Joint Sentence-Document Model for Manifesto Text Analysis

Anonymous ALTA submission

Abstract

Election programs (so-called manifestos) are published verbal declaration of the intentions, motives, or views of the political party. Political scientists use manifestos to understand policy relevant themes discussed and also quantify a 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. Inorder to handle text from multiple languages we exploit continuous neural embeddings for semantic text representation. We have empirically shown that the proposed joint model performs better than state-of-art approaches using manifestos from thirteen countries which are written in six different languages.

1 Introduction

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 Table 5). 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). Inorder to overcome these challenges, fine and coarse-level manifesto sentence classification was addressed using supervised machine learning techniques (Verberne et al., 2014; Zirn et al., 2016). 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 (Slapin and Proksch, 2008). Though sentence level classification and document level quantification tasks are inter-dependent, existing work handle them separately. Whereas, we propose a joint sentence-document model to handle both the tasks together. Overall, the contributions of this work are given as follows:

- We empirically study the utility of multilingual embeddings for cross-lingual manifesto text analysis — at sentence and document level.
- We evaluate the effectiveness of solving sentence and document level tasks together.
- We study the value of *country* information used in conjunction with text for document level regression task.

2 Related Work

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). Also most of the work 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. All these approaches handle sentence and document-level tasks separately.

3 Manifesto Text Analysis

In the CMP, trained annotators categorize manifesto sentences into one of the 57 fine-grained political categories (shown in Table 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 or structured as paragraphs (example is given in Figure 4) which can provide more information related to topic coherence. Also the length of document, measured as number of sentences, varies between manifestos. The typical length (number of sentences) we 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.

A sentence is split into multiple segments, 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, are not common.[1] 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 (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.

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

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 (Budge and Laver, 1992),

$$RILE = \sum_{r \in R} per_r - \sum_{l \in L} per_l \quad (1)$$

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 Table 5, per document. Note that the RILE score is provided almost for all the manifestos in CMP dataset, but the sentence level annotations are provided only for a subset of manifestos.

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 a 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 (e_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.

$$s_i = \frac{1}{|s_i|} \sum_{w \in s_i} e_w$$

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

$$L_S(D_s, Y_s) = -\frac{1}{\sum_{i=1}^{D_s} l_{d_i}} \sum_{i=1}^{D_s} \sum_{j=1}^{l_{d_i}} \sum_{k=1}^{K} Y_{s_{ijk}} \ln \hat{Y}_{s_{ijk}} \quad (2)$$

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 model, we use unrolled (time-distributed) 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.[2]

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

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 $z = W_r^T H_d$, where $H_d = ReLU(W_d^T S_r)$, 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.

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

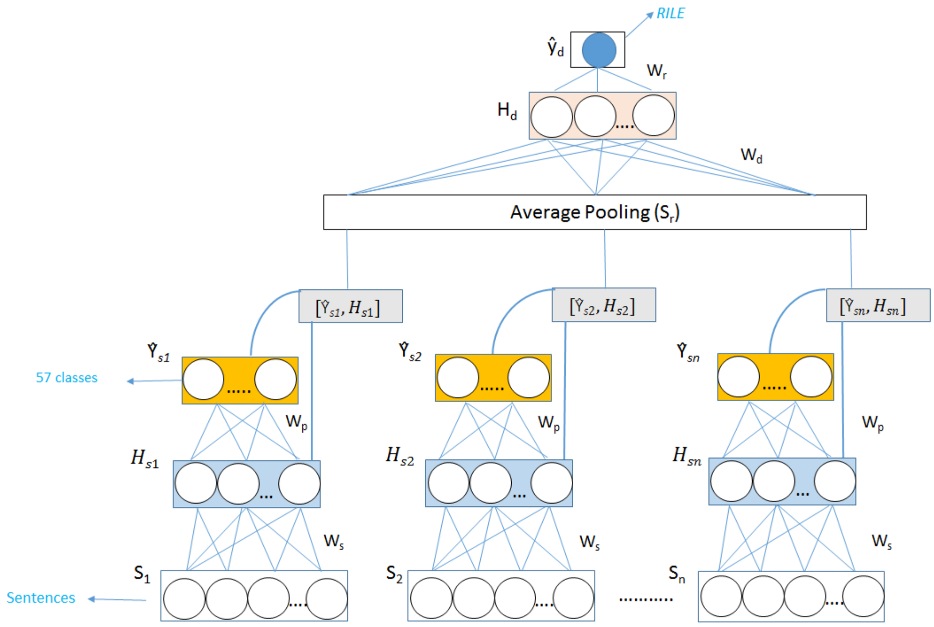

Figure 1: Hierarchical Neural Network for Joint Sentence-Document Analysis. Optimal model parameters we found — $|H_s|$ = 300, $|H_d|$ = 10

$$L_d(\hat{Y}_d, Y_d) = \frac{1}{|D|}(\hat{Y}_d - Y_d)^T(\hat{Y}_d - Y_d) \quad (3)$$

Overall, the combined loss function using (2) and (3) for the joint model is

$$\alpha * L_s(D_s, Y_s) + (1 - \alpha) * L_d(\hat{Y}_d, Y_d) \quad (4)$$

where $0 \leq \alpha \leq 1$.[3] We evaluate both cascaded and joint training for this objective function. We use Adam optimizer (Kingma and Ba, 2014) for parameter estimation.

Cascaded Training: In this approach sentence level model is trained using D_s, to minimize $L_s(D_s, Y_s)$ in (2), 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 $L_d(\hat{Y}_d, Y_d)$ in (3). Here, the sentence level model parameters are fixed when the document level regression model is trained using S_r.

[3] We use α=0.5, since we give equal weight for sentence and document level tasks. We observed a tradeoff in performance with different α, with lesser α (0.1), document level correlation increases while sentence level F-score decreases. Higher value of α (0.9) gives performance closer to cascaded training

Joint Training: Like in cascaded training, sentence level model is pre-trained using labeled sentences. Then the entire network is updated by minimizing the joint loss function (4). Here 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. We also study the effect of quantity of annotated text at both sentence and document level for RILE prediction 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 Experiments

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 (Table 5). We have RILE scores for all the 948 manifestos. Statistics about number of annotated documents and sentences with fine-grained labels (from Table 5) across languages is given below.

Lang.	# Docs (Ann.)	# Sents (Ann.)
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, 'Ann.' refers to annotated. *da - Danish, en - English, fi - Finnish, fr - French, de - German, it - Italian*

We use off-the-shelf multi-lingual word embeddings[4] 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. We first compare traditional bag-of-words discrete representation with distributed neural representation for words for fine-grained thematic classification, under mono-lingual training setting ($Mono$). Hence we compare the following approaches.

Bag of words (BoW-LR, BoW-NN): 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 (AE-NN_m): We build a Neural Network classifier per language, with average neural embedding as sentence representation.

Since distributed representation allows to leverage text across languages, we evaluate the following approaches with combined training sentences across languages ($Cross$).

Convolutional Neural Network (CNN): CNN was shown to be effective for cross-lingual manifesto text coarse-level (7 major domains as shown in Table 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.

Combined average embedding (AE-NN_c): We build a Neural Network classifier with training instances combined across languages, with average neural embedding as sentence representation. This is our proposed approach for sentence-level model.

Commonly for all empirical evaluation, we compute micro-averaged performance with 80-20% train-test ratio across 10 runs with random split (at document level), where the 80% split also contains sentence level annotated documents proportionally. Neural Network model has a single hidden layer commonly for all the sentence and document level approaches. We compute F-score[5] to evaluate sentence classification performance. Sentence classification performance is given in Table 2. In the mono-lingual setting (Table 2), using word embeddings did not provide better performance compared to bag-of-words (except for en which was the source language for obtaining multi-lingual embeddings). Under cross-lingual setting, AE-NN_c is the sentence level Neural Network model. We use AE-NN_c in the *cascaded training* for obtaining document level *RILE* prediction. 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 AE-NN_c. Also *CNN*, AE-NN_c and JT_s use combined training instances across languages. Except for da and it cross-lingual training performs better than mono-lingual setting.

[4] http://128.2.220.95/multilingual

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

Lang.	*BoW-LR*	*BoW-NN*	$AE\text{-}NN_m$	CNN	$AE\text{-}NN_c$	JT_s
da	0.29	**0.35**	0.24	0.30	0.28	0.30
en	0.36	0.38	**0.42**	0.40	**0.42**	0.41
fi	0.21	0.29	0.26	**0.30**	0.27	0.26
fr	0.28	0.36	0.24	0.36	0.37	**0.38**
de	0.30	0.31	0.31	0.31	0.31	**0.33**
it	0.32	**0.33**	0.25	0.30	0.32	0.26
Avg.	0.32	0.34	0.35	0.34	**0.36**	0.35
Setting		Mono			Cross	

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

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 (BoW-NN): We use TF-IDF representation for documents and build a Neural Network model for each language.

Average embedding ($AE\text{-}NN_d$): 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 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 ($AE\text{-}NN_c^{rile}$): Here the predictions of sentence level model in the cascaded approach ($AE\text{-}NN_c$) 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.[6] 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.

Approach	MSE(↓)	r(↑)
$BoW\text{-}NN$	0.054	0.23
$AE\text{-}NN_d$	0.057	0.14
BoC	0.052	0.33
$AE\text{-}NN_c^{rile}$	0.060	0.35
CLS	–	0.24
Cas_d	0.050	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. ↑ indicates higher is better, ↓ indicates lower is better

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 performance, 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.

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

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, $AE\text{-}NN_d$) 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 (r). As the training ratio increases, joint-training leverages both sentence and document level signals effectively.

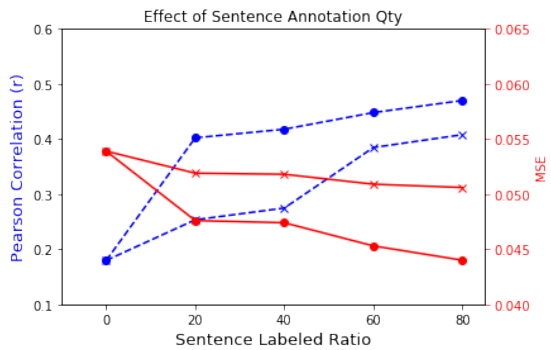

Figure 2: Fixing 80% training documents with RILE score, ratio of documents with sentence level annotations is varied. × denotes Cas_d and ○ denotes JT_d

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 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

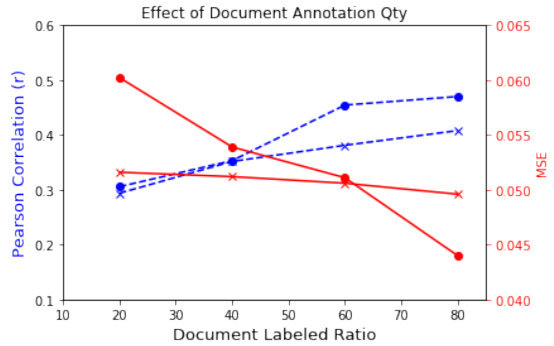

Figure 3: Fixing 80% training documents with sentence level annotations, ratio of documents with RILE score is varied. × denotes Cas_d and ○ denotes JT_d

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 prediction 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, but doesn't provide significant gains for sentence level 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. Off-the-shelf embeddings 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.

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

601
606
410
305

Figure 4: Democratic Party of USA, 2000 — $\int$ denotes sentence segment. 601 - National Way of Life: Positive, 606 - Civic Mindedness: Positive, 410 - Economic Growth: Positive, 305: Political Authority

CMP Coding Scheme	
• Domain 1. External Relations	411 Technology and Infrastructure: Positive
101 Foreign Special Relationships: Positive	412 Controlled Economy
102 Foreign Special Relationships: Negative	413 Nationalisation
103 Anti-Imperialism	414 Economic Orthodoxy
104 Military: Positive	415 Marxist Analysis
105 Military: Negative	416 Anti-Growth Economy: Positive
106 Peace	
107 Internationalism: Positive	• Domain 5: Welfare and Quality of Life
108 European Community/Union: Positive	501 Environmental Protection
109 Internationalism: Negative	502 Culture: Positive
110 European Community/Union: Negative	503 Equality: Positive
	504 Welfare State Expansion
• Domain 2: Freedom and Democracy	505 Welfare State Limitation
201 Freedom and Human Rights	506 Education Expansion
202 Democracy	507 Education Limitation
203 Constitutionalism: Positive	
204 Constitutionalism: Negative	• Domain 6: Fabric of Society
	601 National Way of Life: Positive
• Domain 3: Political System	602 National Way of Life: Negative
301 Decentralisation	603 Traditional Morality: Positive
302 Centralisation	604 Traditional Morality: Negative
303 Governmental and Administrative Efficiency	605 Law and Order: Positive
304 Political Corruption	606 Civic Mindedness: Positive
305 Political Authority	607 Multiculturalism: Positive
	608 Multiculturalism: Negative
• Domain 4: Economy	
401 Free Market Economy	• Domain 7: Social Groups
402 Incentives: Positive	701 Labour Groups: Positive
403 Market Regulation	702 Labour Groups: Negative
404 Economic Planning	703 Agriculture and Farmers: Positive
405 Corporatism/Mixed Economy	704 Middle Class and Professional Groups
406 Protectionism: Positive	705 Underprivileged Minority Groups
407 Protectionism: Negative	706 Non-economic Demographic Groups
408 Economic Goals	
409 Keynesian Demand Management	
410 Economic Growth: Positive	000 No meaningful category applies

Table 5: *Left* topics are given in *red* and *right* topics are given in *blue*

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*. St. Martin's Press New York.

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.

Goran Glavaš, Federico Nanni, and Simone Paolo Ponzetto. 2017. Cross-lingual classification of topics in political texts. Association for Computational Linguistics, pages 42–46.

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* .

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

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR* abs/1412.6980.

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. In *PolText*.

Ludovic Rheault. 2016. Expressions of anxiety in political texts. In *NLP+ CSS 2016*. page 92.

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. In *Information Processing & Management*. 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.