Classifying Topics and Detecting Topic Shifts in Political Manifestos

Cäcilia Zirn[1], Goran Glavaš[1,3], Federico Nanni[1], Jason Eichorst[2], Heiner Stuckenschmidt[1]

[1] Data and Web Science Group, University of Mannheim
B6 26, DE-68161 Mannheim, Germany
[2] Collaborative Research Center SFB 884, University of Mannheim
L13 15-17, DE-68161 Mannheim, Germany
[3] Text Analysis and Knowledge Engineering Lab, University of Zagreb
Unska 3, HR-10000 Zagreb, Croatia
{caecilia,goran,federico,heiner}@informatik.uni-mannheim.de
eichorst@uni-mannheim.de

Abstract

General political topics, like social security and foreign affairs, recur in electoral manifestos across countries. The Comparative Manifesto Project collects and manually codes manifestos of political parties from all around the world, detecting political topics at sentence level. Since manual coding is time-consuming and allows for annotation inconsistencies, in this work we present an automated approach to topical coding of political manifestos. We first train three independent sentence-level classifiers – one for detecting the topic and two for detecting topic shifts – and then globally optimize their predictions using a Markov Logic network. Experimental results show that the proposed global model achieves high classification performance and significantly outperforms the local sentence-level topic classifier.

1 Introduction

The Comparative Manifesto Project (CMP), initiated by Volkens et al. (2011), 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 topically coded by political scientists. Each manifesto sentence has been labeled with one of over fifty political topics, divided into 7 coarse-grained domains.[1] 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 disallow for comparability between manifestos annotated by different coders Mikhaylov et al. (2012).

Automated topic classification of political manifestos does not only save human resources, but it additionally provides comparable and reproducible annotations. Thus, in this work we develop a supervised framework for classifying the broad domain of sentences in political manifestos, with the specific goal of assisting human coders. Our pipeline consists of three different classifiers predicting the domains and domain shifts between pairs of adjacent sentences. They rely on a variety of features including bags-of-words and semantic textual similarity (STS) (Agirre et al., 2012; Šarić et al., 2012). In the second step, we exploit the global context of the manifestos and combine the sentence-level predictions of these three local classifiers in a global Markov Logic-based optimization setting (Richardson and Domingos, 2006), where we introduce additional global information as constraints on the prior distribution of topics, topic shifts and sequences of topics.

We evaluate each of the local classifiers and show that the introduction of global information is justified by the fact that the globally-optimized Markov Logic classifier significantly outperforms the local topic classifier and reaches the satisfactory performance of almost 80% F_1 score.

2 Related Work

The body of work on automated analysis of political texts is substantial (Grimmer and Stewart, 2013). Approaches to classification of political

[1] https://manifestoproject.wzb.eu/coding_schemes/mp_v5

Proceedings of the International Conference on the Advances in Computational Analysis of Political Text (PolText 2016), pages 88–93, Dubrovnik, Croatia, 14–16 July 2016.

texts can be roughly divided into two major groups – dictionary based methods (Kellstedt, 2000; Young and Soroka, 2012) and methods that employ supervised classification models (Purpura and Hillard, 2006; Stewart and Zhukov, 2009; Verberne et al., 2014; Karan et al., 2016). The idea behind all dictionary-based methods is similar – they first identify words that distinguish categories and then measure the occurrence frequencies of those words in texts, regardless of whether the task is recognition of racial policies from media sources (Kellstedt, 2000) or detection of affects and sentiment in political texts (Young and Soroka, 2012).

The counting principle of the dictionary-based approaches might be suitable when classifying larger fragments of text such as paragraphs or whole documents. However, all dictionaries are of limited coverage and are thus unable to capture less obvious indicator terms. This is even more emphasized when classifying short texts (e.g., sentences) as it is unlikely that many dictionary words will appear in such a short text. Along with the fact that sets of indicator words need to be compiled manually, this is why the research focus shifted to supervised classification models. Stewart and Zhukov (2009) label 8000 Russian military statements and train an ensemble of classifiers to predict whether the statements originate from activists or conservatives. Purpura and Hillard (2006) propose a two-level hierarchical classification of US legislative documents using support vector machines and standard TF-IDF weighted bag-of-words features. Karan et al. (2016) propose a very similar approach for classifying Croatian legal documents, using only document titles as input. Considering that titles are significantly shorter pieces of text, they combined traditional bag-of-word features with semantic vector representations (i.e., word embeddings) to avoid the sparseness issues.

Classification of short texts has been shown to be more challenging than document level classification. Short texts contain less words and thus require an additional semantic information, as opposed to only lexical (i.e., symbolic) input. Phan et al. (2008) build a framework for classifying short and sparse text and web snippets. They use external databases, such as MedLine, as the source of semantic knowledge that reveals hidden topics. Similarly, (Hu et al., 2009) exploit world knowledge to cluster short text snippets. The snippets do not provide enough vocabulary overlap when using only bag-of-words representations. Therefore, the authors enrich the text with internal semantics, i.e. deep understanding of the text, and external semantics from resources like Wikipedia and WordNet. The lack of appropriate knowledge bases for the political domain, however, make such approaches not applicable in our case. Instead, besides lexical features, we rely on word embeddings – general vector representations that capture well semantics of words – to topically classify manifesto sentences.

Hachey and Grover (2004) classify the rhetorical status of a sentence for text summarization. Besides lexical features, they add information such as the position of a sentence in the document and named entities. They then apply sequence labeling to predict rhetorical roles for a sequence of sentences in a document. Similarly, in this work we combine various sources of information for local sentence topic classification. We then include these classifiers in a sequence labeling model for identifying globally optimal topic sequences of a given manifesto. We decide to employ Markov logic network as a sequence labeling model because it has been already successfully applied to numerous sequence labeling tasks in natural language processing (Poon, 2010; Che and Liu, 2010; UzZaman et al., 2012; Zirn et al., 2011).

3 Topic Classification of Political Manifestos

Our goal is to support human annotators to assign manifesto sentences to political categories. The CMP distinguishes between over 50 fine-grained political categories that are grouped into seven topical areas: *External Relations*, *Freedom and Democracy*, *Political System*, *Economy*, *Welfare and Quality of Life*, *Fabric of Society* and *Social Groups*.

We first build a local sentence-level classifier that predicts one of the seven topics based on the information extracted from the sentence. Next, we employ two topic-shift classifiers that predict whether two adjacent sentences are on the same topic or not. Finally, we add information on distributions of topics and topic sequences on top of the predictions and combine all components in a global Markov Logic framework, which determines the optimal topical classification for all sentences of a manifesto.

3.1 Local Topic Classifier

The local sentence-level topic classifier makes predictions taking into account only the information from the sentence itself. To this end, a linear SVM classifier with the following set of lexical and numerical features was employed:

1. The bag-of-words term-vector of the sentence;

2. The topic of the preceding sentence;

3. The semantic similarity between the current and preceding sentence, which is computed by greedily aligning most similar words from the two sentences. Let P be the set of greedily aligned pairs (w_1, w_2) of words (where w_1 is from the first sentences, and w_2 is from the second sentence). The raw semantic similarity between the sentences is then given as:

$$sim(s_1, s_2) = \sum_{(w_1,w_2)\in P} \cos(v_{w_1}, v_{w_2})$$

   where v_w is the semantic embedding vector of the word w. We used the pretrained set of 200-dimensional GloVe embeddings[2] (Pennington et al., 2014) to compute the raw semantic similarity score. Because the similarity given by the above-mentioned formula depends on the length of the sentences, we normalized the score by the length of the sentences

4. For each topic class we also computed a numeric feature indicating the level of relative relevance of the sentence words for that class. We computed the relative frequencies of lemmas in sentences belonging to each of the topic classes on the train set. For example, if the word *"social"* appeared n times in all sentences of the train set labeled with the topical class *"Social Fabric"* and these sentences together contain N words, then $\frac{n}{N}$ is the relative relevance of the word *"social"* for the *"social security and welfare"* topic. Let $rr(w, c)$ be the relative relevance of the word w for the topical class t. The relevance score of the sentences s for the class t is then computed as follows:

$$rs(s, c) = \frac{1}{|s|} \sum_{w\in s} rr(w, c)$$

   where $|s|$ is the total number of words in the sentence s. For each sentence, one relevance score (i.e., one feature) is computed for each of the topical classes.

3.1.1 Topic-Shift Classifiers

We employ binary classifiers that predict whether two given adjacent sentences are on the same topic or not. We used the following set of features for the detection of local topic shifts:

1. Bag-of-words term-vector of the first sentence (f^1);

2. Bag-of-words term-vector of the second sentence (f^2);

3. Length (in no. words) of the first sentence (f^3);

4. Length (in no. words) of the second sentence (f^4);

5. Semantic similarity between the two sentences (f^5, cf. Section 3.1);

6. Ngram overlap between the two sentences (f^6) – the number of shared content words, normalized by the length of the sentences.

Considering the large size of the feature space due to the lexical BoW features f^1 and f^2, we first attempted to feed all features to a single linear SVM classifier. However, we observed that the numerical features (f^3–f^6) yield no improvements in classification performance over using only BoW vectors (f^1–f^2). We then fed only the numerical features to the SVM classifier with a non-linear RBF kernel and obtained similar cross-validation performance on the train set as when using the linear SVM classifier with only the bag-of-words features. Considering that the two classifiers – (1) the linear SVM using the bag-of-words features and (2) the RBF SVM with four numeric features – address the same task with completely disjoint sets of features, we decided to incorporate local predictions of both classifiers into the global optimization framework.

3.2 Topic Distribution Information

In addition to the information we gain from the sentence content, we make use of knowledge about the distribution and sequences of topics in manifestos. One salient observation is that topics are usually

[2] http://nlp.stanford.edu/data/glove.6B.zip

tackled in several consecutive sentences, so successive sentences tend to share the same label. If we take this observation a step further, we can measure the probability of topic transitions (e.g., conditional probability of a sentence of topic *Economy* following a sentence of topic *Social fabric*). We estimate these conditional probabilities on the train set. This does not only help us to decide whether two consecutive sentences share the same label, but gives us an estimate for probable sequences of topics.

3.3 Global Optimization

Markov logic (Richardson and Domingos, 2006) can be interpreted as a template language combining first-order logic with maximum entropy models. The user can specify types of data and encode prior knowledge about the information used in the classification scenario, and it searches the most probable world given the evidence.

A Markov network $\mathcal{M}$ is an undirected graph whose nodes represent a set of random variables $\mathbf{X} = \{X_1, ..., X_n\}$ and whose edges model direct probabilistic interactions between adjacent nodes. More formally, a distribution P is a log-linear model over a Markov network $\mathcal{M}$ if it is associated with:

- a set of features $\{f_1(D_1), ..., f_k(D_k)\}$, where each D_i is a clique in $\mathcal{M}$ and each f_i is a function from D_i to $\mathbb{R}$,
- a set of real-valued weights $w_1, ..., w_k$, such that

$$P(\mathbf{X} = \mathbf{x}) = \frac{1}{Z} \exp\left(\sum_{i=1}^{k} w_i f_i(D_i)\right),$$

where Z is a normalization constant.

A Markov logic network is a set of pairs (F_i, w_i) where each F_i is a first-order formula and each w_i a real-valued weight associated with F_i. With a finite set of constants C it defines a log-linear model over possible worlds $\{\mathbf{x}\}$ where each variable X_j corresponds to a ground atom and feature f_i is the number of true groundings (instantiations) of F_i with respect to C in possible world $\mathbf{x}$. Possible worlds are truth assignments to all ground atoms with respect to the set of constants C. We explicitly distinguish between weighted formulas and *deterministic* formulas, that is, formulas that always have to hold.

Given a set of first-order formulas and a set of ground atoms, we wish to find the formulas maximum a posteriori (MAP) weights, that is, the weights that maximize the log-likelihood of the hidden variables given the evidence.

3.3.1 Model

We model each sentence of the manifesto as a constant $s \in S$. In the same manner, topics 1-7 are represented as constants. First, we specify that each sentence s is mapped to exactly one topic t as a deterministic formula:

$$\forall s, t : |t| map(s, t) = 1$$

As we intend to predict the correct mappings, $map(s, t)$ is our hidden predicate. We introduce the predicate $next(s1, s2)$ stating that sentence $s1$ is followed by $s2$ to model the sequences of sentences in a manifesto. This allows us to encode our observation that subsequent sentences share the same topic:

$$\forall s, c : next(s_1, s_2) \wedge map(s_1, t) \Rightarrow map(s_2, t)$$

In contrast to the first formula, this one can be violated with a certain penalty, thus the formula is given a weight. Estimations about the transition between two particular topics are modelled alike by replacing t by particular variables t_1, t_2.

The predictions from the local sentence classifiers are modeled with the predicate $localConf(s, t, conf)$, where $conf$ represents the confidence for sentence s to be mapped to a particular topic t. We use this confidence as the weight for the corresponding formula:

$$\forall s, t : localConf(s, t, conf) \wedge map(s, t)$$

Each of the sentence-pair classifiers is modeled (separately) via a predicate called $flip$.

$$\begin{aligned}\forall s, t :& shift(s_1, s_2, conf) \wedge map(s_1, t)\\ &\Rightarrow \neg map(s_2, t)\end{aligned}$$

It expresses the confidence of a sentence pair belonging to two different topics: the label of the first sentence is “flipped” if the formula is true, i.e. if the confidence by the classifier (included as the weight for the formula) is high enough.

4 Experiments

In our experiments we used six U.S. manifestos (Republican and Democrat manifestos from 2004, 2008, and 2012 elections). In all experiments, we perform folded cross-validation and report the micro-averaged results over folds.

Topic	P	R	F_1
External Rel.	83.7	86.6	85.1
Freedom & Dem.	68.0	59.9	63.7
Pol. system	69.7	65.7	67.6
Economy	73.9	77.4	75.6
Welfare & QoL	72.8	72.8	72.8
Fabric of Soc.	74.8	76.0	75.4
Soc. Groups	71.2	67.9	69.5
Micro-avg.	74.9	74.9	74.9

Table 1: Local topic classification, 10-fold CV (%)

Model	P	R	F_1
Linear, bow feat.	56.6	54.6	55.6
RBF, num. feat.	98.5	27.4	42.9

Table 2: Topic-shift classification, 10-fold CV (%)

Topic Classification Table 1 shows the results of the local topic classifier obtained via the 10-fold CV. The classification performance is best for *External relations* (more easily recognizable due to re-occurring country names) and worst for *Freedom and democracy* (as lexical clues typical for this class tend to frequently appear in sentences of other topics as well).

Topic Shift Classification The performance of the two topic-shift classifiers is given in Table 2. These results indicate that detecting topic shifts is a more difficult task that predicting the topics of individual sentences. This is expected, as correctly identifying the topic shift logically amounts to correctly predicting topics for two consecutive sentences.

Global Classification The predictions of local classifiers are combined with the topic distribution information in a Markov Logic Network (MLN). We use RockIt (Noessner et al., 2013) as the MLN engine.

To evaluate the impact of each component, we start the experiments with a reduced set of formulas and incrementally add more constraints. As a baseline, we simply use the predictions by the local classfier (setting L). In the second setting, we encode rules for transitions (setting T) between particular topics. This is directly compared to a simpler setting S where we just assign consecutive sentences the same label instead of adding an own transition rule for every possible sequence of topics. The results of these combinations applied to the validation set are shown in the first part of the Table 3. Adding the information about consecutive sentences and transitions improves over the local classifier performance for 4 points, reaching 78.3%.

Setting	MaP	MaR	MaF_1	miF_1
L	73.5	72.3	72.8	74.9
L,T	80.7	73.1	75.2	78.3
L,S	78.3	74.5	75.9	78.3
L,S,P_{bow}	74.2	73.0	73.6	75.6
L,S,P_{num}	78.6	76.7	**77.5**	**79.3**
L,S,P_{bow},P_{num}	74.4	73.2	73.7	75.8

Table 3: Global classification (validation-set): MaP/MaR/MaF_1 = Macro precision/recall/F_1-measure; miF_1 = micro F_1-measure

As precision and recall are more balanced for setting S and it needs significantly less rules, we prefer it over setting T for the following experiments. We now employ the predictions of the topic-shift classifiers: P_{BOW} are the predictions of the linear SVM model with BOW features and P_{num} denotes the predictions of the non-linear SVM using numerical features. We first test each one seperately, then both together (setting $L + N$). The lower part of table 3 shows the results. The best performance of 79.3% F_1 score is obtained for the model using predictions P_{BoW}. The combination of both sentence pair classifiers drops performance, which is not surprising due to the performance of classifier P_{num}.

5 Conclusion

We presented an approach for sentence-level topical classification of party manifesto, which can be used to assist human coders in the CMP project and will allow for better reproducibility and comparability of the manually coded manifestos and will speed up the annotation process. We intend to conduct future experiments that evaluate the benefits of the application to the coding process. Furthermore, we showed that the addition of contextual and structural information about the documents improves the topical classification performance. Our approach could benefit from a cross-lingual information, i.e., from exploiting topical sequences common across different countries and languages.

Acknowledgments

The authors thank the DFG for Funding under the SFB 884 Political Economy of Reforms C4 project. Furthermore we thank our master student Xiaochen Zhao for assisting the literature research.

References

Eneko Agirre, Mona Diab, Daniel Cer, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1+2*, pages 385–393. Association for Computational Linguistics.

Wanxiang Che and Ting Liu. 2010. Jointly modeling wsd and srl with markov logic. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 161–169. Association for Computational Linguistics.

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.

Ben Hachey and Claire Grover. 2004. Sentence classification experiments for legal text summarisation. In *Proc. 17th Annual Conference on Legal Knowledge and Information Systems (Jurix-2004)*, pages 29–38.

Xia Hu, Nan Sun, Chao Zhang, and Tat-Seng Chua. 2009. Exploiting internal and external semantics for the clustering of short texts using world knowledge. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 919–928. ACM.

Mladen Karan, Daniela Širinić, Jan Šnajder, and Goran Glavaš. 2016. Analysis of policy agendas: Lessons learned from automatic topic classification of croatian political texts. In *Proceedings of the Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH) at ACL 2016*, page in press.

Paul M Kellstedt. 2000. Media framing and the dynamics of racial policy preferences. *American Journal of Political Science*, pages 245–260.

Slava Mikhaylov, Michael Laver, and Kenneth R Benoit. 2012. Coder reliability and misclassification in the human coding of party manifestos. *Political Analysis*, 20(1):78–91.

Jan Noessner, Mathias Niepert, and Heiner Stuckenschmidt. 2013. Rockit: Exploiting parallelism and symmetry for MAP inference in statistical relational models. *CoRR*, abs/1304.4379.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–43.

Xuan-Hieu Phan, Le-Minh Nguyen, and Susumu Horiguchi. 2008. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th international conference on World Wide Web*, pages 91–100. ACM.

Hoifung Poon. 2010. Markov logic in natural language processing: Theory, algorithms, and applications. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, page 3. Citeseer.

Stephen Purpura and Dustin Hillard. 2006. Automated classification of congressional legislation. In *Proceedings of the 2006 international conference on Digital government research*, pages 219–225. Digital Government Society of North America.

Matthew Richardson and Pedro Domingos. 2006. Markov logic networks. *Mach. Learn.*, 62(1-2):107–136.

Frane Šarić, Goran Glavaš, Mladen Karan, Jan Šnajder, and Bojana Dalbelo Bašić. 2012. Takelab: Systems for measuring semantic text similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1+2*, pages 441–448. Association for Computational Linguistics.

Brandon M Stewart and Yuri M Zhukov. 2009. Use of force and civil–military relations in russia: an automated content analysis. *Small Wars & Insurgencies*, 20(2):319–343.

Naushad UzZaman, Hector Llorens, James Allen, Leon Derczynski, Marc Verhagen, and James Pustejovsky. 2012. Tempeval-3: Evaluating events, time expressions, and temporal relations. *arXiv preprint arXiv:1206.5333*.

Suzan Verberne, Eva Dhondt, Antal van den Bosch, and Maarten Marx. 2014. Automatic thematic classification of election manifestos. *Information Processing & Management*, 50(4):554–567.

Andrea Volkens, Onawa Lacewell, Pola Lehmann, Sven Regel, Henrike Schultze, and Annika Werner. 2011. *The Manifesto Data Collection*. Manifesto Project (MRG/CMP/MARPOR), Wissenschaftszentrum Berlin für Sozialforschung (WZB).

Lori Young and Stuart Soroka. 2012. Affective news: The automated coding of sentiment in political texts. *Political Communication*, 29(2):205–231.

Cäcilia Zirn, Mathias Niepert, Heiner Stuckenschmidt, and Michael Strube. 2011. Fine-grained sentiment analysis with structural features. In *IJCNLP*, pages 336–344.