Adaptive Ensembling: Unsupervised Domain Adaptation for Political Document Analysis

Shrey Desai[1], **Barea Sinno**[2], **Alex Rosenfeld**[3], and **Junyi Jessy Li**[3]

[1]Department of Computer Science
[2]Department of Government, Department of Statistics & Data Science
[3]Department of Linguistics
The University of Texas at Austin
shreydesai@utexas.edu, barea.sinno@utexas.edu
alexbrosenfeld@gmail.com, jessy@austin.utexas.edu

Abstract

Insightful findings in political science often require researchers to analyze documents of a certain subject or type, yet these documents are usually contained in large corpora that do not distinguish between pertinent and non-pertinent documents. In contrast, we can find corpora that label relevant documents but have limitations (e.g., from a single source or era), preventing their use for political science research. To bridge this gap, we present *adaptive ensembling*, an unsupervised domain adaptation framework, equipped with a novel text classification model and time-aware training to ensure our methods work well with diachronic corpora. Experiments on an expert-annotated dataset show that our framework outperforms strong benchmarks. Further analysis indicates that our methods are more stable, learn better representations, and extract cleaner corpora for fine-grained analysis.

1 Introduction

Recent progress in natural language processing and computational social science have pushed political science research into new frontiers. For example, scholars have studied language use in presidential elections (Acree et al., 2018), legislative text in Congress (de Marchi et al., 2018), and similarities in national constitutions (Elkins and Shaffer, 2019). However, datasets used by political scientists are mostly homogeneous in terms of subject (e.g., immigration) or document type (e.g., constitutions). Labeled corpora with pertinent documents usually only stem from a single source; this makes it difficult to generalize conclusions derived from them to other sources. On the other hand, corpora spanning multiple decades and sources tend to be unlabeled. These corpora are largely untouched by political scientists; to illustrate some problems that arise with studying such data, Table 1 shows a sample of topics generated by Latent Dirichlet Allocation (LDA) (Blei et al., 2003), a popular topic model in social science, trained on 60,000 documents sampled from the Corpus of Historical American English (COHA) (Davies, 2008). The generated topics are extremely vague and not specific to politics.

Topic 1	like, day, would, a.m., center
Topic 2	two, samour, family, veronica, son
Topic 3	would, hospital, also, car, hyundai
Topic 4	said, people, one, years, think
Topic 5	city, 6-4, last, wine, york

Table 1: Randomly sampled topics and top keywords derived from a 50-topic LDA model trained on a sample of COHA documents. Topic modeling results on a political subset of COHA are presented in Table 5. Additionally, topic model hyperparameters are detailed in Appendix A.

This paper bridges the gap between labeled and unlabeled corpora by framing the problem as one of domain adaptation. We develop *adaptive ensembling*, an unsupervised domain adaptation framework that learns from a single-source, labeled corpus (the source domain) and utilizes these representations effectively to obtain labels for a multi-source, unlabeled corpus (the target domain). Our method draws upon *consistency regularization*, a popular technique that stabilizes model predictions under input or weight perturbations (Athiwaratkun et al., 2019). At the framework-level, we introduce an adaptive, feature-specific approach to optimization; at the model-level, we develop a novel text classification model that works well with our framework. To better handle the diachronic nature of our corpora, we also incorporate time-aware training and representations.

Our experiments use the New York Times Annotated Corpus (NYT) (Sandhaus, 2008) as our source domain corpus and COHA as our target do-

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 4718–4730, Hong Kong, China, November 3–7, 2019.

main corpus. Concretely, we construct two classification tasks: a *binary task* to determine whether a document is political or non-political; and a *multi-label task* to categorize a document under three major areas of political science in the US: *American Government*, *Political Economy*, and *International Relations* (Goodin, 2009). We subsequently introduce an expert-labeled test set from COHA to evaluate our methods.

Our framework, equipped with our best model, significantly outperforms existing domain adaptation algorithms on our tasks. In particular, adaptive ensembling achieves gains of 11.4 and 10.1 macro-averaged F1 on the binary and multi-label tasks, respectively. Qualitatively, adaptive ensembling conditions the optimization process, learns smoother latent representations, and yields precise but diverse topics as demonstrated by LDA on an extracted political subcorpus of COHA. We release our code and datasets at `http://github.com/shreydesai/adaptive-ensembling`.

2 Motivation from Political Science

Quantitative studies of American public opinion over time have mostly been restricted to surveys such as the American National Election Survey (Baldassarri and Gelman, 2008; Campbell et al., 1980). However, surveys often do not pose well-formed questions, reflect true voter opinion, or capture mass public opinion (Zaller et al., 1992; Bishop, 2004). Therefore, researchers often seek to compare survey findings with those of mass media as the relationship between public opinion and the media has been widely established (Baum and Potter, 2008; McCombs, 2018). Press media, one form of mass media, manifests itself in large, diachronic collections of newspaper articles; such corpora provide a promising avenue for studying public opinion and testing theories, provided scholars can be confident that the measures they obtain over time are substantively invariant (Davidov et al., 2014). However, as alluded to earlier, such diachronic corpora are often unlabeled; political scientists cannot draw conclusions from these corpora in their raw form as they are unable to distinguish between political and non-political articles. We frame this problem as an exchange between two domains: a source, labeled corpus with modern articles (NYT) and a target, unlabeled corpus with decades of articles originating from a multitude of news sources (COHA). Using domain adaptation methods, we can extract a political subcorpus from COHA that would be amenable for the study of public opinion research over time.

3 Unsupervised Domain Adaptation

In this section, we detail the core concepts behind our unsupervised domain adaptation framework. We describe the problem setup (§3.1), an overview of self-ensembling and consistency regularization (§3.2-§3.4), and our novel contributions to this framework (§3.5-§3.6).

3.1 Problem Setup

Let $\mathcal{X}$ and $\mathcal{Y}$ denote the input and output spaces, respectively. We have access to labeled samples $\{x_L^{(i)}, y_L^{(i)}\}_{i=1}^N$ from a source domain $\mathcal{D}_S$ and unlabeled samples $\{x_U^{(i)}\}_{i=1}^M$ from a target domain $\mathcal{D}_T$. The goal of unsupervised domain adaptation is to learn a function $f : \mathcal{X} \rightarrow \mathcal{Y}$ that maximizes the likelihood of the target domain samples by only leveraging supervision from the source domain samples. We also assume the existence of a small amount of labeled target domain samples in order to create a development set, following existing work in unsupervised domain adaptation (Glorot et al., 2011; Chen et al., 2012; French et al., 2018; Zhang et al., 2017).

3.2 Self-Ensembling

Our unsupervised domain adaptation framework builds on top of *self-ensembling* (Laine and Aila, 2017), a semi-supervised learning algorithm based on *consistency regularization*, whereby models are trained to be robust against injected noise (Athiwaratkun et al., 2019).

Self-ensembling is an interplay between two neural networks: a student network $f(x; \theta)$ and a teacher network $f(x; \phi)$. The inputs to both networks are perturbed separately, and the objective is to measure the consistency of the student network's predictions against the teacher's. Both networks share the same base model architecture and initial parameter values, but follow different training paradigms (Laine and Aila, 2017). In particular, the student network is updated via backpropagation, then the teacher network is updated with an exponential average of the student network's parameters (Tarvainen and Valpola, 2017). The networks are trained in an alternating fashion until they converge. During test time, the teacher

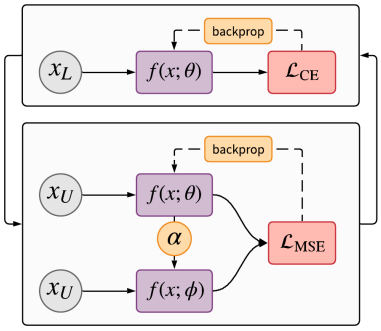


Figure 1: Visualization of the self-ensembling training procedure. Descriptions of individual components are detailed in §3.2-§3.4.

network is used to infer the labels for target domain samples. Figure 1 visualizes the overall training procedure. Further intuition behind self-ensembling is available in Appendix B.

Next, we discuss the training process for the student network (§3.3), the original fixed ensembling method in Tarvainen and Valpola (2017) (§3.4), and our proposed adaptive ensembling method (§3.5).

3.3 Student Training

The student network uses labeled samples from the source domain and unlabeled samples from the target domain to learn domain-invariant features. This is realized by using multiple loss functions, each with its own objective. The **supervised loss** is simply the cross-entropy loss of the student network outputs given source domain samples:

$$\mathcal{L}_{\text{CE}}(\theta) = \sum_{(x,y)\in\mathcal{D}_S} \log p(y|x')$$

However, the supervised loss alone prevents the student network from learning anything useful about the target domain. To address this, Laine and Aila (2017) introduce an **unsupervised loss** to ensure that the student and teacher networks have similar predictions for target domain samples. French et al. (2018) only enforce the consistency constraint for target domain samples, but we propose using both source *and* target domain samples with separately perturbed inputs x' and x''; this provides a balanced source of supervision to train our adaptive constants, discussed in §3.5:

$$\mathcal{L}_{\text{MSE}}(\theta,\phi) = \sum_{x\in\mathcal{D}_S\cup\mathcal{D}_T} ||f(x';\theta) - f(x'';\phi)||^2$$

The overall objective is a combination of the two loss functions:

$$\mathcal{L}(\theta,\phi) = \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{MSE}}$$

3.4 Fixed Ensembling

The teacher network's parameters form an ensemble of the student network's parameters over the course of training:

$$\phi^{(t+1)} \leftarrow \alpha\phi^{(t)} + (1-\alpha)\theta^{(t)}$$

where α is a smoothing factor that controls the magnitude of the parameter updates. Since the labels for the target domain samples are inherently unknown, ensembling parameters in the presence of noise helps the teacher network's predictions converge to the *true* label (Tarvainen and Valpola, 2017).

Limitations Empirically, we find that the highly unstable loss surface presented by textual datasets causes large instabilities in the optimization process. One of the key insights of this paper is that these instabilities are due to the dynamics of the unsupervised loss. Because the unsupervised loss effectively regularizes the source domain representations to work well in the target domain (Laine and Aila, 2017), performance degrades rapidly if this loss fails to converge. This is a strong indicator that self-ensembling fails to learn useful, shared representations for knowledge transfer between textual domains. Qualitative evidence of the unsupervised loss' instability is shown in Figure 6a and further discussed in §7.

3.5 Adaptive Ensembling

We hypothesize that smoothing with a fixed hyperparameter α is responsible for said instabilities. For any given weight matrix (or bias vector), each hidden unit can be conceptualized as controlling one highly specific feature or attribute (Bau et al., 2019). These units may need to be updated with varying degrees throughout the course of training; therefore, smoothing each unit with a fixed constant severely overlooks dynamics at the parameter-level. We propose modifying fixed ensembling by introducing trainable smoothing constants for each unit—hereafter termed *adaptive constants*—as opposed to using a fixed smoothing constant:

$$\phi^{(t+1)} \leftarrow \mathbf{C}^{(t)} \odot \phi^{(t)} + (\mathbf{1} - \mathbf{C}^{(t)}) \odot \theta^{(t)}$$

where a matrix of adaptive constants $\mathbf{C}$ is applied element-wise to ϕ and θ at each step.

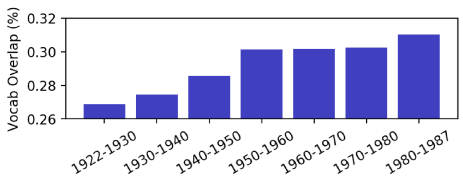


Figure 2: Vocabulary overlap between COHA and NYT, by decade. We collect COHA documents in each decade, create a decade vocabulary, and calculate the percentage overlap between each decade's vocabulary and the overall NYT vocabulary.

Example Assume we are training an arbitrary weight matrix $\mathbf{W} \in \mathbb{R}^{m \times n}$ in the kth layer of a fixed network architecture. Both the student and teacher network have their own copy of $\mathbf{W}$, denoted as $\mathbf{W}_{\text{STU}}$ and $\mathbf{W}_{\text{TEA}}$, respectively. To ensure each parameter $\mathbf{W}_{ij}$ has a corresponding adaptive constant α_{ij}, $\mathbf{C}$ shares the same dimensionality as $\mathbf{W}_{\text{STU}}$ and $\mathbf{W}_{\text{TEA}}$. The previous equation can then be written as:

$$\mathbf{W}_{\text{TEA}}^{(t+1)} \leftarrow \mathbf{C}^{(t)} \odot \mathbf{W}_{\text{TEA}}^{(t)} + (\mathbf{1} - \mathbf{C}^{(t)}) \odot \mathbf{W}_{\text{STU}}^{(t)}$$

Supervision Because the adaptive constants are designed to stabilize training, it is a natural fit to train them using the unsupervised loss:

$$\mathbf{C}^{(t+1)} \leftarrow \mathbf{C}^{(t)} - \epsilon \nabla_{\mathbf{C}} \mathcal{L}_{\text{MSE}}$$

This forms a crucial difference between self-ensembling and adaptive ensembling: in the former method, the teacher network has no say in how its parameters are modified. Adaptive ensembling equips the teacher network with fine-grained control over gradient updates, making it far easier to align activations under a noisy setting.

3.6 Temporal Curriculum

Diachronic datasets important in political science can be difficult to adapt to given the minimal vocabulary overlap between the source and target domain documents. Source and target articles mention named entities and events that, for the most part, do not appear across both datasets. To ease the difficulty of domain adaptation, we exploit the temporal information in our datasets to introduce a curriculum (Bengio et al., 2009).

In particular, each article comes with metadata that includes the year in which the article was published. Figure 2 shows that COHA articles written closer to the time of NYT articles have a larger vocabulary overlap than those written in the distant past. Intuitively, it is easier to learn features from target domain samples that are more like the source domain samples. Hence, we sort the target domain mini-batches by year; the learning task becomes progressively harder as opposed to confusing the models during the early stages of training.

4 Model

In this section, we introduce a new convolutional neural network (CNN) as the plug-in model for our unsupervised domain adaptation framework. We motivate the use of CNNs (§4.1), formalize the model input (§4.2), and introduce several novel components for our task (§4.3).

4.1 Motivation

CNNs have emerged as strong baselines for text classification in NLP (Kim, 2014). CNNs are desirable candidates for our framework as they exhibit a high degree of parameter sharing, significantly reducing the number of parameters to train. In addition, they can be designed to solely optimize the log-likelihood of the training data. Experimentally, we find that models that optimize other distributions (e.g., attention distributions in Transformers (Vaswani et al., 2017) or Hierarchical Attention Networks (Yang et al., 2016)) do not work well with this framework.

4.2 Model Input

Given a discrete input $x = [w_1, \cdots, w_n]$ and vocabulary V, an embedding matrix $\mathbf{E} \in \mathbb{R}^{|V| \times d}$ replaces each word w_i with its respective d-dimensional embedding. The resulting embeddings are stacked row-wise to obtain an input matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$. Following the notion of input perturbation used in consistency regularization algorithms (Athiwaratkun et al., 2019), we design several methods to inject noise into the input layer. Each input is perturbed with additive, isotropic Gaussian noise: $\tilde{\mathbf{X}} = \mathbf{X} + \mathcal{N}(0, \mathbf{I})$. Then, we apply dropout on the perturbed inputs to eliminate dependencies on any one word: $\mathbf{X}' = \tilde{\mathbf{X}} \odot \mathbf{M}$ where $\mathbf{M} \in \mathbb{R}^{n \times d}$ is a Bernoulli mask applied element-wise to the input matrix.

4.3 Model Architecture

Background: 1D Convolutions CNNs for text classification generally use 2D convolutions over the input matrix (Kim, 2014), but architectures using 1D convolutions have also been explored in

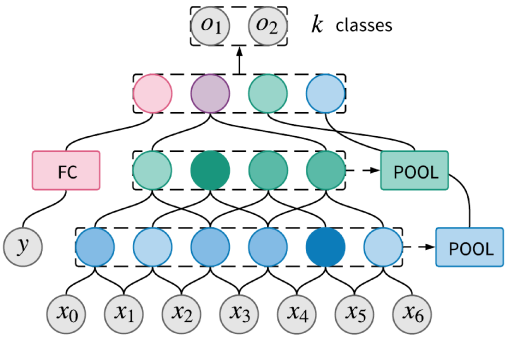


Figure 3: Example three-layer CNN architecture with sequence squeezing, state connections, and time embeddings. Detailed information about individual components is available in §4.3. FC represents a fully-connected layer; POOL represents a pooling layer. Best viewed in color.

other contexts, e.g., sequence modeling (Bai et al., 2018), machine translation (Kalchbrenner et al., 2016), and text generation (Yang et al., 2017). Our model draws upon the latter approach for political document classification. CNNs utilizing 1D convolutions are typically autoregressive in nature; that is, each output y_t only depends on the inputs $x_{<t}$ to avoid information leakage into the future. Two approaches have been proposed to achieve this: history-padding (Bai et al., 2018, 2019) and masked convolutions (Kalchbrenner et al., 2016). Further, each successive convolution uses an exponentially increasing dilation factor, reducing the depth of the network significantly. Below, we elaborate on the components of our model:

Sequence Squeezing Given a model with ℓ layers, previous approaches (Bai et al., 2018, 2019) history-pad the input with $\sum_{i=1}^{\ell} d^{(i-1)}(f-1)$ zeros to obtain an output of length n, where d is the dilation factor and f is the filter size. However, we propose history-padding the input with $(\sum_{i=1}^{\ell} d^{(i-1)}(f-1)) - n + 1$ zeros to ensure the convolutions compress the sequence down to *one* output unit. Formally, this produces an output feature map of dimension $B \times C \times 1$ where B is the batch size and C is the number of channels; one can use a simple `squeeze()` operation to obtain the compact feature matrix $B \times C$. Though this is a subtle difference, our approach yields much richer representations for classification.

State Connections In each layer ℓ_i, a kernel $\mathbf{W}_i$ convolves across an intermediate sequence, inducing a feature map $\mathbf{A}_i$. Because the input is presented as a sequence, the application of $\mathbf{W}_i$ along a one-dimensional axis encourages $\mathbf{A}_i$ to encode temporal features, similar to how the hidden state is formed by applying shared weights across a sequence in recurrent architectures. Further, because the receptive field grows exponentially, the convolutions build hierarchical representations of the input, implying $\mathbf{A}_{i+1}$ builds a more abstract representation of the input than $\mathbf{A}_i$. We exploit this stateful information by pooling each activation map $\mathbf{A}_i$ into a vector and concatenating them row-wise to create a *state matrix*:

$$\mathbf{S} = \text{Concat}(\text{Pool}(\mathbf{A}_1), \cdots, \text{Pool}(\mathbf{A}_{\ell-1}))$$

To the best of our knowledge, our paper is the first to explicitly use the temporal state embedded in causal 1D convolution activations as representations for an end task.

Time Embedding To make our model time aware, we learn representations for the years of the documents (available as metadata in COHA). Such time representations allow the model to reason about content as it appears in different decades. Given a year y (e.g. 1954), we normalize it to the closed unit interval $[0, 1]$ and linearly transform it into a low-dimensional embedding $\mathbf{e}$:

$$\mathbf{e} = \mathbf{W}_e \left[\frac{y - \max_y}{\max_y - \min_y} \right] + b_e$$

where $\max_y$ and $\min_y$ represent the maximum and minimum observed years in the training dataset, respectively.

Overall Architecture We concatenate the various components of our model $[\mathbf{X}'; \mathbf{S}; \mathbf{e}]$ to create a collective representation for classification. We use a 1D convolution ($f = 1$ and $d = 1$) to project this representation to k classes:

$$\mathbf{y} = \text{Conv1D}([\mathbf{X}'; \mathbf{S}; \mathbf{e}]; \mathbf{W}_k)$$

We did not observe any performance advantages from using a fully-connected layer to perform the projection, so we opt to use a fully-convolutional architecture to minimize the number of parameters (Long et al., 2015). Finally, we apply softmax to the output vector $\mathbf{y} \in \mathbb{R}^k$ to obtain a valid probability distribution over the classes. An example of our model architecture is depicted in Figure 3.

5 Datasets

We present a dataset for identifying political documents with manual annotation from political science graduate students. The dataset is constructed for *binary* and *multi-label* tasks: (1) identifying whether a document is political (i.e. containing notable political content) and (2) if so, the area(s) among three major political science subfields in the US: *American Government*, *Political Economy*, and *International Relations* (Goodin, 2009).

Source We use NYT as the source dataset as it contains fine-grained descriptors of article content. We sample 4,800 articles with the descriptor US POLITICS & GOVERNMENT. To obtain non-political articles, we sample 4,800 documents whose descriptors do not overlap with an exhaustive list of political descriptors identified by a political science graduate student. For our multi-label task, the annotator grouped descriptors in NYT that belong to each area label we consider[1].

Target Our target data are historical documents from COHA, which contains a large collection of news articles since the 1800s. To ensure our dataset is useful for diachronic analysis (e.g., public opinion over time), we sample only from news sources that consistently appear across the decades. Further, we ensure there are at least 8,000 total documents in each decade group; this narrows down our time span to 1922–1986. From this subset, we sample ∼250 documents from each decade for annotation. Two political science graduate students each annotated a subset of the data.

To train our unsupervised domain adaptation framework, we use 9,600 unlabeled target examples (same number as NYT). The expert-annotated dataset is divided into three subsets: a training set of 984 documents (*only* for training the In-Domain classifier discussed in §6.2), development set of 246 documents, and test set of 350 documents (50 per decade)[2].

6 Experiments

6.1 Settings

Our CNN has 8 layers, each with 256 channels, $f = 3$, $d = 2^i$ (for the ith layer), and ReLU activation. We enforce a maximum sequence length of 200 and minimum word count from $[1, 2, 3]$ to build the vocabulary. The embedding matrix uses 300-D GloVe embeddings (Pennington et al., 2014) with a dropout rate of 0.5 (Srivastava et al., 2014). We history-pad our input with a zero vector, the state connections are obtained using average pooling, and the time embedding has a dimensionality of 10. The model is optimized with Adam (Kingma and Ba, 2015), learning rate from $[10^{-4}, 5 \cdot 10^{-5}, 10^{-5}]$, and mini-batch size from $[16, 32]$. Hyperparameters were discovered using a grid search on the held-out development set.

[1]These descriptors are available in Appendix C.

[2]The news sources used and label distributions for the expert-annotated dataset are available in Appendix D.

	Binary Task		Multi-Label Task		
Method	Mi-F	Ma-F	Ma-P	Ma-R	Ma-F
Source Only	55.7	46.2	28.8	70.0	39.6
mSDA	57.4	49.7	41.0	63.7	48.1
DANN	68.2	65.8	**50.8**	36.3	42.2
SE	64.0	59.5	42.7	64.1	51.0
+ curriculum	66.4	62.3	44.4	71.7	54.5
AE (ours)	75.1	74.5	46.1	75.3	57.2
+ curriculum	**77.4**	**77.1**	48.2	**83.5**	**61.1**
In-Domain	81.7	81.6	86.5	83.5	84.8

Table 2: Framework results for the binary label task (left) and multi-label task (right). For the binary task, we show micro- and macro-averaged F1 scores. For the multi-label task, we show macro-averaged precision, recall, and F1 scores.

6.2 Framework Results

Using our best model, we benchmark our unsupervised domain adaptation framework against established methods: (1) **Marginalized Stacked Denoising Autoencoders (mSDA):** Denoising autoencoders that marginalize out noise, enabling learning on *infinitely many* corrupted training samples (Chen et al., 2012). (2) **Self-Ensembling (SE):** A consistency regularization framework that stabilizes student and teacher network predictions under injected noise (discussed in detail in §3.2-§3.4) (Laine and Aila, 2017; Tarvainen and Valpola, 2017; French et al., 2018). (3) **Domain-Adversarial Neural Network (DANN):** Multi-component framework that learns domain-invariant representations through adversarial training (Ganin et al., 2016). We also benchmark against **Source Only** (classifier trained on the source domain only) and **In-Domain** (classifier trained on the target domain only) to establish lower and upper performance bounds, respectively (Zhang et al., 2017).

Framework results are presented in Table 2. Our

	Binary Task		Multi-Label Task		
Model	Mi-F	Ma-F	Ma-P	Ma-R	Ma-F
LR	63.7	60.9	53.3	67.3	31.9
BiLSTM	64.8	63.1	36.2	65.0	46.3
CNN (2D)	73.1	72.1	**49.0**	73.8	58.9
CNN (ours)	75.4	75.3	36.9	**91.4**	52.5
+ seq squeeze	75.1	74.6	45.8	79.6	58.2
+ state conn	**80.2**	76.3	45.3	85.5	59.2
+ time emb	77.4	**77.1**	48.2	83.5	**61.1**

Table 3: Model results with adaptive ensembling for the binary label task (left) and multi-label task (right). For the binary task, we show micro- and macro-averaged F1 scores. For the multi-label task, we show macro-averaged precision, recall, and F1 scores.

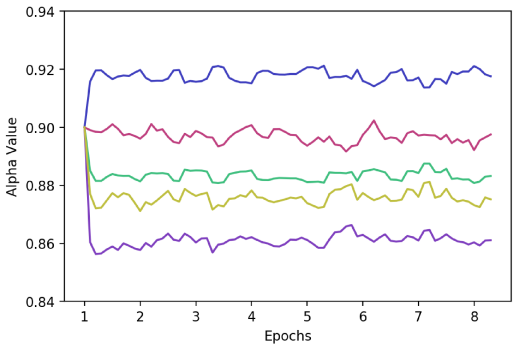


Figure 4: Trajectories of five randomly sampled adaptive constants from our CNN over the course of training. Best viewed in color.

method achieves the highest F1 scores for both tasks. The temporal curriculum further improves our results by a large margin, validating its effectiveness for domain adaptation on diachronic corpora. Although DANN achieves higher precision on the multi-label task, its recall largely suffers.

6.3 Model Results

Next, we ablate the various components of our model and evaluate several other strong text classification baselines under our framework: (1) **Logistic Regression (LR):** We average the word embeddings of each token in the sequence, then use these to train a logistic regression classifier. (2) **Bidirectional LSTM (BiLSTM):** A bidirectional LSTM obtains forwards and backwards input representations $[\overrightarrow{\mathbf{h}_t}; \overleftarrow{\mathbf{h}_t}]$ (Hochreiter and Schmidhuber, 1997); they are concatenated and passed through a fully-connected layer. (3) **CNN (2D):** A CNN using 2D kernels 3×300, 4×300, and 5×300 obtains representations (Kim, 2014). They are max-pooled, concatenated row-wise, and passed through a fully-connected layer.

Model ablations and results are presented in Table 3. Our full model achieves the highest F1 scores on both the binary and multi-label tasks, and each component consistently contributes to the overall F1 score. The 2D CNN also has decent F1 scores, showing that our framework works with standard CNN models. Further, the time embedding significantly improves both F1 scores, indicating the model effectively utilizes the unique temporal information present in our corpora.

7 Analysis

In this section, we pose and qualitatively answer numerous probing questions to further understand the strong performance of adaptive ensembling. We analyze several characteristics of the overall framework (§7.1), then qualitatively inspect its performance on our datasets (§7.2).

7.1 Framework

Are the adaptive constants different across hidden units? We randomly sample five adaptive constants and track their value trajectories over the course of training. Figure 4 shows all of them sharply converge to and bounce around the same general neighboorhood. This is strong evidence that we cannot use a fixed hyperparameter α to smooth each parameter, rather we need per-parameter smoothing constants to account for the functionality and behavior of each unit.

How do the adaptive constants change by layer? Figure 5 shows the distribution of weight and bias parameters of adaptive constants for a top, middle, and bottom layer of our CNN. For the weight parameters, the teacher relies heavily on the student (α is skewed towards smaller smoothing rates) in the top layer, but gradually reduces its dependence by learning target domain features in the lower layers (α is skewed towards larger smoothing rates). For the bias parameters, the teacher prominently shifts the student features to work for the target domain in the top layer, but reduces its dependence on the student in the lower layers. This shows why using a fixed hyperparameter α does not account for layer-wise dynamics, i.e. each layer requires a specific distribution of α values to achieve strong performance.

Do adaptive constants benefit training and latent representations? Figure 6a depicts the unsupervised loss trajectories for self-ensembling

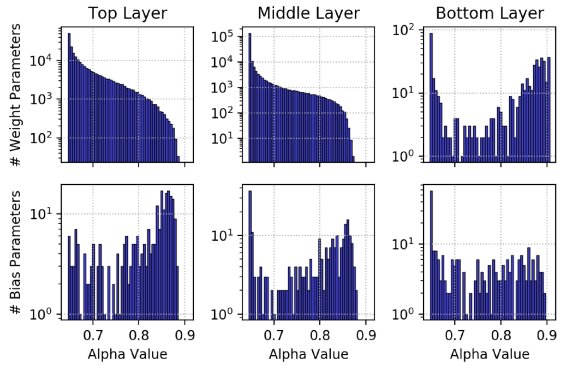
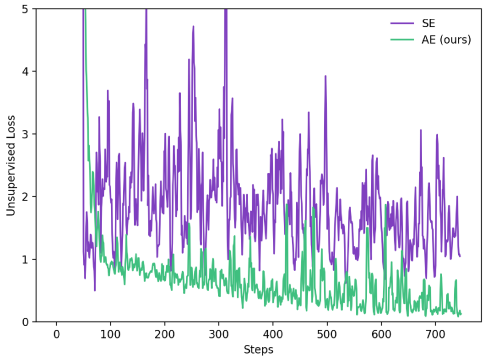


Figure 5: Distribution of teacher network adaptive constants for a top, middle, and bottom layer. We display adaptive constants for both weight (top) and bias (bottom) parameters. The x-axis is shared for both the weight and bias distributions.

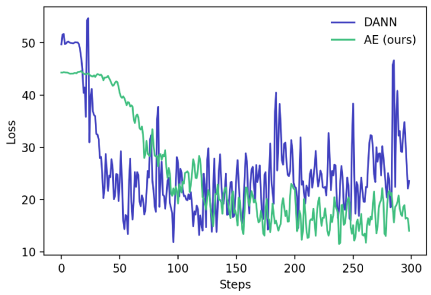
(SE) and adaptive ensembling (AE). Compared to SE, the adaptive constants significantly stabilize the unsupervised loss. Next, Figure 6b shows the general training curves for AE and domain-adversarial neural networks (DANN). The DANN loss oscillates uncontrollably as the adversarial weight increases, but increasing the unsupervised loss weight for AE does not result in as much instability. We also compare the latent representations learned by SE and AE in Figure 7. While SE shows evidence of feature alignment, AE learns a much smoother manifold where source and target domain representations are intertwined.

7.2 Datasets

Does adaptive ensembling yield better topics? In Table 1, we showed that applying LDA directly on COHA yields noisy, unrecognizable topics. Here, we use the SOURCE ONLY model and the adaptive ensembling framework to obtain labels for the unlabeled pool of COHA documents. We extract the political documents, run a topic model on the political subcorpus, and randomly sample topics. The SOURCE ONLY results are shown in Table 4 and the adaptive ensembling results are shown in Table 5. The SOURCE ONLY model has poor recall, as most of the topics are extracted are vague and not inherently political in nature. In contrast, our framework is able to extract a wide range of clean, identifiable political topics. For example, the first topic reflects documents related to the Vietnam conflict while the third topic reflects documents related to important court proceedings.



(a) Loss curves for self-ensembling (SE) and adaptive ensembling (AE) over the course of training.



(b) Loss curves for domain-adversarial neural networks (DANN) and adaptive ensembling (AE). The adversarial loss weight and unsupervised loss weight are annealed for both methods. For fair comparison, we employ the adversarial weight annealing schedule outlined in Ganin et al. (2016).

Topic 1	dr, women, week, medical, doctors
Topic 2	city, police, street, car, avenue
Topic 3	trial, years, police, prison, court
Topic 4	union, strike, workers, lewis, service
Topic 5	like, man, years, little, week

Table 4: Randomly sampled topics and top keywords derived from a 50-topic LDA model trained on 28K COHA articles identified as political using the SOURCE ONLY model.

Does adaptive ensembling preserve the integrity of the original corpus? In order for political scientists to effectively study latent variables—such as political polarization—over time, the extracted political subcorpus must contain a similar integrity as the original corpus. That is, the subcorpus' distribution of documents across years and sources must relatively match that of the original corpus. First, we analyze the document counts for each decade bin, shown in Figure 8. The political subcorpus shows a relatively consistent count across the deacdes, notably also capturing salient peaks from the 1920-1930s. Next, we analyze the document counts for each news source. Once again, the political subcorpus features documents from *all* sources that appear in

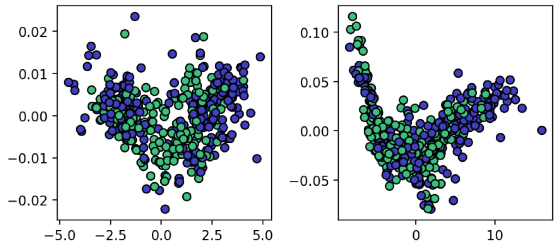


Figure 7: PCA performed on the latent representations of the teacher network in self-ensembling (left) and adaptive ensembling (right). We show representations for both source domain samples (green) and target domain samples (blue). Best viewed in color.

Topic 1	vietnam, hanoi, atomic, bombing, south
Topic 2	germany, britain, france, europe, soviet
Topic 3	court, justice, commission, law, attorney
Topic 4	tax, oil, prices, petroleum, industry
Topic 5	coal, union, strike, workers, miners

Table 5: Randomly sampled topics and top keywords derived from a 50-topic LDA model trained on 28K COHA articles identified as political using ADAPTIVE ENSEMBLING.

the original corpus. In addition, the varied distribution across sources is also captured; Time Magazine (TM) has the most documents whereas Wall Street Journal (WSJ) has the least documents. Together, these results show that the resulting subcorpus is amenable for political science research as it exhibits important characteristics derived from the original COHA corpus.

8 Related Work

Early approaches for unsupervised domain adaptation use shared autoencoders to create cross-domain representations (Glorot et al., 2011; Chen et al., 2012). More recently, Ganin et al. (2016) introduce a new paradigm that create domain-invariant representations through adversarial training. This has gained popularity in NLP (Zhang et al., 2017; Fu et al., 2017; Chen et al., 2018), however the difficulties of adversarial training are well-established (Salimans et al., 2016; Arjovsky and Bottou, 2017). Consistency regularization methods (e.g., self-ensembling) outperform adversarial methods on visual semi-supervised and domain adaptation tasks (Athiwaratkun et al., 2019), but have rarely been applied to textual data (Ko et al., 2019). Finally, Huang and Paul (2018) establish the feasibility of using domain adaptation to label documents from discrete time periods.

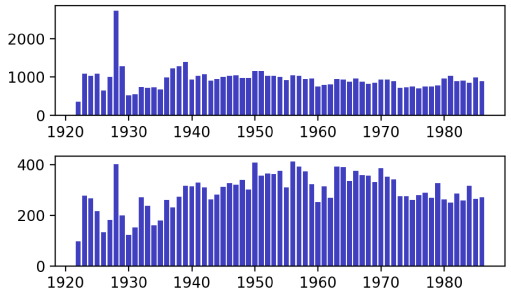


Figure 8: Document counts per decade for the original COHA corpus (top) and the extracted political subcorpus (bottom).

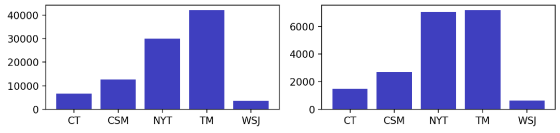


Figure 9: Document counts per news source for the original COHA corpus (left) and the extracted political subcorpus (right). The sources displayed include Chicago Tribute (CT), Christian Science Monitor (CSM), New York Times (NYT), Time Magazine (TM), and Wall Street Journal (WSJ).

Our work departs from previous work by proposing an adaptive, time-aware approach to consistency regularization provisioned with causal convolutional networks.

9 Conclusion

We present *adaptive ensembling*, an unsupervised domain adaptation framework capable of identifying political texts for a multi-source, diachronic corpus by only leveraging supervision from a single-source, modern corpus. Our methods outperform strong benchmarks on both binary and multi-label classification tasks. We release our system, as well as an expert-annotated set of political articles from COHA, to facilitate domain adaptation research in NLP and political science research on public opinion over time.

Acknowledgments

The authors acknowledge the Texas Advanced Computing Center (TACC) at The University of Texas at Austin for providing HPC resources used to conduct this research. Thanks as well to Greg Durrett, Katrin Erk, and the anonymous reviewers for their helpful comments. This work was partially supported by the NSF Grant IIS-1850153.

References

Brice DL Acree, Justin H Gross, Noah A Smith, Yanchuan Sim, and Amber E Boydstun. 2018. Etch-a-sketching: Evaluating the post-primary rhetorical moderation hypothesis. *American Politics Research*.

Martin Arjovsky and Léon Bottou. 2017. Towards principled methods for training generative adversarial networks. In *International Conference on Learning Representations*.

Ben Athiwaratkun, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson. 2019. There are many consistent explanations of unlabeled data: Why you should average. In *International Conference on Learning Representations*.

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*.

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. 2019. Trellis networks for sequence modeling. In *International Conference on Learning Representations*.

Delia Baldassarri and Andrew Gelman. 2008. Partisans without constraint: Political polarization and trends in american public opinion. *American Journal of Sociology*, 114(2):408–446.

Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2019. Identifying and controlling important neurons in neural machine translation. In *International Conference on Learning Representations*.

Matthew A Baum and Philip BK Potter. 2008. The relationships between mass media, public opinion, and foreign policy: Toward a theoretical synthesis. *Annual Review of Political Science*, 11:39–65.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 41–48.

George F Bishop. 2004. *The illusion of public opinion: Fact and artifact in American public opinion polls*. Rowman & Littlefield Publishers.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022.

Angus Campbell, Philip E Converse, Warren E Miller, and Donald E Stokes. 1980. *The American Voter*. University of Chicago Press.

Minmin Chen, Zhixiang Xu, Kilian Q. Weinberger, and Fei Sha. 2012. Marginalized denoising autoencoders for domain adaptation. In *Proceedings of the 29th International Conference on International Conference on Machine Learning*, pages 1627–1634.

Xilun Chen, Yu Sun, Ben Athiwaratkun, Claire Cardie, and Kilian Weinberger. 2018. Adversarial deep averaging networks for cross-lingual sentiment classification. *Transactions of the Association for Computational Linguistics*, 6:557–570.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Eldad Davidov, Bart Meuleman, Jan Cieciuch, Peter Schmidt, and Jaak Billiet. 2014. Measurement equivalence in cross-national research. *Annual review of sociology*, 40:55–75.

Mark Davies. 2008. The corpus of contemporary american english: 450 million words, 1990-present.

Zachary Elkins and Robert Shaffer. 2019. On measuring textual similarity. Work in progress.

Geoff French, Michal Mackiewicz, and Mark Fisher. 2018. Self-ensembling for visual domain adaptation. In *International Conference on Learning Representations*.

Lisheng Fu, Thien Huu Nguyen, Bonan Min, and Ralph Grishman. 2017. Domain adaptation for relation extraction with domain adversarial neural network. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 425–429.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th International Conference on Machine Learning*, pages 513–520.

Robert E Goodin. 2009. *The Oxford handbook of political science*, volume 11. Oxford University Press.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Xiaolei Huang and Michael J Paul. 2018. Examining temporality in document classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 694–699.

Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. 2016. Neural machine translation in linear time. *arXiv preprint arXiv:1610.10099*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference for Learning Representations*.

Wei-Jen Ko, Greg Durrett, and Junyi Jessy Li. 2019. Domain agnostic real-valued specificity prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Samuli Laine and Timo Aila. 2017. Temporal ensembling for semi-supervised learning. In *International Conference for Learning Representations*.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440.

Scott de Marchi, Spencer Dorsey, and Michael J Ensley. 2018. Policy and the structure of roll call voting in the us house. *Available at SSRN 3262316*.

Maxwell McCombs. 2018. *Setting the agenda: Mass media and public opinion*. John Wiley & Sons.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems 29*, pages 2234–2242.

Evan Sandhaus. 2008. The New York Times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems 30*, pages 1195–1204.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30*, pages 5998–6008.

Zichao Yang, Zhiting Hu, Ruslan Salakhutdinov, and Taylor Berg-Kirkpatrick. 2017. Improved variational autoencoders for text modeling using dilated convolutions. In *Proceedings of the 34th International Conference on Machine Learning*, pages 3881–3890.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489.

John R Zaller et al. 1992. *The nature and origins of mass opinion*. Cambridge University Press.

Yuan Zhang, Regina Barzilay, and Tommi Jaakkola. 2017. Aspect-augmented adversarial networks for domain adaptation. *Transactions of the Association for Computational Linguistics*, 5:515–528.

A LDA Topic Model

We experimented with a range of hyperparameters to ensure the Latent Dirichlet Allocation (LDA) model was best optimized for our datasets, leveraging the Gensim[3] library. In particular, we removed all stopwords, extremely rare words (tail 10-20% from a unigram distribution), and set the number of topics to 50.

B Self-Ensembling

The core intuition behind consistency regularization is that ensembled predictions are more likely to be correct than single predictions (Laine and Aila, 2017; Tarvainen and Valpola, 2017). To this end, Laine and Aila (2017) introduce a **student** and **teacher** network that yield single predictions and ensembled predictions, respectively.

After learning from labeled samples, the student may produce varying, dissimilar predictions for unlabeled samples due to the stochastic nature of optimization. One potential solution is to ensemble predictions across time to converge at the *most likely* prediction (Laine and Aila, 2017). Tarvainen and Valpola (2017) improve upon this method by showing that ensembling parameters (as opposed to predictions) results in better predictions. Because the teacher's parameters are smoothed with the student's learned parameters at each iteration, the teacher effectively becomes an ensemble of the student across time.

Further, to ensure that the features learned from the labeled samples are compatible with the unlabeled samples, Laine and Aila (2017); Tarvainen and Valpola (2017); French et al. (2018) motivate a consistency-enforcing approach to bring the student and teacher's predictions closer together. In essence, if a feature learned from samples in the labeled domain is incompatible with samples in the unlabeled domain, the consistency (unsupervised) loss penalizes its incompatibility. Therefore, the interplay between these two networks creates a robust, domain-invariant feature space that characterizes both labeled and unlabeled samples (French et al., 2018). A detailed visualization of the training procedure is presented in Figure 1 in the main body of this paper.

[3] https://radimrehurek.com/gensim/

	Political			Non-Political
	AG	PE	IR	
Train	333	8	156	497
Dev	82	1	33	116
Test	125	8	47	208

Table 6: Distribution of train (In-Domain benchmark *only*), dev, test documents in our expert-annotated COHA subcorpus. For political documents, we break down the distribution into American Government (AG), Political Economy (PE), and International Relations (IR).

C NYT Descriptors

We build a list of "political" descriptors in NYT to determine (a) which labels we can or cannot sample non-political documents from; and (b) which descriptors fall under the three areas of political science we consider for our multi-label task (American Government, Political Economy, and International Relations).

Because documents can be tagged with multiple descriptors, we build a list of descriptors whose documents have significant overlap with US POLITICS & GOVERNMENT. The second author, a political science graduate student, filtered this list to 57 descriptors that are political in nature.

For (a), we sample 4,600 non-political documents whose descriptors do not overlap with the 57 political descriptors described above. For (b), the same political science graduate student assigns each descriptor with one or more area labels. We use this label information to build an NYT dataset for our tasks. The 57 political descriptors and their corresponding area labels are tabulated in Table 7.

D Expert-Annotated Dataset

To create an initial COHA subcorpus of 56,000 documents (8,000 per decade), we sample from the following news sources that consistently appear in across decades: Chicago Tribune, Christian Science Monitor, New York Times, Time Magazine, and Wall Street Journal. Note that these NYT articles (up to year 1986) do not appear in the NYT annotated corpus (Sandhaus, 2008) (starting from year 1987), which we used as our source, training dataset.

From this subcorpus, we perform additional steps to create an expert-annotated dataset (§5). Label distributions for our dataset are presented in Table 6. Although political economy (PE) is

severely underrepresented, we experimentally find that these documents have salient features and are not as difficult to classify. In addition, we employ class imbalance penalties to prevent our model from ignoring these documents.

The source dataset (NYT) was already annotated; to ensure label agreement with our target dataset (COHA), we sampled documents from the source dataset and had our political science graduate students label them to compare against the original label. There were minimal problems here—because NYT has fine-grained labels for their documents, the politically-labeled articles were clearly political and vice-versa.

The target datatset (COHA) was divided into halves and each political science graduate student annotated a half. Prior to annotation, they agreed upon a set of rules to minimize bias in the annotation process. In addition, both of them worked side-by-side during all annotation periods, so they were able to ask each other's opinion in case there was confusion. We also took measures to ensure label correctness after annotation was completed. Each political science graduate student sampled a batch of their political and non-political annotations and sent it to the other to evaluate. Again, there was not much disagreement here as the rules decided upon in the beginning were sufficient to cover most edge cases. Quantitatively, Cohen's $\kappa = 0.95$ as calculated on a mutually annotated subset (Cohen, 1960).

	Area Label		
Topic	**AG**	**PE**	**IR**
International Relations			✓
Presidents and Presidency (US)	✓		
Presidential Elections (US)	✓		
War and Revolution			✓
Presidential Election of 2000	✓		
Presidential Election of 2004	✓		
Law and Legislation	✓		
Civil War and Guerrilla Warfare			✓
International Trade and World Market		✓	
Presidential Election of 1996	✓		
Public Opinion			
Economic Conditions and Trends		✓	
Bombs and Explosives			✓
Arms Sales Abroad			✓
United States Economy		✓	
Missiles and Missile Defense Systems			✓
Oil (Petroleum) and Gasoline		✓	
Appointments and Executive Changes	✓		
Foreign Service			✓
Prisoners of War			✓
War Crimes, Genocide and Crimes Against Humanity			✓
Vice Presidents and Vice Presidency (US)	✓		
Arms Control and Limitation and Disarmament			✓
Military Bases and Installations			✓
Presidential Election of 2008	✓		
Whitewater Case	✓		
Vietnam War	✓		✓
Governors (US)	✓		
Energy and Power		✓	
Stocks and Bonds		✓	
State of the Union Message (US)	✓		
Wages and Salaries		✓	
Church-State Relations	✓		
Shiite Muslims			✓
Special Prosecutors (Independent Counsel)	✓		
White House (Washington, DC)	✓		
Federal Taxes (US)		✓	
Illegal Aliens	✓		
Social Security (US)	✓		
Political Prisoners	✓		✓
Watergate Affair	✓		
Government Employees	✓		
Sunni Muslims			✓
Third World and Developing Countries			✓
Customs (Tariff)		✓	
Welfare (US)		✓	
Gun Control	✓		
Global Warming	✓		
Interest Rates		✓	
Vetoes (US)	✓		
Futures and Options Trading		✓	
Attorneys General	✓		
Layoffs and Job Reductions		✓	
Nazi Policies Toward Jews and Minorities			✓
Government Bonds		✓	
Police Brutality and Misconduct	✓		
Executive Privilege, Doctrine of	✓		

Table 7: Political descriptors in NYT. Each descriptor is categorized under one or more political science areas: American Government (AG), Political Economy (PE), and International Relations (IR).