applied sciences

MDPI

Article

Multi-Stage Prompt Tuning for Political Perspective Detection in Low-Resource Settings

Kang-Min Kim [1,2,†], Mingyu Lee [3,†], Hyun-Sik Won [2], Min-Ji Kim [2], Yeachan Kim [3] and SangKeun Lee [3,4,*]

[1] Department of Data Science, The Catholic University of Korea, Bucheon 14662, Republic of Korea; kangmin89@catholic.ac.kr
[2] Department of Artificial Intelligence, The Catholic University of Korea, Bucheon 14662, Republic of Korea; abugda@catholic.ac.kr (H.-S.W.); kimmin122@catholic.ac.kr (M.-J.K.)
[3] Department of Artificial Intelligence, Korea University, Seoul 02841, Republic of Korea; decon9201@korea.ac.kr (M.L.); yeachan@korea.ac.kr (Y.K.)
[4] Department of Computer Science and Engineering, Korea University, Seoul 02841, Republic of Korea
* Correspondence: yalphy@korea.ac.kr
† These authors contributed equally to this work.

Abstract: Political perspective detection in news media—identifying political bias in news articles—is an essential but challenging low-resource task. Prompt-based learning (i.e., discrete prompting and prompt tuning) achieves promising results in low-resource scenarios by adapting a pre-trained model to handle new tasks. However, these approaches suffer performance degradation when the target task involves a textual domain (e.g., a political domain) different from the pre-training task (e.g., masked language modeling on a general corpus). In this paper, we develop a novel multi-stage prompt tuning framework for political perspective detection. Our method involves two sequential stages: a domain- and task-specific prompt tuning stage. In the first stage, we tune the domain-specific prompts based on a masked political phrase prediction (MP3) task to adjust the language model to the political domain. In the second task-specific prompt tuning stage, we only tune task-specific prompts with a frozen language model and domain-specific prompts for downstream tasks. The experimental results demonstrate that our method significantly outperforms fine-tuning (i.e., model tuning) methods and state-of-the-art prompt tuning methods on the SemEval-2019 Task 4: Hyperpartisan News Detection and AllSides datasets.

Keywords: political bias detection; pre-trained language model; prompt-based learning; prompt tuning; self-supervised learning

Citation: Kim, K.-M.; Lee, M.; Won, H.-S.; Kim, M.-J.; Kim, Y.; Lee, S. Multi-Stage Prompt Tuning for Political Perspective Detection in Low-Resource Settings. *Appl. Sci.* **2023**, *13*, 6252. https://doi.org/10.3390/app13106252

Academic Editor: Andrea Prati

Received: 22 April 2023
Revised: 15 May 2023
Accepted: 18 May 2023
Published: 19 May 2023

1. Introduction

Political perspective detection in news media is more challenging than that in political texts. It is a document-level classification task that seeks to identify author biases (e.g., left-leaning, right-leaning, and no bias) in news articles. Political perspective detection in news media is vital in various applications, such as political polling and news recommendation systems [1,2]. In addition, identifying perspective differences can help lay the foundation for the automatic detection of false content and rumors [2].

Identifying the author biases in news articles may be more difficult than in political texts. This is primarily attributed to the fact that news articles endeavor to maintain credibility and an appearance of impartiality. This may be a challenging task even for people (if they lack political knowledge). Moreover, with long sequences and a large amount of jargon, labeling the political perspective on documents would be labor-intensive and expensive. Under these circumstances, studying political perspective detection systems in low-resource settings plays a critical role.

Prompt-based learning (i.e., discrete prompting and prompt tuning), which freezes all parameters of a pre-trained language model and steers the model by prepending natural

language or continuous prompts to the input, has achieved notable results in low-resource settings [3–5]. In [3], the authors prepend task instructions in a natural language form and a few examples to the input to perform downstream tasks. They achieve promising results in a wide range of natural language processing (NLP) tasks without updating parameters. In [4], they reformulate input examples as cloze-style phrases to help language models understand a given task and achieve significant improvements over fine-tuning (i.e., model tuning), modifying all model parameters, in natural language understanding (NLU) tasks. In another line of work [5], they propose prefix-tuning, which prepends trainable continuous prompts (i.e., prefix) to the input and outperforms fine-tuning in low-data settings for natural language generation (NLG) tasks. Various methods have emerged to learn a new task with very few samples, but prompt-based learning basically benefits from prior knowledge stored in pre-trained language models.

Despite promising results, some challenges remain in applying prompts to political perspective detection tasks. Specifically, prompt-based learning results in performance degradation when the distribution of inputs differs between the source (e.g., the general domain) and target domains (e.g., the political domain) [6]. Intuitively, the significant differences between the source and target domains in the terminologies used and their contextual meanings evoke this phenomenon. For example, "abortion issue" is not a commonly used term but is used relatively frequently in the political domain. Moreover, the expression "right" is generally used to indicate the meaning "correct" or a direction, but in the political domain, it is mainly used to indicate democracy. Therefore, language models pre-trained on a general corpus may suffer from prompt-based learning on political perspective detection tasks.

To address these issues, we develop a multi-stage continuous prompt tuning framework for political perspective detection in low-resource settings. In the first stage, we inject political knowledge into short domain-specific prompts by post-training on an unlabeled political news corpus. We carefully design a post-training task called masked political phrase prediction (MP3). To this end, we construct and utilize background knowledge (i.e., political-issue-specific phrases). Our framework then masks the political-issue-specific phrases in an unlabeled political news corpus. Finally, we train the domain-specific prompts using a frozen language model to predict the contiguous masked tokens corresponding to the selected political phrases. In the second stage, we freeze both the pre-trained language model and domain-specific prompts and only tune the task-specific prompts on the downstream tasks. This allows us to perform prompt-based learning in a political domain using a language model trained with a general corpus. To the best of our knowledge, our current work is one of only a few works that address the political perspective detection in low-resource settings.

We conduct experiments on two political perspective detection benchmarks: SemEval-2019 Task 4: Hyperpartisan News Detection [7] and AllSides [2]. The experimental results show that the proposed multi-stage prompt tuning method yields significantly improved results in political perspective detection. The contributions of this study are as follows:

- We develop a multi-stage continuous prompt tuning framework for political perspective detection in low-resource settings.
- We propose a domain-specific prompt tuning method for the domain adaptation of pre-trained language models through the MP3 task. For the MP3 task, we construct political-issue-specific phrases and mask them from the news corpus.
- The performance evaluation clearly shows that our model outperforms strong baselines in political perspective detection tasks in few-shot settings.

The paper is structured as follows: in Section 2, we give a brief introduction to pre-trained language models that use deep transformer encoders and prompt-based learning methods. Section 3 outlines the proposed framework for political perspective detection. In Sections 4 and 5, we demonstrate the performance evaluation results and conduct an in-depth analysis. Related research is discussed in Section 6, and we conclude the paper in Section 7.

2. Preliminary

2.1. Transformers

Transformers [8] are composed of stacked layers, where each layer contains a multi-head attention module and a fully connected feed-forward network (FFN). The attention function can be formulated as follows:

$$Attention(x) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V, \quad (1)$$

where $Q \in \mathbb{R}^{n \times d_k}$ is the query matrix, $K \in \mathbb{R}^{m \times d_k}$ is the key matrix, and $V \in \mathbb{R}^{m \times d_v}$ is the value matrix. Here, n denotes the number of queries, m denotes the number of keys and values, and d_k and d_v denote the dimensions of the keys and values, respectively. Each query matrix Q, key matrix K, and value matrix V is obtained as follows:

$$\{Q, K, V\}(x) = W_{\{q,k,v\}}x + b_{\{q,k,v\}}, \quad (2)$$

where $W_{\{q,k,v\}}$ and $b_{\{q,k,v\}}$ are learnable weights and biases specific to the query, key, and value matrices, respectively. The multi-head attention performs N heads in parallel and concatenates their outputs to form the input to FFN with a ReLU activation function in between:

$$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2, \quad (3)$$

where W_1 and W_2 are learnable weight matrices, and b_1 and b_2 are learnable biases.

2.2. Pre-Trained Language Models Based on Transformer

We then introduce the base model of the proposed framework. Many studies have utilized pre-trained language models based on transformer architecture [8] for various NLP tasks. In particular, deep transformer encoder-based models, such as the bidirectional encoder representations from transformers (BERT) [9] and the robustly optimized BERT approach (RoBERTa) [10], use self-supervised pre-training approaches for NLU tasks. They optimize pre-training objectives, such as masked language modeling (MLM) and inter-sentence modeling, on an extensive collection of unlabeled text, before fine-tuning them for a particular downstream task.

In BERT [9], the MLM randomly replaces certain tokens in the input sentences with a *[MASK]* token and then trains the model to predict the original vocabulary word that corresponds to each masked token using the contextual information provided by other words in the sentence. They uniformly select 15% of input tokens as possible replacements. However, no masked token is observed during the fine-tuning phase. To alleviate this issue, they use a strategy in which the selected token is replaced with a actual *[MASK]* token 80% of cases, while a random token is used in 10% of cases and the original token is retained in 10% of cases during the masking process. Subsequent studies such as RoBERTa [10] extend the MLM to improve the performance of BERT further. Instead of static masking, they propose dynamic masking. We select RoBERTa to apply our multi-stage prompt tuning framework. We enhance the base model by multi-stage prompt tuning such that the model can understand the semantic meaning of the political article more deeply.

2.3. Prompt-Based Learning Methods

Many studies have utilized prompt-based learning methods, such as discrete prompting [11–13] and prompt tuning (the term "prompt tuning" is used to describe a group of methods rather than a specific method) [5,14,15] for the few-shot and parameter-efficient learning of the pre-trained language models on downstream tasks. Although we adopt the prompt tuning approach for our framework, we explain discrete prompting and prompt tuning for better understanding.

In a pre-trained language model $\mathcal{M}$, a set of discrete input tokens $\mathbf{x}_{1:n} = \{x_0, x_1, ..., x_n\}$ are mapped to input embeddings $\{e(x_0), e(x_1), ..., e(x_n)\}$ by a pre-trained embedding layer $e \in \mathcal{M}$.

2.3.1. Discrete Prompting

Several studies utilize discrete prompts with pre-trained language models to solve downstream tasks without additional parameter tuning [11–13]. In their works, they construct a template T that predicts output $\mathbf{y}$ based on input $\mathbf{x}$ and prompt P. For example, in the political perspective detection task, the template T could be "The political perspective of this document *[Political Document]* is *[MASK]*.", where the prompt is "The political perspective of this document ... is ...". This template is used to predict the political perspective for "*[MASK]*" based on "The political perspective of this document *[Political Document]* is". For simplicity, let $\mathcal{V}$ denote the vocabulary of language model $\mathcal{M}$ and let $[P_i]$ indicate the i-th prompt token of template T. The template $T = \{[P_{0:i}], \mathbf{x}, [P_{i+1:m}], \mathbf{y}\}$ (where $[P_i] \in \mathcal{V}$) is embedded into the vectors as follows:

$$T = \{[e(P_{0:i})], e(\mathbf{x}), [e(P_{i+1:m})], e(\mathbf{y})\} \tag{4}$$

However, the performance of using discrete prompts with vocabulary in the language model is sensitive to choosing examples or templates.

2.3.2. Prompt Tuning

To overcome the limitations of discrete prompting, prompt tuning methods such as prefix-tuning [5] and P-tuning [14] introduce a pseudo token $[P_i]$ that is not limited to vocabulary $\mathcal{V}$ of the language model $\mathcal{M}$. P-tuning [14] uses pseudo tokens as an alternative to discrete prompts and uses bi-directional long short-term memory and two-layer multi-layer perceptron (MLP) to embed them into continuous prompts h, represented as follows:

$$T = \{h_0, ..., h_i, e(\mathbf{x}), h_{i+1}, ..., h_m, e(\mathbf{y})\}, \tag{5}$$

where m is the length of the pseudo tokens, and h_i is the trainable embedding vector of the continuous prompts. Continuous prompts are then trained to find the prompts that minimize the loss function as follows:

$$\hat{h}_{0:m} = \arg\min_{h} \mathcal{L}(\mathcal{M}(\mathbf{x}, \mathbf{y})) \tag{6}$$

This training approach enables effective continuous prompts to be obtained beyond discrete expressions. However, attaching prompts only to the input embeddings results in prompts with minimal influence on the model's prediction. P-tuning v2 [15] modifies the approach by attaching different prompts as prefixes to all layers of the model to address this issue. Continuous prompts used as prefixes $P_k^{(l)}, P_v^{(l)} \in \mathbb{R}^{n \times d/L}$ are created by passing a reparameterization encoder consisting of two-layer MLP.

Here, L is the number of attention heads in the transformer model [8], d is the dimensionality of the hidden representations, n is the length of the input sequence, and the superscript (l) is part of the vector corresponding to the l-th head. The authors then prepend the continuous prompts to keys $K^{(l)}$ and values $V^{(l)}$ of the attention heads in all layers of the transformer model. This results in the computation of a set of multi-head attentions expressed by the following formula:

$$head_l(x) = Attention(e(x)^{(l)} W^{(l)}, [P_k^{(l)} : K^{(l)}], [P_v^{(l)} : V^{(l)}]), \tag{7}$$

where $W^{(l)} \in \mathbb{R}^{d \times d/L}$ is the weight matrix used to generate the queries $Q^{(l)}$. By attaching prompts to all the layers, they also attach prompts to layers close to the output layer that significantly influence the prediction.

3. Methodology

In this section, we describe our multi-stage prompt tuning framework for political perspective detection tasks. Our framework introduces a domain-specific prompt tuning method to adapt a pre-trained language model trained on a general corpus to a specific domain. First, we only tune the domain-specific prompts with a frozen language model on a carefully designed post-training task (i.e., MP3). After tuning the domain-specific prompts, we prepend task-specific prompts to the model for use in downstream tasks. While tuning the task-specific prompts for the downstream task, we freeze both the language model and domain-specific prompts. Figure 1 illustrates the overall process of our framework.

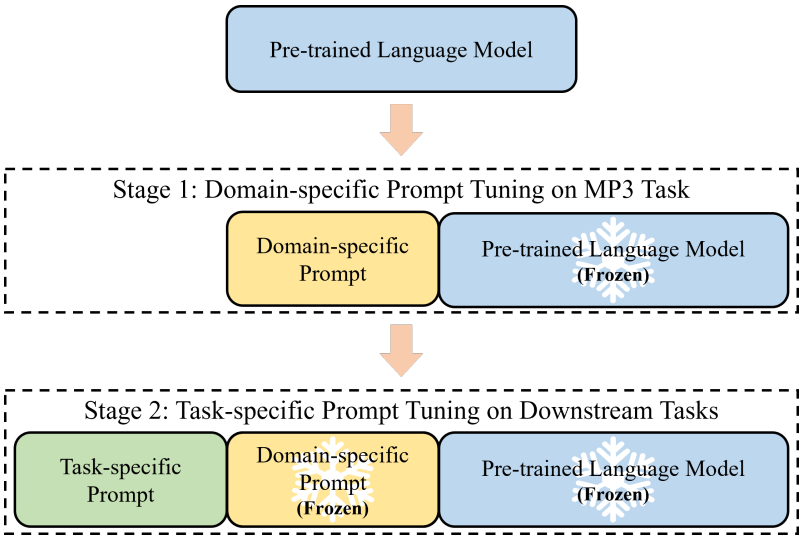


Figure 1. Multi-stage prompt tuning strategy.

3.1. Domain-Specific Prompt Tuning on MP3 Task

3.1.1. Domain-Specific Prompt Tuning

Existing prompt tuning methods attempt to find the optimal prompts after adding prompts to the input to adapt a fixed language model to a specific task. However, the performance of prompt tuning depends primarily on the frequency of the terms in the pre-training corpus [16]. We introduce a domain-specific prompt tuning method that adapts the frozen language model to data distribution in the downstream domain to address this issue. Following [15], domain-specific prompts are applied to all layers of the language model, and then we tune them through a post-training task instead of a downstream task. We first initialize the domain-specific prompts $\mathbf{a} \in \mathbb{R}^{n \times d}$, where n is the length of the domain-specific prompts and d is the embedding size (or the dimensionality of the hidden representations). Domain-specific prompts $\mathbf{a}$ are transformed into $\mathbf{a}_k^{(l)}, \mathbf{a}_v^{(l)} \in \mathbb{R}^{n \times d/L}$ using a reparameterization encoder composed of two-layer MLP with a tanh activation function to prevent unstable training and performance degradation [5]. L represents the number of attention heads in the transformer model, and the superscript (l) represents the vector part corresponding to the l-th head. We prepend these transformed domain-specific prompts to the keys and values of each transformer layer. Domain-specific prompts only adapt the language model to the domain-specific corpus through the MP3 task and are frozen when tuning the model on downstream tasks.

3.1.2. Masked Political Phrase Prediction Task (MP3)

Motivated by the success of masking contiguous spans, noun phrases, and entities [17–19], we introduce the MP3 task, which identifies political-issue-specific phrases and focuses on masking them. An example of MP3 is illustrated in Figure 2.

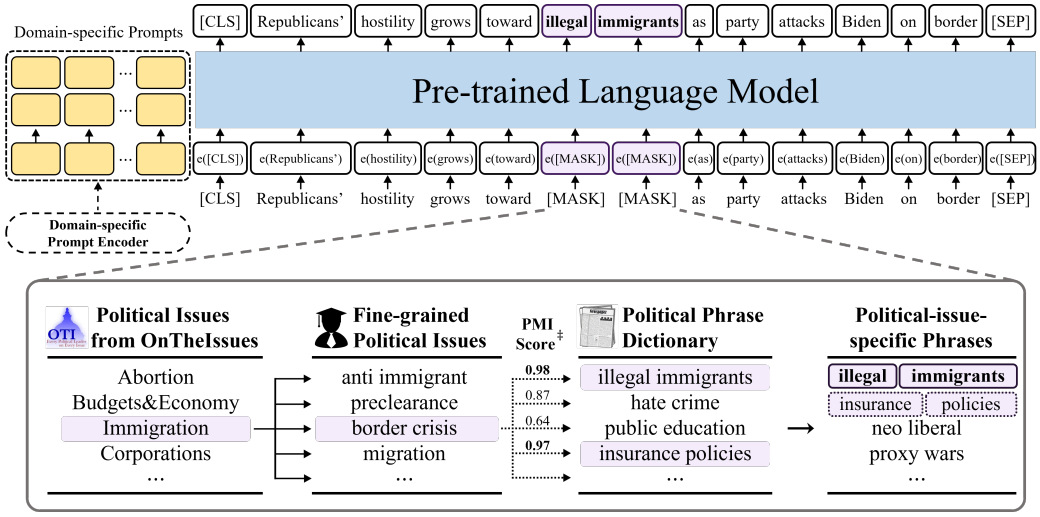


Figure 2. The whole architecture of MP3. ‡ means that the PMI score is not logged for readability.

Constructing Political Lexicon We first construct a lexicon of political terms to inject the political knowledge into domain-specific prompts with MP3. We collect plain news article texts from the politics sections of various news media (we crawled articles from New York Times, Washington Post, Daily Caller, and The Hill) and a benchmark news dataset (we excluded 645 articles with golden labels) [7]. We identify political terms in the corpus by calculating the phrase score (P-score) based on a simple data-driven approach [20] as follows:

$$P - score(\mathrm{w}_i, \mathrm{w}_j) = \frac{count(\mathrm{w}_i\mathrm{w}_j) - \delta}{count(\mathrm{w}_i) \times count(\mathrm{w}_j)}, \tag{8}$$

where w_i and w_j are adjacent words in the corpus. We use δ to prevent excessive phrases consisting of very infrequent words from forming. We use bigrams with scores greater than the threshold as phrases (we empirically set the δ value and threshold to 100). We then repeat the same approach once or twice to identify longer phrases (i.e., trigrams and 4-grams). Finally, the identified terms include 102,849 bigrams, 34,776 trigrams, and 10,161 4-grams.

Extracting Political-Issue Phrase We may not need to use every phrase for post-training because some phrases are meaningless regarding ideological confrontation. Therefore, we extract political-issue-specific phrases from the political lexicon. Inspired by [21], we first collect 23 major and typical political issues in the United States (e.g., abortion, drugs, gun control, health care, and immigration) from OnTheIssues (www.ontheissues.org, accessed on 16 February 2021). Second, we use human political knowledge to extend typical political issues to fine-grained political events or issues. We asked a human expert to provide dozens of fine-grained political issues for each typical issue. We provided hotly debated political events and issue candidates from ISIDEWITH (www.isidewith.com/polls, accessed on 16 February 2021) and Ranker (www.ranker.com, accessed on 16 February 2021). For instance, the human expert can attach fine-grained issues such as "skilled immigration" or "border wall" to "immigration". Finally, we extract the political-issue-specific term by calculating the point-wise mutual information (PMI) [22] with fine-grained political issues. We compute the PMI for a pair of a phrase p and an issue i as follows:

$$PMI(p, i) = log\frac{P(p, i)}{P(p)P(i)}, \tag{9}$$

where $P(i)$ is the probability that i or its fine-grained issues appear in the news articles. We discard phrases that appear in more than 10% of the unlabeled articles. Finally, our proposed model uses 500 top-PMI issue-specific phrases for each of the issues. In addition, we add the single words comprising the above phrases to the lexicon.

Masked Phrase Prediction After extracting the political phrases, we mask them in the new corpus to inject domain knowledge into the domain-specific prompts. First, we search for political phrases mentioned in the corpus. Given a sequence of tokens $\mathbf{T} = (t_1, t_2, ..., t_n)$, we detect political phrases $\mathbf{P}$ and randomly mask them. We mask approximately 50% of the total occurring $\mathbf{P}$. Because political phrases are not always sufficiently present in the sequence, we additionally sample random words until the masking budget is exhausted (approximately 15% of all tokens). We sample sequences of complete words (rather than subword tokens). Language models with domain-specific prompts acquire political knowledge by reconstructing masked political-issue-specific phrases in the corpus. Domain-specific prompts are optimized by minimizing the following loss function:

$$L_{MP3} = -\sum_{\hat{p} \in m(\mathbf{P})} logP(\hat{p}|\mathbf{T}_{\backslash m(\mathbf{P})}), \quad (10)$$

where $m(\mathbf{P})$ and $\mathbf{T}_{\backslash m(\mathbf{P})}$ denote the masked political phrases from $\mathbf{P}$ and the remaining tokens, respectively.

3.2. Task-Specific Prompt Tuning on Downstream Tasks

We observed that using domain-specific prompts directly on downstream tasks decreases performance. This is suspected to be due to catastrophic forgetting, a phenomenon in which acquired task-specific knowledge leads to forgetting previously learned political knowledge. We introduce trainable task-specific prompts for each downstream task to prevent this. Task-specific prompts are created, such as domain-specific prompts. First, we initialize the task-specific prompts $\mathbf{t} \in \mathbb{R}^{m \times d}$, where m is the length of the task-specific prompts. The task-specific prompts $\mathbf{t}$ are transformed into $\mathbf{t}_k^{(l)}, \mathbf{t}_v^{(l)} \in \mathbb{R}^{m \times d/L}$ using a reparameterization encoder composed of two-layer MLP with a tanh activation function. Finally, we prepend the task-specific prompts to domain-specific prompts in each layer of the transformer model.

Unlike domain-specific prompts, task-specific prompts are designed to adapt to downstream tasks by freezing domain-specific prompts and language model parameters during training. In our approach, we feed the *[CLS]* representation of the language model into the output layer to train the task-specific prompts for the political perspective detection task following the work [15]. The sum of the lengths of the domain- and task-specific prompts is less than the length of the prompts in the existing prompt tuning method (which we show in Section 5).

4. Experimental Setup

4.1. Unlabeled News Corpus

First, we train the domain-specific prompts on the collected unlabeled news corpus using the MP3 objective. We collect 69,161 unlabeled news articles from the politics sections of various news media, such as the New York Times (https://www.nytimes.com/section/politics , accessed on 16 February 2021), Washington Post (https://www.washingtonpost.com/politics, accessed on 16 February 2021), Daily Caller (https://dailycaller.com/section/politics/, accessed on 16 February 2021), and The Hill (https://thehill.com/ , accessed on 16 February 2021). In addition, we use news titles and content without labels from a publicly available dataset, SemEval 2019 Task 4—Hyperpartisan News Detection [7]. As previously mentioned, we exclude 645 articles with gold labels used in our evaluation. All the news articles were written in English. They comprise a total of 2.45 GB of plain text.

4.2. Downstream Task Datasets

We conduct experiments on two political perspective detection datasets, SemEval [7] and AllSides [2], adopted as benchmarks in previous studies [2,23–25]. We follow the same evaluation setting as in the work [24] for a fair comparison.

4.2.1. Semeval

This dataset is the official training dataset from SemEval 2019 Task 4: Hyperpartisan News Detection [7]. The objective of this task is to determine if a news article employs hyperpartisan argumentation or not. The dataset contains 645 articles that are labeled manually with a binary label indicating whether they exhibit hyperpartisan behavior. Currently, there is not a test set that is accessible. To compare our results with the top-ranked system, we perform a 10-fold cross-validation on the training set with using the same splits.

4.2.2. Allsides

The AllSides dataset [2] consists of 10,385 news articles from two news aggregation websites on different events, such as terrorism, taxes, the environment, and elections [2]. Websites utilize crowdsourced and editorial-reviewed approaches to indicate the bias of each article. Each article has a political perspective label (e.g., left, center, or right). The statistics for SemEval and AllSides shown in Table 1.

Table 1. Datasets Statistics.

Dataset	# Samples	# Class	Class Distribution
SemEval	645	2	407/238
AllSides	10,385	3	4164/3931/2290

4.3. Baselines

In this section, we compare our model with the following baselines.

4.3.1. Fine-Tuning Methods

- BERT [9] and RoBERTa [10] are large pre-trained language models. They achieve notable results in various NLP tasks by learning language representations using masked language modeling. We use the "large" setting of these models in the experiment.
- MAN [24] utilizes pre-training tasks that integrate social and linguistic information and conducts fine-tuning for political perspective detection.

4.3.2. Prompt-Based Learning Methods

- Lester's prompt tuning [26] uses trainable continuous prompts as an alternative to text prompts. Independent continuous prompts are trained directly for each target task. The backbone of this model is the RoBERTa (large) model used in our experiments.
- P-tuning v2 [15] is the deep prompt tuning method this study adopts. Continuous prompts are applied to each layer of the pre-trained model. The backbone of this model is the RoBERTa (large) model used in our experiments.
- MP-tuning is our proposed multi-stage prompt tuning framework, which uses the RoBERTa (large) model as a backbone.

4.4. Implementation Details

We adopt the RoBERTa large-sized model [10] as a pre-trained language model for prompt tuning and steer it by prepending 30 domain-specific and 40 task-specific prompts (we show the effect of prompt length in Section 5). In addition, we leverage a reparameterization encoder (two-layer and 512 hidden-sized MLP models) to transform trainable embeddings following the previous studies [5,15]. In the MP3 task, we tune only the reparameterization encoder parameters for the domain-specific prompts. We set the learning rate to 10^{-5}, the batch size to 16, and the maximum number of training steps to 10,000 for the MP3 task. All the above-mentioned hyperparameters for MP3 are empirically determined. We then freeze the encoder for domain-specific prompts and tune the encoder for task-specific prompts, which are only used for downstream tasks. We empirically choose the best batch size, number of epochs, and learning rate among {8, 16, 32, 64}, {50, 100, 200},

and $\{10^{-5}, 2 \times 10^{-5}, 5 \times 10^{-5}, 10^{-4}\}$, respectively, based on the development sets of each task.

We use the AdamW optimizer [27] with $\beta 1 = 0.9$, $\beta 2 = 0.999$, an L2 weight decay of 0.01, a learning rate warmup of up to 10% of the maximum training steps, and linear decay of the learning rate. We implement the proposed models using PyTorch [28] and HuggingFace's transformers [29] library. We train the models on a single machine equipped with two Intel Xeon 10-Core processors, 512 GB of RAM, and four NVIDIA TESLA V100 with 32 GB of RAM.

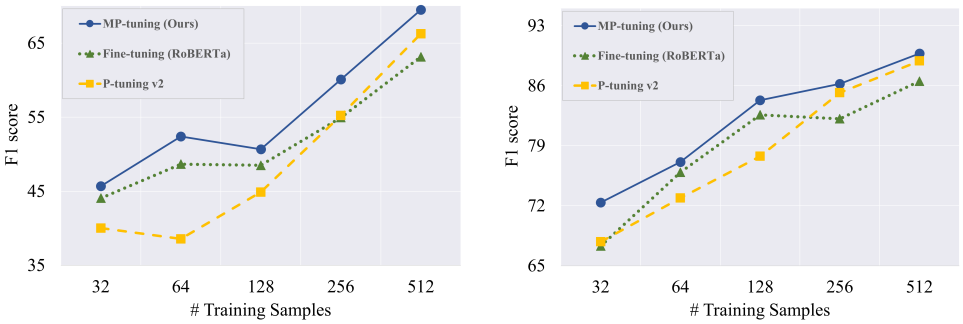
4.5. Evaluation Results

Table 2 shows the performance of our model and the baselines in full data settings. We run five random restarts and report the median accuracy (Acc) and macro-F1 (MaF) scores. As shown in Table 2, MP-tuning outperforms the baselines in terms of the accuracy and macro-F1 score for both datasets. Specifically, MP-tuning achieves 8.0% and 7.0% increases in terms of accuracy and macro-F1, respectively, compared to MAN, the best-performing fine-tuning method on the SemEval dataset. We observe that MP-tuning outperforms MAN in the AllSides dataset as well. Furthermore, MP-tuning outperforms the state-of-the-art prompting method P-tuning v2 in terms of accuracy and macro-F1 score by 1.6% and 2.1%, respectively, on the SemEval dataset and by 2.5% and 4.1%, respectively, on the AllSides dataset.

Table 2. Political perspective detection performance on the SemEval and AllSides dataset. The results marked with ∗ are from our implementations, whereas results marked with † are reported in each reference.

Setting	Model	SemEval		AllSides	
		Acc	MaF	Acc	MaF
Fine-tuning	BERT ∗	86.92	80.71	80.80	79.71
	RoBERTa ∗	87.08	81.34	81.80	80.51
	MAN †	84.66	83.09	81.41	80.44
Prompting	Lester's prompt tuning ∗	82.72	81.35	76.42	74.38
	P-tuning v2 ∗	90.06	87.12	81.18	79.27
	MP-tuning (Ours)	**91.47**	**88.92**	**83.21**	**82.54**

We also investigate the effects of MP-tuning in low-resource settings. We compare MP-tuning with fine-tuning (i.e., original RoBERTa) and the original P-tuning v2. We conduct experiments several times using random sampling on the AllSides and SemEval datasets and report the median F1 score. We systematically evaluate the effect of the number of examples on our models' performance for both datasets, from using 32 examples for training to using 512 examples. As shown in Figure 3, MP-tuning outperforms fine-tuning and P-tuning v2 for each setting. In particular, MP-tuning shows a significant performance improvement compared with fine-tuning, even in significantly low-resource settings (32, 64, 128), whereas P-tuning v2 shows lower performance than fine-tuning. These results demonstrate that our domain-specific prompt tuning on MP3 task alleviates the distribution shift problem of prompt tuning by learning political domain knowledge and adapting a frozen language model. In particular, the outstanding performance of MP-tuning in low-resource settings is advantageous in real-world applications where it is difficult to obtain training data for political perspective detection in news media.



(**a**) AllSides (**b**) SemEval

Figure 3. *k*-shot results on both datasets.

4.6. Ablation Study

We conduct ablation studies for the MP3 task and the domain-specific prompt tuning to understand each component's contribution to our framework. The detailed results are presented in Table 3. An ablation study is conducted using the large-sized RoBERTa model on the AllSides and SemEval datasets. We report the median accuracy and macro-F1 score.

Table 3. Ablation results over components of our models. Here, "MP3" represents political phrase prediction tasks, and "DP" represents domain-specific prompts.

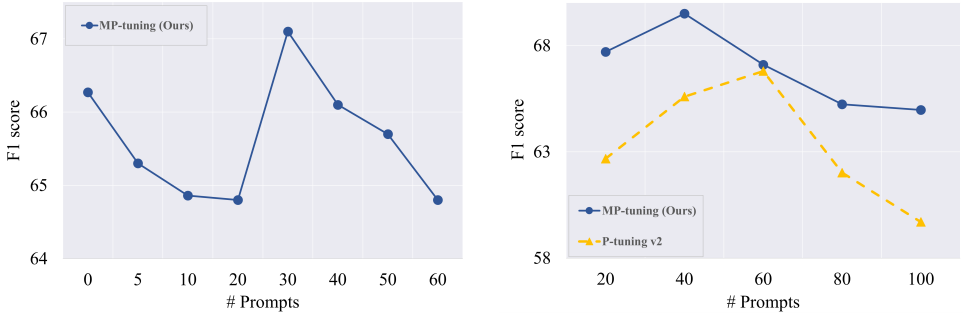
Method	SemEval		AllSides	
	Acc	MaF	Acc	MaF
MP-tuning (Ours)	**91.47**	**88.92**	**83.21**	**82.54**
w/o MP3	91.10	87.72	82.90	81.84
w/o DP	90.81	87.09	81.52	79.97
w/o MP3, DP	90.06	87.12	81.18	79.27

First, the effects of the MP3 task are explored. As shown in Table 3, we can observe that removing the MP3 task (i.e., only MLM without political phrase masking) degrades the performance of MP-tuning on both datasets regarding the accuracy and macro-F1 scores. These results indicate that tuning pre-trained language models with the MP3 task improves the adaptation ability of language models in the political domain. Furthermore, to examine the impact of domain-specific prompts, we compare the results of MP-tuning with those of the model without domain-specific prompts (i.e., task-specific prompts play the role of both task- and domain-specific prompts). As shown in Table 3, MP-tuning without domain-specific prompts exhibits significant performance drops in both tasks. We suspect that even if the prompts learned political knowledge with MP3, learned knowledge would be forgotten during several epochs of training in downstream tasks.

5. Analysis

5.1. Analysis on Domain-Specific Prompt

We analyze the effect of the domain-specific prompt length on downstream task performance. We train the prompts for the RoBERTa model by varying the prompt length in {0, 5, 10, 20, 30, 40, 50, 60} without changing the other settings. In this analysis, we set 60 task-specific prompts. We randomly sample 512 training examples from the AllSides dataset and report the best score. As shown in Figure 4a, MP-tuning exhibits the best performance when the length of the domain-specific prompt is 30. MP-tuning exhibits the worst performance at a domain-specific prompt length of 20. These results indicate that domain knowledge from the domain-specific prompt affects performance more than the missing information caused by the length limitation when the domain-specific prompt is beyond 20 tokens.



(**a**) Domain-Specific Prompts (**b**) Task-Specific Prompts

Figure 4. Analysis for the length of both prompts on Allsides datasets.

5.2. Analysis on Task-Specific Prompt

Because we use additional continuous prompts for specific tasks while freezing the domain-specific prompts, we analyze the effect of task-specific prompt length on downstream task performance. In this analysis, we use 30 tokens for the domain-specific prompt, and the other settings follow those in Section 5.1. As shown in Figure 4b, MP-tuning exhibits the best performance when 40 tokens are used for the task-specific prompt, whereas P-tuning v2 exhibits the best performance when 60 tokens are used. In particular, MP-tuning outperforms P-tuning v2 when 20 tokens are used as task-specific prompts. These results indicate that our proposed method still performs better when using the same number of prompts.

6. Related Work

6.1. Political Perspective Detection

In [30], the problem of automatically identifying the perspective from which a document is written was first addressed. The authors developed a statistical framework to learn the reflection of perspectives in word usage. In [1], they proposed an ideological perspective detection method using word sense disambiguation and latent semantic features. They considered ideologically charged texts, such as political documents or political debates. Recently, few studies have examined political perspective detection in news media. In [2], a graph convolutional network was utilized to contextualize social information (i.e., capturing the dissemination of this information in social networks). However, these approaches are applicable only to news articles shared on social networks. In [31], the authors developed a pre-trained language model-based multi-task learning framework to predict the political perspective of news articles, as well as the party affiliation of politicians and the framing of policy issues. They utilized metaphor and emotion detection as auxiliary tasks to enhance political discourse models. In [24], the authors proposed a framework that pre-trains the model by utilizing various signals from the social and linguistic context, such as entity mentions, news sharing (i.e., social information), and frame indicators. They achieved state-of-the-art performance in political perspective detection. Similarly, we also utilize post-training to enhance the political knowledge of the pre-trained language models. However, unlike the previous approaches, which require social or emotional information, we use only text information from news articles for post-training.

6.2. Prompt-Based Learning

Prompt-based learning prepends natural language prompts or a few trainable parameters to inputs to steer pre-trained language models while keeping these models' parameters frozen. Discrete prompting allows notable performance for pre-trained language models and is effective over fine-tuning in few-shot learning for various tasks. In [3], the authors proposed in-context learning where pre-trained language models are conditioned on input–output examples to perform tasks without optimizing parameters. Various studies have been conducted to improve text prompts using novel techniques, such as prompt

mining [32], gradient-based prompt search [12], and automatic prompt generation [13]. However, these hard prompts are sub-optimal, and the prompting performance is susceptible to the examples or templates used.

Recently, studies on prompt tuning, prepending a few trainable soft prompts to the input, have been actively conducted instead of hard prompts. For example, in [5], the authors tuned prefix activation, prepended to each transformer layer, and achieved promising results on various natural language generation tasks. In another line of work [26], they prepended a few tokens to the input and showed remarkable performance in NLU tasks. Subsequently, P-tuning [15] optimized the prompt tuning to be universally effective across diverse model scales and NLU tasks. SPoT [33] proposed prompt transfer learning, which trains prompts on various source tasks and continuously trains them on target tasks. However, we still observe that even the state-of-the-art prompt-based learning method performs inadequate tuning in a low-resource environment in political domains. In this work, we adapt the model to the political domain through a multi-stage prompt tuning framework without fine-tuning the language model. In particular, unlike previous prompt-based learning approaches, we have utilized prompts not only for downstream tasks but also for domain adaptation.

7. Conclusions

In this paper, we have proposed a novel multi-stage prompt tuning framework for political perspective detection in news media. In particular, we tune domain-specific prompts using a frozen pre-trained language model that learns the MP3 task. We have verified the political perspective detection performance of our methodology using real-world datasets. Our experimental results confirm that our methodology significantly outperforms the strong baseline methods in few-shot and full data settings. In addition, the domain- and task-specific prompts have 2.79% trainable parameters compared to the overall parameters of the language model. This result means that our proposed framework significantly reduces training time, memory cost, and storage cost for domain adaptation, which is useful in real-world applications that require domain-specific language models. We plan to apply our framework to other specific domains such as medicine and finance.

Author Contributions: Conceptualization, S.L.; methodology, K.-M.K.; software, M.L.; validation, M.L.; formal analysis, K.-M.K. and M.L.; investigation, K.-M.K. and M.L.; resources, H.-S.W.; data curation, M.-J.K.; writing—original draft preparation, K.-M.K., M.L. and Y.K.; writing—review and editing, K.-M.K., H.-S.W. and M.-J.K.; visualization, K.-M.K.; supervision, S.L.; project administration, S.L.; funding acquisition, K.-M.K. and S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Basic Research Program through the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A2C3010430), the NRF grant funded by the Korea government (MSIT) (No. 2022R1C1C1010317), the Catholic University of Korea (Research Fund, 2021), and Institute of Information communications Technology Planning Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-00079, Artificial Intelligence Graduate School Program (Korea University)).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The code and the dataset collected by scrapping web pages in this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Elfardy, H.; Diab, M.T.; Callison-Burch, C. Ideological Perspective Detection Using Semantic Features. In Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics (*SEM 2015), Denver, CO, USA, 4–7 June 2015; pp. 137–146.
2. Li, C.; Goldwasser, D. Encoding Social Information with Graph Convolutional Networks for Political Perspective Detection in News Media. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL), Florence, Italy, 28 July–2 August 2019; pp. 2594–2604.
3. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learner. In Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS), Online, 6–12 December 2020; pp. 1877–1901.
4. Schick, T.; Schütze, H. Exploiting Cloze-Questions for Few-Shot Text Classification and Natural Language Inference. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL), Online, 19–23 April 2021; pp. 255–269.
5. Li, X.L.; Liang, P. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL/IJCNLP), Dublin, Ireland, 1–6 August 2021; pp. 4582–4597.
6. Razeghi, Y.; Logan, R.L., IV; Gardner, M.; Singh, S. Impact of Pretraining Term Frequencies on Few-Shot Numerical Reasoning. In Proceedings of the Conference Empirical Methods in Natural Language Processing (EMNLP), Abu Dhabi, United Arab Emirates, 7–11 December 2022; pp. 840–854.
7. Kiesel, J.; Mestre, M.; Shukla, R.; Vincent, E.; Adineh, P.; Corney, D.; Stein, B.; Potthast, M. SemEval-2019 Task 4: Hyperpartisan News Detection. In Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval), Minneapolis, MN, USA, 6–7 June 2019; pp. 829–839.
8. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, L.K.; Polosukhin, L. Attention is all you need. In Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
9. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
10. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692.
11. Schick, T.; Schütze, H. It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Online, 6–11 June 2021; pp. 2339–2352.
12. Shin, T.; Razeghi, Y.; Logan, R.L., IV; Wallace, E.; Singh, S. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL/IJCNLP), Dublin, Ireland, 1–6 August 2021; pp. 3816–3830.
13. Gao, T.; Fisch, A.; Chen, D. Making Pre-Trained Language Models Better Few-Shot Learners. In Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL/IJCNLP), Dublin, Ireland, 1–6 August 2021; pp. 3816–3830.
14. Liu, X.; Zheng, Y.; Du, Z.; Ding, M.; Qian, Y.; Yang, Z.; Tang, J. GPT understands, too. *arXiv* **2021**, arXiv:2103.10385.
15. Liu, X.; Ji, K.; Fu, Y.; Tam, W.; Du, Z.; Yang, Z.; Tang, J. P-Tuning: Prompt Tuning Can Be Comparable to Fine-tuning Across Scales and Tasks. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL), Dublin, Ireland, 22–27 May 2022; pp. 61–68.
16. Schucher, N.; Reddy, S.; de Vries, H. The Power of Prompt Tuning for Low-Resource Semantic Parsing. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL), Dublin, Ireland, 22–27 May 2022; pp. 148–156.
17. Zhang, Z.; Han, X.; Liu, Z.; Jiang, X.; Sun, M.; Liu, Q. ERNIE: Enhanced Language Representation with Informative Entities. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL), Florence, Italy, 28 July–2 August 2019; pp. 1441–1451.
18. Sun, Y.; Wang, S.; Li, Y.; Feng, S.; Chen, X.; Zhang, H.; Tian, X.; Zhu, D.; Tian, H.; Wu, H. ERNIE: Enhanced Representation through Knowledge Integration. *arXiv* **2019**, arXiv:1904.09223.
19. Joshi, M.; Chen, D.; Liu, Y.; Weld, D.S.; Zettlemoyer, L.; Levy, O. SpanBERT: Improving Pre-training by Representing and Predicting Spans. *Trans. Assoc. Comput. Linguist. (TACL)* **2020**, *8*, 64–77. [_a_00300CrossRef]. [CrossRef]
20. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS), Lake Tahoe, NV, USA, 5–10 December 2013; pp. 3111–3119.
21. Roy, S.; Goldwasser, D. Weakly Supervised Learning of Nuanced Frames for Analyzing Polarization in News Media. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–20 November 2020; pp. 7698–7716.

22. Church, K.W.; Hanks, P. Word Association Norms, Mutual Information and Lexicography. *Comput. Linguist.* **1990**, 16, 22–29. [CrossRef]
23. Feng, S.; Chen, Z.; Li, Q.; Luo, M. Knowledge Graph Augmented Political Perspective Detection in News Media. *arXiv* **2021**, arXiv:2108.0386.
24. Li, C.; Goldwasser, D. Using Social and Linguistic Information to Adapt Pretrained Representations for Political Perspective Identification. In Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL/IJCNLP), Dublin, Ireland, 1–6 August 2021; pp. 4569–4579.
25. Zhang, W.; Feng, S.; Chen, Z.; Lei, Z.; Li, J.; Luo, M. KCD: Knowledge Walks and Textual Cues Enhanced Political Perspective Detection in News Media. In Proceedings of the 2022 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), Seattle, WA, USA, 10–15 July 2022; pp. 4129–4140.
26. Lester, B.; Al-Rfou, R.; Constant, N. The Power of Scale for Parameter-Efficient Prompt Tuning. In Proceedings of the Conference Empirical Methods in Natural Language Processing (EMNLP), Punta Cana, Dominican Republic, 7–11 November 2021; pp. 3045–3059.
27. Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. In Proceedings of the International Conference on Learning Representations (ICLR), New Orleans, LA, USA, 6–9 May 2019. [CrossRef]
28. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS), Vancouver, BC, Canada, 8–14 December 2019; pp. 8024–8035.
29. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the Conference Empirical Methods in Natural Language Processing: System Demonstrations (EMNLP Demos), Online, 16–20 November 2020; pp. 38–45.
30. Lin, W.-H.; Wilson, T.; Wiebe, J.; Hauptmann, A. Which Side are You on? Identifying Perspectives at the Document and Sentence Levels. In Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL), New York, NY, USA, 8–9 June 2006; pp. 109–116.
31. Cabot, P.-L.H.; Dankers, V.; Abadi, D.; Fischer, A.; Shutova, E. The Pragmatics behind Politics: Modelling Metaphor, Framing and Emotion in Political Discourse. In *Findings of the Association for Computational Linguistics (EMNLP)*; Online; Association for Computational Linguistics location: Toronto, ON, Canada; pp. 4479–4488.
32. Jiang, Z.; Xu, F.F.; Araki, J.; Neubig, G. How Can We Know What Language Models Know. *Trans. Assoc. Comput. Linguist. (TACL)* **2020**, *8*, 423–438. [CrossRef]
33. Vu, T.; Lester, B.; Constant, N.; Al-Rfou, R.; Cer, D. SPoT: Better Frozen Model Adaptation through Soft Prompt Transfer. In Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL/IJCNLP), Dublin, Ireland, 1–6 August 2021; pp. 5039–5059.