta_{ik} \mid 1 \leq i \leq n\}$, where δ_{ik} is the Kroenecker delta.

a. *quoted datasets* **b.** *brackets-out datasets*

Figure 3 *Fraction of important SIs with year, for various thresholds.*

4 Results

4.1 Nontextual model

The results of training our model on the dataset from 2005–2014 and testing our model predictions on the SIs from 2015 are shown in the radar graph in fig. 4.

We can see from the radar graphs and the associated table that the dataset 'subset-quoted' with threshold 9 gives the best TPR of 0.8. As all the features except the age of the SI are binary features, we can use the coefficients as a measure of importance of the features. The ten most important features, together with the twenty least important features, can be in appendix.

According to these tables in the appendix, it is the SI subject which is crucial to the importance of the SI – this is not surprising as we would intuitively expect some subjects to be much more important than others.

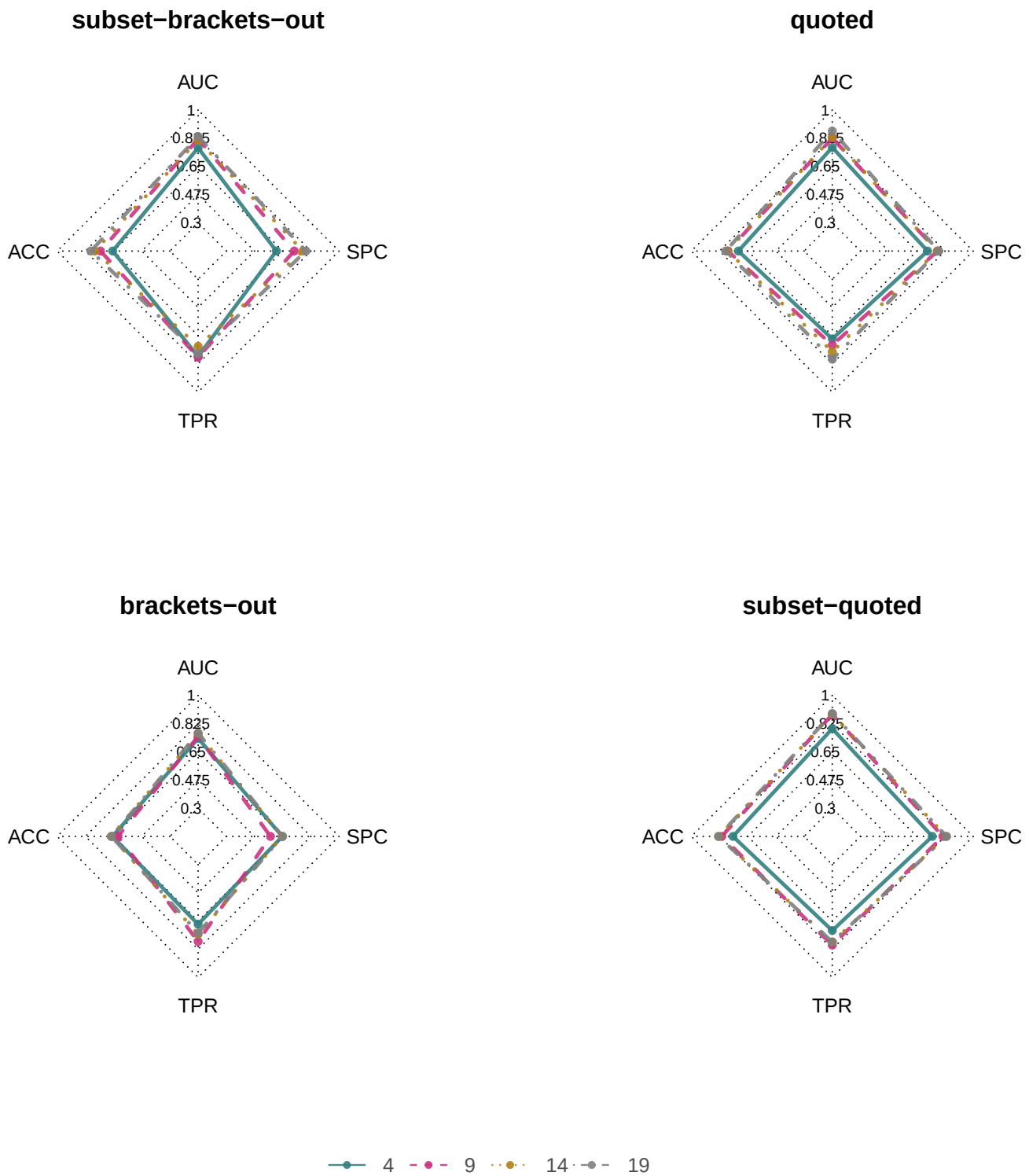

Figure 4 *Radar graphs for the nontextual model. The four metrics of AUC, accuracy (ACC), true positive rate (TPR) and specificity (SPC) are shown on the four axes.*

4.2 Model with textual features

The radar graph for the models with textual features only is shown in fig. 5. As the radar graph shows however, the TPR is less than 0.5. Nonetheless, we get a large improvement in SPC, which is not a surprise as it is easier to detect EMs (and thus their corresponding SIs) with no impact by the presence of bigrams from the set texts, like "minimal impact". The feature contributions, which in this case are the bigrams, that contribute most to increasing and decreasing impact can be found in the appendix.

4.3 Model with nontextual and textual features

As the textual features greatly improve detection of SIs without impact (increase in specificity), we would expect a corresponding effect when combined with the nontextual features. Indeed, this turns out to be the case as shown in fig. 6. In contrast to the previous best case (threshold 9, on the *uk-quoted* dataset), the best TPR here (for the same case) decreases to 0.7636 from 0.8000. However we get increases in specificity (0.8119 to 0.8833), accuracy (from 0.8112 to 0.8770) and AUC (from 0.8821 to 0.8988), so this is a better result overall and an acceptable tradeoff, which improves accuracy significantly. The metrics and confusion matrix for our best model are presented in table 1.

AUC	0.8988	ACC	0.8770
TPR	0.7636	SPC	0.8833

a. *Metrics*

	no impact	impact
no impact	**878**	116
impact	13	**42**

b. *Confusion matrix, numbers in bold indicate correct classification*

Table 1 *Metrics and confusion matrix for the best model (threshold 9, Google dataset subset-quoted)*

Feature importances in the full model. It is interesting to note the relative feature importances in the full model, which considers both nontextual and textual features. Since the textual features and nontextual features are on different scales (nontextual features are binary, except age, whereas textual features are the normalized tf-idf vectors), we look at the top and bottom ten nontextual (tables 2 and 3) and textual features (tables 4 and 5) separately. As expected, the important features from the nontextual model generally retain their position; and subject remains a key determining factor, both in terms of increasing the probability of impact (top 10), or decreasing this probability (bottom 10). Using the bigrams allows the classifier to learn from the impact statement – this shows up in features like "ria attached" (ria is short for regulatory impact assessment), "sector impact" and "saving million". On the other hand, bigrams like "minimal impact", "nil impact", "no regulatory" (from the set text "no regulatory impact on businesses etc.") reduce the probability of impact.

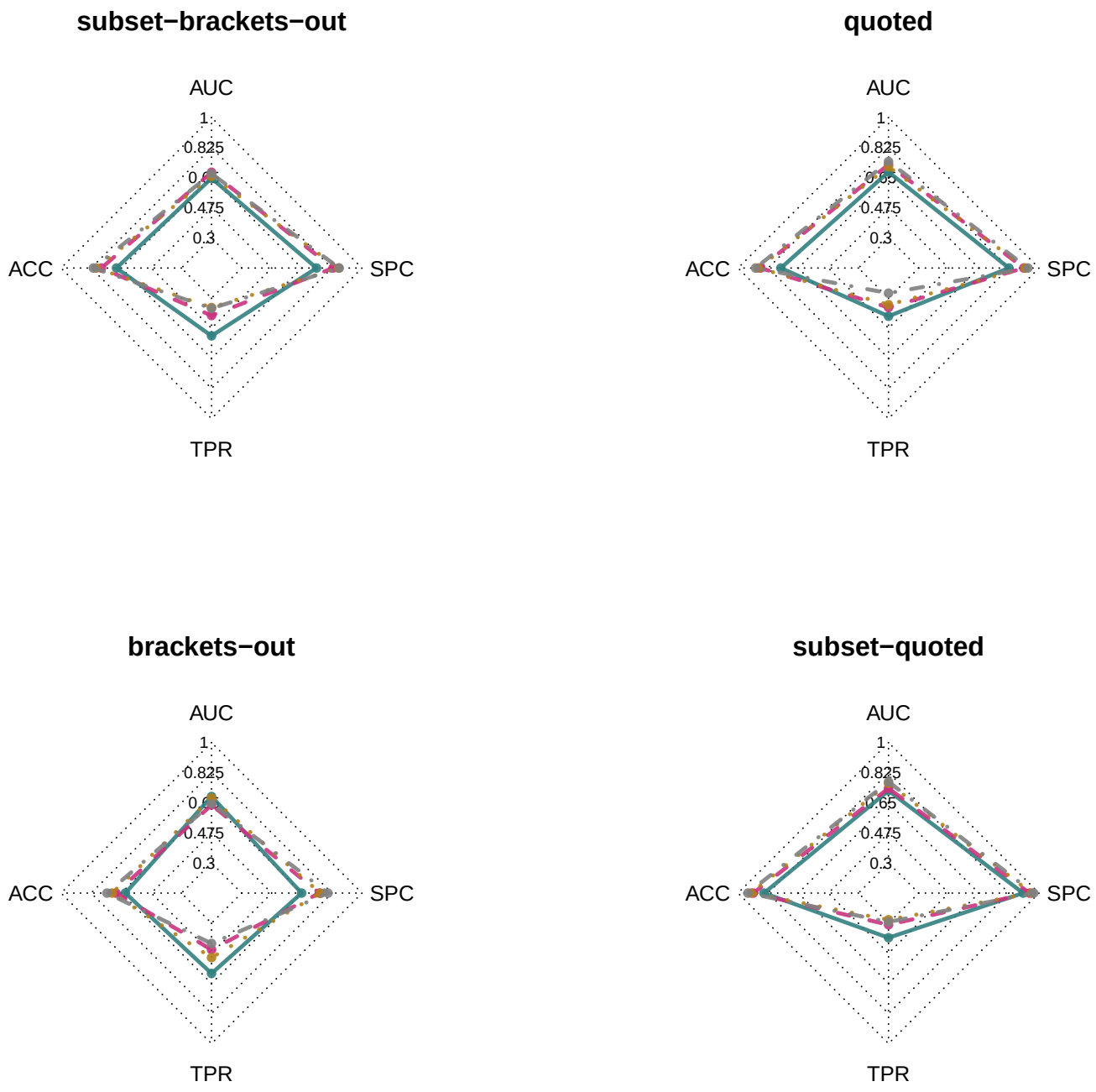

Figure 5 *Radar graphs for the model with textual features only. The four metrics of AUC, accuracy (ACC), true positive rate (TPR) and specificity (SPC) are shown on the four axes.*

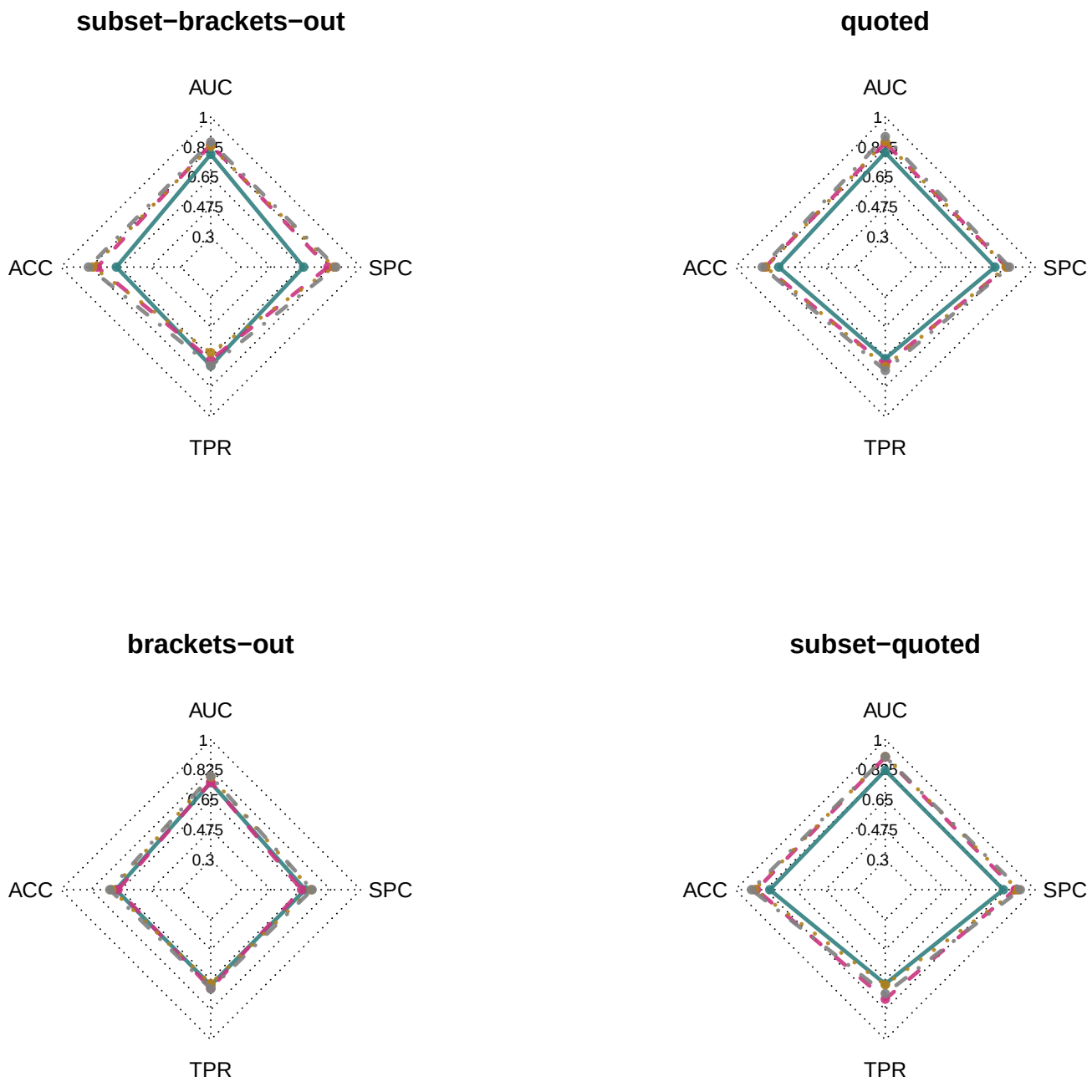

Figure 6 *Radar graphs for the full model with both nontextual and textual features. The four metrics of AUC, accuracy (ACC), true positive rate (TPR) and specificity (SPC) are shown on the four axes.*

coefficient	labels
2.945382	s/public-procurement
2.869478	s/dentists
2.531924	s/financial-service
2.015708	s/prevention-nuclear-proliferation
2.007383	s/south-atlantic-territories
1.750775	s/health-safety
1.707233	s/caribbean-north-atlantic-territories
1.642955	s/civil-aviation
1.637356	s/marine-management
1.589784	s/inheritance-tax

Table 2 *Feature importances for the full model, nontextual features, top ten.*

coefficient	labels
-1.097465	s/customs
-1.097990	s/social-security
-1.135556	s/energy-conservation
-1.160055	l/england amendments
-1.188670	s/employment-training
-1.196328	laid-before-parliament d/energy-climate-change
-1.231507	s/land-registration
-1.244139	s/council-tax
-1.258940	s/nationality
-1.271093	s/european-communities

Table 3 *Feature importances for the full model, nontextual features, bottom ten.*

coefficient	labels
2.763053	w/ria attached
2.034421	w/set new
2.004422	w/memorandum appendix
1.913661	w/instrument negligible
1.774121	w/attached no
1.725723	w/body instrument
1.697452	w/result order
1.683182	w/sector impact
1.669292	w/saving million
1.496011	w/negligible additional

Table 4 *Feature importances for the full model, textual features, top ten.*

coefficient	labels
-0.961068	w/zero impact
-0.973862	w/assessment prepared
-0.976673	w/body nil
-1.001358	w/additional cost
-1.014267	w/foreseen impact
-1.051905	w/has not
-1.069880	w/negligible impact
-1.073856	w/full impact
-1.084724	w/cost on
-1.151050	w/body foreseen

Table 5 *Feature importances for the full model, textual features, bottom ten.*

5 Conclusion

Classifying legislation according to its impact or importance is no easy task. Traditionally, it has involved intensive and time-consuming hand-coding by a number of different human coders (e.g. Mayhew, 1991). Given this classification challenge, *predicting* the importance of legislation has proven an insurmountable obstacle. In this paper, we developed a logistic regression machine learning model that has enabled us to identify the textual and non-textual features of UK Statutory Instruments (and their accompanying EMs) between 2005 and 2014 that contribute to classification of importance or impact, as characterized by Google hit data, of UK SIs. We used this model to predict the impact of UK SIs from 2015, with a high degree of accuracy. Our method and machine learning model could be used to predict the impact or importance of new and future legislation, within minutes of its publication.

We contribute to existing work on the classification of legislation in two main ways. Firstly, we have a developed a method for classifying important legislation that obviates the need for time and labor-intensive hand-coding. By using Google hit data, we eliminate the need for human coding. What is more, while existing work on legislative importance has primarily focused on the US Congress, and has developed 'raters' of importance that are often idiosyncratic to that legislative setting, our proposed method can very easily and quickly be replicated for other legislatures in other countries. As such, we believe that our classification efforts here can help advance work concerned with legislative outcomes (e.g. Mayhew, 1991; Coleman, 1999; Edwards III *et al.* , 1997; Binder, 1999; Clinton & Lapinski, 2006, 2007; Döring *et al.* , 1995; Tsebelis, 1999). Although it remains to be seen how well our method of classifying impact might correlate with that of earlier efforts, our classification method is quick, relatively easy and intuitive.

Secondly, by developing our machine learning model, we believe we can advance existing work on legislative importance, by *predicting* when new legislation is likely to have an impact. In this manner, we are effectively preempting the retrospective raters of (Mayhew, 1991) and in doing so, we think our work may be of interest to other areas of political science that are interested in prediction (e.g. Tumasjan *et al.* , 2010; Bermingham & Smeaton, 2011; Huberty, 2013; Livne

et al. , 2011; Sang & Bos, 2012; Beauchamp, 2016).

Finally, we think the machine learning model we have developed here will be of interest to the wider work on text analysis in political science (e.g. Monroe & Maeda, 2004; Slapin & Proksch, 2008; Benoit & Däubler, 2014; Lowe, 2016), which is increasingly employing machine learning methods to classify political text (e.g. Grimmer & Stewart, 2013; Hillard *et al.* , 2008).

Of course, all that remains is to test the accuracy of our predictions into the future.

6 Acknowledgments

The authors would like to acknowledge the use of the University of Oxford Advanced Research Computing (ARC) facility in carrying out this work[4].

[4] http://dx.doi.org/10.5281/zenodo.22558

References

Abramowitz, Alan I. 1988. An improved model for predicting presidential election outcomes. *PS: Political Science & Politics*, **21**(04), 843–847.

Adler, E Scott, & Wilkerson, John. 2008. *Congressional Bills Project: NSF 00880066 and 00880061.*

Baumgartner, Frank R, & Jones, Bryan D. 2002. *Policy dynamics.* University of Chicago Press.

Baumgartner, Frank R, Green-Pedersen, Christoffer, & Jones, Bryan D. 2013. *Comparative studies of policy agendas.* Routledge.

Beauchamp, Nicholas. 2016. Predicting and Interpolating State-Level Polls Using Twitter Textual Data. *American Journal of Political Science.*

Benoit, Kenneth, & Däubler, Thomas. 2014. Putting Text in Context: How to Estimate Better Left-Right Positions by Scaling Party Manifesto Data using Item Response Theory. *In: Prepared for the âĂIJMapping Policy Preferences from TextsâĂİ Conference, May 15–16, 2014, Berlin.*

Bermingham, Adam, & Smeaton, Alan F. 2011. On using Twitter to monitor political sentiment and predict election results.

Binder, Sarah A. 1999. The dynamics of legislative gridlock, 1947–96. *American Political Science Review*, **93**(03), 519–533.

Bishop, C.M. 2006. *Pattern Recognition and Machine Learning.* Information Science and Statistics. Springer.

Blanpain, Roger. 1977. *International Encyclopaedia for Labour Law and Industrial Relations: Section general: International Monographs, National Monographs; Codex: International labour law; Case Law; European Works Council; Legislation.* Kluwer.

Blondel, Jean. 1970. Legislative Behaviour: Some Steps towards a Cross-National Measurement. *Government and Opposition*, **5**(1), 67–85.

Blumenthal, Mark. 2014. Polls, forecasts, and aggregators. *PS: Political Science & Politics*, **47**(02), 297–300.

Brody, Richard, & Sigelman, Lee. 1983. Presidential popularity and presidential elections: An update and extension. *Public Opinion Quarterly*, **47**(3), 325–328.

Chamberlain, Lawrence H. 1946. The President, Congress, and Legislation. *Political Science Quarterly*, **61**(1), 42–60.

Clinton, Joshua D, & Lapinski, John S. 2006. Measuring legislative accomplishment, 1877–1994. *American Journal of Political Science*, **50**(1), 232–249.

Clinton, Joshua D, & Lapinski, John S. 2007. Measuring significant legislation, 1877–1948. *Pages 361–78 of: Process, Party, and Policymaking: Further New Perspectives on the History of Congress*, vol. 2.

Coleman, John J. 1999. Unified government, divided government, and party responsiveness. *American Political Science Review*, **93**(04), 821–835.

Döring, Herbert, *et al.* . 1995. *Parliaments and majority rule in Western Europe*. Campus Frankfurt.

Edwards III, George C, Barrett, Andrew, & Peake, Jeffrey. 1997. The legislative impact of divided government. *American journal of political science*, 545–563.

Grimmer, Justin, & King, Gary. 2011. General purpose computer-assisted clustering and conceptualization. *Proceedings of the National Academy of Sciences*, **108**(7), 2643–2650.

Grimmer, Justin, & Stewart, Brandon M. 2013. Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political analysis*, 267–297.

Hibbs Jr, Douglas A. 1982. President Reagan's Mandate from the 1980 Elections: A Shift to the Right? *American Politics Quarterly*, **10**(4), 387–420.

Hillard, Dustin, Purpura, Stephen, & Wilkerson, John. 2008. Computer-assisted topic classification for mixed-methods social science research. *Journal of Information Technology & Politics*, **4**(4), 31–46.

Huberty, Mark Edward. 2013. Multi-cycle forecasting of congressional elections with social media. *Pages 23–30 of: Proceedings of the 2nd workshop on Politics, Elections and Data*. ACM.

John, Peter, Bertelli, Anthony, Jennings, Will, & Bevan, Shaun. 2013. *Policy agendas in British politics*. Springer.

Lapinski, John S. 2008. Policy substance and performance in American lawmaking, 1877–1994. *American Journal of Political Science*, **52**(2), 235–251.

Lewis-Beck, Michael S, & Rice, Tom W. 1984. Forecasting presidential elections: A comparison of naive models. *Political Behavior*, **6**(1), 9–21.

Livne, Avishay, Simmons, Matthew P, Adar, Eytan, & Adamic, Lada A. 2011. The Party Is Over Here: Structure and Content in the 2010 Election. *ICWSM*, **11**, 17–21.

Lowe, Will. 2016. Scaling things we can count. *Online verfügbar unter http://dl. conjugateprior. org/preprints/scaling-things-we-can-count. pdf, zuletzt geprüft am*, **16**(2016), 99–132.

Mayhew, David R. 1991. *Divided we govern*. Yale University.

Mayr, Philipp, & Tosques, Fabio. 2006. Google Web APIs-an instrument for Webometric analyses? *arXiv preprint cs/0601103*.

Monroe, Burt L, & Maeda, Ko. 2004. TalkâĂŹs cheap: Text-based estimation of rhetorical ideal-points. *Pages 29–31 of: annual meeting of the Society for Political Methodology*.

Ng, Andrew Y, & Jordan, Michael I. 2002. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in neural information processing systems*, **2**, 841–848.

Polsby, NW. 1963. Community power and political theory.

Sang, Erik Tjong Kim, & Bos, Johan. 2012. Predicting the 2011 dutch senate election results with twitter. *Pages 53–60 of: Proceedings of the workshop on semantic analysis in social media*. Association for Computational Linguistics.

Scholtz, Evi, & Trantas, Georgios. 1995. Legislation on benefits and on regulatory matters: social security and labor matters. *Parliaments and Majority Rule in Western Europe, ed. Herbert Doering. New York: St. Martin's*, 628–53.

Silver, Nate. 2010. Pollster Ratings v4. 0: Methodology. *FiveThirtyEight: Politics Done Right*.

Slapin, Jonathan B, & Proksch, Sven-Oliver. 2008. A scaling model for estimating time-series party positions from texts. *American Journal of Political Science*, **52**(3), 705–722.

Tsebelis, George. 1999. Veto players and law production in parliamentary democracies: An empirical analysis. *American Political Science Review*, **93**(03), 591–608.

Tumasjan, Andranik, Sprenger, Timm Oliver, Sandner, Philipp G, & Welpe, Isabell M. 2010. Predicting elections with twitter: What 140 characters reveal about political sentiment. *ICWSM*, **10**(1), 178–185.