CTTA-T: Continual Test-Time Adaptation for Text Understanding via Teacher-Student with a Domain-aware and Generalized Teacher

Anonymous ACL submission

Abstract

Text understanding often suffers from domain shifts. To handle testing domains, domain adaptation (DA) is trained to adapt to a fixed and observed testing domain; a more challenging paradigm, test-time adaptation (TTA), cannot access the testing domain during training and online adapts to the testing samples during testing, where the samples are from a fixed domain. We aim to explore a more practical and underexplored scenario, continual test-time adaptation (CTTA) for text understanding, which involves a sequence of testing (unobserved) domains in testing. Current CTTA methods struggle in reducing error accumulation over domains and enhancing generalization to handle unobserved domains: 1) Noise-filtering reduces accumulated errors but discards useful information, and 2) accumulating historical domains enhances generalization, but it is hard to achieve adaptive accumulation. In this paper, we propose a **CTTA-T**[1] (**c**ontinual **t**est-**t**ime **a**daptation for **t**ext understanding) framework adaptable to evolving target domains: it adopts a teacher-student framework, where the teacher is domain-aware and generalized for evolving domains. To improve teacher predictions, we propose a refine-then-filter based on dropout-driven consistency, which calibrates predictions and removes unreliable guidance. For the adaptation–generalization trade-off, we construct a domain-aware teacher by dynamically accumulating cross-domain semantics via incremental PCA, which continuously tracks domain shifts. Experiments show CTTA-T excels baselines.

1 Introduction

Text understanding has a wide range of applications. Most existing models rely on offline training with fixed source data and perform well when the testing and training domains are aligned (Devlin et al., 2019; Hu et al., 2018). However, in real-world text understanding applications, testing domains shift over time, a phenomenon known as continual domain shift (Wang et al., 2022). This occurs, for instance, when a model trained in a source domain (i.e., movie reviewing) is later deployed for inference on various target domains (i.e., restaurant reviewing or book reviewing). Most current models do not improve themselves as they are applied to new target domains. It is sometimes necessary for models to continually enhance themselves as they encounter evolving target domains.

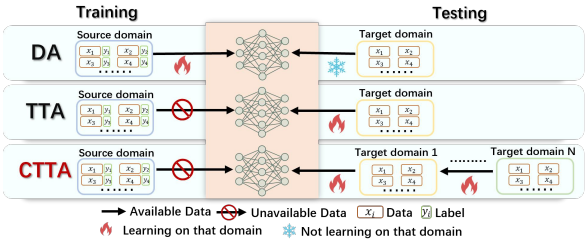


Figure 1: Comparison of three settings for adapting to domain shift. DA, TTA, and CTTA represent progressively more challenging and practical scenarios.

To mitigate the domain shift, some researchers explore domain adaptation (DA) in text understanding. DA improves model performance by reducing the gap between source (training) and target (testing) domains (Cao et al., 2020) to mitigate a single domain shift, which requires observed information of the target domain during training. However, in practice, domain shifts are often observed during testing, which limits the applicability of DA. A new task, test-time adaptation (TTA), enables models to adapt to domain shift during testing by learning online on unlabeled testing data from the fixed target domain. However, real-world scenarios always involve continual domain shifts (i.e., continuous multiple shifts), where domains shift over time, and TTA methods are inadequate in handling such scenarios. To overcome the limitation, we study continual test-time adaptation (CTTA) in text un-

[1] Code is: anonymous.4open.science/r/CTTA-T-CF28.

derstanding, where unlabeled testing samples arrive sequentially from continuously shifting domains. In this setting, the model must predict and adapt online without access to training samples or past and future testing samples. The illustration of the three settings (DA, TTA, and CTTA) is in Fig. 1.

Although TTA has explored techniques such as entropy minimization (Wang et al., 2021), teacher-student self-training (Lee et al., 2013; Ye et al., 2022), and contrastive learning (Chen et al., 2022), these methods are not fully applicable under CTTA for text understanding due to two limitations: 1) continual domain shift amplifies noise accumulation over evolving domains, eventually causing collapse (Guo et al., 2017; Chen et al., 2019). 2) TTA focuses on fixed domain shifts but lacks mechanisms to adapt to evolving textual patterns for evolving domains, leading to poor generalization.

To fix noise accumulation and generalization issues in CTTA mentioned above, researchers propose sample filtering and teacher-student framework methods: (1) To mitigate noise accumulation, researchers proposed sample filtering methods (Lee et al.; Wang et al., 2024) that remove noisy samples during adaptation to improve robustness. However, these methods often struggle to distinguish noise from useful signals, and directly filtering samples may harm performance by discarding target-domain information. (2) To address generalization issues, researchers proposed a teacher-student framework (Karim et al., 2023; Lyu et al., 2024) that accumulates information from previous domains to build a teacher for generating pseudo-labels to guide the student. This enhances generalization across domains: the teacher, updated by partially absorbing the student's parameters, provides cross-domain understanding ability, while the student adapts to each specific domain. When teacher accumulates historical domain information from student, the contribution of each domain remains fixed at every update and cannot adapt dynamically to domain-specific semantic information.

In summary, when applying existing CTTA methods (i.e., teacher-student with sample filtering) to text understanding, models struggle to balance the useful and noisy information from the target domain. We argue that we should carefully and adaptively absorb the sample-level and domain-level information: (1) considering uncertainty in sample filtering, and (2) dynamically accumulating domain-specific semantic information.

In this paper, we propose the **c**ontinual **t**est-**t**ime **a**daptation for **t**ext understanding **(CTTA-T)** framework: It adopts a teacher-student framework, equipping the teacher with both domain awareness and generalization to adapt to evolving domains, while applying a consistency-based refine-then-filter module to improve the quality of its predictions. Specifically, we use the teacher-student framework as the foundation for our implementation in CTTA for text understanding. We propose a refine-then-filter module to improve teacher prediction reliability by leveraging consistency from multiple stochastic forward passes under dropout. It first refines the teacher's softmax outputs via consistency-based reweighting, then filters low-consistency predictions. We propose a domain-aware teacher update module that employs IPCA to compute semantic shift, enabling the teacher to perform cross-domain accumulation by dynamically adjusting its integration of target-domain information, balancing domain-specific adaptation and cross-domain generalization. We apply stochastic restoration to part of the teacher's parameters by injecting source general knowledge to improve teacher generalization. Since there is currently no available benchmark for CTTA in text understanding, we further construct a benchmark that covers multiple domains and task sequences across robust QA, reading comprehension, and cross-lingual QA. Experiments show that baselines struggle with noise accumulation and generalization, while ours maintains stable performance in CTTA, with average improvements of 3.96% in EM and 4.19% in F1 over the strongest baseline.

Our contributions are: **(1)** We propose **CTTA-T**, a continual test-time adaptation (CTTA) framework for text understanding, and introduce a benchmark tailored for evaluating CTTA. **(2)** We introduce a refine-then-filter module to improve teacher prediction reliability based on prediction consistency. **(3)** We propose a domain-aware teacher update module that applies dynamically cross-domain accumulation via IPCA to balance adaptation and generalization. **(4)** Experiments on the CTTA benchmark demonstrate that CTTA-T achieves SOTA.

2 Related Work

Domain Adaptation in Text Understanding aims to improve generalization across domains in text understanding. It falls into two types: **1) Data-based methods** adapt to the target domain by generating or selecting relevant data. CANMD (Yue

et al., 2022) generates high-confidence pseudo-labels. T-SAS (Jeong et al., 2023) mitigates noise by generating and filtering answers. Domain-Inv (Khandelwal, 2024) aligns target and source domains without target labels. **2) Self-training-based methods** mitigate domain gaps by modifying model architecture or training protocols. TTT-NN (Hardt and Sun, 2024) retrieves neighbors during testing. STAF (Xiong et al., 2024) uses multi-stream perturbation consistency and memory. CrossIn (Lin et al., 2025) enables cross-lingual tuning using translated pairs. However, DA assumes access to domain shifts during training, while in practice, such shifts are observable at test time.

Test-Time Adaptation is a testing process that mitigates domain shift (Wang et al., 2021). It falls into two types: **1) Non-backpropagation-based methods** update modules or outputs solely during testing. LAME (Boudiaf et al., 2022) adjusts probabilities with Laplace maximum likelihood. FOA (Niu et al., 2024) uses covariance adaptation to learn input prompts. **2) Backpropagation-based methods** update the model by optimizing using test samples. Tent (Wang et al., 2021) adapts during testing by minimizing entropy. OIL (Ye et al., 2022) uses a teacher-student framework to improve adaptability. CoTTA (Wang et al., 2022) uses a teacher-student framework and causal inference. SAR (Niu et al., 2023) conducts sharpness-aware entropy minimization. SoTTA (Gong et al., 2023) buffers class-balanced samples and sharpens entropy minimization. DSS (Wang et al., 2024) mitigates noise through positive-negative learning. REM (Han et al., 2025) builds the prediction difficulty via a progressive masking strategy. Existing work focuses primarily on CV, and our goal is to achieve strong performance on continual domain shift in text understanding.

3 Method

The CTTA-T framework comprises four modules as shown in Fig. 2. (1) We employ a teacher-student framework as the foundation of our CTTA approach for text understanding (§3.1): the teacher absorbs cross-domain information, and students learn domain-specific information. (2) We apply a refine-then-filter module (§3.2) via consistency to improve teacher prediction reliability, which reduces the teacher-predicted uncertainty according to its consistency. (3) We propose a domain-aware teacher module (§3.3) that dynamically accumulates cross-domain semantic information from the student. We adopt incremental principal component analysis (IPCA) to extract and update the semantic space that reflects the evolving domain information. (4) We introduce a stochastic restoration module (§3.4) to enhance teacher generalization by randomly injecting source-domain knowledge.

3.1 Teacher-Student Model for CTTA in Text Understanding

To achieve continual test-time adaptation (CTTA) for text understanding, we build a CTTA backbone using a teacher-student framework where the teacher continuously guides the student during testing. This is motivated by text understanding, which involves context-sensitive semantics and domain-specific terminology, making continual adaptation crucial as domains shift during testing.

Let θ_t^T and θ_t^S denote the teacher and student models at time t, both initialized from the source model θ_0. At each time, we enforce consistency between their predictions by minimizing the cross-entropy of their softmax probability outputs:

$$\mathcal{L}_{\theta_t^S}(x_t) = -\sum_y p\left(y|x_t, \theta_t^T\right) \log p\left(y|x_t, \theta_t^S\right). \quad (1)$$

After the student is updated via Eq. 1, the teacher updates by partially absorbing the student with a fixed weight. In this framework, the student model θ_t^S acts as a fast adapter to the current domain, capturing domain-specific semantics from the textual embedding of x_t by directly updating its parameters via Eq. 1. The teacher model θ_t^T, in contrast, serves as a stable knowledge aggregator, progressively integrating the student's updates to maintain a general understanding of semantics across testing domains. This allows the teacher to provide reliable pseudo-labels to guide the student, even as the input domain shifts continuously.

3.2 Refine-then-Filter Teacher Prediction via Consistency

We propose a *Refine-then-Filter Teacher Prediction* (RFP) module that enhances teacher guidance by improving the reliability of teacher predictions before using them to guide the student. In CTTA, adaptation under continual domain shifts amplifies two types of uncertainty: *epistemic uncertainty* (EU), caused by model uncertainty, which leads to unstable predictions on unseen domains, *aleatoric uncertainty* (AU), caused by input noise, which introduces random noise even for familiar domain

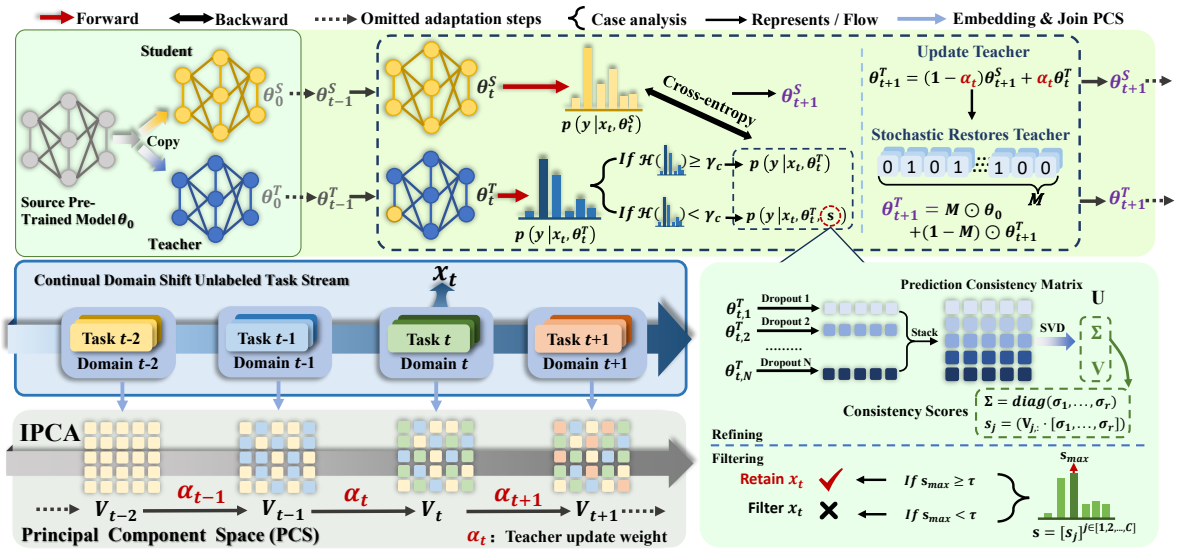


Figure 2: Overview of **CTTA-T**. When a textual sample x_t arrives (blue box), teacher and student predict on it. Low-entropy teacher outputs guide the student via cross-entropy; otherwise, $\mathbf{s}_{\max}$ is computed (bottom-right). Low-$\mathbf{s}_{\max}$ samples are discarded, while high ones combine the consistency distribution with the teacher output for refined student update. The student then updates the teacher (upper top-right), with update weight measured by IPCA domain distance (bottom-left). Finally, part of the teacher's parameters is randomly restored (lower top-right).

inputs (Kendall and Gal, 2017). As the EU stems from insufficient model confidence and the AU from irreducible data noise, handling them jointly is ineffective. To mitigate their effects, our RFP operates in two stages: the **refine** stage reduces EU by leveraging prediction consistency (i.e., the agreement among multiple stochastic forward passes under our dropout) to calibrate the teacher's outputs, which stabilizes predictions in unfamiliar domains. The **filter** stage suppresses AU by removing predictions with low consistency, as these likely arise from intrinsically noisy; eliminating them prevents such randomness from passing to the student. Hence, explicitly modeling EU and AU becomes essential to prevent accumulated uncertainty from destabilizing continual adaptation in CTTA.

Refinement via Prediction Consistency Distribution. To identify samples with high EU, we first use entropy as a screening mechanism and then perform refinement based on prediction consistency. We compute the entropy of the teacher's prediction as a proxy to detect high-EU samples, following established practice (Kendall and Gal, 2017). Specifically, for a given sample x_t, we compute the entropy $\mathcal{H}(p(y|x_t, \theta_t^T))$ using θ_t^T, and mark it as high-EU if the entropy exceeds a threshold γ_c. Once these high-EU samples are identified, we introduce a three-step prediction refinement process:

Step 1: Construct the prediction consistency matrix. We take N stochastic dropout masks on the teacher model θ_t^T at time t, yielding N sampled models $\theta_n^{dp^{n\in[1,2,\ldots,N]}}$. Each θ_n^{dp} belongs to a variational distribution $q(\theta)$ that approximates the true posterior. This enables MC sampling to approximate the model's predictive distribution and quantify EU through the variability of predictions across different dropout-induced parameter samples. For input x_t, we obtain the probability $p_n = P(y|x_t, \theta_n^{dp})$ from each sampled model. We then construct a prediction consistency matrix P_C by stacking these probability vectors column-wise: $P_C = [p_1, \ldots, p_N)]^T \in \mathbb{R}^{N\times C}$, where each row corresponds to one stochastic forward pass and each column corresponds to a class. Finally, we apply SVD to P_C to capture the principal consistency patterns among predictions:

$$P_C = U\Sigma V^T, \quad \Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r), \quad (2)$$

where $r = \mathrm{rank}(P_C)$, Each row of V^T represents a principal direction in the class space, whose corresponding singular value σ_i quantifies its importance. A large v_{ji} indicates a strong correlation between class j and principal direction i. By constructing and decomposing P_C, the model identifies coherent uncertainty patterns across MC samples, thereby isolating EU from unstable model beliefs.

Step 2: Obtain prediction consistency distribution. We define the prediction consistency distribution $\mathbf{s}$ as a class-wise summary of these principal components $V_{:,i}$. For each class j, its score $\mathbf{s}_j$ is

computed as the weighted sum of the principal directions $\{V_{:,i}\}_{i=1}^{r}$ associated with that class, using singular values $\{\sigma_i\}_{i=1}^{r}$ as weights:

$$\mathbf{s} = [\mathbf{s}_j]^{j\in[1,2,\ldots,C]}, \tag{3}$$

where $\mathbf{s}_j = \sum_{i=1}^{r} V_{j,i}\sigma_i$, and $V_{j,:}$ denotes the j-th row of V. The singular values capture how dominant each principal direction is among all stochastic predictions, while $V_{j,:}$ measures how much class j aligns with these stable directions. Therefore, a higher $\mathbf{s}_j$ indicates that class j consistently receives similar predictions across multiple MC-dropout samples, corresponding to lower EU.

Step 3: Refine the sample with prediction consistency distribution. We define the refined probability $p(y|x_t, \theta_t^T, \mathbf{s})$ as Eq. 4, where the prediction consistency distribution $\mathbf{s}$ is softmax-transformed to re-weight the original probabilities.

$$p(y|x_t, \theta_t^T, \mathbf{s}) = \text{softmax}(\text{softmax}(\mathbf{s}) \odot p(y|x_t, \theta_t^T)), \tag{4}$$

where $\odot$ is Hadamard product. For high-EU samples, we replace teacher's prediction in Eq. 1 with refined probability $p(y|x_t, \theta_t^T, \mathbf{s})$ to reduce EU.

Filtering via Prediction Consistency Score. Since AU originates from data noise and cannot be directly mitigated, we apply a filter guided by the prediction consistency score $\mathbf{s}$ to reduce AU. Specifically, we filter out samples if its prediction consistency score $\mathbf{s}_{\text{max}} = \max_j [\mathbf{s}_j]^{j\in[1,2,\ldots,\mathcal{C}]}$ is below the threshold τ, where $[\mathbf{s}_j]^{j\in[1,2,\ldots,\mathcal{C}]}$ (as eq. 3) is the prediction consistency distribution. When the sample x_t is dominated by data noise, the consistency matrix P_C exhibits random behavior and causes a drop in $\mathbf{s}_{\text{max}}$. Therefore, setting a threshold τ to filter out smaller $\mathbf{s}_{\text{max}}$ enables effective masking of samples with high AU (The proof of the upper bound of consistency score is in App. C.1).

3.3 Cross-Domain Accumulation for Domain-Aware Teacher

We propose a *Cross-Domain Accumulation for Domain-Aware Teacher (CDA)* module that enables a domain-aware teacher to dynamically accumulate cross-domain semantic information for text understanding in CTTA. Most teacher-student CTTA models adopt a fixed-weight exponential moving average (EMA) update (Tarvainen and Valpola, 2017). We argue that a domain-aware teacher should dynamically absorb student knowledge based on cross-domain shifts. Traditional EMA updates the teacher θ_t^T at each time t as:

$$\theta_{t+1}^T = \alpha\theta_t^T + (1-\alpha)\theta_{t+1}^S, \tag{5}$$

where the θ_{t+1}^S is updated student model through Eq. 1. However, traditional EMA, using a static accumulation rate α, fails to reflect domain shifts in streaming data, leading to either over-adaptation or under-adaptation in evolving text distributions. To facilitate domain-aware accumulation, CDA dynamically adjusts the update rate α by measuring the semantic distance between all historical and current domains via incremental principal component analysis (IPCA) (Ross et al., 2008) in two steps:

Step 1: Measuring domain semantic distance. We use IPCA to extract low-dimensional semantic representations C and measure the semantic shift distance between the current and historical domains. The motivation for using IPCA lies in its ability to incrementally accumulate domain semantics into a compact representation, enabling continual tracking of cross-domain evolution without storing past data. Specifically, we treat the covariance matrix as a proxy for domain-level semantics, since it captures co-occurrence patterns among features, which reflect the structural semantics of domain. At each time t, we compute the covariance matrix C_t from all historical samples $X_{\leq t}$, and apply singular value decomposition (SVD) to obtain the principal component space V_t ($C_t = V_t\Lambda_t V_t^\top$). When new data X_{t+1} arrives, IPCA integrates the new domain's semantic statistics into the historical covariance matrix, updating C_t to C_{t+1} as follows (see App. B.1 for detailed IPCA derivation):

$$C_{t+1} \approx \frac{N_t C_t + n\hat{C} + \frac{N_t n}{N_t+n}(\mu_t - \hat{\mu})(\mu_t - \hat{\mu})^\top}{N_t + n}, \tag{6}$$

where N_t, μ_t denote the count and mean of historical samples, and n, $\hat{\mu}$, $\hat{C}$ are the numbers, mean, and covariance of current samples. This allows us to incrementally maintain an up-to-date semantic representation C_{t+1} without storing all samples. The corresponding principal component matrices V_t and V_{t+1} represent the main semantic axes of the historical and current domains. Since the principal component space captures the densest semantic directions, the Frobenius norm of their variation,

$$distance \approx \|\Delta V\|_F^2, \tag{7}$$

where $\Delta V = V_t - V_{t+1}$, measures how much the domain semantics have shifted over time (see App. B.2 for $distance$ detailed derivation). The Frobenius norm summarizes the squared difference of each component axis, offering an intuitive and efficient measure of domain-level variation. In conclusion, IPCA allows us to accumulate cross-domain

semantics in a low-dimensional space and dynamically quantify domain shift over time, forming the basis for domain-aware teacher adaptation in CDA.

Step 2: Obtain teacher weight α according to domain semantic distance. We obtain the teacher's update weight α based on the semantic distance measured in Step 1. Intuitively, a large domain distance (large ΔV) indicates that the current domain introduces substantial semantic change. To address it, the teacher needs to obtain more up-to-date domain information provided by the student (θ^S_{t+1}) to adapt effectively. Conversely, a small ΔV suggests semantic similarity between the current and historical domains, where retaining prior knowledge is more beneficial and relying more on previous parameters (θ^T_t). To formalize the relation, we define $\mathcal{F}$ that reflects the trade-off between aligning with the current student and preserving past information:

$$\begin{aligned}\mathcal{F}(\theta^T_{t+1}) =& \| \Delta V \|^2_F \cdot (\theta^T_{t+1} - \theta^S_{t+1})^2 \\ &+(1- \| \Delta V \|^2_F) \cdot (\theta^T_{t+1} - \theta^T_t)^2.\end{aligned} \tag{8}$$

It shows that when the distance between all historical domains and the current domain increases (i.e., large ΔV), the term $(\theta^T_{t+1} - \theta^S_{t+1})^2$ is emphasized, forcing the teacher to align with the student and adapt to the new domain. When ΔV is small, the term $(\theta^T_{t+1} - \theta^T_t)^2$ is emphasized, forcing the teacher to retain historical domains information by pushing teacher to its previous parameters. Then, we differentiate Eq. 8 to obtain the gradient for updating the teacher model. Finally, we combine the result of the differentiation with Eq. 5 to derive the weight as $\alpha = 1 - \textit{distance}$ (derivation in App. B.4). This derivation provides a theoretically grounded mechanism to replace the fixed α in prior work to result in a domain-aware update rule.

When ΔV is large, the update weight α decreases, prompting the teacher to absorb more from the student and better adapt to the current domain; vice versa, preserving historical parameters to prevent overfitting and maintain general information.

CDA continuously adjusts α to modulate the trade-off between general knowledge retention and new domain-specific semantics in CTTA.

3.4 Stochastic Restoration for Generalized Teacher

We propose a *Stochastic Restoration for Generalized Teachers (SRT)* module that enhances the generalization ability of the teacher model by randomly restoring a portion of its parameters to their initial source values. The motivation is to enhance the generalization of the teacher by injecting source-domain knowledge, thereby preventing it from overfitting to transient and potentially noisy test-time domains. Specifically, for each parameter tensor, we independently sample a binary mask $M \in \{0,1\}^d$ from a Bernoulli distribution with restoration probability p, i.e., $M \sim \text{Bernoulli}(p)$. The restoration operation is defined as:

$$\theta^T_{t+1} = M \odot \theta_0 + (1 - M) \odot \theta^T_{t+1}, \tag{9}$$

where $\odot$ denotes Hadamard product, θ_0 is source parameters, and θ^T_{t+1} is teacher parameter updated via Eq. 5. By partially restoring the teacher's parameters, SRT constrains the deviation from general representations to improve generalization.

Note that SRT is compatible with our cross-domain accumulation module (CDA). While CDA adapts the teacher to recent test-time samples, SRT injects generalization signals from the source parameters. Together, they balance adaptation and generalization. SRT is inspired by CoTTA (Wang et al., 2022), which aims to combat overfitting under continual shifts. However, unlike CoTTA, which applies restoration to the student and disrupts critical prediction parameters, we apply restoration to the teacher. This design avoids direct impact on inference and focuses on enhancing the teacher's generalization, thereby improving the quality of pseudo-labels used for student adaptation.

4 Experiments

4.1 Experimental Settings

Tasks and Benchmarks. We establish a CTTA benchmark for text understanding covering 3 tasks and 11 subtasks: Robust QA (Ravichander et al., 2021) (speech, keyboard, translation noise), Reading Comprehension (Fisch et al., 2019) (Search, Trivia), and Cross-lingual QA (Artetxe et al., 2020) (speech, keyboard, translation noise). We construct short (7-task) and long (11-task) streams with one standard order (Order 4) and two randomized variants (Orders 5–6) via Fisher–Yates shuffle; Orders 1–3 follow the same strategy (details in App. E).

Baselines. *Tent* (Wang et al., 2021) minimizes the entropy of samples. *OIL* (Ye et al., 2022) employs a teacher-student framework and causal inference. *CoTTA* (Wang et al., 2022) combines a teacher-student framework, mean predictions over augmentations. *SAR* (Niu et al., 2023) filters low-entropy samples and sharpens entropy minimiza-

	Short Sequence						Long Sequence							
Order	**Order1 (t⟶)**		**Order2 (t⟶)**		**Order3 (t⟶)**		**Order4 (t⟶)**		**Order5 (t⟶)**		**Order6 (t⟶)**		**Avg**	
Methods	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
base	52.52	65.45	52.52	65.45	52.52	65.45	55.94	68.27	55.94	68.27	55.94	68.27	54.23	66.86
Tent	28.83	34.43	29.66	39.89	38.34	51.23	36.04	43.63	19.69	27.19	8.27	12.20	26.81	34.76
OIL	39.53	53.31	36.89	47.69	46.60	56.55	37.77	44.43	43.48	53.42	28.07	35.09	38.72	48.42
CoTTA	53.81	65.75	44.19	55.74	49.15	61.01	47.65	57.68	23.77	31.84	15.99	22.14	39.09	49.03
SAR	51.80	61.68	52.13	64.50	52.52	64.69	53.08	65.10	52.77	64.55	50.36	62.42	52.11	63.82
SoTTA	35.23	45.67	45.43	56.60	48.83	60.20	47.37	58.56	38.15	50.20	30.25	41.97	40.88	52.20
REM	39.45	49.16	44.31	55.25	44.93	54.07	45.08	54.63	35.28	44.94	34.55	46.31	40.60	50.73
CTTA-T (ours)	**54.14**	**66.57**	**53.88**	**66.01**	**54.07**	**66.12**	**57.74**	**69.33**	**57.63**	**69.33**	**57.71**	**69.24**	**55.86**	**67.77**
xTune	56.96	68.98	56.96	68.98	56.96	68.98	59.93	71.44	59.93	71.44	59.93	71.44	58.45	70.21
Tent	50.72	62.13	44.79	56.64	44.52	54.47	39.06	49.06	23.06	29.26	55.91	67.63	43.01	53.20
OIL	44.59	57.01	48.78	60.34	54.37	65.64	41.08	48.04	51.00	61.19	39.20	47.28	46.50	56.58
CoTTA	51.49	63.57	53.24	65.77	52.02	64.18	54.80	66.70	55.57	67.61	56.28	68.04	53.90	65.98
SAR	55.50	66.86	56.19	67.54	56.15	67.56	58.10	69.02	55.59	66.78	49.99	61.21	55.25	66.50
SoTTA	42.41	53.79	56.34	67.54	54.76	65.89	54.72	65.71	47.04	57.79	45.91	56.62	50.20	61.22
REM	49.72	61.07	49.72	62.24	51.57	62.82	52.34	63.76	50.69	62.15	45.94	58.77	50.00	61.80
CTTA-T (ours)	**57.85**	**69.72**	**57.99**	**69.76**	**58.00**	**69.80**	**60.84**	**72.08**	**60.92**	**72.18**	**60.86**	**71.96**	**59.41**	**70.92**

Table 1: Main results (%). We report EM and F1, and the final column shows the average score across all orders. **Bold**: the best. Underline: the second-best. "t →": the adaptation on a temporally ordered task stream. All results are statistically significant based on the t-test ($p < 0.01$), details in App. L. Hyperparameter analysis is in App. K

					Short Sequence						Long Sequence							
Order	TS-CTTA	CDA	RFP	SRT	**Order1 (t⟶)**		**Order2 (t⟶)**		**Order3 (t⟶)**		**Order4 (t⟶)**		**Order5 (t⟶)**		**Order6 (t⟶)**		**Avg**	
base					EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
A	✓				37.24	48.11	44.72	56.44	49.32	60.93	40.83	51.05	42.18	52.18	29.81	37.83	40.68	51.09
B	✓	✓			45.91	56.17	47.95	59.05	49.79	60.81	44.48	53.56	47.89	58.75	35.67	45.15	45.28	55.58
C	✓	✓	✓		48.12	59.35	46.37	57.29	51.22	62.78	48.71	58.74	51.90	63.17	39.46	49.76	47.63	58.52
CTTA-T (ours)	✓	✓	✓	✓	**54.14**	**66.57**	**53.88**	**66.01**	**54.07**	**66.12**	**57.74**	**69.33**	**57.63**	**69.33**	**57.71**	**69.24**	**55.86**	**67.77**
xTune					EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
A	✓				46.43	55.97	47.74	58.03	53.33	63.86	45.02	53.62	46.15	54.35	37.38	47.41	46.01	55.54
B	✓	✓			50.01	60.28	53.77	64.78	55.76	65.96	54.72	65.43	57.75	68.69	51.52	61.99	53.92	64.52
C	✓	✓	✓		55.12	66.58	55.45	66.85	57.22	68.85	55.36	65.71	57.46	68.49	53.68	64.58	55.72	66.84
CTTA-T (ours)	✓	✓	✓	✓	**57.85**	**69.72**	**57.99**	**69.76**	**58.00**	**69.80**	**60.84**	**72.08**	**60.92**	**72.18**	**60.86**	**71.96**	**59.41**	**70.92**

Table 2: Ablation study of the four modules in our framework. **Bold**: the best results.

tion. *SoTTA* (Gong et al., 2023) buffers high-confidence, class-balanced samples and sharpens entropy minimization. *REM* (Han et al., 2025) leverages entropy ranking and masked consistency to enforce difficulty-aware adaptation. See **Implementation Details** and **Metrics** in App. D.

4.2 Main Results

Tab. 1 reports overall results across task streams (Orders 1 to 6), evaluated using exact match (EM) and F1 scores. Each score denotes the average performance after applying CTTA under the corresponding order. Our method achieves SOTA performance with both backbone models, where the backbone refers to the pre-trained model used in the initial adaptation stage. Averaged over backbones, our method exhibits improvement over SAR (+3.96% in EM, +4.19% in F1) and REM (+12.31% , +13.08%), while outperforming vanilla forward, Tent, OIL, CoTTA, and SoTTA, confirming the practical effectiveness of our approach. We further compare with strong baselines (See App. J). Two observations are: **1) Our method shows better stability across orders.** In contrast, baselines exhibit heavy fluctuations. For example, Tent with the base model shows a 9.51% EM gap between Orders 1 and 3, ours just shows 0.07%. Analysis of the stability is presented in App. H. **2) The sensitivity of the method to the backbone is different.** A more robust backbone could improve performance. Analysis of backbone sensitivity is in the App G.

4.3 Ablation Studies

We follow standard CTTA evaluation protocols (Wang et al., 2022; Jiang et al., 2024) for ablation studies. Tab. 2 summarizes the contribution of each module. Row A (TS-CTTA, §3.1) serves as the baseline. Adding CDA enhances performance via dynamic cross-domain accumulation, enabling the teacher to gain domain awareness through IPCA-based semantic tracking. RFS further improves stability and accuracy by enforcing prediction con-

sistency and filtering uncertain guidance. However, as shown in Row C, prediction errors still accumulate and degrade performance over time. SRT alleviates this issue by periodically restoring the teacher to the source state, reintroducing general knowledge. Further analysis is provided in App. I.

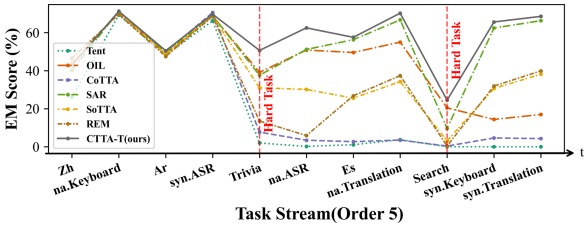


Figure 3: Study of error accumulation. The performance of our method and the baselines on 5.

4.4 Analysis Studies

Analysis of Error Accumlation. To analyze error accumulation, we evaluate all methods on short and long task sequences, particularly focusing on hard tasks (e.g., Search and Trivial), where error accumulation tends to be more severe. As shown in Fig. 3, all baselines suffered from error accumulation, which led to collapse, while ours remained stable or showed only minimal decline. Highlighting ours effectively mitigates error accumulation. Baseline collapse arises from uncontrolled error accumulation due to noisy pseudo-labels, though the underlying reasons differ across methods. More analysis is found in the App. F.

Table 3: Comparison of the cost of different methods.

Backbone	CTTA-T Time (s)	REM Time (s)	SoTTA Time (s)	SAR Time (s)	CoTTA Time (s)
Base	**0.043**	0.061	0.051	0.044	0.196
xTune	**0.040**	0.062	0.050	0.042	0.200

Cost Comparison. We evaluate the method cost (per-sample adaptation plus inference time (s)). Since baselines like Tent and OIL target fixed shifts and fail under continual shift (see Tab. 1), they are excluded from cost comparison. For fairness, we compare only with CTTA-specific baselines: CoTTA, SAR, SoTTA, and REM. Tab. 3 reports average inference plus adaptation time across Orders.

Analysis of Filtering Strategies. To assess the RFP's effectiveness, we compare it with alternative filtering strategies in three settings: 1) No Filter, 2) Confidence Filter (replaces RFP with confidence-based filtering), and 3) RFP w/o Filtering (disables RFP's filtering). As shown in Fig. 4, the RFP (red)

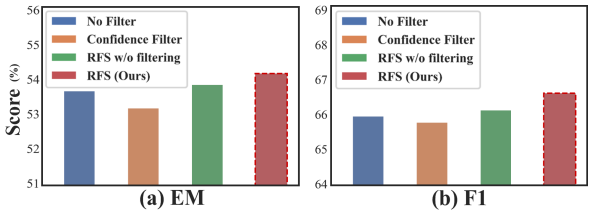


Figure 4: Performance of different filtering strategies.

outperformed all strategies, confirming its effectiveness in improving prediction by refining prediction and filtering unreliable guidance. Using only refining (green) ranks second, showing that refining improves prediction, while the Confidence Filter (orange) underperforms No Filter (blue), indicating that directly relying on prediction confidence is harmful in CTTA, where domain shifts cause overconfident wrong predictions.

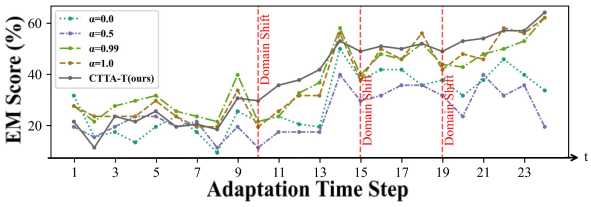


Figure 5: Performance of our method and model using fixed weights under domain shift (report EM each step).

Analysis of CDA versus Fixed Teacher Updates. To assess CDA under CTTA, we compared it with fixed weights (α=0/0.5/0.99/1.0) on a stream with domain shifts at t=10/15/19. As shown in Fig. 5, our method maintains or improves performance under shifts, while fixed-weight baselines drop sharply and recover slowly. This advantage comes from CDA's dynamic cross-domain accumulation, which balances domain-specific and domain-general semantic information, whereas fixed weights fail to adapt and cause instability.

5 Conclusion

We propose CTTA-T, a novel CTTA framework for text understanding. CTTA-T adopts a teacher-student architecture to support continuous test-time adaptation. A refine-then-filter module based on prediction consistency improves prediction reliability, while a domain-aware teacher dynamically accumulates cross-domain semantic knowledge and is partially restored to its source state to enhance generalization. Moreover, we introduce a benchmark for CTTA in text understanding. Experiments show that CTTA-T achieves strong performance.

Limitations

Although CTTA-T shows strong and stable performance across diverse continual domain shift settings, there remain several natural limitations. First, our benchmark primarily covers QA, reading comprehension, and cross-lingual QA; extending CTTA to other text understanding tasks (e.g., dialogue or code comprehension) is left for future work.

While we theoretically establish key properties (e.g., prediction consistency bound and domain-aware update rule), more general theoretical guarantees under arbitrary domain shifts remain open.

Importantly, these limitations do not undermine the practical effectiveness of our framework but instead suggest fruitful avenues for future research.

References

Mikel Artetxe, Sebastian Ruder, Dani Yogatama, and Liu. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637.

Maurice Stevenson Bartlett. 1937. Properties of sufficiency and statistical tests. *Proceedings of the Royal Society of London. Series A-Mathematical and Physical Sciences*, 160(901):268–282.

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

Yu Cao, Meng Fang, Baosheng Yu, and Joey Tianyi Zhou. 2020. Unsupervised domain adaptation on reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7480–7487.

Chaoqi Chen, Weiping Xie, Wenbing Huang, Yu Rong, Xinghao Ding, Yue Huang, Tingyang Xu, and Junzhou Huang. 2019. Progressive feature alignment for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 627–636.

Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. 2022. Contrastive test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 295–305.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019. Mrqa 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 1–13.

Taesik Gong, Yewon Kim, Taeckyung Lee, Sorn Chottananurak, and Sung-Ju Lee. 2023. Sotta: Robust test-time adaptation on noisy data streams. *Advances in Neural Information Processing Systems*, 36:14070–14093.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR.

Jisu Han, Jaemin Na, and Wonjun Hwang. 2025. Ranked entropy minimization for continual test-time adaptation. In *Forty-second International Conference on Machine Learning*.

Moritz Hardt and Yu Sun. 2024. Test-time training on nearest neighbors for large language models. In *The Twelfth International Conference on Learning Representations*.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International conference on machine learning*, pages 4411–4421. PMLR.

Minghao Hu, Yuxing Peng, Zhen Huang, Xipeng Qiu, Furu Wei, and Ming Zhou. 2018. Reinforced mnemonic reader for machine reading comprehension. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 4099–4106. International Joint Conferences on Artificial Intelligence Organization.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Hwang, and Jong Park. 2023. Test-time self-adaptive small language models for question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15459–15469, Singapore. Association for Computational Linguistics.

Jincen Jiang, Qianyu Zhou, Yuhang Li, Xinkui Zhao, Meili Wang, Lizhuang Ma, Jian Chang, Jian Zhang, Xuequan Lu, and 1 others. 2024. Pcotta: Continual test-time adaptation for multi-task point cloud understanding. *Advances in Neural Information Processing Systems*, 37:96229–96253.

Nazmul Karim, Niluthpol Chowdhury Mithun, Abhinav Rajvanshi, Han-pang Chiu, Supun Samarasekera, and Nazanin Rahnavard. 2023. C-sfda: A curriculum learning aided self-training framework for efficient source free domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24120–24131.

Alex Kendall and Yarin Gal. 2017. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30.

Anant Khandelwal. 2024. Domaininv: Domain invariant fine tuning and adversarial label correction for unsupervised qa domain adaptation. In *Proceedings of the 9th Workshop on Representation Learning for NLP (RepL4NLP-2024)*, pages 13–25.

Dong-Hyun Lee and 1 others. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta.

Jonghyun Lee, Dahuin Jung, Saehyung Lee, Junsung Park, Juhyeon Shin, Uiwon Hwang, and Sungroh Yoon. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. In *The Twelfth International Conference on Learning Representations*.

Geyu Lin, Bin Wang, Zhengyuan Liu, and Nancy F. Chen. 2025. CrossIn: An efficient instruction tuning approach for cross-lingual knowledge alignment. In *Proceedings of the Second Workshop on Scaling Up Multilingual & Multi-Cultural Evaluation*, pages 12–23, Abu Dhabi. Association for Computational Linguistics.

Y Liu. 2000. Statistical behavior of the eigenvalues of random matrices. In *Proceedings of the Mathematical Junior Seminar*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Fan Lyu, Kaile Du, Yuyang Li, Hanyu Zhao, Zhang Zhang, Guangcan Liu, and Liang Wang. 2024. Variational continual test-time adaptation. *arXiv preprint arXiv:2402.08182*.

Shuaicheng Niu, Chunyan Miao, Guohao Chen, Pengcheng Wu, and Peilin Zhao. 2024. Test-time model adaptation with only forward passes. In *International Conference on Machine Learning*, pages 38298–38315. PMLR.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. 2023. Towards stable test-time adaptation in dynamic wild world. In *The Eleventh International Conference on Learning Representations*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Abhilasha Ravichander, Siddharth Dalmia, Maria Ryskina, Florian Metze, Eduard Hovy, and Alan W Black. 2021. NoiseQA: Challenge set evaluation for user-centric question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2976–2992, Online. Association for Computational Linguistics.

David A. Ross, Jongwoo Lim, Ruei Sung Lin, and Ming Hsuan Yang. 2008. Incremental learning for robust visual tracking. *International Journal of Computer Vision*, 77(1-3):125–141.

Yi Su, Yixin Ji, Juntao Li, Hai Ye, and Min Zhang. 2023. Beware of model collapse! fast and stable test-time adaptation for robust question answering. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12998–13011, Singapore. Association for Computational Linguistics.

Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. 2021. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*.

Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. 2022. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7201–7211.

Yanshuo Wang, Jie Hong, Ali Cheraghian, Shafin Rahman, David Ahmedt-Aristizabal, Lars Petersson, and Mehrtash Harandi. 2024. Continual test-time domain adaptation via dynamic sample selection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1701–1710.

Haoyu Xiong, Xinchun Zhang, Leixin Yang, Yu Xiang, and Gang Fang. 2024. Staf: Pushing the boundaries of test-time adaptation towards practical noise scenarios. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15226–15237.

Hai Ye, Yuyang Ding, Juntao Li, and Hwee Tou Ng. 2022. Robust question answering against distribution shifts with test-time adaption: An empirical study. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6179–6192.

Zhenrui Yue, Huimin Zeng, Ziyi Kou, Lanyu Shang, and Dong Wang. 2022. Contrastive domain adaptation for early misinformation detection: A case study on covid-19. In *Proceedings of the 31st ACM international conference on information & knowledge management*, pages 2423–2433.

Bo Zheng, Li Dong, Shaohan Huang, Wenhui Wang, Zewen Chi, Saksham Singhal, Wanxiang Che, Ting Liu, Xia Song, and Furu Wei. 2021. Consistency regularization for cross-lingual fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3403–3417.

A Problem Definition

Let the data distribution of the source domain be denoted as P_s, and the data distribution of the target domain vary over time as P_t, where t indicates the time step. Given an initial model θ_0, it is pre-trained on the source domain dataset $\{X^s, Y^s\} \sim P_s$. At each time step t, the model receives an unlabeled test sample $x_t^T \in X_t^T$ from the target domain and produces a prediction $y_t^T \in Y_t^T$, where $\{X_t^T, Y_t^T\} \sim P_t$. The model then immediately updates its parameters θ_t using the pseudo-label y_t^T. It is important to note that during the adaptation process, the model has no access to the source training data X^s, past test samples $X_{<t}^T$, or future test samples $X_{>t}^T$; it can only utilize the current test instance X_t^T for online adaptation. Our objective is to enable the model to continuously adapt to the evolving target distribution during test time, thereby enhancing its prediction performance in the target domain. The overall procedure of the CTTA setting is illustrated in Fig. 6.

B Theoretical derivation of the CDA module

B.1 The derivation process of IPCA

Let the input embedding at time step t be denoted by $X_t \in \mathbb{R}^{n \times d}$, where n represents the number of samples and d is the embedding dimension. The symmetric covariance matrix C_t and the mean μ_t of the historical data are defined as

$$\begin{aligned} C_t &= \frac{1}{N_t} \sum_{i=1}^{t} (X_i - \mu_t)^\top (X_i - \mu_t), \\ \mu_t &= \frac{1}{N_t} \sum_{i=1}^{t} \sum_{j=1}^{n} X_i^{(j)}. \end{aligned} \tag{10}$$

where N_t denotes the number of historical samples. SVD is applied to $C_t = V_t \Lambda_t V_t^\mathsf{T}$, extracting the principal component space $V_t \in \mathbb{R}^{d \times k}$. When the new samples X_{t+1} arrives, the statistics are recursively updated using IPCA:

$$\begin{aligned} C_{t+1} &\approx \frac{N_t C_t + n C_{\text{new}}}{N_t + n} \\ &\quad + \frac{N_t n}{N_t + n} \frac{(\mu_t - \mu_{\text{new}})(\mu_t - \mu_{\text{new}})^\top}{N_t + n}, \\ \mu_{t+1} &= \frac{N_t \mu_t + n \mu_{\text{new}}}{N_t + n}. \end{aligned} \tag{11}$$

where $C_{\text{new}} = \frac{1}{n}(X_{t+1} - \mu_{\text{new}})^\top (X_{t+1} - \mu_{\text{new}})$ denotes the covariance matrix after the new sample X_{t+1} is added. $\mu_{\text{new}} = \frac{1}{n} \sum_{j=1}^{n} X_{t+1}^{(j)}$ denotes the mean of the embedding layer after the new sample X_{t+1} is added.

B.2 Proof of Domain Semantic Distance between Historical and Current Domain

Let $\Delta V = V_{t+1} - V_t$ denote the change in the principal directions after incorporating a new batch of data into the principal component space. The principal component space of the historical data are denoted as $V = [v_{t,1}, v_{t,2}, \ldots, v_{t,i}, \ldots, v_{t,k}]$, while after adding the new data, the updated principal component space are given by $V_{t+1} = [v_{t+1,1}, v_{t+1,2}, \ldots, v_{t+1,i}, \ldots, v_{t+1,k}]$, where $v_{t+1,i}$ represents the i-th principal component direction. We define the principal component similarity as the average cosine similarity between the old and new principal component spaces:

$$similarity = \frac{1}{k} \sum_{i=1}^{k} \cos\theta_i = \frac{1}{k} \sum_{i=1}^{k} (v_{t,i}^\top \cdot v_{t+1,i}), \tag{12}$$

where θ_i denotes the angle between $v_{t,i}$ and $v_{t+1,i}$. Since both $v_{t,i}$ and $v_{t+1,i}$ are unit vectors, the cosine similarity can be expressed as $\cos\theta_i = v_{t,i}^\mathsf{T} \cdot v_{t+1,i}$. Given that the new batch is small relative to the historical data, the change in principal directions is slight, implying that θ_i is also small. Thus, using a Taylor expansion, we

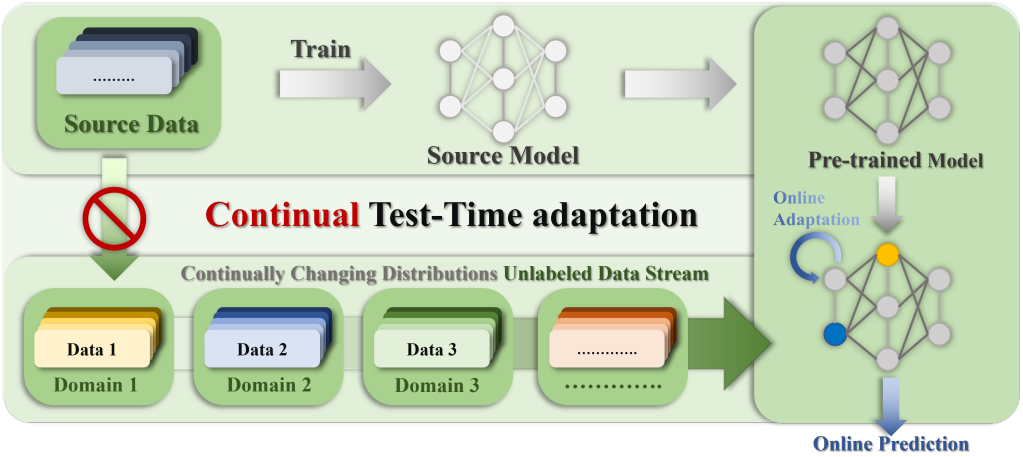


Figure 6: Illustrating the setting of Continual Test-Time Adaptation (CTTA). To better reflect real-world environment, the target test data arrives sequentially, with its domain shifting continuously over time. At each time step, the model only accesses the current data. A model pre-trained on the source domain is used to perform online prediction and adaptation to the current target data, without access to any source domain data.

have the approximation

$$\cos\theta_i \approx 1 - \frac{\theta_i^2}{2} \tag{13}$$

Let Δv_i denote the perturbation of the i-th principal component direction, so that $v_{t+1,i} = v_{t,i} + \Delta v_i$. Since both $v_{t,i}$ and $v_{t+1,i}$ are unit vectors, it follows that

$$\begin{aligned} \|v_{t+1,i}\|^2 = 1 = \|v_{t,i} + \Delta v_i\|^2 \\ = 1 + 2v_{t,i}^\top \Delta v_i + \|\Delta v_i\|^2. \end{aligned} \tag{14}$$

which leads to $v_{t,i}^\top \Delta v_i = -\frac{\|\Delta v_i\|^2}{2}$. The cosine similarity between $v_{t,i}$ and $v_{t+1,i}$ can thus be expanded as

$$\cos\theta_i = v_{t,i}^\top \cdot v_{t+1,i} = v_{t,i}^\top (v_{t,i} + \Delta v_i) = 1 + v_{t,i}^\top \Delta v_i, \tag{15}$$

hence:

$$\begin{aligned} 1 + v_{t,i}^\top \Delta v_i = 1 - \frac{\theta_i^2}{2} \Longrightarrow \\ v_{t,i}^\top \Delta v_i = -\frac{\|\Delta v_i\|^2}{2} \approx -\frac{\theta_i^2}{2}. \end{aligned} \tag{16}$$

Consequently, the angle θ_i can be approximated as $\theta_i \approx \| \Delta v_i \|$. By substituting this approximation into the Taylor expansion of the cosine similarity, we obtain:

$$\cos\theta_i \approx 1 - \frac{\| \Delta v_i \|^2}{2}. \tag{17}$$

Substituting this expression into the definition of principal component similarity yields:

$$similarity = \frac{1}{k}\sum_{i=1}^{k} \cos\theta_i \tag{18}$$

$$\approx \frac{1}{k}\sum_{i=1}^{k} \left(1 - \frac{\| \Delta v_i \|^2}{2}\right) \tag{19}$$

$$= 1 - \frac{1}{2k}\sum_{i=1}^{k} \| \Delta v_i \|^2 . \tag{20}$$

Let $\Delta V = [\Delta v_1, \Delta v_2, \ldots, \Delta v_k]$ denote the perturbation matrix, whose Frobenius norm satisfies

$$\| \Delta V \|_F^2 = \sum_{i=1}^{k} \| \Delta v_i \|^2, \tag{21}$$

where $\| \cdot \|_F$ denotes the Frobenius norm. Accordingly, the principal component similarity can be approximated as:

$$similarity \approx 1 - \frac{1}{2k} \| \Delta V \|_F^2 . \tag{22}$$

To intuitively capture domain shifts, we use the distance as a proxy for domain similarity and simplify Eq. 22:

$$distance = 2k \cdot (1 - similarity) = \| \Delta V \|_F^2 \tag{23}$$

B.3 Proof for the CDA Module's Sensitivity to Minor Domain Shifts

Lemma 1. *ΔV lies in the orthogonal complement of V_t, it sensitively captures subtle shifts, enabling accurate quantification.*

Proof. Since both V_t and V_{t+1} are orthogonal matrices, they satisfy the orthogonality condition $V_{t+1}^{\mathrm{T}} V_{t+1} = I$, where I denotes the identity matrix. By substituting $V_{t+1} = V_t + \Delta V$ into the orthogonality condition, we obtain:

$$\begin{aligned} (V_t + \Delta V)^\top (V_t + \Delta V) = I \\ V_t{}^\top V_t + V_t{}^\top \Delta V + \Delta V^\top V_t + \Delta V^\top \Delta V = I. \end{aligned} \tag{24}$$

Given that $V_t^{\mathrm{T}} V_t = I$ and neglecting the second-order infinitesimal term $\Delta V^\top \Delta V \approx 0$, it follows that: $V_t^{\mathrm{T}} \Delta V + \Delta V^{\mathrm{T}} V_t = 0$. Rearranging the terms, we conclude that ΔV resides in the orthogonal complement space of V.

As a result, the variation in principal component direction, represented by ΔV, is local and confined to directions orthogonal to the current principal component subspace, thereby enabling the model to sensitively capture subtle changes in the data distribution.

B.4 Derivation of the Relationship between CDA Weight α and Domain Semantic Distance

We construct the following regularization function $\mathcal{F}$:

$$\begin{aligned} \mathcal{F}(\theta_{t+1}^T) = &\parallel \Delta V \parallel_F^2 \ (\theta_{t+1}^T - \theta_{t+1}^S)^2 \\ &+ (1 - \parallel \Delta V \parallel_F^2) \, (\theta_{t+1}^T - \theta_t^T)^2 \end{aligned} \tag{25}$$

By differentiating $\mathcal{F}$ concerning θ_{t+1}^T and setting the derivative to zero, we obtain the optimal update direction for the teacher parameters:

$$\begin{aligned} \frac{\partial \mathcal{F}}{\partial \theta_{t+1}^T} &= 2 \ \parallel \Delta V \parallel_F^2 \ (\theta_{t+1}^T - \theta_{t+1}^S) \\ &\quad + 2\,(1 - \parallel \Delta V \parallel_F^2)(\theta_{t+1}^T - \theta_t^T) \\ &= 0 \end{aligned} \tag{26}$$

Rearranging the terms, we have:

$$\begin{aligned} \theta_{t+1}^T &= \frac{(1 - \parallel \Delta V \parallel_F^2)\theta_t^T + \parallel \Delta V \parallel_F^2 \ \theta_{t+1}^S}{(1 - \parallel \Delta V \parallel_F^2) + \parallel \Delta V \parallel_F^2} \\ &= (1 - \parallel \Delta V \parallel_F^2)\theta_t^T + \parallel \Delta V \parallel_F^2 \ \theta_{t+1}^S. \end{aligned} \tag{27}$$

Meanwhile, the standard EMA update formula is given by:

$$\theta_{t+1}^T = \alpha \theta_t^T + (1 - \alpha)\theta_{t+1}^S. \tag{28}$$

By applying the method of undetermined coefficients, it follows that:

$$\alpha = 1 - \parallel \Delta V \parallel_F^2 = 1 - distance. \tag{29}$$

Since α varies with time step t, it is appropriate to define α as α_t. Due to the model's high sensitivity to parameter changes, we map α_t to the range $[\alpha_{\text{bound}}, 1]$.

C Theoretical derivation of the RFP module

C.1 The Proof of the Upper Bound of the Consistency Score

Given a classification model with C output classes and a stochastic dropout mechanism, we denote by $p_i \in \mathbb{R}^C$ the predicted softmax probability vector of a single input sample after the i-th stochastic forward pass (with dropout enabled). Repeating this process for N stochastic passes yields a prediction consistency matrix P (The main text uses P_C):

$$P = \begin{bmatrix} p_1^\top \\ p_2^\top \\ \vdots \\ p_N^\top \end{bmatrix} \in \mathbb{R}^{N \times C}. \tag{30}$$

Each row of P corresponds to the model's probabilistic prediction under a specific dropout mask, and collectively, P encodes the model's belief distribution consistency under randomness.

We focus on high-noise or ambiguous samples, which are typically difficult to classify due to a lack of discriminative information. Models tend to produce less confident and more uniform predictions for such inputs. For a high-noise sample, each row vector $p_i \in \mathbb{R}^C$ in P satisfies: $\forall i \in \{1, \ldots, N\}$, $\mathbf{p}_i \approx \mathbf{u}_C$, where $\mathbf{u}_C = \left[\frac{1}{C}, \ldots, \frac{1}{C}\right] \in \mathbb{R}^C$ denotes the uniform distribution over classes.

We perform singular value decomposition (SVD) on P:

$$P = U \Sigma V^\top, \tag{31}$$

where $\mathbf{U} \in \mathbb{R}^{N \times N}$ is the matrix of left singular vectors, $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r) \in \mathbb{R}^{N \times C}$ where $U \in \mathbb{R}^{N \times N}$ is the diagonal matrix of singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$, $V \in \mathbb{R}^{C \times C}$ is the matrix of right singular vectors, which corresponds to directions in the class space. Since $V^\top$ spans the class space, it captures how predictions distribute and align along different class directions. Specifically, each row $v_j \in \mathbb{R}^C \ in \ V$ represents a class-wise projection of a principal direction in the model's output distribution.

We define a class-level prediction consistency vector $\vec{s} \in \mathbb{R}^C$, where each element s_j reflects

Table 4: Task sequences (Orders 1–6) in the CTTA benchmark. Orders 1–3 are short sequences comprising 7 tasks, and Orders 4–6 are long sequences comprising 11 tasks.

Order	Task Sequence
1	syn.Keyboard → syn.ASR → syn.Translation → Search → Zh → Ar → Es
2	syn.ASR → Es → Zh → syn.Keyboard → Search → syn.Translation → Ar
3	Es → syn.ASR → syn.Keyboard → Zh → syn.Translation → Search → Ar
4	syn.Keyboard → syn.ASR → syn.Translation → na.Keyboard → na.ASR → na.Translation → Search → Trivia → Zh → Ar → Es
5	Zh → na.Keyboard → Ar → syn.ASR → Trivia → na.ASR → Es → na.Translation → Search → syn.Keyboard → syn.Translation
6	syn.Translation → Zh → Trivia → Ar → na.ASR → Search → na.Translation → syn.Keyboard → syn.ASR → Es → na.Keyboard

how strongly the principal components align with class j. The score is computed as a weighted sum over principal components projected onto the class j:

$$\mathbf{s}_j = \sum_{k=1}^{r} \sigma_k \cdot v_{k,j}, \tag{32}$$

where $v_{k,j}$ is the j-th entry of the k-th right singular vector (i.e., the projection of the k-th component of the class j), σ_k is the singular value corresponding to the k-th component, which serves as its importance weight. A peaky consistency score $\mathbf{s}_j$ (i.e., one class dominates) indicates that the principal directions of the model's predictions are aligned toward a particular class, reflecting high agreement across dropout samples. A flat $\mathbf{s}_j$ suggests that the model's predictions fluctuate in multiple directions and no class dominates, indicating low consistency and potentially high noise.

We define a consistency score for each test sample:

$$\mathbf{s}_{\max} = \max_{j \in \{1,\ldots,C\}} \mathbf{s}_j, \tag{33}$$

We set a filtering threshold τ and **discard samples whose $\mathbf{s}_{\max} < \tau$**. Below, we provide theoretical justification for this filtering rule by proving that noisy samples are likely to exhibit lower $\mathbf{s}_{\max}$ values.

Define the **average prediction** vector:

$$\bar{p} = \frac{1}{N} \sum_{i=1}^{N} p_i. \tag{34}$$

Then, the consistency score satisfies:

$$\mathbf{s}_{\max} \leq \max_j \bar{p}_j + \epsilon, \tag{35}$$

where ϵ is a function of the variance among the p_i and alignment between singular vectors and class axes. When p_i is stable and aligned (low-noise), singular vectors align with dominant class j^*, and softmax amplifies this; when $\mathbf{p}_i$ is diverse (high-noise), $\bar{p}$ becomes flat and singular directions point to mixed class subspaces; this reduces the $\vec{s}_j$ value for any j, leading to lower $\mathbf{s}_{\max}$. In conclusion, when the sample x_t is dominated by data noise, the consistency matrix P exhibits random behavior and causes a drop in $\mathbf{s}_{\max}$. Therefore, setting a threshold τ to filter out the lower $\mathbf{s}_{\max}$ enables effective masking of samples with high AU.

D Implementation Details and Metrics

Implementation Details. In our main experiments, we employ both XLM-RoBERTa-Base (Conneau et al., 2020) and XLMR-Base-xTune as backbone models, following configurations used in prior studies(Ye et al., 2022)(Su et al., 2023). The latter uses xTune(Zheng et al., 2021), a strong robustness tuning method, to train a source model on XLM-RoBERTa-base. All backbone models are pretrained on the SQuAD(Rajpurkar et al., 2016), which serves as the source domain, using the default configuration provided by XTREME (Hu et al., 2020). During the CTTA phase, target test samples are sequentially fed into the model following the designed input order. At each time step t, the model is exposed solely to the current target test data; neither past nor future data is accessible. Each input sample is first evaluated and then immediately used for adaptation, with the student model's prediction adopted as the final prediction. Our method is implemented using PyTorch, and all experiments are conducted on an NVIDIA N800 GPU. We set the batch size to 16 and utilize the AdamW optimizer (Loshchilov and Hutter, 2019). The learning rate is uniformly set to 1e-5 across all methods. For our approach, the hyperparameter γ is set to 0.4; τ is set to 1.2 when using XLMR-Base and 0.6 when using XLMR-Base-xTune. Baseline hyperparameters follow the default hyperparameters reported in the original works. All experiments are repeated three times with different random seeds, and the average results are reported. Importantly, as there is no existing work on CTTA in NLP, all baseline

methods are adapted from closely related TTA approaches. Specifically, Tent and OIL—originally designed for test-time adaptation on a fixed target domain—are modified to operate on the long and short sequence target task streams (Order 1 to 6) defined above. For CoTTA, which was originally developed for CTTA in computer vision, we adapt the task from image classification to text understanding and replace the image-based augmentation with text-based augmentation, such as random token replacement and deletion.

Metrics. *EM* follows a strict criterion: a prediction is considered correct if it exactly matches the ground truth (i.e., a character-level exact match); otherwise, it is deemed incorrect. *F1* score, being the harmonic mean of Precision and Recall, offers a balanced measure of the model's performance. EM emphasizes exact correspondence between the predicted answer and the reference answer, where a prediction is considered correct only if it matches the ground truth at the character level. This reflects the model's ability to achieve strict semantic alignment. In contrast, the F1 captures partial overlaps between the prediction and the reference. Even if the prediction is not entirely correct, it can still receive credit as long as it contains key information, making F1 more tolerant to minor discrepancies.

E Tasks and CTTA Benchmark

We establish a CTTA benchmark for text understanding, covering multiple domains and adaptation orders. The benchmark includes three tasks: We establish a CTTA benchmark for text understanding across multiple domains and adaptation orders, covering three tasks: Robust QA (Ravichander et al., 2021), Reading Comprehension (Fisch et al., 2019), and Cross-lingual QA (Artetxe et al., 2020). The Robust QA task has two subsets (natural and synthetic noise), each with three subtasks (speech recognition, keyboard, and translation errors). The Reading Comprehension task includes Search and Trivia subtasks, while Cross-lingual QA includes translations into Chinese, Arabic, and Spanish. Each subtask is treated independently, resulting in 11 tasks. We construct short-sequence (7 tasks) and long-sequence (11 tasks) target task streams to evaluate performance under CTTA. To construct long and short sequence target task streams, we use the following method: For the long-sequence setting (Orders 4 to 6), we organize the three task categories—Robust QA, Reading Comprehension, and Cross-lingual QA—into a fixed order (Order 4), numbering tasks 1 to 11. We then apply the Fisher-Yates shuffle (Liu, 2000) twice to generate two randomized long-sequence streams (Orders 5 and 6). The short-sequence stream (Order 1) and its two randomized variants (Orders 2 and 3) are constructed similarly, excluding Robust QA-na and the Trivia subtask from Reading Comprehension. Details of Orders 1 to 6 are provided in Tab. 4.

F Further Analysis Study of Error Accumulation

To further analyze error accumulation, we compare our method with baselines on both short and long task sequences, with an emphasis on hard tasks (e.g., Search and Trivial) where accumulated errors are most likely to dominate. Specifically, the short sequence includes Search, while the long sequence includes both Search and Trivial. As shown in Fig. 7, we report the EM variations of each method under continuous domain shifts from Order 1 to Order 6 (excluding Order 5).

All baselines exhibited severe error accumulation on hard tasks, whereas our method remained stable or showed only minimal degradation, highlighting its robustness in CTTA. OIL, SAR, SoTTA, and REM suffered from progressive error accumulation due to noisy and degrading pseudo-labels, which irreversibly propagated over time. Tent, by involving all samples in its entropy minimization, rapidly degenerated into a trivial solution where outputs were biased toward specific classes, leading to the earliest failure. CoTTA deteriorated even earlier in Order 2, as its recovery strategy overwrote sensitive parameters, thereby accelerating error accumulation.

G Analysis Study of Backbone Sensitivity

Since the teacher model is initialized from a pretrained backbone, its quality may influence the overall CTTA performance. To study this factor, we evaluate different methods under two types of backbone initialization: a standard pretrained model (base) and a robustness-enhanced model (xTune). The results are summarized in Tab. 1. We observe that our method achieves consistently stable performance across both backbones. In contrast, other CTTA baselines perform unstably when using the weaker `base` backbone, and their results become relatively more stable with `xTune`, but still lag be-

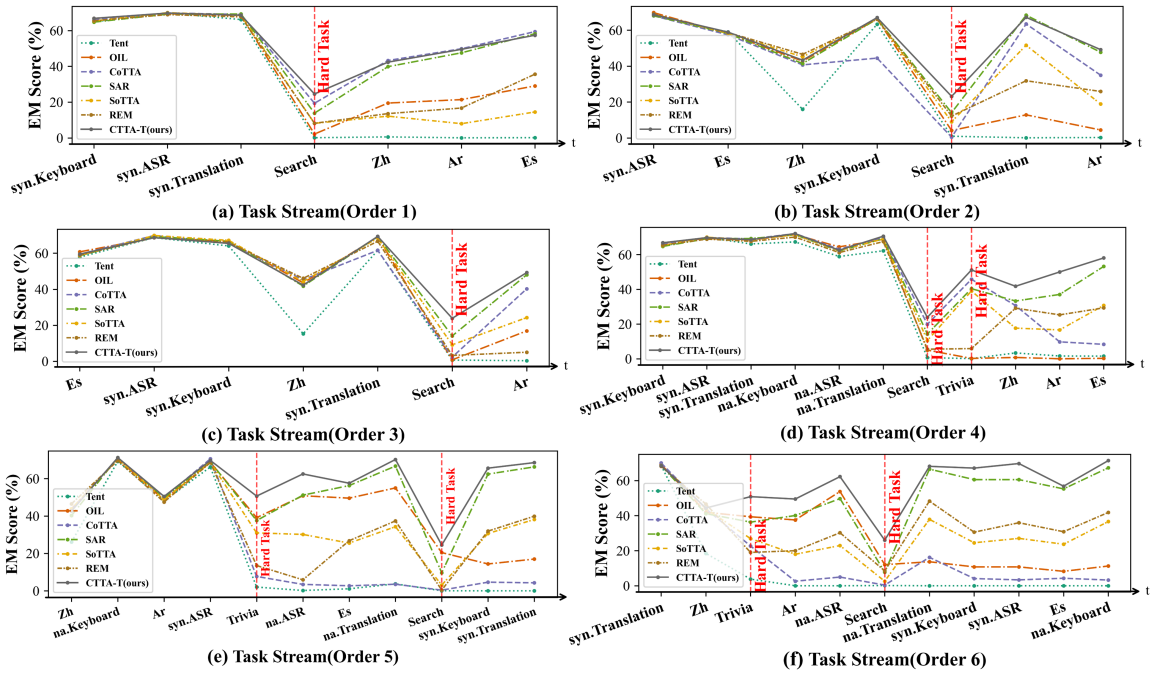


Figure 7: Further Analysis Study of Model Collapse. The EM variations of each method under continuous domain shifts from Order 1 to Order 6 (excluding order 5)

hind our method. This demonstrates that a stronger backbone indeed improves the quality of pseudo-labels and benefits all approaches, but our method is less sensitive to the initial teacher quality and remains reliable even with a weaker backbone.

H Analysis Study of the Stability of each method

To further assess stability under different adaptation orders, we report the variance of EM and F1 in Table 5. The variance is computed within each short-sequence (Orders 1–3) and long-sequence (Orders 4–6) setting, reflecting the sensitivity of each method to order variations. A smaller variance indicates more consistent performance across different orders. Our method (CTTA-T) consistently achieves the lowest variance across both metrics and sequence types, while baseline methods often display substantially larger variances. These results confirm that our method maintains stable adaptation behavior and is less affected by the sequence order, thereby validating its robustness in CTTA.

I Ablation Study of Effect of Each Component

In Tab. 6, we assess the contribution of each component by removing it from the full model. Replacing the CDA module (Sec. 3.4) with a fixed EMA weight ($\alpha = 0.99$) leads to a performance drop, indicating that dynamically adjusting EMA weights in response to domain shift enhances adaptation in CTTA settings. Similarly, removing the RFP module results in degraded performance, suggesting that the RFP module improves pseudo-label quality and effectively filters noisy samples. Notably, removing the SRT module causes the most substantial decline in performance, demonstrating its crucial role in breaking the chain of noise-induced erroneous updates and improving generalization.

J Strong Baseline Comparison

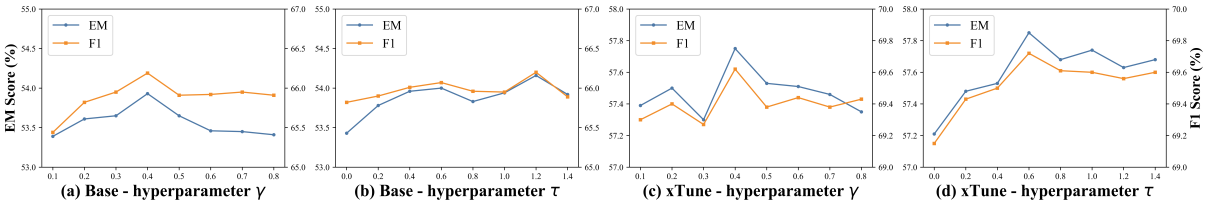
To further evaluate the adaptability of our method under the CTTA setting, we design a rigorous comparative experiment by selecting Anti-CF (Su et al., 2023), a state-of-the-art method for test-time adaptation (TTA) in NLP, as a strong baseline. Anti-CF regularizes the adaptation process using outputs from the source model, which effectively mitigates the issue of model collapse commonly observed in TTA. However, it relies on a critical assumption: the availability of source training data during testing to warm up a side network—this fundamentally violates the CTTA principle of source-free test-time adaptation. Despite this violation, we include Anti-CF in our comparison to provide a comprehensive assessment of our method. As shown in Tab. 7, our method surpasses Anti-CF with an

Table 5: Variance across long and short sequences. We report the variance of three adaptation orders within each sequence, calculated on a per-sequence basis. **Bold**: the smallest variance.

Short Seq														
	CTTA-T (ours)		REM		SoTTA		SAR		CoTTA		OIL		Tent	
Method	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**
base	**0.017**	**0.081**	6.004	6.954	0.087	1.894	33.396	38.172	16.804	13.398	16.804	37.931	18.497	48.961
xtune	**0.005**	**0.001**	23.050	18.332	0.100	0.106	38.785	37.577	0.537	0.860	16.050	12.628	8.186	10.392
Long Seq														
	CTTA-T (ours)		REM		SoTTA		SAR		CoTTA		OIL		Tent	
Method	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**	**V(EM)**	**V(F1)**
base	**0.003**	**0.003**	0.761	0.530	1.478	1.336	48.946	45.872	181.460	224.987	40.462	56.005	129.879	164.758
xtune	**0.002**	**0.012**	7.361	4.324	1.157	0.862	15.319	16.303	0.365	0.312	26.798	40.776	179.894	245.460

Table 6: Ablation study on the short and long sequence benchmarks. w/o CDA, w/o RFP, and w/o SRT indicate the removal of the corresponding modules from our framework. **Bold**: the best results.

	Short Sequence						Long Sequence							
Order	**Order1 (t⟶)**		**Order2 (t⟶)**		**Order3 (t⟶)**		**Order4 (t⟶)**		**Order5 (t⟶)**		**Order6 (t⟶)**		**Avg**	
base	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
CTTA-T (ours)	**54.14**	**66.57**	**53.88**	**66.01**	**54.07**	**66.12**	**57.74**	**69.33**	**57.63**	**69.33**	**57.71**	**69.24**	**55.86**	**67.77**
w/o CDA	53.51	65.83	53.40	65.55	53.14	65.40	57.30	68.86	57.43	68.95	57.36	68.83	55.36	67.24
w/o RFP	53.66	65.94	53.69	66.12	53.72	65.97	57.50	68.09	57.45	68.94	57.30	68.84	55.55	67.32
w/o SRT	48.12	59.35	46.37	57.29	51.22	62.78	48.71	58.74	51.90	63.17	39.46	49.76	47.63	58.52
xTune	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
CTTA-T (ours)	**57.85**	**69.72**	**57.99**	**69.76**	**58.00**	**69.80**	**60.84**	**72.08**	**60.92**	**72.18**	**60.86**	**71.96**	**59.41**	**70.92**
w/o CDA	57.53	69.60	57.69	69.59	57.92	69.59	60.33	71.50	60.88	71.92	60.58	71.72	59.16	70.65
w/o RFP	57.42	69.41	57.42	69.49	57.65	69.46	60.52	71.75	60.74	71.83	60.58	71.80	59.06	70.62
w/o SRT	55.12	66.58	55.45	66.85	57.22	68.85	55.36	65.71	57.46	68.49	53.68	64.58	55.72	66.84



(a) Base - hyperparameter γ (b) Base - hyperparameter τ (c) xTune - hyperparameter γ (d) xTune - hyperparameter τ

Figure 8: Effect of different hyperparameters γ and τ in the RFP module.

Table 7: Performance of our method and a strong baseline, Anti-CF, on Orders 1 - 6. **Bold**: the best results.

	Short Sequence						Long Sequence							
Order	**Order1 (t⟶)**		**Order2 (t⟶)**		**Order3 (t⟶)**		**Order4 (t⟶)**		**Order5 (t⟶)**		**Order6 (t⟶)**		**Avg**	
Methods	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
base													54.23	66.86
Anti-CF	52.83	65.57	53.35	65.69	53.23	65.64	56.47	68.50	56.22	68.37	56.18	68.32	54.71	67.02
CTTA-T (ours)	**54.14**	**66.57**	**53.88**	**66.01**	**54.07**	**66.12**	**57.74**	**69.33**	**57.63**	**69.33**	**57.71**	**69.24**	**55.86**	**67.77**
xTune													58.45	70.21
Anti-CF	56.47	68.94	56.82	68.96	56.67	68.95	59.73	71.49	59.47	71.19	59.54	71.11	58.12	70.11
CTTA-T (ours)	**57.85**	**69.72**	**57.99**	**69.76**	**58.00**	**69.80**	**60.84**	**72.08**	**60.92**	**72.18**	**60.86**	**71.96**	**59.41**	**70.92**

Table 8: The p-values of the t-test on our method.

Order	**Order1**		**Order2**		**Order3**		**Order4**		**Order5**		**Order6**	
base	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
CTTA-T	0.0001	0.0012	0.0002	0.0005	1.80e-5	0.0020	9.48e-5	0.0003	6.68e-8	0.0013	0.0003	0.0008
xtune	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
CTTA-T	7.08e-5	4.41e-6	2.34e-7	4.16e-5	2.78e-6	2.09e-5	0.0001	9.89e-5	6.74e-7	1.16e-5	1.27e-6	0.0006

average improvement of 1.22% in EM and 0.78% in F1. Notably, when using xTune as the backbone model, Anti-CF even underperforms vanilla inference. This observation highlights the intrinsic difficulty of achieving stable adaptation under continually shifting domains, further demonstrating the robustness and practical value of our CTTA method.

K Hyperparameter analysis

We conduct a hyperparameter search for γ_c and τ in the RFP module (Sec. 3.3) using Order. The threshold used in the main text is γ_c, and $\gamma_c = \gamma \log C$. Therefore, we only analyze γ. Since γ and τ are interdependent, jointly searching them incurs a computational complexity of $\mathcal{O}(n^2)$. To address this, we first simplify the second-stage filtering in the RFP module. Specifically, instead of applying the `max` operation (i.e., $\max_j \pi_j = \pi_{\text{max}} < \gamma$), we sort $\pi(y^T)$ in descending order and compute the average of the top-k values π_{avg} that account for 30% of the distribution mass. Samples with $\pi_{\text{avg}} < 0.2$ are filtered out. This modification approximates the model's consistency scoring across all samples as being uniform, effectively decoupling the influence of τ and enabling the isolated searching of γ. After selecting an appropriate γ, we restore the original second-stage filtering of the RFP module and proceed with τ tuning. Fig. 8 (a) and (b) present the EM and F1 scores results for different values of γ and τ using the base backbone, and the optimal configuration is $\gamma = 0.4$ and $\tau = 1.2$. Similarly, Fig. 8 (c) and (d) show the results for the xTune backbone, with the selected hyperparameters being $\gamma = 0.4$ and $\tau = 0.6$.

L Significance Test

To assess whether the performance improvements achieved by our method are statistically significant across various evaluation metrics, we perform t-tests (Bartlett, 1937) on the main experimental results. This test evaluates whether the differences in means between our method and baseline approaches are significant across different task orders. As shown in Tab. 8, the computed p-values for all orders are below 0.01, indicating that the observed performances are consistent and statistically significant rather than due to random chance.

M Societal Impacts

The proposed CTTA-T framework addresses a critical challenge in real-world natural language processing systems: continual domain shift. By enabling the model to adapt continuously to changing test distributions, our method enhances the robustness and long-term viability of text understanding systems across various scenarios. These advancements could benefit users in multilingual, underrepresented, or rapidly changing environments, reducing performance degradation and decreasing dependence on expensive retraining or annotated data. However, adaptive models that update without explicit human control may become challenging to interpret and audit. While our method provides a solid foundation for continuous test-time adaptation in test understanding tasks, it is crucial to consider responsible and ethical usage policies in broader applications, particularly in areas involving sensitive data or high-risk decisions.