Joint Sentence-Document Model for Manifesto Text Analysis

Anonymous ALTA submission

Abstract

Political parties are essential institutions of democracy. Election programs (so-called manifestos) is a published verbal declaration of the intentions, motives, or views of the political party.[1] Political scientists use manifestos to understand sentence level policy relevant themes discussed and also quantify party's position on the left-right spectrum. Rather than handling the two tasks separately, we propose a joint sentence-document model for sentence level thematic classification and document level position quantification using manifestos in different languages. In-order to handle text from multiple languages we exploit continuous neural embeddings for semantic text representation. We have empirically shown the effectiveness of proposed approach using manifestos from thirteen countries which are written in six different languages.

1 Introduction

Language is the medium for politics and political conflict. During election campaigns, individual candidates and political parties articulate their policy positions through party platforms and manifestos. Elected representatives debate legislation and vote for bills. After laws are passed, bureaucrats solicit comments before they issue regulations. The public discuss and express their views through social media (Twitter, blogs, Facebook), and they also launch campaigns for the change they wish to see through online petitions. News reports document the day-to-day affairs of regional and international issues, in a way it covers information from many other sources. It is apparent that understanding text is needed to know what political actors are speaking or writing. Though political scientists have long recognized the need for using text as data, it has not been explored due the difficulty in handling large amounts of text — since hiring coders to manually annotate all documents was the common approach used. With automated text analysis, use of text for political data analysis is getting increased attention among political scientists (Ruedin, 2013; Gentzkow et al., 2017).

Among many actors, political parties are at the core of contemporary democratic systems. One of the widely used dataset by political scientists includes the Comparative Manifesto Project (CMP) dataset, initiated by (Volkens et al., 2011), that collects party election programs (so-called manifestos) from elections in many countries around the world. The goal of the project is to provide a large data collection to support political studies on electoral processes. A sub-part of the manifestos has been manually annotated at sentence-level with one of over fifty fine-grained political themes, divided into 7 coarse-grained topics (schema is given in Figure 5). Each sentence is segmented if it discusses more than one topic and labeled. While manual annotations are very useful for political analyses, they come with two major drawbacks. First, it is very time-consuming and labor-intensive to manually annotate each sentence with the correct category from a complex annotation scheme. Secondly, coders' preferences towards particular categories might cause annotation inconsistencies and affect comparability between manifestos annotated by different coders (Mikhaylov et al., 2012). (Verberne et al., 2014) is one of the early works using manifesto text for automatic thematic classification, using a topic hier-

[1] https://en.wikipedia.org/wiki/Manifesto

archy defined by Dutch political experts. In (Zirn et al., 2016) authors worked with CMP dataset to do coarse-level topic classification (defined by CMP). They used Markov Logic Networks (MLN) to model sentence adjacency topic smoothness constraint. Nonetheless, manually coded manifestos remain the crucial data source for studies in computational political science (Lowe et al., 2011; Nanni et al., 2016).

Other than the sentence level labels, the manifesto text also has document level signals which quantifies its position in the left-right spectrum — rile, expert survey and voter survey scores (Slapin and Proksch, 2008). Though sentence level classification and document level quantification tasks are inter-dependent, existing works handle them separately. Whereas, we propose a joint sentence-document model to handle both the tasks together. We use quantification and regression inter-changeably in this document.

2 Related Works

The recent adoption of NLP methods had led to significant advances in the field of Computational Social Science (Lazer et al., 2009) including political science (Grimmer and Stewart, 2013). Some popular tasks addressed with political text include party position analysis (Biessmann, 2016); political leaning categorization (Akoglu, 2014; Zhou et al., 2011); stance classification (Sridhar et al., 2014); identifying keywords, themes & topics (Karan et al., 2016; Nallapati et al., 2004; Ding et al., 2011); emotion analysis (Rheault, 2016) and sentiment analysis (Bakliwal et al., 2013). These works use manifestos, political speech, news articles, floor debates and social media posts.

With the advancement of computational resources, large scale comparative political text analysis has gained the attention of political scientists, where the objective is to analyze large amounts of data uniformly in order to make it comparable (Lucas et al., 2015). For example, rather than analyzing the political manifesto of a particular party during an election, mining different manifestos across countries over time can provide deeper comparative insights on political change.

Existing classification models, except (Glavaš et al., 2017), utilize discrete representation of text (i.e., bag of words) and can thus exploit only monolingual data (i.e., train and predict same language instances). Hence, most of the works analyze manifesto text at country level. Recent work has shown the use of neural embeddings for multilingual manifesto text coarse-level topic classification (7 major categories) (Glavaš et al., 2017). In (Glavaš et al., 2017) authors show that multilingual embeddings are more effective for cross-lingual coarse-level manifesto text topic classification using labeled data across languages. In this work, we focus on cross-lingual fine-grained thematic classification (57 categories in total), where we have labeled data across all the languages.

For document level quantification task, many works use label count aggregation of manually annotated sentences as features (Lowe et al., 2011; Benoit and Däubler, 2014) and other works use dictionary based supervised methods (wordscores), unsupervised factor analysis based techniques (wordfish) (Hjorth et al., 2015; Bruinsma and Gemenis, 2017). Since the latter techniques use discrete word representation, they cannot be used for multi-lingual setting. In (Glavas et al., 2017), authors leverage neural embeddings for EU parliament speech text quantification task with two pivot texts for extreme left and right positions. They represent the documents using word embeddings averaged with TF-IDF scores as weights. Rather than handling sentence and document-level tasks separately, we evaluate the utility of solving them together using joint sentence-document model under cross-lingual setting.

3 Background

In the CMP, trained annotators categorize manifesto sentences into one of the 57 fine-grained political categories (shown in Figure 5) that are grouped into seven policy areas: External Relations, Freedom and Democracy, Political System, Economy, Welfare and Quality of Life, Fabric of Society and Social Groups. Political parties either write their promises as a list of sentences (example is given in Figure 4a) or structured as paragraphs (example is given in Figure 4b) which can provide more information related to topic coherence. Also the length of document, measured as number of sentences, varies between manifestos. Mean and standard deviation computed using digitized manifestos (948 in total) from 13 countries — Austria, Australia, Denmark, Finland, France, Germany, Italy, Ireland, New Zealand, South Africa, Switzerland, United Kingdom and United States, is 516.7 ± 667. Variance in the number of sentences across documents in conjunction with class imbalance makes the automated thematic classification a challenging task.

Secondly, a sentence is split into multiple sentences, if it discusses unrelated topics or different aspects of a larger policy. An example sentence split into two is given below

> We need to address our close ties with our neighbours (107) as well as the unique challenges facing small business owners in this time of economic hardship. (402)

Scenarios where split sentences discuss different topics, as in the example given above, is not high in number[2]. Also the segmentation was shown to be inconsistent and to have no effect on quantifying proportion of sentences discussing various topics and document level regression tasks (especially *rile* quantification) (Däubler et al., 2012). Hence, similar to previous works (Biessmann, 2016; Glavaš et al., 2017), we consider the sentence level classification as a multi-class single label problem. We use the segmented text when available (especially for evaluation), and complete sentences otherwise.

A manifesto as a whole can be positioned on the left-right spectrum based on the proportion of topics discussed. We use the *rile* score, which is defined as the difference between count of sentences discussing left and right topics (formulation is given below) (Budge and Laver, 1992).

$$rile = \sum_{r \in R} per_r - \sum_{l \in L} per_l \tag{1}$$

where, R = {104, 201, 203, 305, 401, 402, 407, 414, 505, 601, 603, 605, 606} and L = {103, 105, 106, 107, 202, 403, 404, 406, 412, 413, 504, 506, 701}, "per_{xyz}" denotes share of each topic (xyz) as given in Figure 5, per document.

4 Proposed Approach

We propose a joint sentence-document model to classify manifesto sentences into one out of 57 categories and also quantify document level *rile* score. The joint formulation is not only to capture the task inter-dependencies but also to efficiently use annotations at different levels of granularity (sentence and document) — *rile* score is available for 948 digitized manifestos from 13 countries, whilst sentence level annotations are available only for 235 manifestos. We use Hierarchical Neural Network for modeling sentence level classification and document level regression tasks. The proposed architecture is given in Figure 1. Since the text across countries is multi-lingual in nature, we use neural embeddings to represent words ($emb(w)$). We refer to the total set of manifestos available for training as D, subset of it which are annotated with sentence level labels (one out of 57 classes) as D_s. We denote each manifesto as d, which has l_d sentences s_1, s_2, ..., s_{l_d}.

4.1 Sentence Level Model

We represent each sentence using average embedding of its constituent words.

$$s_i = \frac{1}{|s_i|} \sum_{w \in s_i} emb(w)$$

The average embedding representation is given as input to hidden layer with rectified linear activation units (ReLU) to get the hidden representation, which is defined as

$$H_{s_i} = \max(0, a)$$

where, $a = W_s^T s_i$. Finally, the predictions are obtained using a softmax layer, which takes the hidden representation as input and gives probability of 57 classes as output.

$$\hat{Y}_{s_{ik}} = \frac{e^{H_{s_i}^T w_{pk}}}{\sum_{j=1}^{K} e^{H_{s_i}^T w_{pj}}}$$

We use cross-entropy loss function for sentence level model. For sentences in D_s, with ground truth labels Y_s (one-of-K encoding), the loss function is given as follows

$$L_S(D_s, Y_s) = -\frac{1}{\sum_{i=1}^{D_s} l_{d_i}} \sum_{i=1}^{D_s} \sum_{j=1}^{l_{d_i}} \sum_{k=1}^{K} Y_{s_{ijk}} \ln \hat{Y}_{s_{ijk}}$$

4.2 Joint Sentence-Document Model

Using the Hierarchical Neural Network, we model both sentence level classification and document level regression tasks together. In the combined

[2] In (Däubler et al., 2012), using a sample of 15 manifestos, authors noted that around 7.7% split sentences discuss different topics

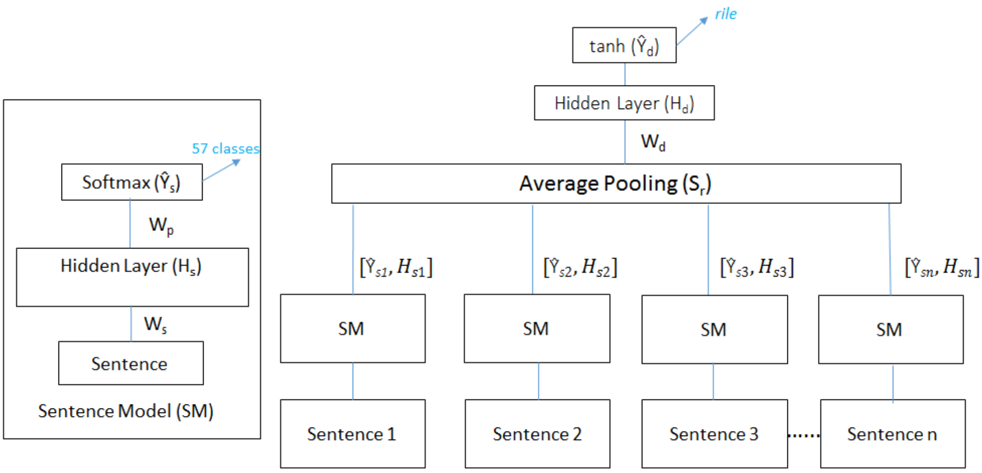

Figure 1: Hierarchical Neural Network for Joint Sentence-Document Analysis. Optimal model parameters we found — $|H_s|$ = 300, $|H_d|$ = 10

model, we use unrolled (time-distributed[3]) neural network model for the sentences in a manifesto. Here, the model minimizes cross-entropy loss for sentences over each temporal layer (1 to n). We use average-pooling to create document representation with individual sentence representations. We concatenate hidden representation (H_s) and predicted output distribution ($\hat{Y}_s$) of sentences before average-pooling[4].

$$S_r = \frac{1}{|l_{d_i}|} \sum_{s \in d_i} [\hat{Y}_s, H_s]$$

rile varies from -100 to +100, we scale it to a range between -1 and +1. So, we use tanh layer finally to get regressed output with $z = H_d = ReLU(W_d^T S_r)$ as input.

$$\hat{Y}_d = tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

Since it is a regression task, we minimize mean-squared error loss function between the predicted $\hat{Y}_d$ and actual rile scores Y_d.

$$L_d(\hat{Y}_d, Y_d) = \frac{1}{|D|} \sum_{d=1}^{D} (\hat{Y}_d - Y_d)^2$$

Overall, the combined loss function for the joint model is

$$L_s(D_s, Y_s) + L_d(\hat{Y}_d, Y_d) \qquad (2)$$

We evaluate both cascaded and joint training for this objective function. We use Adam optimizer[5] for parameter estimation.

- Cascaded Training: In this approach sentence level model is trained using D_s, to minimize $L_s(D_s, Y_s)$, and the pre-trained model is used to obtain document level representation S_r for all the manifestos in training set, D. Then the document level regression task is trained to minimize $L_d(\hat{Y}_d, Y_d)$. Here, the sentence level model parameters are fixed when the document level regression model is trained using S_r.

- Joint Training: Here the entire network is updated by minimizing the joint loss function (2). The sentence level model uses both labeled and unlabeled data, where the updates using unlabeled data is influenced by document level objective.

The proposed architecture evaluates the effectiveness of posing sentence-level topic classification task as a precursor to solve document level *rile* prediction, rather than learning a model directly. Secondly, we also study the utility of having more document level annotations — one-dimensional scores on the left-right spectrum, compared to sentence level annotations, especially

[3] https://keras.io/layers/wrappers/#timedistributed

[4] We observed that the concatenated representation performed better than using either hidden representation or output distribution

[5] https://keras.io/optimizers/#adam

for sentence level thematic classification task. Finally, the document level *rile* score estimation can be replaced with any other task in the proposed architecture — expert survey or voter survey scores.

5 Experimental Results

As mentioned earlier, we use manifestos collected and annotated by political scientists at CMP. In this work, we used digitized manifestos from 13 countries, 948 in total which are written in 6 different languages — Danish (Denmark), English (Australia, Ireland, New Zealand, South Africa, United Kingdom, United States), Finnish (Finland), French (France), German (Austria, Germany, Switzerland) and Italian (Italy). Out of the 948 manifestos, 235 are annotated with sentence level labels (Figure 5). We have *rile* scores for all the 948 manifestos. Statistics about number of annotated documents and sentences with fine-grained labels (from Figure 5) across languages is given below.

Lang.	# Docs (Antd.)	# Sents (Antd.)
da	175 (36)	32161 (8762)
en	312 (94)	227769 (73682)
fi	97 (16)	18717 (8503)
fr	53 (10)	24596 (5559)
de	216 (65)	146605 (79507)
it	95 (14)	40010 (4918)
Total	948 (235)	489858 (180931)

Table 1: Statistics of dataset, 'Antd.' refers to annotated. *da - Danish, en - English, fi - Finnish, fr - French, de - German, it - Italian*

We use off-the-shelf multi-lingual word embeddings[6] to represent words. We chose embeddings trained using translation invariance approach (Ammar et al., 2016), with size 512 for our work, since we found it empirically better compared to other approaches. As mentioned in Section 4.1, we use average embedding as sentence representation. Following are the sentence classification baseline techniques that we compare our proposed approach against.

Bag of words: We use TF-IDF representation for sentences and build a model for each language separately. We use Logistic Regression classifier, which is the state-of-art approach for fine-grained (57 classes) manifesto sentence classification (Biessmann, 2016). We refer to this approach as *LR*. We also use Neural Network classifier, which we refer as *NN*.

Language-wise average embedding: We build a Neural Network (LW-NN) classifier per language, with average neural embedding as sentence representation.

Convolutional Neural Network: CNN was shown to be effective for cross-lingual manifesto text coarse-level (7 major domains as shown in Figure 5) topic classification (Glavaš et al., 2017). So, we evaluate CNN with a similar architecture — single convolution (32 filters with window size 3), single max pooling layer and finally a softmax layer. We use neural embeddings to represent words. Similar to (Glavaš et al., 2017) we combined training instances across all the languages.

For document level regression task, following are the baseline approaches. Note that we use *tanh* output for all the models, since the range of re-scaled *rile* is from -1 to +1.

Bag of words: We use TF-IDF representation for documents and build a regression model for each country and language separately. We use Linear Regression; referred to as *LR-CTR* and *LR-Lang* which are models built per country and language respectively. Since language-wise model performed better than country-wise model, we also build a Neural Network model (to capture non-linear patterns) for each language, *NN-Lang*.

Average embedding (Doc-Avg): We use average embedding of words in the document as representation, with Neural Network model.

Bag-of-Centroids (BoC): Here the word embeddings are clustered into K different clusters using K-Means clustering algorithm, and words in each document are assigned to clusters based on its euclidean-distance ($dist$) to cluster-centroids (C).

$$cluster(word) = \operatorname*{argmin}_k dist(C_k, word)$$

Finally, each document is represented by the distribution of words mapped to different

[6] http://128.2.220.95/multilingual

clusters (1 X K vector). We use a Neural Network regression model with bag-of-centroids representation. Results with K=1000, which performed best is given in Table 3.

Sentence level model and RILE formulation *(SRILE)*: Here the predictions of sentence level model in the cascaded approach are used directly with *rile* formulation (equation (1)) to derive $rile$ score for manifestos.

Cross-lingual scaling (CLS): This is a recent unsupervised approach for multi-lingual political speech text positioning task (Glavas et al., 2017). Authors use average word-embeddings weighed by TF-IDF score to represent documents[7]. Then a graph is constructed using pair-wise distance of documents. Given two pivots texts for extreme left and right positions [-1, +1], label propagation approach is used to quantify other documents in the graph.

We compute average performance with 80-20% train-test ratio across 10 runs with random split, where the 80% split also contains sentence level annotated documents proportionally. We compute average F-score[8] to evaluate sentence classification performance. Sentence classification performance is given in Table 2. The proposed approach trained using *cascaded training* is referred to *Cas-S*. Note that in *cascaded training*, sentence and document level models are trained separately in a cascaded fashion. Joint-training results where the sentence model is trained in a semi-supervised way together with document level regression task is referred to as *JT-S*. We observed that *JT-S* has a comparable performance with *Cas-S*. Also *CNN*, *Cas-S* and *JT-S* use combined training instances across languages compared to other approaches. Hence other than en and de which have sufficient labeled data, models that use combined training data perform better. Secondly, in the mono-lingual setting, using word embeddings provides better performance compared to bag-of-words for en, fi, de and it.

[7]We use this aggregate representation since it was shown to be better than word alignment and scoring approach (Glavas et al., 2017)

[8]Harmonic mean of precision and recall, `https://en.wikipedia.org/wiki/F1_score`

Approach	MSE	r
LR-CTR	0.874	0.18
LR-Lang	0.684	0.24
NN-Lang	0.054	0.23
Doc-Avg	0.057	0.14
BoC	0.052	0.33
SRILE	0.060	0.35
CLS	–	0.24
Cas-D	0.051	0.41
JT-D	**0.044**	**0.47**

Table 3: $rile$ score prediction performance. Best scores are given in bold. CLS assigns -1 and +1 for extreme manifestos and propagates the values on a graph. Since they solve it as a classification problem, MSE is not applicable

rile score regression performance results are given in Table 3. We evaluate document level performance using mean-squared-error (MSE) and Pearson correlation (r). The proposed approach using cascaded training with fixed sentence model parameters is referred to as *Cas-D* and jointly trained model is referred to as *JT-D*. Overall we the jointly trained model performs best for document level task, with a comparable performance at sentence level.

5.1 Quantity of Annotation

We measure the importance of annotated text at sentence and document level for *rile* score regression task. We vary the percentage of labeled data, while keeping the test sample size at 20% as before. In the first setting, we keep the training ratio of documents at 80%, within that 80% we increase the proportion of documents with sentence level annotations — from 0 (document average embedding setting, *Doc-Avg*) to 80%. Results are given in Figure 2. Similarly, in the other setting, we keep the training set with 80% sentence level annotated documents (which is ∼20% of the total data), and add documents (with only *rile* score), increasing the training set from 20 to 80%. Results of this study are given in Figure 3. We observe that, jointly-trained model uses sentence level annotations more effectively than cascaded approach (Figure 2) — even with less sentence level annotations. Also, with less document level signal (upto 40%) for training, both the approaches perform similarly. As the training ratio increases, joint-training leverages both sentence and docu-

Lang.	LR	NN	LW-NN	CNN	Cas-S	JT-S
da	0.29	**0.36**	0.24	0.30	0.28	0.30
en	0.24	0.29	**0.43**	0.40	0.42	0.41
fi	0.21	0.23	0.26	**0.30**	0.27	0.26
fr	0.28	0.28	0.24	0.36	0.37	**0.38**
de	0.20	0.23	0.31	0.31	0.31	**0.33**
it	0.14	0.22	0.25	0.30	**0.32**	0.26
Avg.	0.22	0.26	0.35	0.34	**0.36**	0.35

Table 2: Micro-Averaged F-measure. Best scores are given in bold under each language setting

ment level signals effectively.

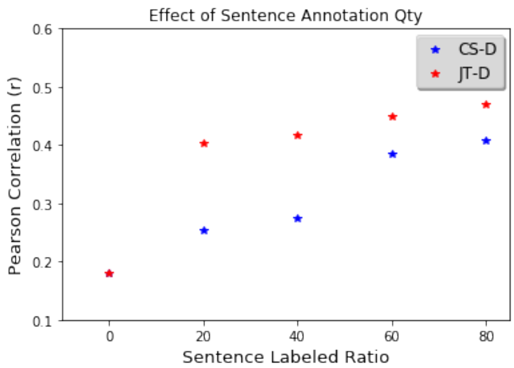

Figure 2: Fixing 80% training documents with rile score, ratio of documents with sentence level annotations is varied

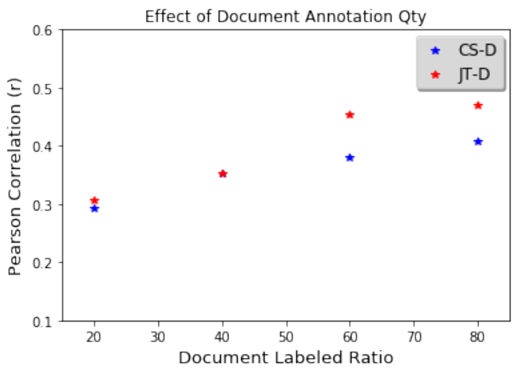

Figure 3: Fixing 80% training documents with sentence level annotations, ratio of documents with rile score is varied

5.2 Use of Country Information

Since the definition of left-right varies between countries, we study the influence of *country* information in the proposed model (*JT-D*) with *joint-training*. We use two ways to incorporate country information: (a) *stack* — one-hot encoding (13 countries, 1 X 13 vector) of each manifesto's *country* is concatenated with hidden representation of the document (S_r in Figure 1) (b) *non-linear stack* — one-hot-encoded country vector is passed through a hidden layer with *tanh* non-linear activation and concatenated with S_r. With both the models we observed mild improvement in correlation (given in Table 4).

Approach	MSE	r
stack	0.045 (0.001 ↓)	0.49 (0.02 ↑)
non-linear stack	0.048 (0.004 ↓)	0.48 (0.01 ↑)

Table 4: $rile$ score performance with country information. Difference compared to *JT-D* is given within paranthesis. ↑ – improvement, ↓ – decrease in performance

6 Conclusion and Future Work

In this work we evaluated the utility of a joint sentence-document model for sentence level thematic classification and document level *rile* score regression tasks. Our observations are as follows: (a) joint model performs better than state-of-art approaches (b) joint-training leverages sentence level annotations more effectively than cascaded approach for *rile* score regression task. There are many extensions possible to the current work. First is to handle class imbalance in the dataset with a cost-sensitive objective function. Secondly, CNN gave a comparable performance with Multi-layer Perceptron (NN), which motivates the need to evaluate an end-end sequential architecture. We used off-the-shelf embeddings, which leads to out-of-vocabulary scenarios. It could be beneficial to adapt word-embeddings with manifesto corpus. Finally, background information such as country can be leveraged more effectively.

Under my leadership we will continue to work in the international coalition, 104
with President Bush and Prime Minister Blair and others, to find the terrorists
and bring them to justice.

We're proud we have such well-trained and well-equipped men and women to 104
send on this difficult mission.

They go with our blessing, and our heartfelt appreciation of their courage and 104
commitment, and that of their families.

Our security also requires that we protect Australia's borders. People 601
smugglers are criminals who must be hounded out of business. 605

It's why we plan to introduce a proper Coast Guard – a cop on the beat 52 605
weeks of the year.

We need to work harder at finding diplomatic solutions with our neighbours -- 107
to take the boats back, or stop them before they leave.

(a) Australian Labor Party, 2001

These are accomplishments, not accidents. They came about because Democrats – from the 305
White House, to the Congress, to State Houses all across America – brought new thinking and 305
new action to our most pressing challenges. We used government as a catalyst to engage the best 305
ideas and energies of the American people. We asked citizens to get involved and they did. They 202
tutored in their children's schools, patrolled on neighborhood crime watches, volunteered in 202
local hospitals, and voiced their opinion on every issue. They shaped effective solutions to real 202
problems. It will take more of this brand of new thinking if we are to build on this record of 305
achievement.

During our nation's darkest hours, Americans have strived mightily and succeeded in meeting 601
the challenges of their times. The question before us is whether we will do the same during this 606
bright moment; whether we will seize this moment to bring more prosperity and progress to 410
more Americans than ever before; whether, having finally conquered our financial deficits, we 305
will have the courage to conquer the other deficits – in health care, in education, in the
environment – that challenge us today.

In this Platform, today's Democratic Party lays out its plans to do just that. This platform was 305
not written in a dark backroom, but in the light of day; in an open, democratic process that was 305
interactive and inclusive. It was developed both with the guidance of the brightest Democratic
leaders and with the voices of thousands of ordinary Americans around the country who 305
contributed their thoughts, ideas, beliefs, and dreams to this platform in person, on paper, and 305
over the Internet. This is a 21st century platform for the 21st century's party. A people's 202
platform for the people's party.

(b) Democratic Party of USA, 2000

Figure 4: Sample Manifestos — $\int$ denotes sentence segment

Domain 1: External Relations
101 Foreign Special Relationships: Positive
102 Foreign Special Relationships: Negative
103 Anti-Imperialism
104 Military: Positive
105 Military: Negative
106 Peace
107 Internationalism: Positive
108 European Community/Union: Positive
109 Internationalism: Negative
110 European Community/Union: Negative
Domain 2: Freedom and Democracy
201 Freedom and Human Rights
202 Democracy
203 Constitutionalism: Positive
204 Constitutionalism: Negative
Domain 3: Political System
301 Decentralisation
302 Centralisation
303 Governmental and Administrative Efficiency
304 Political Corruption
305 Political Authority
Domain 4: Economy
401 Free Market Economy
402 Incentives: Positive
403 Market Regulation
404 Economic Planning
405 Corporatism/Mixed Economy
406 Protectionism: Positive
407 Protectionism: Negative
408 Economic Goals
409 Keynesian Demand Management
410 Economic Growth: Positive
411 Technology and Infrastructure: Positive
412 Controlled Economy
413 Nationalisation
414 Economic Orthodoxy
415 Marxist Analysis
416 Anti-Growth Economy: Positive
Domain 5: Welfare and Quality of Life
501 Environmental Protection
502 Culture: Positive
503 Equality: Positive
504 Welfare State Expansion
505 Welfare State Limitation
506 Education Expansion
507 Education Limitation
Domain 6: Fabric of Society
601 National Way of Life: Positive
602 National Way of Life: Negative
603 Traditional Morality: Positive
604 Traditional Morality: Negative
605 Law and Order: Positive
606 Civic Mindedness: Positive
607 Multiculturalism: Positive
608 Multiculturalism: Negative
Domain 7: Social Groups
701 Labour Groups: Positive
702 Labour Groups: Negative
703 Agriculture and Farmers: Positive
704 Middle Class and Professional Groups
705 Underprivileged Minority Groups
706 Non-economic Demographic Groups

000 No meaningful category applies

Figure 5: *Left* topics are given in red, *right* topics are given in blue and the rest are given in black

References

Leman Akoglu. 2014. Quantifying political polarity based on bipartite opinion networks.

Waleed Ammar, George Mulcaire, Yulia Tsvetkov, Guillaume Lample, Chris Dyer, and Noah A Smith. 2016. Massively multilingual word embeddings. *arXiv preprint arXiv:1602.01925* .

Akshat Bakliwal, Jennifer Foster, Jennifer van der Puil, Ron O'Brien, Lamia Tounsi, and Mark Hughes. 2013. Sentiment analysis of political tweets: Towards an accurate classifier. Association for Computational Linguistics.

Kenneth Benoit and Thomas Däubler. 2014. Putting text in context: How to estimate better left-right positions by scaling party manifesto data using item response theory.

Felix Biessmann. 2016. Automating political bias prediction.

B. Bruinsma and K. Gemenis. 2017. Validating Wordscores. *ArXiv e-prints* .

Ian Budge and Michael Laver. 1992. Party policy and government coalitions / edited by m.j. laver and ian budge .

Thomas Däubler, Kenneth Benoit, Slava Mikhaylov, and Michael Laver. 2012. Natural sentences as valid units for coded political texts. *British Journal of Political Science* 42(4):937–951.

Zhuoye Ding, Qi Zhang, and Xuanjing Huang. 2011. Keyphrase extraction from online news using binary integer programming.

Matthew Gentzkow, Bryan T Kelly, and Matt Taddy. 2017. Text as data.

Goran Glavaš, Federico Nanni, and Simone Paolo Ponzetto. 2017. Cross-lingual classification of topics in political texts. Association for Computational Linguistics, pages 42–46. https://doi.org/10.18653/v1/W17-2906.

Goran Glavas, Federico Nanni, and Simone Paolo Ponzetto. 2017. Unsupervised cross-lingual scaling of political texts. In *EACL*.

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

Frederik Hjorth, Robert Klemmensen, Sara Hobolt, Martin Ejnar Hansen, and Peter Kurrild-Klitgaard. 2015. Computers, coders, and voters: Comparing automated methods for estimating party positions. *Research & Politics* 2(2):2053168015580476.

Mladen Karan, Jan Šnajder, Daniela Širinic, and Goran Glavaš. 2016. Analysis of policy agendas: Lessons learned from automatic topic classification of croatian political texts.

David Lazer, Alex Sandy Pentland, Lada Adamic, Sinan Aral, Albert Laszlo Barabasi, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, et al. 2009. Life in the network: the coming age of computational social science. *Science (New York, NY)* 323(5915):721.

Will Lowe, Kenneth Benoit, Slava Mikhaylov, and Michael Laver. 2011. Scaling policy preferences from coded political texts. Wiley Online Library, volume 36, pages 123–155.

Christopher Lucas, Richard A Nielsen, Margaret E Roberts, Brandon M Stewart, Alex Storer, and Dustin Tingley. 2015. Computer-assisted text analysis for comparative politics. *Political Analysis* 23(2):254–277.

Slava Mikhaylov, Michael Laver, , and Kenneth R Benoit. 2012. Coder reliability and misclassification in the human coding of party manifestos. In *Political Analysis*.

Ramesh Nallapati, James Allan, and Sridhar Mahadevan. 2004. Extraction of key words from news stories.

Federico Nanni, Cäcilia Zirn, Goran Glavaš, Jason Eichorst, and Simone Paolo Ponzetto. 2016. Topfish: topic-based analysis of political position in us electoral campaigns. volume PolText.

Ludovic Rheault. 2016. Expressions of anxiety in political texts. page 92.

Didier Ruedin. 2013. The role of language in the automatic coding of political texts. Wiley Online Library, volume 19, pages 539–545.

Jonathan B Slapin and Sven-Oliver Proksch. 2008. A scaling model for estimating time-series party positions from texts. Wiley Online Library, volume 52, pages 705–722.

Dhanya Sridhar, Lise Getoor, and Marilyn Walker. 2014. Collective stance classification of posts in online debate forums. *ACL 2014* .

Suzan Verberne, Eva D'hondt, Antal van den Bosch, and Maarten Marx. 2014. Automatic thematic classification of election manifestos. Elsevier, volume 50, pages 554–567.

Andrea Volkens, Onawa Lacewell, and Sven Regel Henrike Schultzeand Annika Werner. Pola Lehmann. 2011. The manifesto data collection. manifesto project (mrg/cmp/marpor). In *Wissenschaftszentrum Berlin fur Sozialforschung (WZB)*.

Daniel Xiaodan Zhou, Paul Resnick, and Qiaozhu Mei. 2011. Classifying the political leaning of news articles and users from user votes.

Cäcilia Zirn, Goran Glavaš, Federico Nanni, Jason Eichorts, and Heiner Stuckenschmidt. 2016. Classifying topics and detecting topic shifts in political manifestos. University of Zagreb.