AETTA: Label-Free Accuracy Estimation for Test-Time Adaptation

Taeckyung Lee[†] Sorn Chottananurak[†] Taesik Gong[‡] Sung-Ju Lee[†]

[†]KAIST [‡]Nokia Bell Labs

{taeckyung,sorn111930,profsj}@kaist.ac.kr, taesik.gong@nokia-bell-labs.com

Abstract

Test-time adaptation (TTA) has emerged as a viable solution to adapt pre-trained models to domain shifts using unlabeled test data. However, TTA faces challenges of adaptation failures due to its reliance on blind adaptation to unknown test samples in dynamic scenarios. Traditional methods for out-of-distribution performance estimation are limited by unrealistic assumptions in the TTA context, such as requiring labeled data or re-training models. To address this issue, we propose AETTA, a label-free accuracy estimation algorithm for TTA. We propose the prediction disagreement as the accuracy estimate, calculated by comparing the target model prediction with dropout inferences. We then improve the prediction disagreement to extend the applicability of AETTA under adaptation failures. Our extensive evaluation with four baselines and six TTA methods demonstrates that AETTA shows an average of 19.8%p more accurate estimation compared with the baselines. We further demonstrate the effectiveness of accuracy estimation with a model recovery case study, showcasing the practicality of our model recovery based on accuracy estimation. The source code is available at https://github.com/taeckyung/AETTA.

1. Introduction

The rise of deep learning has impacted various fields with remarkable achievements [4, 13, 17, 32, 33]. In real-world deep learning applications, the divergence between training and test data, known as domain shifts, often leads to poor accuracy. For instance, object detection models encountering previously unseen data (*e.g.*, variations of objects) or distributional shifts (*e.g.*, weather changes) might suffer from performance degradation. To overcome this challenge, Test-Time Adaptation (TTA) [2, 11, 12, 28, 29, 34–36] has been regarded as a promising solution recently and actively studied. TTA aims to adapt pre-trained models to domain shifts on the fly with only unlabeled test data.

Despite recent advances in TTA, significant challenges hinder its practical applications. The core issue is that TTA's reliance on unlabeled test-domain samples makes TTA susceptible to adaptation failures, especially in dynamic environments where the domain continuously changes [29, 30]. Although recent TTA studies deal with dynamic test streams in TTA [11, 12, 29, 35, 36], the inherent risk of TTA–blind adaptation to unseen test samples without ground-truth labels–remains a critical vulnerability. Notably, the absence of ground-truth labels makes it difficult to monitor the correctness of the adaptation. While various out-of-distribution performance estimation approaches have been proposed [1, 5, 15, 27], such methods necessitate labeled train data for accuracy estimation, which is impractical for TTA scenarios.

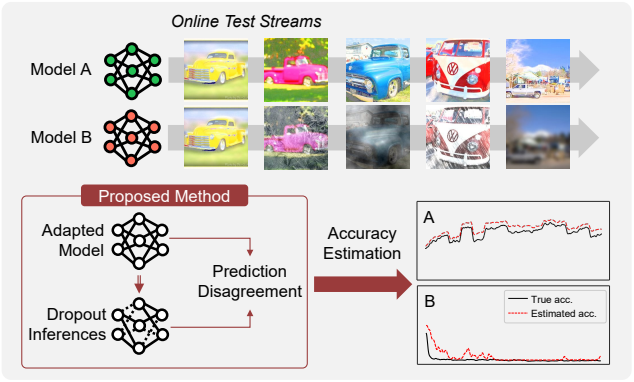


Figure 1. AETTA estimates the model's accuracy after adaptation using unlabeled test data without needing source data or ground-truth labels. AETTA can be integrated into existing TTA methods to estimate their accuracy under various scenarios.

In light of these challenges, we propose AETTA (Accuracy Estimation for Test-Time Adaptation), a novel accuracy estimation method designed for TTA without reliance on labeled data or source data access (Figure 1). AETTA leverages *prediction disagreement with dropout inferences*, where the prediction disagreement between the adapted model and dropout inferences serves as a basis for performance estimation. To enhance AETTA's robustness to adaptation failure scenarios, we propose *robust disagreement equality* that dynamically adjust the accuracy estimates based on model

failures. The key idea is to extend the well-calibration assumption (*i.e.*, predicted probabilities of expected model predictions are neither over-/under-confident [21]) to cover over-confident models (*e.g.*, adaptation failures) via adaptive scaling of the predicted probability. In addition, we provide theoretical analysis on how AETTA can estimate accuracy with unlabeled test data.

We evaluate AETTA on three TTA benchmarks (CIFAR10-C, CIFAR100-C, and ImageNet-C [18]) with two scenarios of fully TTA (*i.e.*, adapting to each corruption) [34] and continual TTA (*i.e.*, continuously adapting to 15 corruptions) [35]. We evaluate the accuracy estimation of AETTA integrated with six state-of-the-art TTA algorithms [12, 28, 29, 34–36]. We compare AETTA with four baselines that could be applied in the TTA setting. The result illustrates that AETTA shows an average of 19.8%p more accurate estimation compared with the baselines in various TTA methods and evaluation scenarios.

Furthermore, we explore the impact of performance estimation in TTA through a case study where we avoided undesirable accuracy drops in TTA based on AETTA. We propose a simple model recovery algorithm, which resets the model when consecutive estimated accuracy degradation or sudden accuracy drop are observed. Our case study shows that our model recovery algorithm with accuracy estimation achieved 11.7%p performance improvement, outperforming the best baseline that knows when distribution changes by 3.0%p. The result shows an example where accuracy estimation could benefit TTA in practice.

2. Preliminaries

2.1. Test-Time Adaptation (TTA)

Consider the source data distribution $\mathcal{D}^{\mathcal{S}}$, and the target data distribution $\mathcal{D}^{\mathcal{T}}$ and its random variable (X, Y), where Y is typically unknown to the learning algorithm, and K is total number of classes. The covariate shift assumption [31] asserts a disparity between the source and target data distributions, defined by $\mathcal{D}^{\mathcal{S}}(\mathbf{x}) \neq \mathcal{D}^{\mathcal{T}}(\mathbf{x})$ while maintaining consistency in the conditional label distribution: $\mathcal{D}^{\mathcal{S}}(y|\mathbf{x}) = \mathcal{D}^{\mathcal{T}}(y|\mathbf{x})$.

Let $h \sim \mathcal{H}_{\mathcal{A}}$ denote a hypothesis that predicts a single class for a single input and f denote a corresponding softmax value before class prediction. We define the hypothesis space $\mathcal{H}_{\mathcal{A}}$ as a hypothesis space $\mathcal{H}$ induced by a stochastic training algorithm $\mathcal{A}$ [21]. The stochasticity could arise from a different random initialization or data ordering.

Assuming an off-the-shelf model $h_0 \sim \mathcal{H}_{\mathcal{A}}$ pre-trained on $\mathcal{D}^{\mathcal{S}}$, the goal of (fully) test-time adaptation (TTA) [34] is to adapt h_0 for the target distribution $D^{\mathcal{T}}$ to produce h, using a batch of the unlabeled test set in an online manner.

2.2. Accuracy Estimation in TTA

We adopt a common TTA setup where source data is unavailable and target test data lacks labels [12, 28, 29, 34–36]. The objective of TTA accuracy estimation is to predict the test accuracy (or error) with unlabeled test streams.

Given an adapted model $h(\cdot; \Theta)$ at time t, we denote the test error of model $h(\cdot; \Theta)$ by:

$$\texttt{Err}_{\mathcal{D}^{\mathcal{T}}}(h) \triangleq \mathbb{E}_{\mathcal{D}^{\mathcal{T}}}[\mathbb{1}(h(X) \neq Y)]. \tag{1}$$

Note that we use the terms test accuracy and test error depending on the context, and the sum of them is 1. Given the temporal nature of TTA, we consider estimating the accuracy of the model $h(\cdot; \Theta)$–which has been updated before time t–with the test batch $\mathbf{X}_t$. Following the estimation, the test batch $\mathbf{X}_t$ is used for adaptation.

3. Methodology

3.1. Disagreement Equality

We introduce an approach for estimating the test error of a model that is adapted at test time. The key idea is to compare the model's output against outputs generated through dropout inference. Remarkably, this estimation process does not rely on access to the original training or labeled test data, which contrasts with existing accuracy estimation methods [1, 5, 15, 21, 27]. For example, generalization disagreement equality (GDE) [21] proposes a theoretical ground for estimating model error by measuring the disagreement rate between two networks. However, GDE requires multiple pre-trained models from different training procedures to calculate the disagreement rate.

Instead of multiple pre-trained models, our strategy utilizes dropout inference sampling, a technique where random parts of a model's intermediate layer outputs are omitted during the inference process [9]. From a single adapted model, we simulate the behavior of independent and identically distributed (i.i.d.) models by dropout inference sampling.

Definition 3.1. The hypothesis space $\mathcal{H}_{\mathcal{A}}$ satisfies the **dropout independence** if for any $h \sim \mathcal{H}_{\mathcal{A}}$, h and its dropout inference samples are i.i.d. over $\mathcal{H}_{\mathcal{A}}$.

To estimate the accuracy of the model, we propose **prediction disagreement with dropout inferences (PDD)** that calculates a disagreement between the adapted model $h(\cdot; \Theta)$ and the dropout inferences $h(\cdot; \Theta^{\texttt{dropout}})$ with respect to test samples as:

$$\texttt{PDD}_{\mathcal{D}^{\mathcal{T}}}(h) \triangleq \mathbb{E}_{\mathcal{D}^{\mathcal{T}}}\left[\frac{1}{N}\sum_{i=1}^{N}\mathbb{1}\left[h(X;\Theta) \neq h(X;\Theta^{\texttt{dropout}_i})\right]\right], \tag{2}$$

where N is the number of dropout inferences.

We now provide the theoretical background to estimate test error with PDD. We first define the expectation function $\tilde{h}$ [21] over hypothesis space $\mathcal{H}_\mathcal{A}$, which produces probability vector of size K. For k-th element $\tilde{h}_k(\mathbf{x})$, we define:

$$\tilde{h}_k(\mathbf{x}) \triangleq \mathbb{E}_{h\sim\mathcal{H}_\mathcal{A}}[\mathbb{1}[h(\mathbf{x}) = k]], \tag{3}$$

which indicates the probability of a sample $\mathbf{x}$ sampled from $D^\mathcal{T}$ being classed as the class k. Note that the expectation function does not represent the model's accuracy; it indicates the probability of the input being classified as a particular class, regardless of the ground truth labels.

Then, we define a confidence-prediction calibration assumption, indicating that the value of $\tilde{h}$ for a particular class equals the probability of the sample having the same ground-truth label [21].

Definition 3.2. The hypothesis space $\mathcal{H}_\mathcal{A}$ and corresponding expectation function $\tilde{h}$ satisfies **confidence-prediction calibration**[1] on $\mathcal{D}^\mathcal{T}$ if for any confidence value $q \in [0, 1]$ and class $k \in [1, \cdots, K]$:

$$p(Y = k|\tilde{h}_k(X) = q) = q. \tag{4}$$

With PDD and the assumption of dropout independence and confidence-prediction calibration, we are able to estimate the model's prediction error h (Theorem 3.1). Detailed proof is provided in the Appendix A.1.

Theorem 3.1 (Disagreement Equality)**.** *If the hypothesis space $\mathcal{H}_\mathcal{A}$ and corresponding expectation function $\tilde{h}$ satisfies dropout independence and confidence-prediction calibration, prediction disagreement with dropouts (PDD) approximates the test error over $\mathcal{H}_\mathcal{A}$:*

$$\mathbb{E}_{h\sim\mathcal{H}_\mathcal{A}}[\texttt{Err}_{\mathcal{D}^\mathcal{T}}(h)] = \mathbb{E}_{h\sim\mathcal{H}_\mathcal{A}}[\texttt{PDD}_{\mathcal{D}^\mathcal{T}}(h)]. \tag{5}$$

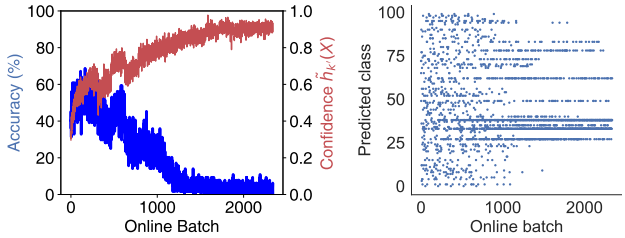
3.2. Robust Disagreement Equality

Adaptation failures in TTA are often coupled with over-confident incorrect predictions. Figure 2 shows an illustrative example of this case; as the expectation function's accuracy drops, the confidence increases, and predictions are skewed towards a few classes[2]. This violates the confidence-prediction calibration, leading to a high misalignment between test error and PDD (red lines in Figure 3).

To tackle the issue, we propose a **robust confidence-prediction calibration** to provide the theoretical ground of accuracy estimation for both well-calibrated or over-confident expectation function $\tilde{h}$.

[1]We rename the term from class-wise calibration [21] to clearly state the purpose of the calibration.

[2]Using the probabilistic property of expectation of dropout inferences [9], we approximate $\tilde{h}(X)$ as $\mathbb{E}_{\mathcal{H}_\mathcal{A}}[\mathbb{E}_{\texttt{dropout}}[h(X; \Theta^{\texttt{dropout}})]]$.



(a) Test batch accuracy and confidence. (b) Predicted class distribution.

Figure 2. Batch-wise accuracy, confidence, and prediction distribution when a model failed to adapt. TENT [34] is used on CIFAR100-C with continually changing domains. The model becomes over-confident, and predictions are skewed.

Definition 3.3. The hypothesis space $\mathcal{H}_\mathcal{A}$ and corresponding expectation function $\tilde{h}$ satisfies **robust confidence-prediction calibration** on $\mathcal{D}^\mathcal{T}$ if for any confidence value $q \in [0, 1]$, any class $k \in [1, \cdots, K]$, and the over-confident class k', there exists a weighting constant $b \geq 1$ and corresponding $0 \leq a \leq 1$ that satisfies:

$$p(Y = k'|\tilde{h}_{k'}(X) = q) = aq, \tag{6}$$

and

$$p(Y = k|\tilde{h}_k(X) = q) = bq \quad \text{for } k \neq k'. \tag{7}$$

Robust confidence-prediction calibration adjusts the over-confident expectation function $\tilde{h}$ to have a lower probability on the misclassified class k' via multiplying $a \leq 1$. Note that we can easily expand Definition 3.3 for multiple over-confident classes. Then, we estimate the test error with Theorem 3.2 (detailed proof in the Appendix A.2).

Theorem 3.2 (Robust Disagreement Equality)**.** *If the hypothesis space $\mathcal{H}_\mathcal{A}$ and corresponding expectation function $\tilde{h}$ satisfies dropout independence and robust confidence-prediction calibration with a weighting constant b, prediction disagreement with dropouts (PDD) approximates the test error over $\mathcal{H}_\mathcal{A}$:*

$$\mathbb{E}_{h\sim\mathcal{H}_\mathcal{A}}[\texttt{Err}_{\mathcal{D}^\mathcal{T}}(h)] = b\ \mathbb{E}_{h\sim\mathcal{H}_\mathcal{A}}[\texttt{PDD}_{\mathcal{D}^\mathcal{T}}(h)] - C, \tag{8}$$

where

$$C = \int_{q\in[0,1]} (b - a)\ q(1 - q)\ p(\tilde{h}_{k'}(X) = q)dq. \tag{9}$$

3.3. Accuracy Estimation for TTA

With Theorem 3.2, we propose an empirical approach to estimate the single model test error. Our experiments show that a single model's disagreement (and the test error) lies close to the robust disagreement equality. This aligns with the previous finding that a single pair of differently-trained models' disagreement rate (and the test error) lies close to

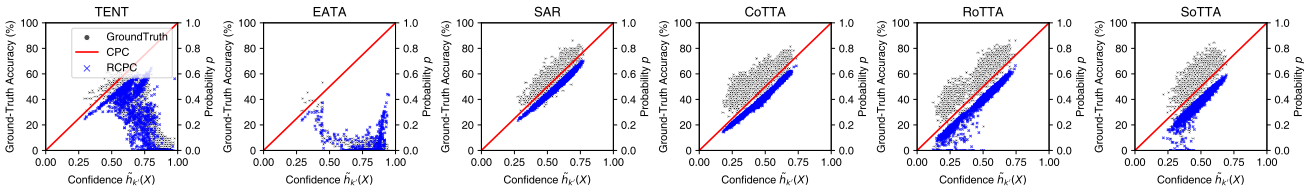


Figure 3. Correlations between the confidence value of estimated expectation function $\tilde{h}$ and (1) ground-truth accuracy (GroundTruth), (2) conditional probability $p(Y = k'|\tilde{h}_{k'}(X) = q)$ of confidence-prediction calibration (CPC), and (3) robust confidence-prediction calibration (RCPC). We used six TTA methods in CIFAR100-C with continual domain changes. We observed accuracy degradation in TENT and EATA and improvement in SAR, CoTTA, RoTTA, and SoTTA. When models failed to adapt, the original CPC misaligned with the ground truth. In contrast, our WCPC dynamically scaled the probability p, thus showing better alignment.

the disagreement equality [21]. Therefore, we approximate a single model test error as:

$$\texttt{Err}_{\mathcal{D}^\mathcal{T}}(h) \approx b\,\texttt{PDD}_{\mathcal{D}^\mathcal{T}}(h), \tag{10}$$

where we omit C due to the insufficient information regarding the true value of $p(\tilde{h}_{k'}(X) = q)$. Note that $C \approx 0$ for models with calibration.

Now, we discuss selecting a proper weighting constant b. Note that a desirable b should dynamically suppress the over-confident expectation function depending on the context so that the confidence-prediction calibration assumption holds. To this end, we use the skewness of the predicted outputs as an indicator of model over-confidence. Out intuition is based on the observation that the predicted class distribution is highly skewed when the adaptation fails (Figure 2b), which aligns with the findings from prior studies [19, 24]. Specifically, we estimate the skewness of predictions by calculating the entropy (Ent) of the batch-aggregated softmax values from the dropout inferences over a test batch $\mathbf{X}_t$:

$$E^{\texttt{avg}} = \texttt{Ent}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{|\mathbf{X}_t|}\sum_{\mathbf{x}\in\mathbf{X}_t} f(\mathbf{x};\Theta^{\texttt{dropout}_i})\right), \tag{11}$$

where $E^{\texttt{avg}}$ would maximize as $E^{\texttt{max}} = \texttt{Ent}(\vec{1}_K/K)$ with uniform predictions among the batch (*e.g.*, no failures); while the minimum value would be 0 when entire batch predicts a single class (*e.g.*, adaptation failures).

We then model b with $E^{\texttt{avg}}$ as:

$$b = \left(\frac{E^{\texttt{avg}}}{E^{\texttt{max}}}\right)^{-\alpha}, \tag{12}$$

where $\alpha \in [0, \infty)$ is a hyperparameter. If the adaptation does not fail, predictions are uniformly distributed as $E^{\texttt{avg}} = E^{\texttt{max}}$ and $b = 1$. Note that $a = b = 1$ drives Theorem 3.2 to be equivalent to Theorem 3.1. We found that modeling b with the average batch-wise entropy effectively corrects the correlation between confidence and prediction probability, as illustrated in Figure 3 (blue dots).

Finally, with Equation 10 and Equation 12, we propose **Accuracy Estimation for TTA (AETTA)**:

$$\texttt{Err}_{\mathcal{D}^\mathcal{T}}(h) \approx \left(\frac{E^{\texttt{avg}}}{E^{\texttt{max}}}\right)^{-\alpha}\texttt{PDD}_{\mathcal{D}^\mathcal{T}}(h). \tag{13}$$

Observe that $\alpha = 0$ and ∞ result in $\texttt{Err}_{\mathcal{D}^\mathcal{T}}(h) = \texttt{PDD}_{\mathcal{D}^\mathcal{T}}(h)$ and $\texttt{Err}_{\mathcal{D}^\mathcal{T}}(h) = 1$, respectively. Setting a small α would result in a lesser penalty with adaptation failures. On the other hand, choosing a high α would undesirably penalize model improvement cases. Our experiment found that accuracy estimation is not too sensitive to α (Figure 5b), and we chose $\alpha = 3$ for the other experiments.

Algorithm 1 AETTA: batchwise TTA accuracy estimation

Input: Test batch $\mathbf{X}_t$, model f, number of dropout inferences N

$\texttt{PDD} \leftarrow 0$
$\mathbf{Y}^{\texttt{avg}} \leftarrow \vec{0}$
$\hat{\mathbf{Y}} \leftarrow f(\mathbf{X}_t;\Theta)$
for $i \in \{1,\cdots,N\}$ **do**
  $\hat{\mathbf{Y}}^d \leftarrow f(\mathbf{X}_t;\Theta^{\texttt{dropout}_i})$
  $\mathbf{Y}^{\texttt{avg}} \leftarrow \mathbf{Y}^{\texttt{avg}} + \texttt{Avg}(\hat{\mathbf{Y}}^d)$
  $\texttt{PDD} \leftarrow \texttt{PDD} + \texttt{Avg}(\mathbb{1}[\arg\max(\hat{\mathbf{Y}}) \neq \arg\max(\hat{\mathbf{Y}}^d)])$
$\mathbf{Y}^{\texttt{avg}} \leftarrow \frac{1}{N}\mathbf{Y}^{\texttt{avg}}$
$\texttt{PDD} \leftarrow \frac{1}{N}\texttt{PDD}$ ▷ Avg. over dropouts
$E^{\texttt{avg}} \leftarrow \texttt{Ent}(\mathbf{Y}^{\texttt{avg}})$ ▷ Entropy of avg. batch
$\texttt{Err} \leftarrow \left(\frac{E^{\texttt{avg}}}{E^{\texttt{max}}}\right)^{-\alpha}\texttt{PDD}$ ▷ $\texttt{Err}_{\mathcal{D}^\mathcal{T}}(h)$
$\texttt{Acc} \leftarrow 1 - \texttt{Err}$

We summarize the accuracy estimation procedure in Algorithm 1. We first infer with the adapted model for the current test batch $\mathbf{X}_t$. Then, we repeatedly perform dropout inference sampling. With N samples from dropout inferences, we estimate the entropy of the batch-aggregated softmax output $E^{\texttt{avg}}$. Finally, we calculated the expected error of the model by AETTA. We apply the exponential moving average to the final accuracy estimation for stable error estimation.

Table 1. Mean absolute error (MAE) (%) of the accuracy estimation on fully TTA (adapting to each corruption type). **Bold** numbers are the lowest error. Averaged over three different random seeds for 15 types of corruption.

Dataset	Method	TTA Method						
		TENT [34]	EATA [28]	SAR [29]	CoTTA [35]	RoTTA [36]	SoTTA [12]	Avg. (↓)
Fully CIFAR10-C	SrcValid	18.37 ± 0.29	14.37 ± 0.33	21.28 ± 0.27	18.43 ± 0.16	20.35 ± 1.31	13.13 ± 0.85	17.66 ± 0.24
	SoftmaxScore [7]	6.26 ± 0.49	4.78 ± 0.12	5.21 ± 0.22	10.96 ± 0.28	6.01 ± 0.23	4.97 ± 0.50	6.37 ± 0.10
	GDE [21]	18.69 ± 0.28	16.95 ± 0.22	21.25 ± 0.27	14.50 ± 0.03	23.27 ± 0.43	16.45 ± 0.21	18.52 ± 0.13
	AdvPerturb [23]	23.06 ± 1.17	24.97 ± 1.00	21.89 ± 0.95	18.00 ± 0.82	19.35 ± 0.99	23.68 ± 0.85	21.83 ± 0.92
	AETTA	4.00 ± 0.03	3.87 ± 0.14	3.89 ± 0.07	6.83 ± 0.47	6.44 ± 1.35	5.28 ± 0.87	**5.05 ± 0.46**
Fully CIFAR100-C	SrcValid	38.96 ± 0.22	10.71 ± 0.31	42.68 ± 0.21	44.58 ± 0.30	23.50 ± 0.51	19.34 ± 0.63	29.96 ± 0.09
	SoftmaxScore [7]	17.34 ± 0.10	27.86 ± 1.11	24.56 ± 0.25	34.50 ± 0.35	24.18 ± 0.19	23.98 ± 0.21	25.40 ± 0.23
	GDE [21]	40.11 ± 0.05	71.53 ± 2.12	42.51 ± 0.23	33.21 ± 0.24	48.02 ± 0.56	34.24 ± 0.12	44.94 ± 0.23
	AdvPerturb [23]	24.17 ± 0.41	8.22 ± 0.56	22.91 ± 0.60	20.53 ± 0.14	17.84 ± 0.65	25.77 ± 0.47	19.91 ± 0.26
	AETTA	6.89 ± 0.15	20.15 ± 1.70	6.54 ± 0.15	6.05 ± 0.12	6.88 ± 0.10	5.29 ± 0.18	**8.63 ± 0.24**
Fully ImageNet-C	SrcValid	39.13 ± 0.89	35.89 ± 0.79	29.77 ± 0.94	41.09 ± 0.53	10.28 ± 0.28	16.00 ± 0.33	28.69 ± 0.54
	SoftmaxScore [7]	20.67 ± 0.01	21.06 ± 0.03	24.42 ± 0.08	19.62 ± 0.02	21.03 ± 0.04	23.60 ± 0.07	21.73 ± 0.03
	GDE [21]	70.58 ± 0.01	66.17 ± 0.07	63.48 ± 0.03	72.76 ± 0.02	66.39 ± 0.04	52.74 ± 0.02	65.35 ± 0.02
	AdvPerturb [23]	12.56 ± 0.03	14.52 ± 0.01	18.76 ± 0.06	11.05 ± 0.02	12.93 ± 0.04	22.90 ± 0.02	15.45 ± 0.02
	AETTA	6.14 ± 0.03	6.48 ± 0.02	6.43 ± 0.09	6.02 ± 0.03	14.82 ± 0.01	17.40 ± 0.26	**9.55 ± 0.07**

Table 2. Mean absolute error (MAE) (%) of the accuracy estimation on continual TTA (continuously adapting to 15 consecutive corruptions). **Bold** numbers are the lowest error. Averaged over three different random seeds for 15 types of corruption.

Dataset	Method	TTA Method						
		TENT [34]	EATA [28]	SAR [29]	CoTTA [35]	RoTTA [36]	SoTTA [12]	Avg. (↓)
Continual CIFAR10-C	SrcValid	10.84 ± 1.83	11.06 ± 0.11	21.29 ± 0.26	18.30 ± 0.25	13.37 ± 0.89	9.40 ± 0.85	14.04 ± 0.58
	SoftmaxScore [7]	41.10 ± 11.66	15.40 ± 4.73	5.21 ± 0.22	12.96 ± 0.37	12.57 ± 0.43	4.37 ± 0.09	15.27 ± 2.51
	GDE [21]	46.29 ± 10.93	26.44 ± 5.16	21.25 ± 0.27	14.69 ± 0.15	17.50 ± 0.30	17.03 ± 0.70	23.87 ± 2.43
	AdvPerturb [23]	15.56 ± 1.53	20.93 ± 2.83	21.88 ± 0.93	17.79 ± 0.74	22.95 ± 0.82	23.63 ± 0.78	20.45 ± 1.17
	AETTA	9.05 ± 1.02	7.13 ± 3.33	3.89 ± 0.06	5.82 ± 0.30	5.36 ± 1.22	4.73 ± 0.34	**6.00 ± 0.35**
Continual CIFAR100-C	SrcValid	11.00 ± 0.58	1.68 ± 0.18	38.20 ± 0.22	46.09 ± 0.38	19.43 ± 1.17	17.16 ± 1.57	22.32 ± 0.52
	SoftmaxScore [7]	58.29 ± 1.82	76.58 ± 0.71	24.05 ± 0.29	36.27 ± 0.68	27.19 ± 0.12	21.89 ± 0.35	40.71 ± 0.43
	GDE [21]	80.87 ± 1.29	94.01 ± 0.43	39.21 ± 0.22	35.43 ± 0.30	41.68 ± 0.45	35.29 ± 0.27	54.41 ± 0.18
	AdvPerturb [23]	10.12 ± 0.24	1.97 ± 0.33	24.93 ± 0.57	19.62 ± 0.15	21.18 ± 0.71	25.12 ± 0.39	17.16 ± 0.32
	AETTA	5.85 ± 0.36	4.18 ± 0.82	6.67 ± 0.12	6.55 ± 0.17	5.86 ± 0.10	5.32 ± 0.18	**5.74 ± 0.13**
Continual ImageNet-C	SrcValid	33.30 ± 0.93	36.42 ± 0.76	22.30 ± 0.55	41.06 ± 0.54	9.56 ± 0.26	14.28 ± 0.28	26.15 ± 0.53
	SoftmaxScore [7]	19.34 ± 0.02	20.16 ± 0.05	21.91 ± 0.16	19.63 ± 0.01	17.56 ± 0.08	19.67 ± 0.50	19.71 ± 0.53
	GDE [21]	68.30 ± 0.01	66.58 ± 0.03	64.36 ± 0.15	72.81 ± 0.07	73.76 ± 0.22	55.76 ± 0.45	66.93 ± 0.14
	AdvPerturb [23]	14.82 ± 0.02	14.15 ± 0.06	19.17 ± 0.14	11.06 ± 0.02	11.05 ± 0.05	20.83 ± 0.39	15.18 ± 0.09
	AETTA	5.66 ± 0.05	6.73 ± 0.03	6.68 ± 0.04	5.98 ± 0.04	11.19 ± 0.12	19.22 ± 0.79	**9.24 ± 0.14**

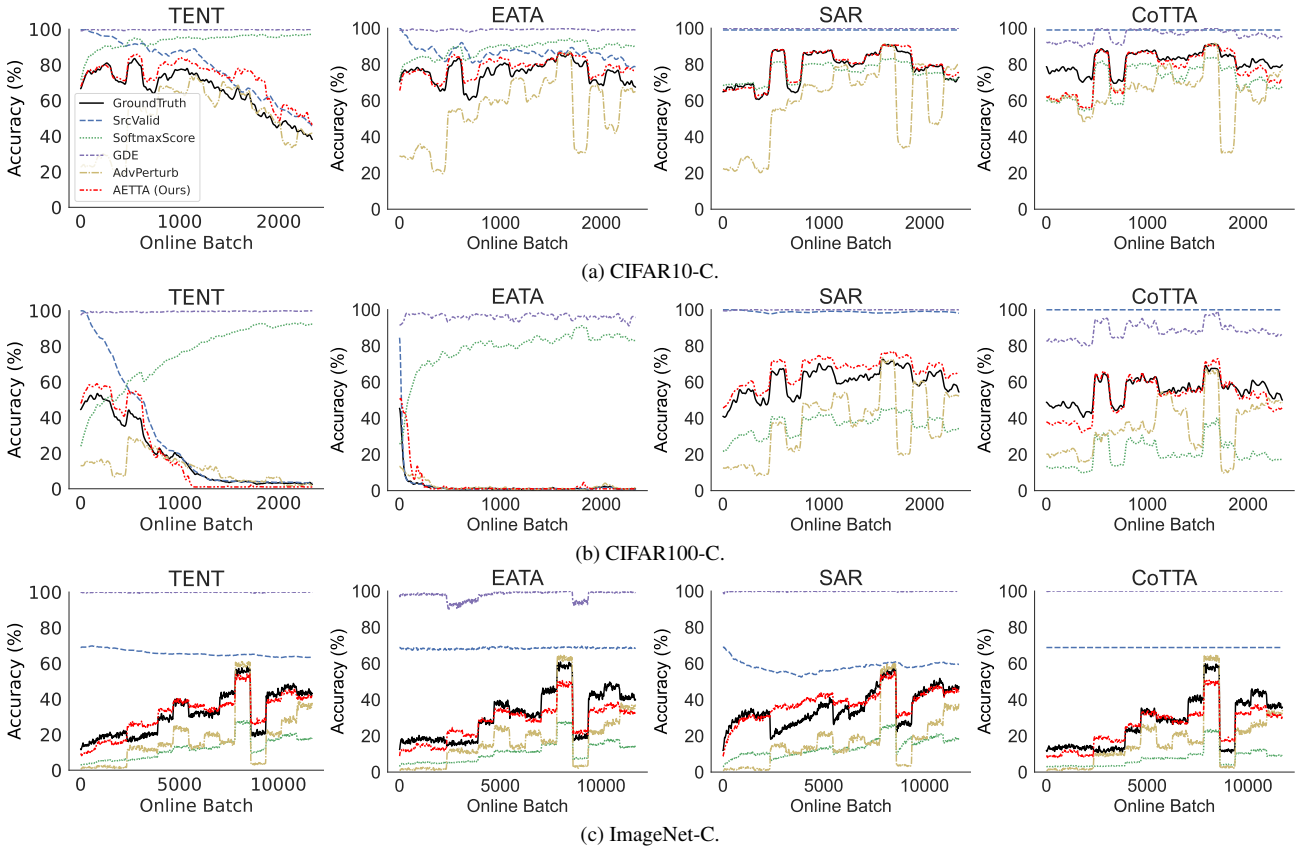
4. Experiments

We describe our experimental setup and present the results. Please refer to the Appendix D for further details.

Scenario. We consider both fully (non-continual) and continual test-time adaptation scenarios. In the fully TTA setting, target domains are each corruption type [34], while in the continual setting, the target domain continually changes to 15 different corruptions [35]. During adaptation, we calculate the accuracy estimation for every batch and report the mean absolute error between the ground-truth batch-wise accuracy. We ran experiments with three random seeds (0, 1, 2) and reported the average values. We use the test batch size 64 for all TTA baselines, with a memory size 64 for RoTTA [36] and SoTTA [12]. We specify further details of the hyperparameters in the Appendix D.2.

Datasets. We use three standard benchmarks for test-time adaptation: **CIFAR10-C**, **CIFAR100-C**, and **ImageNet-C** [18]. Each dataset contains 15 different corruptions with five levels of corruption, where we use corruption level 5. CIFAR10-C/CIFAR100-C/ImageNet-C contains 10/100/1,000 classes with 10,000/10,000/50,000 test data, respectively. We use pre-trained ResNet18 [17] as an adaptation target, following a recent study [12].

TTA Methods. We consider six state-of-the-art TTA methods. TENT [34] updates BN parameters with entropy minimization. EATA [28] utilizes entropy thresholding-based sample filtering and anti-forgetting regularization. SAR [29] also adapts sample filtering with sharpness minimization [8]. CoTTA [35] addresses the continual setting by augmentations and stochastic restoration of model weights to avoid catastrophic forgetting. RoTTA [36] adapts with robust batch normalization and category-balanced sampling with timeliness and uncertainty. SoTTA [12] utilizes high-confidence uniform-sampling and entropy-sharpness minimization for robust adaptation in noisy data streams [8].



(a) CIFAR10-C.

(b) CIFAR100-C.

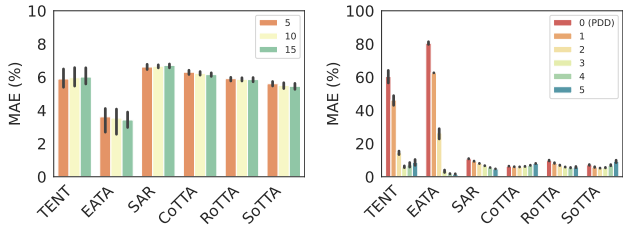
(c) ImageNet-C.

Figure 4. Qualitative results on continual CIFAR10-C, CIFAR100-C, and ImageNet-C.

Accuracy Estimation Baselines. We evaluate four distinct accuracy estimation baselines that could be applied to TTA settings: SrcValid, SoftmaxScore, GDE, and AdvPerturb.

- **SrcValid** is a widely used technique that validates performance by leveraging labeled source data. It computes the accuracy using a hold-out labeled source dataset to estimate the target performance. Importantly, the hold-out source data for validation were not used for training in other baselines to ensure they do not affect the model performance. Note that TTA usually assumes that source data are unavailable during test time; hence, this baseline is unrealistic in TTA. We nonetheless include SrcValid as one of our baselines to understand its performance when the source data are accessible.
- **SoftmaxScore** [7] utilizes the confidence scores derived from the last softmax layer as the model's accuracy, which is also a widely used baseline [5, 6]. It estimates the target domain accuracy by averaging softmax confidence scores computed from the current test batch. In addition, we apply temperature scaling [16] to improve the estimation performance [10].
- Generalization disagreement equality (**GDE**) [21] aims to estimate test accuracy by quantifying the (dis)agreement rate between predictions on a test batch generated by a pair of models. Since training multiple models is impractical, we compare the current adapted model and the previous model right before the adaptation. We also report a comparison with the original GDE and multiple pre-trained models in Appendix B.
- Adversarial perturbation (**AdvPerturb**) [23] also aims to estimate the OOD accuracy by calculating the agreement between the domain-adapted model and the source model, where adversarial perturbations on a test batch are applied to penalize the unconfident samples near the decision boundary. We note that the original paper aims to predict the accuracy of the source model, while our goal is to predict the accuracy of the adapted model.

Results. Table 1 and Table 2 show the results on the fully and continual TTA settings. We observe that none of the baselines could reliably predict the accuracy among different scenarios. On the other hand, AETTA achieves the lowest mean absolute error, including adaptation failure cases (*e.g.*, TENT in continual CIFAR10/100-C). On average, AETTA outperforms baselines by 19.8%p, validating the effectiveness of our robust prediction disagreement in diverse scenarios. More details are in the Appendix F.



(a) Number of dropout inferences N. (b) Scaling hyperparameter α.

Figure 5. Impact of hyperparameters on the accuracy estimation performance.

Qualitative Analysis. We qualitatively analyze the results of the baselines and AETTA to understand the behavior. Figure 4 visualizes the ground-truth accuracy and the estimated accuracy from the baselines and AETTA under adaptation failure and non-failure cases. The Gaussian filter is applied for visualization. We observe that AETTA generally shows a reliable estimation of the ground-truth accuracy in diverse scenarios (fully and continual) and datasets (CIFAR10/100-C and ImageNet-C). SrcValid correctly estimated when model accuracy decreases; however, it consistently predicted high accuracy when the adaptation did not fail. This limitation might be due to the distributional gap between source and target data. SoftmaxScore [7] captures the trend of ground-truth accuracy in some cases, but it overestimates the accuracy when the model accuracy drops. This is mostly due to the over-confident predictions from the model. GDE [21] showed to constantly predict high values among different TTA methods. Note that GDE was originally designed to utilize various pre-trained models. To use GDE in TTA, we utilize adapted models sampled at different stages of adaptation. The result suggests that utilizing multiple models from the single stochastic learning process might not be sufficient to consist of independent and identically distributed (i.i.d.) ensembles, leading to inaccurate estimation. AdvPerturb [23] shows accuracy estimations when ground-truth accuracy decreases but shows high errors in other cases. We believe this happens because it aims to evaluate the performance of the source model, not the adapted model. We found similar patterns were observed with different TTA methods.

Impact of Hyperparameter N. The number of dropout inferences, N, is a hyperparameter for calculating the test error. We conducted an ablation study in continual CIFAR100-C with varying $N \in \{5, 10, 15\}$. As shown in Figure 5a, we found the effect of hyperparameter N is negligible. We interpret this result as the effect of calculating prediction disagreement over sufficient batch size with dropout independence, which could reduce the probabilistic variances from dropout inference sampling. We adopt a single value of $N = 10$ for the other experiments.

Impact of Hyperparameter α. We investigate the impact of α, a hyperparameter to control the strength of robust confidence-prediction calibration. We conduct an ablation study in continual CIFAR100-C with varying $\alpha \in \{0, \dots, 5\}$, where $\alpha = 0$ indicates no weighting, thus $\texttt{Err} = \texttt{PDD}$. Figure 5b shows the result. Note that estimations are often inaccurate when $\alpha = 0$, which shows the importance of our robust equality. Setting a reasonable α is important to predict failed adaptation cases (TENT and EATA) properly, but it is generally robust after certain values. We adopt $\alpha = 3$ for the other experiments.

5. Case Study: Model Recovery

The deployment of TTA algorithms encounters a significant challenge when exposed to extreme test streams, such as continuously changing corruptions [35]. Several TTA algorithms (*e.g.*, TENT [34]) were not designed to exhibit robustness under such extreme conditions. Consequently, the model weights are poorly updated, leading to performance degradation, even worse than the source model. Although recent studies attempt to manage dynamic test streams [11, 12, 35], TTA algorithms are still susceptible to adaptation failures [30]. To tackle the issue, we perform a case study of model recovery based on the accuracy estimation.

Recovery Algorithm. We introduce a simple reset algorithm based on our accuracy estimation with AETTA. Our reset algorithm detects two cases: (1) consecutive low accuracies and (2) sudden accuracy drop. First, we reset the model if the five recent consecutive estimated accuracies (*e.g.*, $t-4, \cdots, t$) are lower than the five previous consecutive estimations (*e.g.*, $t-9, \cdots, t-5$). This way, we can detect the gradual degradation of TTA accuracy. Second, we apply hard lower-bound thresholding, which resets the model if the estimated accuracy is below the threshold (*e.g.*, 0.2). This could prevent catastrophic failure of TTA algorithms.

Baselines. Some TTA studies covered the model recovery/reset as a part of the TTA algorithm: Episodic resetting (**Episodic**) [37], where the model resets after every batch; Model Recovery Scheme (**MRS**) [29], where the model resets when the moving average of entropy loss falls below a certain threshold; Stochastic restoration (**Stochastic**) [35], where a small number of model weights are stochastically restored to the initial weight of the source model; and Fisher information based restoration (**FisherStochastic**) [3], which applies stochastic restoration for layer importance measured by Fisher information matrix. We also include a baseline (**DistShift**), which assumes that the model knows when the distribution changes and thus acts as an oracle. DistShift resets the model when the test data distribution (corruption) changes, which is not feasible in practice.

Table 3. Average accuracy improvement (%p) with model recovery. **Bold** number is the highest improvement. Averaged over three different random seeds for 15 types of corruption.

Method	TTA Method						
	TENT [34]	EATA [28]	SAR [29]	CoTTA [35]	RoTTA [36]	SoTTA [12]	Avg. (↑)
Episodic [37]	33.58 ± 1.04	51.28 ± 0.52	-7.00 ± 0.26	1.65 ± 0.10	-22.57 ± 0.85	-26.40 ± 0.51	5.09 ± 0.24
MRS [29]	24.12 ± 2.11	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	-1.97 ± 2.23	0.00 ± 0.00	3.69 ± 0.22
Stochastic [35]	35.93 ± 0.78	-0.01 ± 0.47	-2.00 ± 0.48	0.00 ± 0.00	-2.55 ± 0.49	0.35 ± 0.51	5.29 ± 0.19
FisherStochastic [3]	40.27 ± 1.29	0.12 ± 1.16	-4.85 ± 0.13	0.13 ± 0.03	-2.89 ± 0.13	-1.36 ± 0.51	5.24 ± 0.29
DistShift	38.93 ± 1.15	22.17 ± 2.38	-3.25 ± 0.10	1.51 ± 0.09	-7.63 ± 0.23	0.68 ± 0.19	8.74 ± 0.55
AETTA	36.79 ± 1.20	48.64 ± 0.74	-5.66 ± 0.20	1.64 ± 0.11	-6.03 ± 0.89	-4.97 ± 1.58	**11.73 ± 0.34**

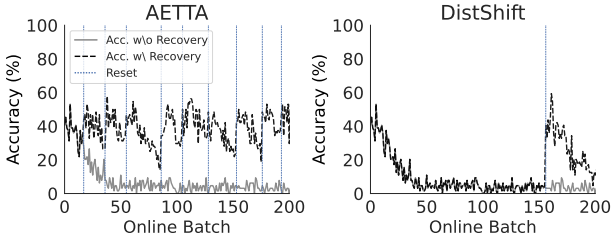


Figure 6. An example of model recovery compared with DistShift. Reset points are marked over the x-axis.

Results. Our simple recovery algorithm outperforms the baselines, including DistShift, which relies on an impractical assumption of knowing when the corruption changes. Episodic [37] showed high accuracy improvements under adaptation failures; however, it prevents continuous adaptation, even without adaptation failures. MRS [29] fails to recover among various TTA methods due to the hard-coded threshold of loss value. Stochastic [35] and FisherStochastic [3] show marginal improvements while failing to recover EATA. Our proposed reset algorithm successfully recovers from adaptation failures while minimizing the negative effect on TTA without failures.

Qualitative Analysis. Figure 6 shows an example of our model recovery compared with DistShift. Notably, our recovery algorithm resets only when an accuracy degradation trend is detected. On the other hand, DistShift failed to recover in the early steps since it resets the model only on distribution shifts. This implies that estimating performance degradation is more beneficial than knowing when the domain changes to improve TTA performance.

6. Related Work

Test-Time Adaptation. Recent progress in the field of test-time adaptation (TTA) has focused on improving model robustness [2, 11, 12, 28, 29, 35, 36] and addressing novel forms of domain shifts [11, 12, 35]. On the other hand, an analysis [30] pointed out the conventional TTA approaches remain prone to adaptation failures and demonstrated the importance of model recovery. In alignment with this insight, our work not only showcases the feasibility of accuracy estimation for TTA but also investigates a promising model recovery solution to enhance the robustness of TTA.

Accuracy Estimation. Existing accuracy estimation approaches mainly focus on the ensemble of multiple pre-trained models [1, 5, 15, 21, 27]. Accuracy-on-the-line [27] and Agreement-on-the-line [1] have demonstrated a notable linear relationship between performances in a wide range of models and distribution shifts, relying on the consistency of model predictions between in-distribution (ID) and out-of-distribution (OOD) data. The Difference of Confidence (DoC) [15] leverages differences in the model's confidence between ID and OOD data to estimate the accuracy gap under distribution shifts for calculating the final OOD accuracy. Self-training ensemble [5] estimates the accuracy of the pre-trained classifier by iteratively learning an ensemble of models with a training dataset, unlabeled test dataset, and wrongly classified samples. All these methods require labeled ID data to estimate OOD accuracy. To our knowledge, no existing studies target the accuracy estimation in TTA where source data and labels are unavailable.

7. Conclusion

We proposed a label-free TTA performance estimation method without access to source data and target labels. Based on the dropout inference sampling, we proposed calculating the prediction disagreement to estimate the TTA accuracy. We further improved the method with robust disagreement equality by utilizing the batch-aggregated distribution to penalize skewed predictions. Our method outperformed the baselines in diverse scenarios and datasets. Finally, our case study of model recovery showed the practicality of accuracy estimation. Our findings suggest that accuracy estimation is not only feasible but also a valuable tool in advancing the field of TTA without the need for labeled data.

Acknowledgements

This work was supported by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2022-0-00495, On-Device Voice Phishing Call Detection).

References

[1] Christina Baek, Yiding Jiang, Aditi Raghunathan, and J. Zico Kolter. Agreement-on-the-line: Predicting the performance of neural networks under distribution shift. In *Advances in Neural Information Processing Systems*, pages 19274–19289. Curran Associates, Inc., 2022. 1, 2, 8

[2] Malik Boudiaf, Romain Mueller, Ismail Ben Ayed, and Luca Bertinetto. Parameter-free online test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8344–8353, 2022. 1, 8

[3] Dhanajit Brahma and Piyush Rai. A probabilistic framework for lifelong test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3582–3591, 2023. 7, 8, 17, 24

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, pages 1877–1901. Curran Associates, Inc., 2020. 1

[5] Jiefeng Chen, Frederick Liu, Besim Avci, Xi Wu, Yingyu Liang, and Somesh Jha. Detecting errors and estimating accuracy on unlabeled data with self-training ensembles. In *Advances in Neural Information Processing Systems*, pages 14980–14992. Curran Associates, Inc., 2021. 1, 2, 6, 8

[6] Ching-Yao Chuang, Antonio Torralba, and Stefanie Jegelka. Estimating generalization under distribution shifts via domain-invariant representations. In *Proceedings of the 37th International Conference on Machine Learning*, pages 1984–1994. PMLR, 2020. 6

[7] Hady Elsahar and Matthias Gallé. To annotate or not? predicting performance drop under domain shift. 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)*, pages 2163–2173, 2019. 5, 6, 7, 14, 15, 18, 19, 20, 21, 22, 23

[8] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021. 5, 16

[9] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 1050–1059, New York, New York, USA, 2016. PMLR. 2, 3

[10] Saurabh Garg, Sivaraman Balakrishnan, Zachary Chase Lipton, Behnam Neyshabur, and Hanie Sedghi. Leveraging unlabeled data to predict out-of-distribution performance. In *International Conference on Learning Representations*, 2022. 6

[11] Taesik Gong, Jongheon Jeong, Taewon Kim, Yewon Kim, Jinwoo Shin, and Sung-Ju Lee. NOTE: Robust continual test-time adaptation against temporal correlation. In *Advances in Neural Information Processing Systems*, 2022. 1, 7, 8, 16

[12] Taesik Gong, Yewon Kim, Taeckyung Lee, Sorn Chottananurak, and Sung-Ju Lee. SoTTA: Robust test-time adaptation on noisy data streams. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 1, 2, 5, 7, 8, 14, 15, 16, 18, 19, 20, 21, 22, 23, 24

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Commun. ACM*, 63(11):139–144, 2020. 1

[14] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 15

[15] Devin Guillory, Vaishaal Shankar, Sayna Ebrahimi, Trevor Darrell, and Ludwig Schmidt. Predicting with confidence on unseen distributions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1134–1144, 2021. 1, 2, 8

[16] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017. 6, 15

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 5, 15, 16

[18] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. 2, 5, 16

[19] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017. 4

[20] Junyuan Hong, Lingjuan Lyu, Jiayu Zhou, and Michael Spranger. MECTA: Memory-economic continual test-time model adaptation. In *The Eleventh International Conference on Learning Representations*, 2023. 15

[21] Yiding Jiang, Vaishnavh Nagarajan, Christina Baek, and J Zico Kolter. Assessing generalization of sgd via disagreement. In *International Conference on Learning Representations*, 2021. 2, 3, 4, 5, 6, 7, 8, 11, 12, 14, 15, 18, 19, 20, 21, 22, 23

[22] Diederick P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 16

[23] JoonHo Lee, Jae Oh Woo, Hankyu Moon, and Kwonho Lee. Unsupervised accuracy estimation of deep visual models using domain-adaptive adversarial perturbation without source samples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16443–16452, 2023. 5, 6, 7, 14, 15, 18, 19, 20, 21, 22, 23

[24] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *Advances in*

Neural Information Processing Systems, pages 21464–21475. Curran Associates, Inc., 2020. 4

[25] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations (ICLR)*, 2017. 16

[26] TorchVision maintainers and contributors. Torchvision: Pytorch's computer vision library. https://github.com/pytorch/vision, 2016. 16

[27] John P Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *Proceedings of the 38th International Conference on Machine Learning*, pages 7721–7735. PMLR, 2021. 1, 2, 8

[28] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *Proceedings of the 39th International Conference on Machine Learning*, pages 16888–16905. PMLR, 2022. 1, 2, 5, 8, 14, 15, 16, 18, 19, 20, 21, 22, 23, 24

[29] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. In *The Eleventh International Conference on Learning Representations*, 2023. 1, 2, 5, 7, 8, 14, 16, 17, 18, 19, 20, 21, 22, 23, 24

[30] Ori Press, Steffen Schneider, Matthias Kümmerer, and Matthias Bethge. Rdumb: A simple approach that questions our progress in continual test-time adaptation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 1, 7, 8

[31] Joaquin Quiñonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D Lawrence. *Dataset shift in machine learning*. Mit Press, 2008. 2

[32] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. 1

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. 1

[34] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. 1, 2, 3, 5, 7, 8, 14, 16, 18, 19, 20, 21, 22, 23, 24

[35] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7201–7211, 2022. 1, 2, 5, 7, 8, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24

[36] Longhui Yuan, Binhui Xie, and Shuang Li. Robust test-time adaptation in dynamic scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15922–15932, 2023. 1, 2, 5, 8, 14, 16, 18, 19, 20, 21, 22, 23, 24

[37] Marvin Zhang, Sergey Levine, and Chelsea Finn. Memo: Test time robustness via adaptation and augmentation. In *Advances in Neural Information Processing Systems*, pages 38629–38642. Curran Associates, Inc., 2022. 7, 8, 17, 24