ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-4485-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICASSP48485.2024.10447730

LEVERAGING BIASES IN LARGE LANGUAGE MODELS: "BIAS-KNN" FOR EFFECTIVE FEW-SHOT LEARNING

Yong Zhang[1 †], *Hanzhang Li*[1 2 †], *Zhitao Li*[1], *Ning Cheng*[1 *], *Ming Li*[1 3], *Jing Xiao*[1], *Jianzong Wang*[1]

[1]Ping An Technology (Shenzhen) Co., Ltd., China
[2]Lanzhou University, China
[3] University of Maryland

ABSTRACT

Large Language Models (LLMs) have shown significant promise in various applications, including zero-shot and few-shot learning. However, their performance can be hampered by inherent biases. Instead of traditionally sought methods that aim to minimize or correct these biases, this study introduces a novel methodology named "bias-kNN". This approach capitalizes on the biased outputs, harnessing them as primary features for kNN and supplementing with gold labels. Our comprehensive evaluations, spanning diverse domain text classification datasets and different GPT-2 model sizes, indicate the adaptability and efficacy of the "bias-kNN" method. Remarkably, this approach not only outperforms conventional in-context learning in few-shot scenarios but also demonstrates robustness across a spectrum of samples, templates and verbalizers. This study, therefore, presents a unique perspective on harnessing biases, transforming them into assets for enhanced model performance.

Index Terms— LLM, Model Bias, Bias Leverage, kNN Methods, Zero-Shot Learning, Few-shot Learning

1. INTRODUCTION

Large language models (LLMs) have emerged as powerful tools showcasing impressive zero-shot and few-shot capabilities [1, 2]. Leveraging templates and verbalizers [3] to align an LLM's output probability distribution with task-specific labels allows the model to address downstream classification tasks in zero-shot or few-shot contexts.

However, these models are not without their challenges, predominantly stemming from biases. These biases influence both the model's inherent discriminative abilities and its output, skewing probability values for specific categories. Moreover, they can also disrupt conventional decision boundaries, thereby compromising their reliability [4, 5]. Research identifies these biases mainly as vanilla label bias, where frequently encountered words during pre-training get prediction preference, and domain label bias, where the bias manifestation varies based on content domain [6]. Another noteworthy phenomenon is the surface form competition [5], where semantically similar words vie for identical probability space, leading to distributional conflicts.

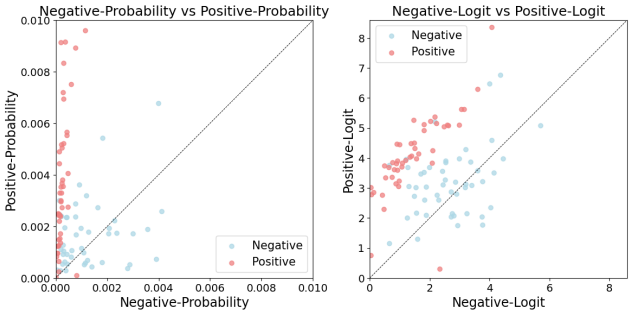

Fig. 1: Zero-shot probability and logit results from the CR train dataset, visualizing 50 samples each from the Positive and Negative categories. The model exhibits a clear bias towards the Positive category. The dashed line $y = x$ denotes the decision boundary for these categories.

Addressing biases in LLMs necessitates a diverse strategy. Initially, some methods focus on direct bias measurement and recalibration. Take, for instance, Contextual Calibration [4] which uses neutral test inputs, such as "N/A" to recalibrate model outputs. In a similar vein, Domain-Context Calibration [6] leverages random in-domain tokens to gauge the bias probability of individual labels. While potent, these approaches sometimes apply a broad-brush correction, occasionally missing the nuanced biases specific to certain test samples.

In another category, methods like PROCA [7] strategize around defining an optimal classification boundary. They draw on the model's contextual insights and employ a Gaussian Mixture Model (GMM) to understand the data spread. Similarly, approaches such as KNN-C [8] and kNN-prompting [9] harness the model representations, emphasizing its capability for representation over prediction [10], to navigate around biases rather than confront them directly.

As depicted in Figure 1, biases in LLMs frequently result in lower probabilities for verbalizers, causing dense clusters and category overlaps. However, the evident directionality

†Equal contribution
*Corresponding authors: Ning Cheng (chengning211@pingan.com.cn)

differences between categories hint at an intriguing opportunity: utilizing biased outputs together with Nearest Neighbor methods to enhance sample inference.

Moving away from traditional strategies that aim to minimize or correct biases, we present a novel approach termed "bias-kNN". In this method, we utilize biased outputs as primary features for kNN, enriched by gold labels. Our evaluations, which encompass various domain text classification datasets and different GPT-2 model sizes [11], demonstrate that "bias-kNN" not only surpasses the performance of traditional in-context learning in few-shot settings but also exhibits robustness across diverse templates and verbalizers.

The contributions of this paper are:

- We unveil a pioneering approach, "bias-kNN", which diverges from traditional strategies that aim to minimize or correct biases. Instead, this method capitalizes on biased outputs by using them as primary features for kNN, further enriched by gold labels.
- Our rigorous evaluations cover a range of domain text classification datasets and span different GPT-2 model sizes, reinforcing the versatility and adaptability of the "bias-kNN" approach.
- The "bias-kNN" approach consistently outperforms traditional in-context learning in few-shot scenarios and exhibits enhanced stability across varied labeled data samples. Furthermore, its proven robustness with diverse templates and verbalizers highlights its resilience and broad applicability.

2. METHODOLOGY

In this section, we detail our kNN modeling technique that enhances text classification by harnessing biases in logit outputs from language models. The method's structure is depicted in Figure 2.

2.1. Bias Output based kNN Modeling

Our approach is based on the idea that model outputs, even with their challenges, have valuable distinguishing features. With this in mind, we adapted the kNN method.

Given a model $\mathcal{M}_\theta$, a domain-specific labeled dataset $\mathcal{A} = \{(x_i, y_i)\}_{i=1}^{|A|}$, a template $\mathcal{T}$, the label set of the domain data $\mathcal{Y} = \{y_j^*\}_{j=1}^{|\mathcal{Y}|}$ and a verbalizer $\mathcal{V}$ to map each label word of $\mathcal{Y}$ to a word v in the $\mathcal{M}_\theta$'s vocabulary. We employed the template $\mathcal{T}$ to structure each x_i as $\mathcal{T}(x_i)$ and feed to the model $\mathcal{M}_\theta$ to get the probability output $\mathcal{P}$, representing the probability $p_{\left(y_j^* \mid \mathcal{T}(x_i)\right)}$ of each y_j^* in the target label set $\mathcal{Y}$.

$$\left\{p_{\left(y_j^* \mid \mathcal{T}(x_i)\right)}\right\}_{j=1}^{|\mathcal{Y}|} = \mathrm{M}_\theta(\mathrm{y} \mid \mathbf{x}) \propto \mathrm{M}_\theta(\mathrm{V}(\mathrm{y}) \mid \mathcal{T}(x)) \quad (1)$$

Using the model's output probabilities, which encapsulate the biases, we transform them into features that can be utilized in a kNN framework. During the prediction phase, our approach retrives the k most similar samples from the datastore $\mathcal{K} = \mathrm{kNN}\left(\mathcal{A}, \mathcal{P}\right)$ using the cosine distance metric. The definitive label y_{pred} for the input sample is then ascertained through a majority vote.

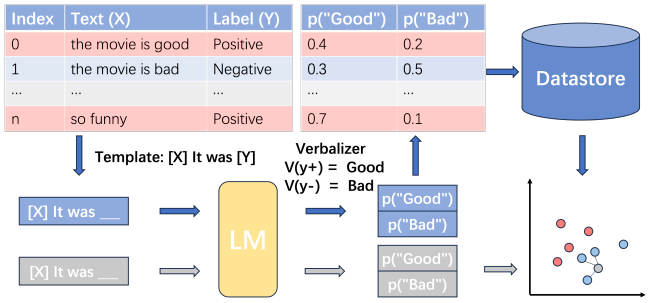

Fig. 2: The architecture of our proposed model

$$y_{\text{pred}} = \arg\max_{y_j^* \in \mathcal{Y}} \sum_{i \in \mathrm{NN}^k(\mathcal{K}, x_i)} \mathbf{1}\left(y_i = y_j^*\right) \quad (2)$$

In essence, our method harnesses the biases typically found in LLM outputs, transforming potential shortcomings into features that empower a kNN-based classification approach.

3. EXPERIMENT

3.1. Setup

3.1.1. Datasets

We evaluated our approach on six classification tasks, spanning four distinct task families and covering a wide range of data domains. For **Sentiment Classification**, we employed datasets like the Stanford Sentiment Treebank (SST-2) [12], Movie Reviews (MR) [13], and CommitmentBank (CR) [14], all of which categorize sentiments into binary classes. **Topic Classification** was addressed using the AGNews dataset [15], which classifies articles into one of four news categories. The **Subjectivity Classification** was conducted using the Subj dataset [16], differentiating sentences from movie statements into subjective or objective categories. Lastly, for **Entailment Analysis**, we harnessed the Recognizing Textual Entailment (RTE) dataset [17], a resource specifically curated for textual entailment tasks.

3.1.2. Evaluation

Our evaluation was designed to rigorously and comprehensively assess the effectiveness of our proposed method over the diverse datasets. For each dataset, we present results detailing the mean, minimum, and standard deviation of accuracy.

3.1.3. Baselines

To ascertain the efficacy of our methodology, we juxtaposed it against several benchmark techniques:

Table 1: Main results on classification tasks

Method		SST-2	MR	CR	Subj	RTE	AGNews
gpt2-medium	Zero-LM	58.4/58.4/0.0	57.4/57.4/0.0	66.7/66.7/0.0	73.4/73.4/0.0	56.7/56.7/0.0	44.5/44.5/0.0
	Raw-ICL$_{m=3}$	65.9/50.9/13.4	66.2/50.8/9.8	71.4/38.1/13.4	46.6/38.8/3.0	48.5/45.8/1.9	54.8/43.5/5.3
	bias-kNN$_{m=3}$	75.3/71.6/2.4	74.0/69.0/3.1	77.7/73.4/3.0	72.0/67.4/2.3	48.1/44.0/3.3	52.2/50.1/2.0
gpt2-large	Zero-LM	75.0/75.0/0.0	71.1/71.1/0.0	68.5/68.5/0.0	55.2/55.2/0.0	53.4/53.4/0.0	63.4/63.4/0.0
	Raw-ICL$_{m=3}$	63.3/50.9/14.7	62.3/50.0/12.1	69.9/62.0/9.6	57.1/48.5/7.8	54.9/51.3/2.0	63.1/42.1/10.0
	bias-kNN$_{m=3}$	79.5/77.8/2.7	76.8/75.0/1.7	83.9/81.0/2.1	51.8/45.0/6.0	48.9/43.7/2.4	67.3/66.0/0.9
gpt2-XL	Zero-LM	67.2/67.2/0.0	65.0/65.0/0.0	66.0/66.0/0.0	58.6/58.6/0.0	53.4/53.4/0.0	56.1/56.1/0.0
	Raw-ICL$_{m=3}$	61.9/51.0/10.1	55.7/50.0/9.1	68.7/62.2/6.1	30.8/21.9/5.9	53.2/52.7/1.1	78.5/71.3/3.2
	bias-kNN$_{m=3}$	75.9/66.2/5.7	79.2/74.5/2.3	81.7/72.9/5.0	57.6/36.9/9.7	50.6/45.8/3.5	56.6/53.6/1.9

Notes: The three digits in each cell represent the mean, min, and standard deviation of accuracy.

- **Zero-LM**: By leveraging manual prompts and verbalizers, this method gleans predictions directly from the language model (LM) without necessitating training samples. The chosen label is the one with the highest outcome.
- **Raw-ICL**: This is a straightforward implementation of In-Context Learning (ICL). It employs m demonstration samples in conjunction with a prompt and verbalizer. The label is pegged to the most pronounced outcome. Notably, for comparative purposes, the same labeled data sample used for bias-kNN serves as a demonstration.

3.1.4. Implementation Details

We utilized three GPT2 variants distinguished by their capacities: GPT2-medium(0.3B), GPT2-large(0.8B), and GPT2-XL(1.5B). For our kNN modeling, the number of training samples for each category is denoted as $\boldsymbol{m}$. Samples of sizes 2, 3, 4, 5, 6, 7, 8, 16, 32, and 64 were selected to construct our kNN datastore. Both construction and inference utilized the same model. The chosen templates and verbalizers are itemized in Table 1. To offset the potential influence of randomness, we executed five random samplings for kNN datastore construction for each m and a separate set of five for the Raw-ICL demonstration order given the specified m. For a fair assessment, identical templates and verbalizers were adopted for Zero-LM and Raw-ICL. The kNN-Prompt results were sourced directly from the relevant paper. In all cases, the number of neighbors, k, was set to 3.

Table 2: Templates and verbalizers

Dataset	Template	Verbalizer
SST2 MR CR	Review: [X] Sentiment: [Y]	Positive, Negative
AGNews	Input: [X] Type: [Y]	World, Sports, Business, Tech
Subj	Input: [X] Type: [Y]	Objective, Subjective
RTE	[X1] Hypothesis: [X2] Prediction: [Y]	True, False

3.2. Main Results

Table 1 presents a detailed comparison of our method based on the gpt2 models against current baselines over various datasets. The performance of bias-kNN improves with an increase in the value of m, displaying higher mean accuracy and smaller standard deviation, as illustrated in the subsequent figures.

In the table, we report the smallest m that yields good performance. For the gpt2-large model, it is evident that with $m = 3$, the minimum accuracy values of bias-kNN consistently surpass those of Zero-LM and Raw-ICL across all classification datasets. Moreover, the bias-kNN method demonstrates greater stability compared to the Raw-ICL method. This suggests that utilizing labeled data via bias-kNN might be preferable to ICL in certain scenarios. It is worth noting that both Raw-ICL and bias-kNN encounter challenges with the Subj and RTE tasks, corroborating findings from a previous study [9].

Different sizes of the gpt models exhibit patterns analogous to the gpt2-large model, albeit with minor variations in accuracy across datasets. Interestingly, we observe that, in some cases, smaller models outperform their larger counterparts, which could be due to the randomness of instance selection and the effects of the template and verbalizer, as shown in [4, 5].

3.3. Ablation Study and Analysis

In this section, we conduct experiments to analyze the impact of various choices related to templates, verbalizers, output features, and distance metrics of the kNN. We base these experiments on the CR dataset using GPT2-large and consine distance metric.

3.3.1. Robustness of Templates and Verbalizers

Selecting an optimal verbalizer, as corroborated by previous studies, can be a demanding task in terms of resources [3, 18, 19]. Figure 3 showcases the chosen verbalizers which are prominent token pairs from the vocabulary. Results strongly suggest that bias-kNN outdoes Zero-LM when $m > 2$. Remarkably, bias-kNN also outperforms Zero-LM for certain verbalizer choices associated with causal tokens at $m = 2$. This highlights the potential of bias-kNN in simplifying the task of selecting an effective verbalizer.

Moreover, as depicted in Figure 4, altering templates can lead to variability in the performance of bias-kNN. However,

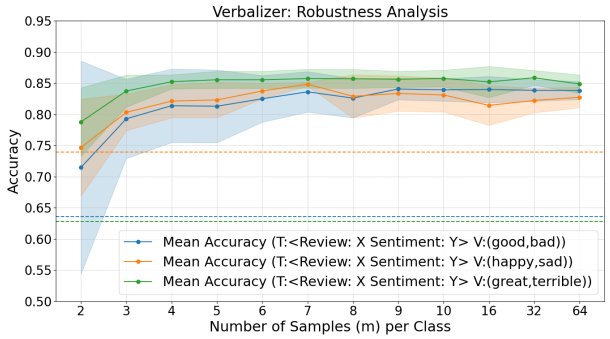

Fig. 3: Verbalizer: robustness analysis. The shaded region denotes the standard deviation. All figures are consistent. Dashed lines of the same color indicate the Zero-LM accuracy for the corresponding settings.

there's a notable enhancement in performance when $m > 2$. This further suggests that the bias-kNN approach can alleviate the challenges associated with template selection [20].

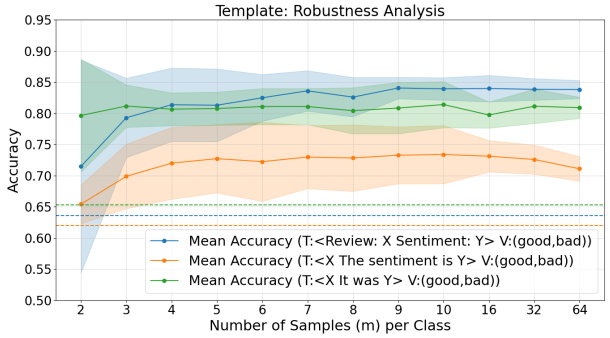

Fig. 4: Template: robustness analysis

3.3.2. *Biased Logit as a Feature*

Figure 1 illustrates the directionality of the logit. While it aligns more closely with probability, it exhibits a wider distribution. The biased logit reveals a diminished accuracy and an increased standard deviation across various m values compared to probabilities under the cosine metric.

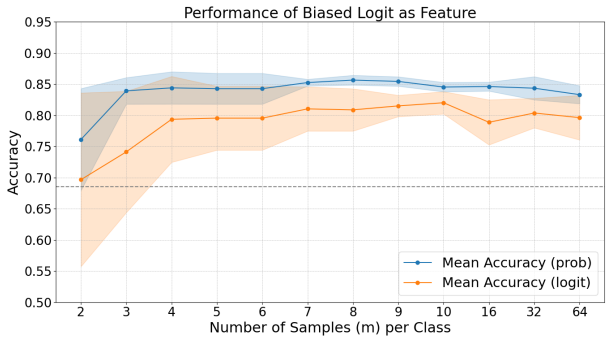

Fig. 5: Performance of biased logit as feature

3.3.3. *Impact of Distance Metrics*

We evaluated multiple distance metrics including "euclidean", "manhattan", "chebyshev", and "cosine". Among these, the "cosine" metric showcased the best performance across various setups. This might be attributed to the model's inherent bias that amplifies the likelihood of labels in line with the bias direction.

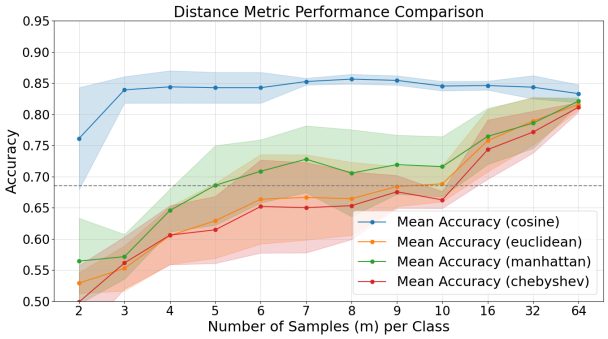

Fig. 6: Distance metric performance comparison

3.3.4. *Impact of the Number of Nearest Neighbors (k)*

It becomes evident that increasing the value of k generally does not yield improved performance when m is small. However, it demonstrates superior results when m exceeds 16.

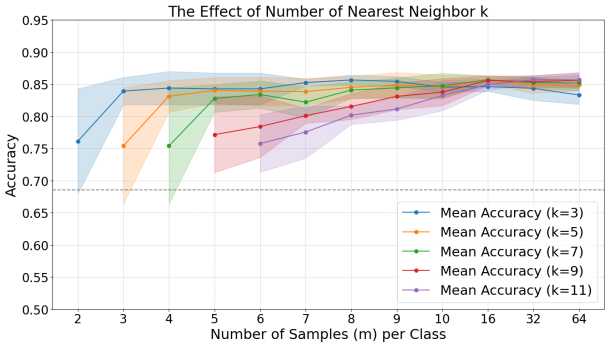

Fig. 7: The effect of number of nearest neighbor k

4. CONCLUSION

In this study, we presented the innovative "bias-kNN" approach, harnessing biases in large language models to bolster text classification. Through rigorous evaluations across diverse datasets and GPT-2 model variants, our method consistently outperformed conventional in-context learning strategies. The adaptability of "bias-kNN" was further underscored by its robust performance over a range of templates and verbalizers. Contrary to the prevailing perception of biases as solely detrimental in machine learning, our research highlights the potential advantages of strategically leveraging them in specific contexts.

5. ACKNOWLEDGEMENT

This paper is supported by the Key Research and Development Program of Guangdong Province under grant No.202 1B0101400003. The corresponding author is Ning Cheng from Ping An Technology (Shenzhen) Co., Ltd (chengning211@pingan.com.cn).

6. REFERENCES

[1] F. Petroni, T. Rocktäschel, and S. e. a. Riedel, "Language models as knowledge bases?" in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 2463–2473.

[2] T. Brown, B. Mann, N. Ryder, and S. et al., "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 1877–1901.

[3] T. Schick and H. Schütze, "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: Main Volume*, 2021, pp. 255–269.

[4] Z. Zhao, E. Wallace, S. Feng, D. Klein, and S. Singh, "Calibrate before use: Improving few-shot performance of language models," in *International Conference on Machine Learning*, 2021, pp. 12 697–12 706.

[5] A. Holtzman, P. West, V. Shwartz, Y. Choi, and L. Zettlemoyer, "Surface form competition: Why the highest probability answer isn't always right," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 7038–7051.

[6] Y. Fei, L. Cui, S. Yang, W. Lam, Z. Lan, and S. Shi, "Enhancing grammatical error correction systems with explanations," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2023, pp. 7489–7501.

[7] Z. Han, Y. Hao, L. Dong, Y. Sun, and F. Wei, "Prototypical calibration for few-shot learning of language models," in *The Eleventh International Conference on Learning Representations*, 2023.

[8] F. Nie, M. Chen, Z. Zhang, and X. Cheng, "Improving few-shot performance of language models via nearest neighbor calibration," *arXiv preprint arXiv:2212.02216*, 2022.

[9] B. Xu, Q. Wang, Z. Mao, Y. Lyu, Q. She, and Y. Zhang, "kNN prompting: Beyond-context learning with calibration-free nearest neighbor inference," in *The Eleventh International Conference on Learning Representations*, 2023.

[10] U. Khandelwal, O. Levy, D. Jurafsky, L. Zettlemoyer, and M. Lewis, "Generalization through memorization: Nearest neighbor language models," in *International Conference on Learning Representations*, 2019.

[11] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[12] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1631–1642.

[13] B. Pang and L. Lee, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, 2005, pp. 115–124.

[14] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004, pp. 168–177.

[15] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[16] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, 2004, pp. 271–278.

[17] I. Dagan, O. Glickman, and B. Magnini, "The pascal recognising textual entailment challenge," in *Machine learning challenges workshop*, 2005, pp. 177–190.

[18] T. Schick and H. Schütze, "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*, 2021, pp. 2339–2352.

[19] S. Hu, N. Ding, H. Wang, Z. Liu, J. Wang, J. Li, W. Wu, and M. Sun, "Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2022, pp. 2225–2240.

[20] T. Shin, Y. Razeghi, R. L. Logan IV, E. Wallace, and S. Singh, "Autoprompt: Eliciting knowledge from language models with automatically generated prompts," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 4222–4235.