Uncertainty Quantification with Pre-trained Language Models: A Large-Scale Empirical Analysis

Yuxin Xiao[1], **Paul Pu Liang**[2], **Umang Bhatt**[3],
Willie Neiswanger[4], **Ruslan Salakhutdinov**[2], **Louis-Philippe Morency**[2]
[1]Massachusetts Institute of Technology, [2]Carnegie Mellon University,
[3]University of Cambridge, [4]Stanford University
[1]yuxin102@mit.edu [2]{pliang,rsalakhu,morency}@cs.cmu.edu
[3]usb20@cam.ac.uk [4]neiswanger@cs.stanford.edu

Abstract

Pre-trained language models (PLMs) have gained increasing popularity due to their compelling prediction performance in diverse natural language processing (NLP) tasks. When formulating a PLM-based prediction pipeline for NLP tasks, it is also crucial for the pipeline to minimize the *calibration error*, especially in safety-critical applications. That is, the pipeline should reliably indicate when we can trust its predictions. In particular, there are various considerations behind the pipeline: (1) the choice and (2) the size of PLM, (3) the choice of uncertainty quantifier, (4) the choice of fine-tuning loss, and many more. Although prior work has looked into some of these considerations, they usually draw conclusions based on a limited scope of empirical studies. There still lacks a holistic analysis on how to compose a well-calibrated PLM-based prediction pipeline. To fill this void, we compare a wide range of popular options for each consideration based on three prevalent NLP classification tasks and the setting of domain shift. In response, we recommend the following: (1) use ELECTRA for PLM encoding, (2) use larger PLMs if possible, (3) use Temp Scaling as the uncertainty quantifier, and (4) use Focal Loss for fine-tuning.

1 Introduction

PLMs (Qiu et al., 2020; Min et al., 2021) have achieved state-of-the-art performance on a broad spectrum of NLP benchmarks (Rajpurkar et al., 2016, 2018; Wang et al., 2019a,b) and are increasingly popular in various downstream applications such as question answering (Yoon et al., 2019; Garg et al., 2020), text classification (Arslan et al., 2021; Limsopatham, 2021), and relation extraction (Zhou et al., 2021; Xiao et al., 2022). Consequently, it is paramount for PLMs to faithfully communicate when to (or not to) rely on their predictions for decision-making, especially in high-stakes scenarios. In these cases, we need PLMs to quantify their uncertainty accurately and calibrate well (Abdar et al., 2021), meaning that their predictive confidence should be a valid estimate of how likely they are to make a correct prediction. Consider an example of medical question answering (Yoon et al., 2019; Zhang et al., 2021) where a PLM is asked to assist doctors when diagnosing diseases. If the PLM is 90% sure that a patient is healthy, the predicted outcome should occur 90% of the time in practice. Otherwise, it may adversely affect doctors' judgment and lead to catastrophic consequences. Hence, since PLMs have become the de facto paradigm for many NLP tasks, it is necessary to assess their calibration quality.

When constructing a well-calibrated PLM-based prediction pipeline for NLP tasks, various considerations are involved. To name a few:

1. Due to the use of diverse pre-training datasets and strategies, different PLMs may behave differently regarding calibration.
2. The model size of PLMs may also affect their capability in calibration.
3. Leveraging uncertainty quantifiers (e.g., Temp Scaling (Guo et al., 2017) and MC Dropout (Gal and Ghahramani, 2016)) alongside PLMs in the pipeline may reduce calibration error.
4. Some losses (e.g., Focal Loss (Mukhoti et al., 2020) and Label Smoothing (Müller et al., 2019)) may fine-tune PLMs to calibrate better.

Although some of these considerations have been studied before, the ideal choice for each consideration remains obscure. On the one hand, Desai and Durrett (2020) report unconventional calibration behavior for PLMs, which casts doubts on the prior beliefs drawn on traditional neural networks by Guo et al. (2017). On the other hand, existing work (Desai and Durrett, 2020; Dan and Roth, 2021) on PLMs' empirical calibration performance often looks at a single consideration and concludes by comparing only one or two types of PLMs.

Therefore, in this paper, we present a comprehensive analysis of the four pivotal considerations

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 7302 - 7313

introduced above via large-scale empirical evaluations. To ensure that our analysis is applicable to various NLP tasks and resilient to domain shift, we set up three NLP tasks (i.e., Sentiment Analysis, Natural Language Inference, and Commonsense Reasoning) and prepare both in-domain and out-of-domain testing sets for each task. In addition to the explicit metrics of prediction and calibration error, we also utilize two evaluation tasks to examine calibration qualities implicitly. Selective prediction lowers prediction error by avoiding uncertain testing points, and out-of-domain detection checks if a pipeline is less confident on unseen domains. By comparing four to five options for each consideration, we recommend the following:

1. Use ELECTRA (Clark et al., 2020) as the PLM to encode input text sequences.
2. Use the larger version of a PLM if possible.
3. Use Temp Scaling (Guo et al., 2017) for post hoc uncertainty recalibration.
4. Use Focal Loss (Mukhoti et al., 2020) during the fine-tuning stage.

Compared to prior work, our extensive empirical evaluations also reveal the following novel observations that are unique to PLM-based pipelines:

- The calibration quality of PLMs is relatively consistent across tasks and domains, except XLNet (Yang et al., 2019) being the most vulnerable to domain shift.
- In contrast to other NLP tasks, larger PLMs are better calibrated in-domain in Commonsense Reasoning.
- Uncertainty quantifiers (e.g., Temp Scaling) are generally more effective in improving calibration out-of-domain.
- Ensemble (Lakshminarayanan et al., 2017) is less effective in PLM-based pipelines.

To encourage future work towards better uncertainty quantification in NLP, we release our code and large-scale evaluation benchmarks containing 120 PLM-based pipelines based on four metrics (prediction and calibration error, selective prediction, and out-of-domain detection). These pipelines consist of distinct choices concerning the four considerations and are tested on all three NLP tasks under both in- and out-of-domain settings.[1]

[1]Our data and code are available at https://github.com/xiaoyuxin1002/UQ-PLM.git.

2 Background

2.1 Problem Formulation

Datasets. In this work, we focus on utilizing PLMs for NLP classification tasks. More specifically, consider such a task where the training set $\mathbb{D}_{\text{train}} = \{(x_i, y_i)\}_{i=1}^{N_{\text{train}}}$ consists of pairs of a text sequence $x_i \in \mathcal{X}_{\text{in}}$ and an associated label $y_i \in \mathcal{Y}$. Similarly, the validation set $\mathbb{D}_{\text{val}}$ and the in-domain testing set $\mathbb{D}_{\text{in}}$ come from the same domain $\mathcal{X}_{\text{in}}$ and share the same label space $\mathcal{Y}$. We also prepare an out-of-domain testing set $\mathbb{D}_{\text{out}}$, which differs from the others by coming from a distinct domain $\mathcal{X}_{\text{out}}$.

PLM-based Pipeline. We apply a PLM M to encode an input text sequence x_i and feed the encoding vector to a classifier F, which outputs a predictive distribution $\mathbf{u}_i$ over the label space $\mathcal{Y}$ via the softmax operation. Here, parameters in M and F are fine-tuned by minimizing a loss function ℓ on $\mathbb{D}_{\text{train}}$. It is optional to modify the distribution $\mathbf{u}_i$ post hoc by an uncertainty quantifier Q to reduce calibration error. We define the predicted label as $\hat{y}_i = \arg\max_{j \in \{1,\dots,|\mathcal{Y}|\}} \mathbf{u}_{ij}$ with the corresponding confidence $\hat{c}_i = \mathbf{u}_{i\hat{y}_i}$.

Calibration. One crucial goal of uncertainty quantification is to improve calibration. That is, the predicted confidence should match the empirical likelihood: $P(y_i = \hat{y}_i \mid \hat{c}_i) = \hat{c}_i$. We follow Guo et al. (2017) by using the expected calibration error (ECE) to assess the calibration performance. The calculation of ECE is described in Section 3.1. To reduce ECE, our main experimental evaluation lies in examining four considerations involved in a PLM-based pipeline: (1) the choice of PLM M (Section 3), (2) the size of PLM M (Section 4), (3) the choice of uncertainty quantifier Q (Section 5), and (4) the choice of loss function ℓ (Section 6).

2.2 Related Work

Uncertainty quantification has drawn long-lasting attention from various domains (Bhatt et al., 2021), such as weather forecasting (Brier et al., 1950; Raftery et al., 2005), medical practice (Yang and Thompson, 2010; Jiang et al., 2012), and machine translation (Ott et al., 2018; Zhou et al., 2020; Wei et al., 2020). Researchers have approached this question from both Bayesian (Kendall and Gal, 2017; Depeweg et al., 2018) and frequentist perspectives (Alaa and Van Der Schaar, 2020a,b). They have also proposed different techniques to improve uncertainty calibration for classification (Kong et al., 2020; Krishnan and Tickoo, 2020) and

regression (Kuleshov et al., 2018; Cui et al., 2020; Chung et al., 2021) tasks. Recent work has investigated connections between uncertainty and other properties, such as model interpretability (Antoran et al., 2021; Ley et al., 2022), selective prediction (Xin et al., 2021; Varshney et al., 2022a,b), and out-of-domain generalization (Wald et al., 2021; Qin et al., 2021).

PLMs (Qiu et al., 2020; Min et al., 2021) have achieved state-of-the-art prediction performance on diverse NLP benchmarks (Rajpurkar et al., 2016, 2018; Wang et al., 2019a,b) and demonstrated many desired properties like stronger out-of-domain robustness (Hendrycks et al., 2020) and better uncertainty calibration (Desai and Durrett, 2020). They typically leverage a Transformer architecture (Vaswani et al., 2017) and are pre-trained by self-supervised learning (Jaiswal et al., 2021).

Although Guo et al. (2017) report that larger models tend to calibrate worse, PLMs have been shown to produce well-calibrated uncertainty in practice (Desai and Durrett, 2020), albeit for giant model sizes. Their unusual calibration behavior puts the observations drawn on traditional neural networks (Ovadia et al., 2019; Mukhoti et al., 2020) or pre-trained vision models (Minderer et al., 2021) in doubt. Prior work (Desai and Durrett, 2020; Dan and Roth, 2021) on the calibration of PLMs often explores only one or two types of PLMs and ignores uncertainty quantifiers and fine-tuning losses beyond Temp Scaling and Cross Entropy, respectively. As a result, there lacks a holistic analysis that explores the full set of these considerations in a PLM-based pipeline. Therefore, our paper aspires to fill this void via extensive empirical studies.

3 Which Pre-trained Language Model?

3.1 Experiment Setup

To evaluate the calibration performance of PLMs, we consider a series of NLP classification tasks:

1. **Sentiment Analysis** identifies the binary sentiment of a text sequence. We treat the IMDb *movie review* dataset (Maas et al., 2011) as in-domain and the Yelp *restaurant review* dataset (Zhang et al., 2015) as out-of-domain.
2. **Natural Language Inference** predicts the relationship between a hypothesis and a premise. We regard the Multi-Genre Natural Language Inference (MNLI) dataset (Williams et al., 2018) covering a range of genres of *spoken and written text* as in-domain and the Stanford Natural Language Inference (SNLI) dataset (Bowman et al., 2015) derived from *image captions* only as out-of-domain.
3. **Commonsense Reasoning** determines the most reasonable continuation of a sentence among four candidates. We view the Situations With Adversarial Generations (SWAG) dataset (Zellers et al., 2018) as in-domain and its *adversarial* variant (HellaSWAG) (Zellers et al., 2019) as out-of-domain.

	Sentiment Analysis	Natural Language Inference	Commonsense Reasoning
$\mathcal{X}_{\text{in}}$	IMDb	MNLI	SWAG
$\mathcal{X}_{\text{out}}$	Yelp	SNLI	HellaSWAG
$\lvert\mathcal{Y}\rvert$	2	3	4
$\lvert\mathbb{D}_{\text{train}}\rvert$	25,000	392,702	73,546
$\lvert\mathbb{D}_{\text{val}}\rvert$	12,500	4,907	10,003
$\lvert\mathbb{D}_{\text{in}}\rvert$	12,500	4,908	10,003
$\lvert\mathbb{D}_{\text{out}}\rvert$	19,000	4,923	5,021

Table 1: In- and out-of-domain datasets, label space size, and each data split size of the three NLP tasks.

Hugging Face Name	Model Size	Pre-training Corpus Size	Pre-training Task
bert-base-cased	109M	16G	Masked LM, NSP
xlnet-base-cased	110M	161G	Permuted LM
electra-base-discriminator	110M	161G	Replacement Detection
roberta-base	125M	161G	Dynamic Masked LM
deberta-base	140M	85G	Dynamic Masked LM
bert-large-cased	335M	16G	Masked LM, NSP
xlnet-large-cased	340M	161G	Permuted LM
electra-large-discriminator	335M	161G	Replacement Detection
roberta-large	335M	161G	Dynamic Masked LM
deberta-large	350M	85G	Dynamic Masked LM

Table 2: Model size, pre-training corpus size, and pre-training task of the five PLMs, separated into the base (upper) and the large (lower) versions.

For each task, we construct $\mathbb{D}_{\text{train}}$, $\mathbb{D}_{\text{val}}$, and $\mathbb{D}_{\text{in}}$ from the corresponding in-domain dataset, and $\mathbb{D}_{\text{out}}$ from the corresponding out-of-domain dataset. The original validation set of each dataset is split in half randomly to form a held-out non-blind testing set (i.e., $\mathbb{D}_{\text{in}}$ or $\mathbb{D}_{\text{out}}$). Table 1 describes the task details.

To understand which PLM delivers the lowest calibration error, we examine five popular options:

1. **BERT** (Devlin et al., 2019) utilizes a bidirectional Transformer architecture pre-trained by masked language modeling (LM) and next sentence prediction (NSP).
2. **XLNet** (Yang et al., 2019) proposes a two-stream self-attention mechanism and a pre-training objective of permuted LM.
3. **ELECTRA** (Clark et al., 2020) pre-trains a

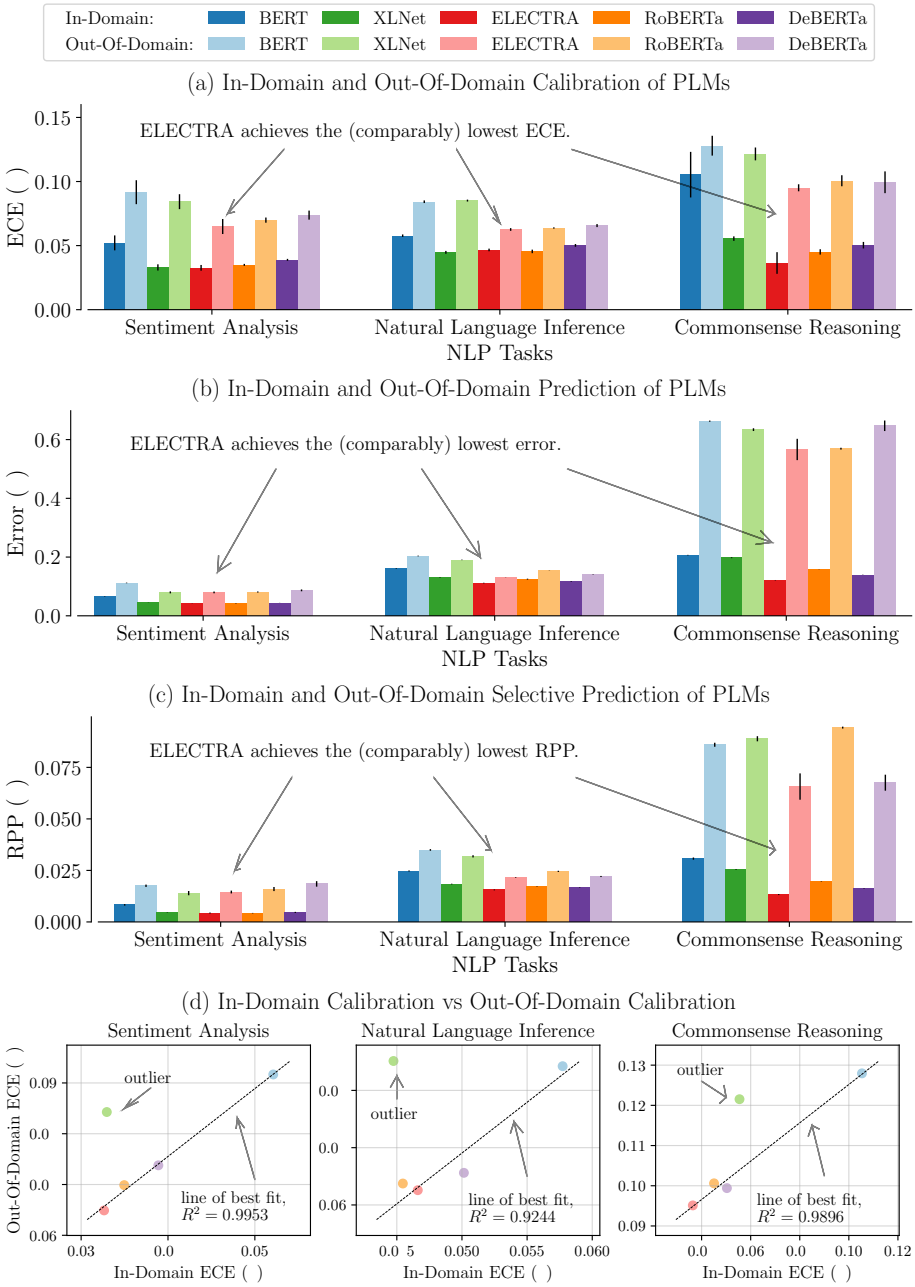

Figure 1: Calibration and (selective) prediction performance of five PLMs in three NLP tasks under two domain settings. The calibration quality of the five PLMs is relatively consistent across tasks and domains, while XLNet is the least robust to domain shift. ELECTRA stands out due to its lowest scores in ECE, prediction error, and RPP.

discriminative model to detect tokens replaced by a generative model.

4. **RoBERTa** (Liu et al., 2019) builds on BERT by pre-training based on dynamic masked LM only and tuning key hyperparameters.
5. **DeBERTa** (He et al., 2020) further improves RoBERTa via a disentangled attention mechanism and an enhanced mask decoder.

We use the base version of each PLM, which has a similar model size and is initialized from the corresponding Hugging Face (Wolf et al., 2020) pre-trained checkpoint. Table 2 details these PLMs. After receiving the encoding vector of the classification token `[CLS]` for an input text sequence from the PLM, we pass it through a classifier to obtain a predictive distribution. Regarding the classifier configuration, we follow the default practice in Hugging Face by utilizing a two-layer neural network with tanh non-linear activation.

The learning rate for each model-dataset combination is tuned based on the validation set among $\{5e{-}6, 1e{-}5, 2e{-}5, 5e{-}5\}$. We leverage AdamW

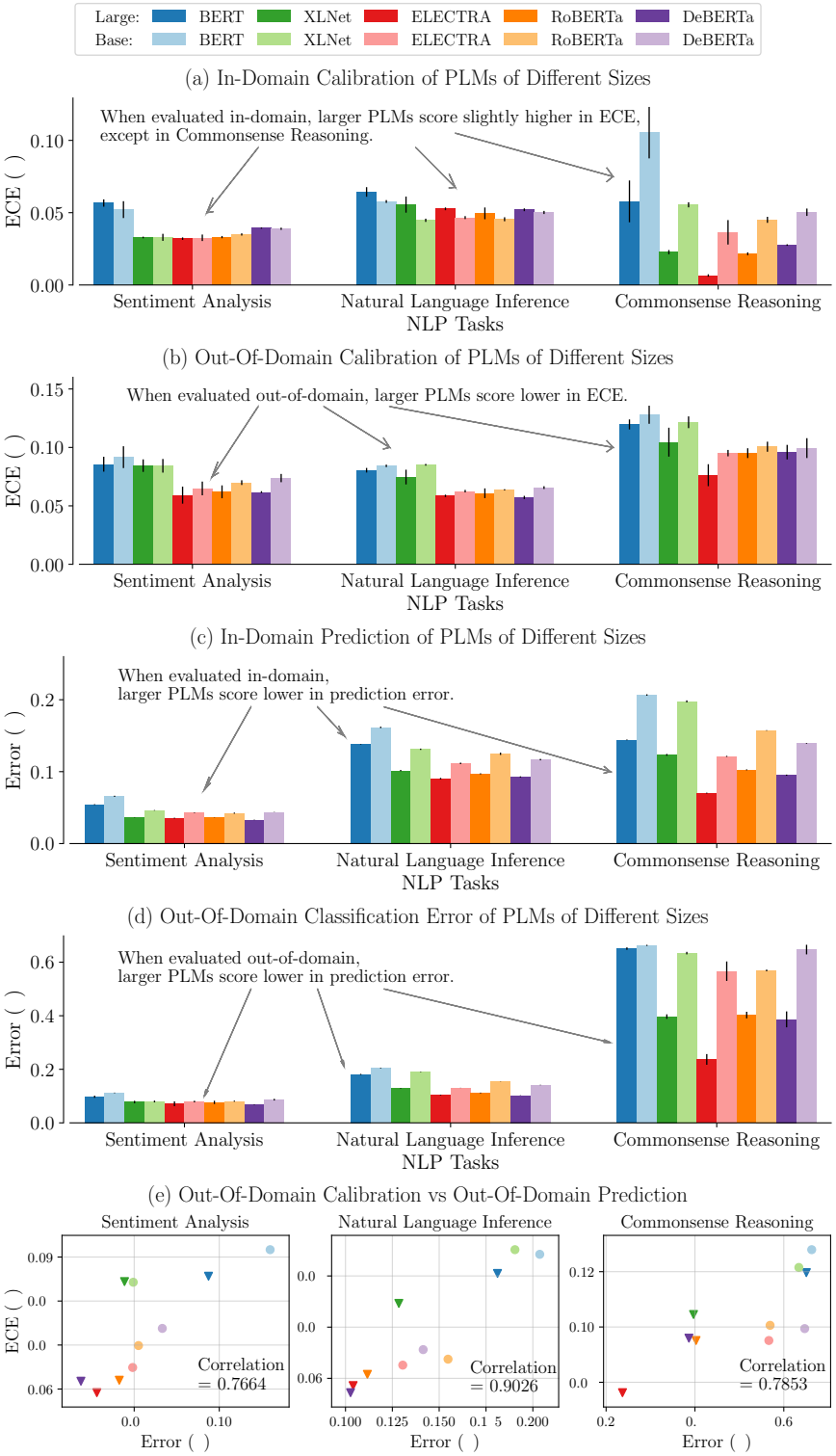

Figure 2: Calibration and prediction performance of large and base PLMs in three NLP tasks under two domain settings. Larger PLMs calibrate better than their respective base versions when evaluated out-of-domain, while calibrating slightly worse in-domain with one exception in Commonsense Reasoning. If the computational budget permits, larger PLMs constitute more powerful pipelines given their lower out-of-domain ECE along with lower prediction error. We also observe a positive correlation between calibration and prediction error out-of-domain.

(Loshchilov and Hutter, 2018) to minimize the cross-entropy loss on $\mathbb{D}_{\text{train}}$ for five epochs with early stopping and a linearly decaying scheduler (Goyal et al., 2017) whose warm-up ratio $= 10\%$. Batch size is 16, and the model gradients are clipped to a maximum norm of 1. We perform

our experiments on a Tesla A6000 GPU and report the mean and one standard error by conducting six trials with different seeds.

To explicitly evaluate calibration performance by ECE, we first stratify N predictions into K bins of equal width based on the sorted confidence values. Then ECE is a weighted average of the absolute difference between the accuracy and confidence of each bin: $\text{ECE} = \sum_{k=1}^{K} \frac{|B_k|}{N} |\text{acc}(B_k) - \text{conf}(B_k)|$, where $\text{acc}(B_k)$ and $\text{conf}(B_k)$ are the average accuracy and confidence of predictions in bin B_k, respectively. We set $K = 10$ in our experiments.

To implicitly assess calibration quality based on selective prediction, we deploy the metric of reversed pair proportion (RPP) (Xin et al., 2021). More specifically, for a dataset of size N, $\text{RPP} = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \mathbb{1}[\hat{c}_i < \hat{c}_j, y_i = \hat{y}_i, y_j \neq \hat{y}_j]$. It measures the proportion of prediction pairs with a reversed confidence-error relationship. A lower RPP indicates that the pipeline is more confident on correct predictions.

3.2 Empirical Findings

As shown in Figure 1(a), the calibration performance of all five PLMs deteriorates from in-domain to out-of-domain. This phenomenon coincides with the finding made by Ovadia et al. (2019) on traditional neural networks. In addition, **the ranking among the five PLMs based on ECE is generally consistent,** which implies that their calibration quality is transferable across tasks and domains. More specifically, for all three tasks under the in-domain setting, XLNet, ELECTRA, RoBERTa, and DeBERTa outperform BERT in terms of lower ECE, suggesting that a larger pre-training corpus may improve the calibration quality (see Table 2). **When moving to the out-of-domain setting, XLNet sees the largest increase in ECE,** which makes it an outlier in Figure 1(d). This observation may indicate that the pre-training task of permuted LM is vulnerable to domain shift.

ELECTRA stands out among the five examined PLMs in encoding input text sequences. Not only does it achieve the (comparably) lowest ECE in all three tasks under both in- and out-of-domain settings, it also delivers the lowest prediction error in Figure 1(b) and the lowest RPP for selective prediction in Figure 1(c). We hypothesize its success to the unique pre-training paradigm of replaced token detection, which preserves the token distribution by avoiding the artificial `[MASK]` tokens in masked LM and enhances the computational efficiency by learning from all input tokens.

4 What Model Size?

4.1 Experiment Setup

To investigate how the size of PLMs affects the calibration performance, we compare the large versions of the five PLMs mentioned in Section 3.1 against their respective base versions. We keep the rest of the setup the same as in Section 3.1.

4.2 Empirical Findings

Figures 2(a) and (b) demonstrate that larger PLMs tend to produce a slightly higher ECE compared to their respective base versions when evaluated in-domain, while calibrating better out-of-domain. This observation based on five PLMs verifies the conclusion made by Dan and Roth (2021) solely based on BERT. However, **there is a notable exception that larger PLMs are significantly better calibrated in-domain in Commonsense Reasoning than their respective base versions,** which implies that larger PLMs are more aware of their uncertainties during the reasoning process.

Larger PLMs constitute more powerful PLM-based pipelines, if computational budget permits. Although sometimes they suffer slightly in in-domain calibration compared to their smaller counterparts, larger PLMs achieve a lower ECE out-of-domain. They also deliver lower in- and out-of-domain prediction errors in Figures 2(c) and (d), respectively. In addition, we observe a positive correlation between calibration and prediction errors under the out-of-domain setting in Figure 2(e), suggesting that pipelines calibrating well out-of-domain are more accurate under domain shift as well. This reflects the finding in Wald et al. (2021) that multi-domain calibration leads to better out-of-domain prediction performance.

5 Which Uncertainty Quantifier?

5.1 Experiment Setup

As discussed in Section 2.1, we can further adjust the vanilla predictive distribution post hoc via an uncertainty quantifier. Therefore, we study four uncertainty quantifiers based on the setup in Section 3.1 to inspect which improve the calibration performance in our problem formulation:

1. **Temp Scaling** (Guo et al., 2017) learns a scalar parameter T_{temp} based on $\mathbb{D}_{\text{val}}$ and "soft-

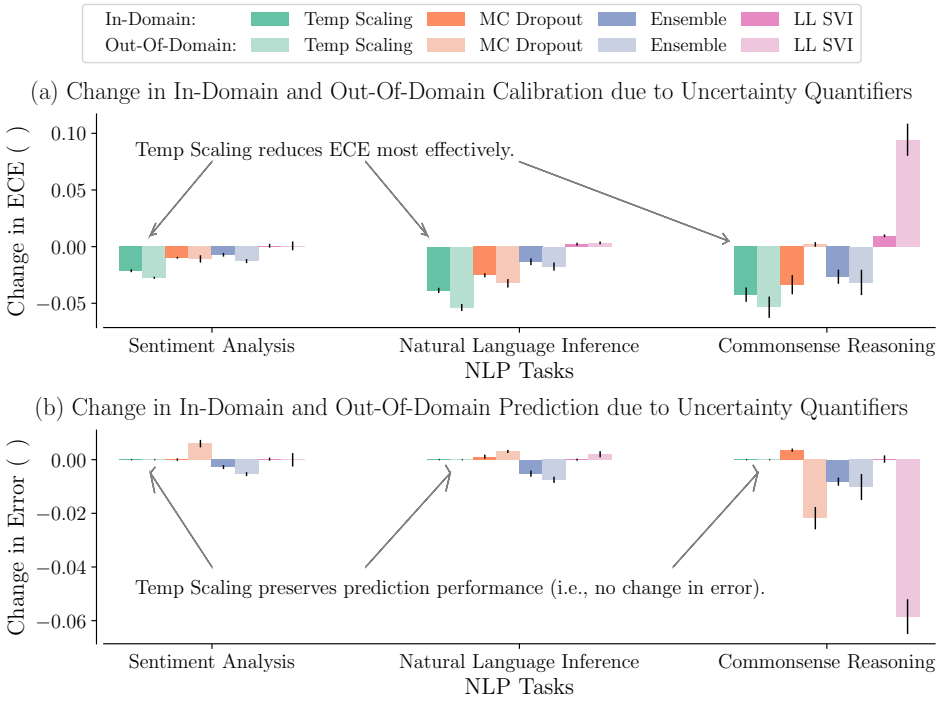

Figure 3: Change in calibration and prediction performance due to the use of four uncertainty quantifiers. The effectiveness of these quantifiers in reducing ECE follows the descending order of Temp Scaling, MC Dropout, Ensemble, and LL SVI. The drop in ECE is more significant out-of-domain. Temp Scaling is the most compelling fine-tuning loss due to its largest reduction in ECE, preservation of prediction results, and little computational cost.

ens" the vanilla logit output with T_{temp} to obtain a new predictive distribution.

2. **MC Dropout** (Gal and Ghahramani, 2016) approximates the expectation of a posterior predictive distribution by averaging T_{mc} forward passes with dropout turned on.
3. **Ensemble** (Lakshminarayanan et al., 2017) averages the predictive distributions of T_{en} independently trained models.
4. **LL SVI** (Last-Layer Stochastic Variational Inference) (Blundell et al., 2015) implements variational layers with reparameterized Monte Carlo estimators based on the Bayesian-Torch package (Krishnan et al., 2022). It approximates the expectation of a posterior predictive distribution by averaging T_{svi} forward passes through the Bayesian classification layers.

Here, we follow Lakshminarayanan et al. (2017) by setting $T_{\text{en}} = 5$. We use $T_{\text{mc}} = 10$ and $T_{\text{svi}} = 50$ due to computational constraints during inference. The dropout rate in MC Dropout is the same as the default dropout rate of each PLM.

5.2 Empirical Findings

In Figure 3, we plot the change in calibration and prediction performance due to the use of uncertainty quantifiers compared to the vanilla results in Section 4.1. **The improvement in calibration is more significant out-of-domain.** More specifically, **the degree to which these quantifiers decrease ECE follows the descending order of Temp Scaling, MC Dropout, Ensemble, and LL SVI.** In fact, LL SVI even hurts the calibration in terms of an increase in ECE, suggesting that variational classifiers with reparameterized Monte Carlo estimators cannot capture uncertainties well when used only at the fine-tuning stage. Unlike Ovadia et al. (2019), **we find Ensemble less effective in PLM-based pipelines,** possibly because individual learners in Ensemble are initialized from the same pre-trained model checkpoint and, consequently, the strong correlation among them limits the power of Ensemble (Liu and Yao, 1999).

Meanwhile, Temp Scaling preserves prediction results, and Ensemble lowers prediction error, as expected. Although MC Dropout and LL SVI reduce the prediction error out-of-domain in Commonsense Reasoning by producing sharper predictive distributions, they usually end up being overconfident, which leads to the rise in ECE in Figure 3(a).

Temp Scaling is the most appropriate uncertainty quantifier for PLM-based pipelines. Com-

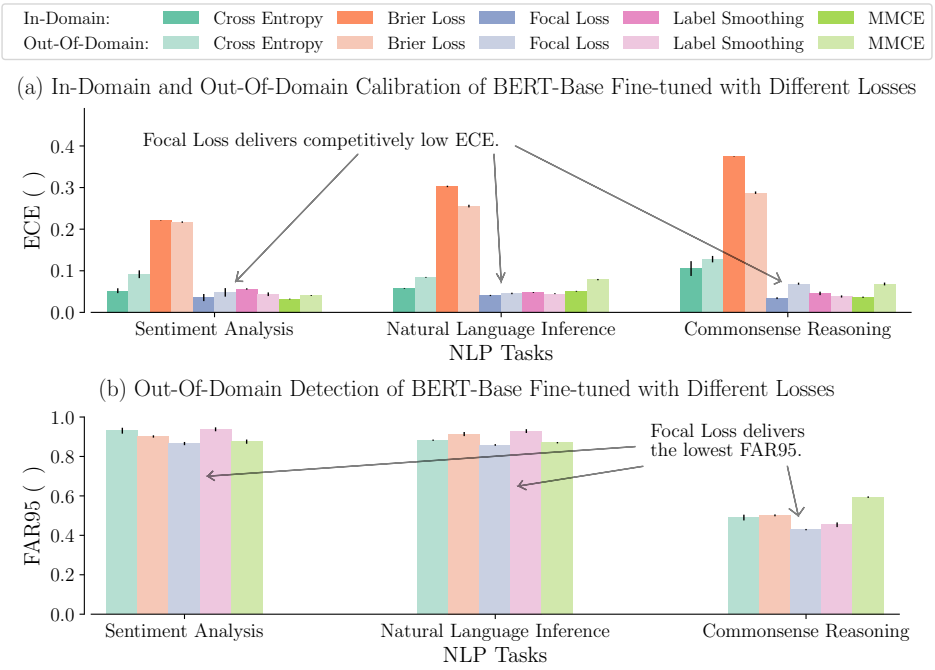

Figure 4: Calibration and out-of-domain detection performance of BERT base models fine-tuned by five losses. Focal Loss, Label Smoothing, and MMCE are more capable of fine-tuning well-calibrated models compared to Cross Entropy and Brier Loss. Focal Loss is the best option due to its competitively low ECE and FAR95.

pared to LL SVI, Temp Scaling diminishes ECE and maintains the competitive prediction quality of PLMs. Moreover, the post hoc recalibration manner of Temp Scaling adds little to the computational burden. In contrast, Ensemble or MC Dropout significantly increases the computational cost during fine-tuning or inference, respectively. Note that this distinction is of great importance given the enormous computational burdens of PLMs.

6 Which Fine-tuning Loss?

6.1 Experiment Setup

Besides cross-entropy loss, we consider four other losses when fine-tuning a BERT base model and compare their calibration performance based on the setup in Section 3.1.

1. **Cross Entropy** (Good, 1952) is the negative log likelihood of ground-truth classes.
2. **Brier Loss** (Brier et al., 1950) is the squared difference between predictive distributions and one-hot ground-truth vectors.
3. **Focal Loss** (Mukhoti et al., 2020) applies a modulating term to cross-entropy loss to focus model learning on hard misclassified samples.
4. **Label Smoothing** (Müller et al., 2019) produces targeting distributions by allocating probability mass to non-ground-truth classes.
5. **MMCE** (Maximum Mean Calibration Error) (Kumar et al., 2018) is a differentiable proxy to regularize calibration error, usually used alongside cross-entropy loss.

We use a smoothing factor of 0.1, and follow the practice in Mukhoti et al. (2020) by setting the focal hyperparameter to 5 when the predictive probability for the ground-truth class $\in [0, 0.2)$ and to 3 when the probability $\in [0.2, 1]$.

In addition, we leverage out-of-domain detection to implicitly examine the quality of uncertainty quantification. We want models to be less confident on $\mathbb{D}_{\text{out}}$ than on $\mathbb{D}_{\text{in}}$ and, hence, report the false alarm rate at 95% recall (FAR95) (Hendrycks et al., 2020). This metric tells the ratio of samples in $\mathbb{D}_{\text{in}}$ whose confidence is lower than the 95th percentile of samples in $\mathbb{D}_{\text{out}}$.

6.2 Empirical Findings

As shown in Figure 4(a), **Label Smoothing, Focal Loss, and MMCE generate better-calibrated BERT base models compared to Cross Entropy and Brier Loss.** While models fine-tuned by Cross Entropy, Focal Loss, or MMCE calibrate better in-domain, Brier Loss and Label Smoothing enjoy a decrease in ECE when evaluated out-of-domain. This observation matches the findings in Desai and Durrett (2020); Dan and Roth (2021) and is in-

tuitive for Label Smoothing since it deliberately alleviates overconfidence during fine-tuning.

Focal Loss is the most compelling fine-tuning loss for PLM-based pipelines. Among the five examined options, Focal Loss delivers competitively low ECE, both in- and out-of-domain for all three tasks. Moreover, it scores the lowest in FAR95, as illustrated in Figure 4(b), meaning that models fine-tuned by Focal Loss are most alert to domain shift. We note that FAR95 scores are relatively high in Sentiment Analysis and Natural Language Inference, probably because these pipelines also predict well out-of-domain in Figure 2(d).

7 Conclusion

In this paper, we contribute a comprehensive analysis on how to reduce calibration error in a PLM-based pipeline. We establish four key considerations behind the pipeline and compare a broad range of prevalent options for each consideration. Our empirical evaluations consist of three distinct NLP classification tasks and two different domain settings. Based on our large-scale systematic analysis, we recommend the following:

1. Use ELECTRA for PLM encoding.
2. Use larger PLMs if possible.
3. Use Temp Scaling for post hoc recalibration.
4. Use Focal Loss during the fine-tuning stage.

Compared to existing work, we also observe the following novel phenomena that are unique to PLM-based pipelines:

- The relative calibration quality of PLMs is consistent in general across tasks and domains, with an exception of XLNet, which is the least robust to domain shift.
- Larger PLMs are better calibrated under the in-domain setting in Commonsense Reasoning, unlike in the other NLP tasks.
- Uncertainty quantifiers are generally more effective in improving calibration performance under the out-of-domain setting.
- Ensemble is less effective in reducing calibration error when used with PLM-based pipelines, despite their convincing performance with traditional models.

8 Limitation

Due to computational constraints, we are unable to pre-train PLMs from scratch with other combinations of pre-training corpora and tasks. Consequently, while our analysis is applicable to existing widely-used PLMs, we do not claim its generalization to new combinations of pre-training corpora and tasks. We believe that this does not invalidate our claims which are primarily targeted toward real-world practitioners using existing PLMs. It is possible that techniques catering to the special needs of PLM-based pipelines (Kong et al., 2020) can mitigate calibration error further.

Moreover, although our setup involves domain shift, we do not focus on inspecting how the degree of domain shift affects the calibration performance of PLM-based pipelines. It is also interesting to consider how to construct a well-calibrated PLM-based pipeline for other types of NLP tasks such as cross-lingual text classification and generation, which we leave to future work.

Acknowledgements

This material is based upon work partially supported by National Science Foundation (Awards #1722822 and #1750439) and National Institutes of Health (Awards #R01MH125740, #R01MH096951, and #U01MH116925). PPL is partially supported by a Facebook PhD Fellowship and a Carnegie Mellon University's Center for Machine Learning and Health Fellowship. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the sponsors, and no official endorsement should be inferred.

References

Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U Rajendra Acharya, et al. 2021. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*.

Ahmed Alaa and Mihaela Van Der Schaar. 2020a. Discriminative jackknife: Quantifying uncertainty in deep learning via higher-order influence functions. In *ICML*.

Ahmed Alaa and Mihaela Van Der Schaar. 2020b. Frequentist uncertainty in recurrent neural networks via blockwise influence functions. In *ICML*.

Javier Antoran, Umang Bhatt, Tameem Adel, Adrian Weller, and José Miguel Hernández-Lobato. 2021. Getting a CLUE: A method for explaining uncertainty estimates. In *ICLR*.

Yusuf Arslan, Kevin Allix, Lisa Veiber, Cedric Lothritz, Tegawendé F Bissyandé, Jacques Klein, and Anne Goujon. 2021. A comparison of pre-trained language models for multi-class text classification in the financial domain. In *WWW Companion*.

Umang Bhatt, Javier Antorán, Yunfeng Zhang, Q Vera Liao, Prasanna Sattigeri, Riccardo Fogliato, Gabrielle Melançon, Ranganath Krishnan, Jason Stanley, Omesh Tickoo, et al. 2021. Uncertainty as a form of transparency: Measuring, communicating, and using uncertainty. In *AIES*.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. 2015. Weight uncertainty in neural network. In *ICML*.

Samuel Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *EMNLP*.

Glenn W Brier et al. 1950. Verification of forecasts expressed in terms of probability. *Monthly weather review*.

Youngseog Chung, Willie Neiswanger, Ian Char, and Jeff Schneider. 2021. Beyond pinball loss: Quantile methods for calibrated uncertainty quantification. In *NeurIPS*.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

Peng Cui, Wenbo Hu, and Jun Zhu. 2020. Calibrated reliable regression using maximum mean discrepancy. In *NeurIPS*.

Soham Dan and Dan Roth. 2021. On the effects of transformer size on in-and out-of-domain calibration. In *EMNLP (Findings)*.

Stefan Depeweg, Jose-Miguel Hernandez-Lobato, Finale Doshi-Velez, and Steffen Udluft. 2018. Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. In *ICML*.

Shrey Desai and Greg Durrett. 2020. Calibration of pre-trained transformers. In *EMNLP*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *ICML*.

Siddhant Garg, Thuy Vu, and Alessandro Moschitti. 2020. Tanda: Transfer and adapt pre-trained transformer models for answer sentence selection. In *AAAI*.

I.J. Good. 1952. Rational decisions. *Journal of the Royal Statistical Society. Series B (Methodological)*.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *ICML*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. In *ICLR*.

Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness. In *ACL*.

Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. 2021. A survey on contrastive self-supervised learning. *Technologies*.

Xiaoqian Jiang, Melanie Osl, Jihoon Kim, and Lucila Ohno-Machado. 2012. Calibrating predictive model estimates to support personalized medicine. *Journal of the American Medical Informatics Association*.

Alex Kendall and Yarin Gal. 2017. What uncertainties do we need in bayesian deep learning for computer vision? In *NeurIPS*.

Lingkai Kong, Haoming Jiang, Yuchen Zhuang, Jie Lyu, Tuo Zhao, and Chao Zhang. 2020. Calibrated language model fine-tuning for in-and out-of-distribution data. In *EMNLP*.

Ranganath Krishnan, Pi Esposito, and Mahesh Subedar. 2022. Bayesian-torch: Bayesian neural network layers for uncertainty estimation. `https://github.com/IntelLabs/bayesian-torch`.

Ranganath Krishnan and Omesh Tickoo. 2020. Improving model calibration with accuracy versus uncertainty optimization. In *NeurIPS*.

Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. 2018. Accurate uncertainties for deep learning using calibrated regression. In *ICML*.

Aviral Kumar, Sunita Sarawagi, and Ujjwal Jain. 2018. Trainable calibration measures for neural networks from kernel mean embeddings. In *International Conference on Machine Learning*.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *NeurIPS*.

Dan Ley, Umang Bhatt, and Adrian Weller. 2022. Diverse, global and amortised counterfactual explanations for uncertainty estimates. In *AAAI*.

Nut Limsopatham. 2021. Effectively leveraging bert for legal document classification. In *EMNLP Workshop*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Yong Liu and Xin Yao. 1999. Ensemble learning via negative correlation. *Neural networks*.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *ICLR*.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *ACL*.

Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heinz, and Dan Roth. 2021. Recent advances in natural language processing via large pre-trained language models: A survey. *arXiv preprint arXiv:2111.01243*.

Matthias Minderer, Josip Djolonga, Rob Romijnders, Frances Hubis, Xiaohua Zhai, Neil Houlsby, Dustin Tran, and Mario Lucic. 2021. Revisiting the calibration of modern neural networks. In *NeurIPS*.

Jishnu Mukhoti, Viveka Kulharia, Amartya Sanyal, Stuart Golodetz, Philip Torr, and Puneet Dokania. 2020. Calibrating deep neural networks using focal loss. In *NeurIPS*.

Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. 2019. When does label smoothing help? In *NeurIPS*.

Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. 2018. Analyzing uncertainty in neural machine translation. In *ICML*.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. 2019. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *NeurIPS*.

Yao Qin, Xuezhi Wang, Alex Beutel, and Ed Chi. 2021. Improving calibration through the relationship with adversarial robustness. In *NeurIPS*.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*.

Adrian E Raftery, Tilmann Gneiting, Fadoua Balabdaoui, and Michael Polakowski. 2005. Using bayesian model averaging to calibrate forecast ensembles. *Monthly weather review*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *ACL*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *EMNLP*.

Neeraj Varshney, Swaroop Mishra, and Chitta Baral. 2022a. Investigating selective prediction approaches across several tasks in iid, ood, and adversarial settings. In *ACL (Findings)*.

Neeraj Varshney, Swaroop Mishra, and Chitta Baral. 2022b. Towards improving selective prediction ability of nlp systems. In *ACL Workshop*.

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 *NeurIPS*.

Yoav Wald, Amir Feder, Daniel Greenfeld, and Uri Shalit. 2021. On calibration and out-of-domain generalization. In *NeurIPS*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019a. Superglue: A stickier benchmark for general-purpose language understanding systems. In *NeurIPS*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2019b. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*.

Xiangpeng Wei, Heng Yu, Yue Hu, Rongxiang Weng, Luxi Xing, and Weihua Luo. 2020. Uncertainty-aware semantic augmentation for neural machine translation. In *EMNLP*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *EMNLP*.

Yuxin Xiao, Zecheng Zhang, Yuning Mao, Carl Yang, and Jiawei Han. 2022. Sais: Supervising and augmenting intermediate steps for document-level relation extraction. In *NAACL*.

Ji Xin, Raphael Tang, Yaoliang Yu, and Jimmy Lin. 2021. The art of abstention: Selective prediction and error regularization for natural language processing. In *ACL*.

Huiqin Yang and Carl Thompson. 2010. Nurses' risk assessment judgements: A confidence calibration study. *Journal of Advanced Nursing*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*.

Wonjin Yoon, Jinhyuk Lee, Donghyeon Kim, Minbyul Jeong, and Jaewoo Kang. 2019. Pre-trained language model for biomedical question answering. In *ECML-PKDD*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. In *EMNLP*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *ACL*.

Taolin Zhang, Zerui Cai, Chengyu Wang, Minghui Qiu, Bite Yang, and Xiaofeng He. 2021. Smedbert: A knowledge-enhanced pre-trained language model with structured semantics for medical text mining. In *ACL*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NeurIPS*.

Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. 2021. Document-level relation extraction with adaptive thresholding and localized context pooling. In *AAAI*.

Yikai Zhou, Baosong Yang, Derek F Wong, Yu Wan, and Lidia S Chao. 2020. Uncertainty-aware curriculum learning for neural machine translation. In *ACL*.

A Responsible NLP Research

In this paper, we aim to identify the best choice for each consideration in constructing a well-calibrated PLM-based pipeline via extensive empirical studies. Our empirical analysis involves training multiple large-scale PLMs and, consequently, consumes a fair amount of computational power. However, we believe that the takeaways from our analysis will benefit NLP practitioners at large, which will write off the computational cost in the future.

In particular, the Hugging Face package leveraged in our experiments utilizes the Apache License 2.0, and the Bayesian-Torch package utilizes the BSD 3-Clause License. We focus on PLM-based pipelines targeting English and assess them based on six NLP datasets, which aligns with the intended use of these datasets. We also release the evaluation benchmarks of our empirical analysis to illustrate the performance of different PLM-based pipelines based on diverse metrics. The benchmarks do not contain information that uniquely identifies individual people or offensive content.