AdaDPI: Document-level Translation Adaptive Agent via Dynamic Parametric Internalization

Hong Ren, Liting Deng, Shaolin Zhu*, Deyi Xiong*

TJUNLP Lab, School of Computer Science and Technology, Tianjin University, China
{rhong,dengliting,zhushaolin,dyxiong}@tju.edu.cn

Abstract

Large Language Models (LLMs) have demonstrated remarkable capabilities in machine translation. However, maintaining discourse coherence and terminological consistency remains a persistent challenge in document-level translation (DocMT). Existing solutions, such as memory-based agents, predominantly rely on explicit context concatenation. This paradigm treats historical context as a static external resource, which often leads to context dilution, high inference latency, and superficial knowledge integration. To address these limitations, we propose **AdaDPI**, an adaptive agentic framework that shifts the DocMT paradigm from static retrieval to dynamic parametric internalization. Specifically, we design a linguistic uncertainty monitor (LUM) to actively detect critical discourse discontinuities by the model's epistemic uncertainty. Upon detection, a context-to-parameter integrator (CPI) compiles retrieved external constraints directly into the model's intrinsic state via an online parameter adaptation mechanism. Through the online parameter adaptation on a lightweight adapter, AdaDPI internalizes document-specific norms into the model's intrinsic representations, enabling a progressive evolution of the translation strategy as the discourse unfolds. Extensive experiments on the discourse-rich GuoFeng and IWSLT2017 datasets demonstrate that AdaDPI significantly outperforms the SoTA baselines by more than 5 points on the consistency metric.

1 Introduction

The advent of Large Language Models (LLMs) has marked a paradigm shift in Machine Translation (MT), particularly in handling long-context document translation (DocMT) (Wu et al., 2024; Wang et al., 2023a). Unlike sentence-level systems, DocMT requires maintaining discourse coherence, lexical consistency, and stylistic unity across extended narratives (Maruf et al., 2021; Karpinska and Iyyer, 2023). While LLMs exhibit remarkable zero-shot capabilities, their performance on long documents often degrades due to the context maintenance bottleneck (Ma et al., 2024; Chen et al., 2024). As the discourse expands, the model struggles to recall upstream constraints effectively, leading to issues such as inconsistent terminology, dropped pronouns, ambiguous antecedents, and rigid stylistic shifts (Bao et al., 2021; Alves et al., 2024).

To mitigate these issues, recent research has bifurcated into two dominant streams: context-window extension and retrieval-augmented generation (RAG). The former relies on scaling the input length (e.g., via sliding windows or recurrence), yet frequently struggles with uneven attention distribution (Liu et al., 2024), where crucial constraints embedded in the mid-context are often ignored, and inference costs are prohibitive. The latter, exemplified by agentic frameworks such as DELTA (Wang et al., 2024c) and GRAFT (Dutta et al., 2025), uses explicit memory banks to store and retrieve relevant information. While effective, these methods suffer from a fundamental cognitive limitation: they treat translation as a static querying process rather than a dynamic learning one (Zhang et al., 2025b; Zhou et al., 2025). This reliance on static external retrieval hinders the model from constructing a holistic discourse representation of the document, resulting in high latency and fragmented reasoning.

Insights from translation process research suggest that high-quality document translation relies on internalizing context rather than merely accessing it (Maruf et al., 2021; Karpinska and Iyyer, 2023). This aligns with the concept of ad-hoc norm acquisition (Toury, 2012): as human translators progress through a text, they dynamically adapt their internal representations to capture the specific linguistic norms and stylistic constraints of

*Corresponding authors.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 37912–37929

the document. Consequently, translation consistency typically improves in later chapters as the translator becomes increasingly specialized in the domain (Wu et al., 2024).

To bridge the gap between static knowledge retrieval and dynamic cognitive adaptation, we propose **AdaDPI**, an adaptive agentic framework that shifts document translation from static retrieval to dynamic internalization. Specifically, we first design a linguistic uncertainty monitor (LUM). By quantifying the model's epistemic uncertainty about discourse coherence via the divergence of attention distributions and token-level entropy, LUM dynamically scrutinizes internal confidence to detect critical discourse discontinuities (e.g., ambiguous anaphora or terminology mismatches). This ensures that adaptation is activated strictly when the model's internal representations prove insufficient. Upon detection, we design the context-to-parameter integrator (CPI) to internalize external knowledge into the model's intrinsic state directly. Unlike standard methods that burden the input window with retrieved snippets, CPI functions as a cognitive assimilation mechanism. It bridges the gap between explicit rules and implicit generation by consolidating external constraints into the model's latent states, thereby enabling the agent to adapt its translation policy dynamically without expanding the context length.

The main contributions of this work are as follows: (i) We propose the **AdaDPI** adaptive agent, which shifts the DocMT paradigm from static retrieval to dynamic internalization, allowing models to learn document-specific norms on the fly. (ii) We introduce a CPI module combined with an LUM, which compresses explicit discourse constraints into latent memory and updates model parameters in real-time during inference. (iii) We evaluate our method on challenging document-level translation benchmarks, including the discourse-rich GuoFeng Webnovel dataset and IWSLT2017. Extensive experiments show that AdaDPI significantly reduces dependency on external retrieval as translation progresses while achieving SoTA performance on document translation tasks.

2 Related Work

LLM-Based DocMT: The application of LLMs to DocMT has primarily focused on two paradigms: context-aware prompting and supervised fine-tuning. Early strategies capitalized on the extended context windows of LLMs by concatenating preceding sentences or surrounding paragraphs to model inter-sentential dependencies (e.g., anaphora resolution) (Karpinska and Iyyer, 2023; Chitale et al., 2024). To handle ultra-long documents that exceed standard context limits, sliding window mechanisms and hierarchical encoding strategies have been widely adopted (Pal et al., 2024; Wu et al., 2024; Cui et al., 2024). In parallel with these inference-time techniques, instruction tuning on document-level parallel corpora has proven effective in enhancing the general discourse modeling capabilities of base models (Alves et al., 2024; Li et al., 2024). For instance, Wu et al. (2024) demonstrated that fine-tuning LLMs on document pairs significantly reduces discourse-level errors compared to sentence-level systems. However, these approaches face inherent limitations in the utilization of context. Merely expanding the context window often exacerbates positional attention biases, leading models to neglect constraints in the intermediate segments of long sequences (Liu et al., 2024). Moreover, SFT remains static during inference. While recent work such as LanderMT (Zhu et al., 2024b) has explored selective fine-tuning by routing language-aware neurons to enhance general translation performance, it still cannot adapt to the specific stylistic drifts or ad hoc terminological constraints unique to unseen documents in real-time.

RAG and Agent for DocMT: To overcome the fixed-context bottleneck, RAG and autonomous agentic frameworks have emerged as robust alternatives (Wang et al., 2024a; Ramos et al., 2025). Standard RAG methods improve consistency by retrieving relevant terminology or similar translation pairs from an external datastore and prepending them to the input as few-shot demonstrations (Zhu et al., 2024c; Tang et al., 2025). Beyond basic retrieval, recent efforts have also focused on enhancing the robustness of such in-context learning (ICL) approaches for translation (Zhu et al., 2024a). Besides, recent agent-based systems have introduced more sophisticated workflows. Wang et al. (2024c) propose DELTA, an agent that uses multi-level memory structures, such as proper-noun records and bilingual summaries, to manage discourse constraints explicitly. Similarly, Dutta et al. (2025) introduced GRAFT, a graph-based agentic framework that models dependencies between discourse segments to enhance coherence. While these methods effectively introduce external knowl-

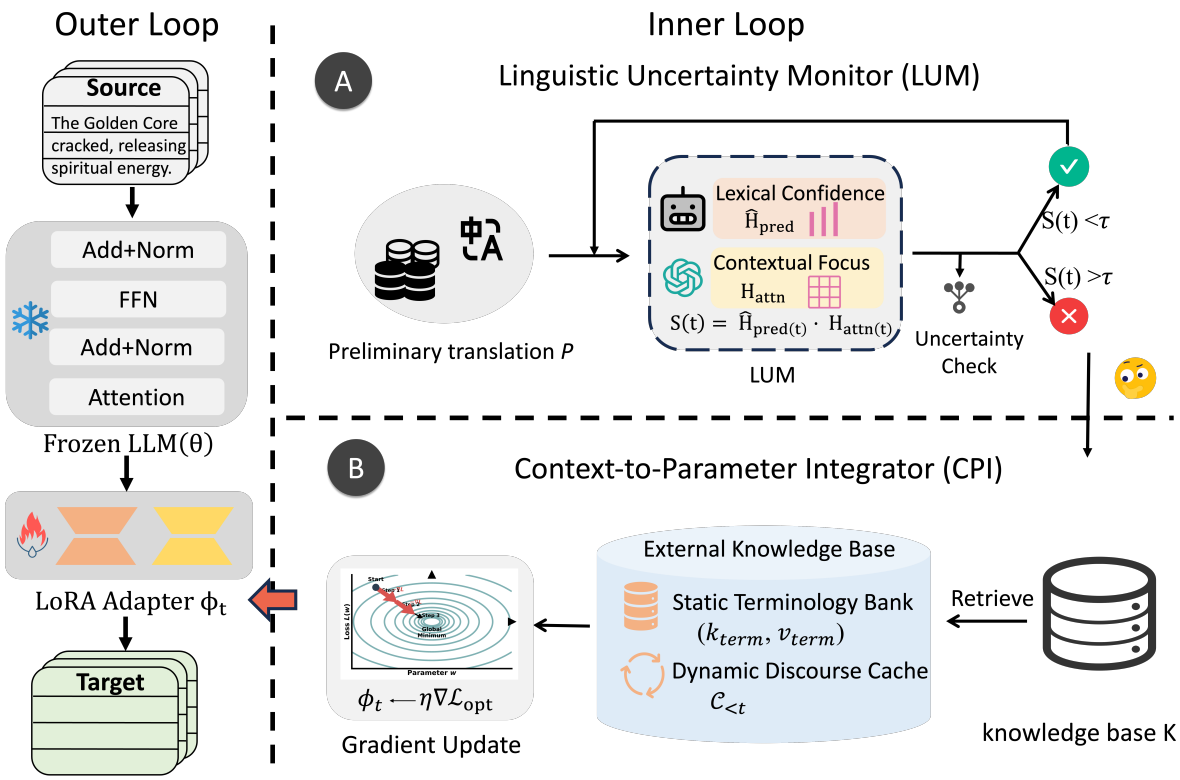

Figure 1: Illustration of the proposed AdaDPI.

edge, they predominantly rely on explicit context concatenation (Ji et al., 2024). Treat memory as a separate textual module that the model must implement at every step, which not only consumes valuable context-window capacity but also incurs high inference latency due to repeated processing of retrieved tokens. In contrast, AdaDPI shifts the paradigm from static retrieval to dynamic parametric internalization, directly compiling external constraints into the model's parameters, thereby enabling efficient adaptation.

3 Methodology

In this section, we present **AdaDPI**, an adaptive framework that bridges the gap between static knowledge retrieval and dynamic cognitive adaptation in DocMT. Unlike conventional paradigms that rely on fixed parameters or transient context windows, AdaDPI treats the translation process as a continuous state-tracking problem, in which the agent's internal representations actively evolve alongside the unfolding discourse. As illustrated in Figure 1, AdaDPI formalizes the translation task in a dynamic parametric setting, then employs an LUM to actively detect discourse discontinuities and resolve these ambiguities via a CPI that internalizes external constraints through online self-evolution.

3.1 Problem Formulation and Preliminaries

We formalize DocMT as a sequence-to-sequence generation task. Let $\mathcal{D}_x = \{x_1, x_2, \ldots, x_N\}$ denote a source document consisting of N sentences, and $\mathcal{D}_y = \{y_1, y_2, \ldots, y_N\}$ be its corresponding target translation. For the t-th sentence y_t, the generation is conditioned on the current source sentence x_t and the historical context $\mathcal{C}_{<t}$, which encompasses previous source and target segments as:

$$P(\mathcal{D}_y|\mathcal{D}_x;\theta) = \prod_{t=1}^{N} P(y_t \mid x_t, \mathcal{C}_{<t};\theta) \quad (1)$$

A critical limitation of this formulation is that the parameter θ of LLM remains invariant throughout the entire inference process. Consequently, the model must rely solely on the explicit context $\mathcal{C}_{<t}$ (often truncated due to window limits) to maintain discourse coherence, leading to the aforementioned context dilution and lack of adaptation. To overcome this, AdaDPI reformulates DocMT as a dynamic state-tracking process. We introduce a lightweight, time-dependent parameter set ϕ_t (e.g., LoRA) that evolves alongside document traversal, while keeping the massive LLM

parameters θ frozen. The translation of the t-th sentence is thus governed by the evolving joint parameters $\{\theta, \phi_t\}$:

$$P(\mathcal{D}_y|\mathcal{D}_x; \theta, \phi_0 \dots \phi_N) = \prod_{t=1}^{N} P(y_t \mid x_t, \mathcal{C}_{<t}; \theta, \boldsymbol{\phi}_t) \tag{2}$$

where ϕ_t represents the internalized state of the agent at step t. The core objective of AdaDPI is to define an update mechanism $\mathcal{F}$ such that $\phi_{t+1} \leftarrow \mathcal{F}(\phi_t, \mathcal{K}_t)$, where $\mathcal{K}_t$ denotes the discourse constraints extracted at the current step. This formulation allows the model to progressively accumulate document-specific knowledge into ϕ, shifting the burden of consistency maintenance from explicit context $\mathcal{C}_{<t}$ to implicit parameters.

3.2 Linguistic Uncertainty Monitor (LUM)

Continuous parameter updates at every decoding step incur unnecessary computational overhead and may introduce noise from non-content tokens. Therefore, AdaDPI employs the LUM, which computes real-time statistical metrics from the model's intermediate states to determine whether the current context is sufficient for accurate translation. Only when specific uncertainty criteria are met does LUM activate the subsequent parameter integration process. We formulate this decision process based on two complementary indicators: Lexical Confidence (LC) and Contextual Focus (CF).

LC quantifies the model's uncertainty about token selection. High uncertainty in the output layer often correlates with complex terminologies or ambiguous translations. For a generated token y_t at step t, we compute the Shannon entropy of the probability distribution over the vocabulary $\mathcal{V}$:

$$\hat{\mathcal{H}}_{\text{pred}}(t) = \frac{-\sum_{v \in \mathcal{V}} P(v|\mathbf{h}_t) \log P(v|\mathbf{h}_t)}{\log |\mathcal{V}|} \tag{3}$$

where $P(v|\mathbf{h}_t)$ is the probability of candidate word w given the current hidden state $\mathbf{h}_t$. A high $\mathcal{H}_{\text{pred}}$ means a primary signal that the static parameters θ struggle to predict the next token with high confidence. To distinguish discourse-level ambiguity, we further analyze the attention mechanism. Attention heads usually exhibit sharp focus on relevant antecedent tokens when resolving dependencies. Conversely, a flat, uniform attention distribution indicates that the model fails to identify supportive context. We quantify this as CF by calculating the entropy of the attention weights $\alpha_{t,j}$ over the context window $\mathcal{C}_{<t}$, averaged across K representative attention heads:

$$\mathcal{H}_{\text{attn}}(t) = \frac{1}{K} \sum_{k=1}^{K} \left(- \sum_{j \in \mathcal{C}_{<t}} \alpha_{t,j}^{(k)} \log \alpha_{t,j}^{(k)} \right) \tag{4}$$

A higher $\mathcal{H}_{\text{attn}}$ indicates a dispersed attention pattern, suggesting that the model lacks explicit guidance from the current context $\mathcal{C}_{<t}$.

LUM integrates the two metrics to form a robust triggering condition. We define a composite uncertainty score $\mathcal{S}(t) = \mathcal{H}_{\text{pred}}(t) \cdot \mathcal{H}_{\text{attn}}(t)$. The CPI is activated strictly when this score exceeds a pre-defined threshold τ. Details are provided in Appendix A.1. By doing so, LUM ensures that computationally expensive parameter updates are reserved solely for discourse-critical moments (e.g., ambiguous entities or stylistic consistency) to optimize the trade-off between performance and efficiency.

3.3 Context-to-Parameter Integrator (CPI)

Once the LUM signals a deficit in internal representation, the CPI is activated. Conceptually, CPI functions as a knowledge compiler. It transforms discrete and symbolic constraints (e.g., retrieved text) into continuous and functional modifications (parameter updates) within the model's neural circuitry.

Upon activation at step t, CPI first queries an external knowledge base $\mathcal{K}$ (Appendix A.2). The retrieval module returns a set of $\mathcal{K}_t = \{(k_i, v_i)\}_{i=1}^{M}$, where pairs represent specific terminology definitions, stylistic exemplars, or antecedent-pronoun mappings relevant to the current ambiguity. To enable rapid internalization without disrupting the pre-trained knowledge θ, we define a mutable parameter subspace. We adopt a LoRA (Hu et al., 2022) as the dynamic carrier for evolution. Building on the concept of parameter-based knowledge transfer demonstrated in MLAS-LoRA (Dong et al., 2025), we extend this paradigm to the real-time internalization of ad-hoc discourse constraints via a mutable parameter subspace. Let ϕ_t denote the parameters of these adapters at step t. The forward pass of the linear layers in the LLM is modified as:

$$\mathbf{H}_{\text{out}} = \mathbf{W}_\theta \mathbf{H}_{\text{in}} + \Delta \mathbf{W}_{\phi_t} \mathbf{H}_{\text{in}} \tag{5}$$

θ of LLM remains frozen, ensuring general translation capability, while ϕ_t serves as a dedicated scratchpad for document-specific adaptation. The

core innovation of CPI lies in transforming constraints into optimization targets. Unlike in-context learning, which treats $\mathcal{K}_t$ as passive input tokens, AdaDPI uses $\mathcal{K}_t$ to compute a gradient update and directly embeds its information into the adapter parameters. Specifically, we formulate a temporary optimization objective $\mathcal{L}_{\text{opt}}$:

$$\mathcal{L}_{\text{opt}}(\boldsymbol{\phi}_t) = -\sum_{(k,v)\in\mathcal{K}_t} \log P(v \mid k, \mathcal{C}_{<t}; \theta, \boldsymbol{\phi}_t) \tag{6}$$

To integrate this knowledge $\mathcal{K}_t$, we perform a single-step or few-step gradient descent update on the adapter parameters $\boldsymbol{\phi}_t$:

$$\boldsymbol{\phi}'_t \leftarrow \boldsymbol{\phi}_t - \eta \cdot \nabla_{\phi} \mathcal{L}_{\text{opt}}(\boldsymbol{\phi}_t) \tag{7}$$

This optimization ensures that the updated adapter $\boldsymbol{\phi}'_t$ is aligned with the retrieved discourse constraints.

3.4 Online Dual-Loop Adaptation

The parameters are optimized during training and frozen during inference in the LLM (see details in A.3). However, it prevents the model from adapting to the non-stationary linguistic distributions inherent in DocMT. Therefore, AdaDPI implements a novel dual-loop adaptation protocol that effectively transforms the inference phase into a continuous learning trajectory.

The Inner Loop: It addresses the challenge of integrating external knowledge into a model without ground-truth supervision. When the LUM triggers an adaptation event at step t, the system suspends generation and enters an optimization state. Although the target sentence y_t is unknown, the retrieved $\mathcal{K}_t$ (e.g., term pairs or syntactic templates) serve as high-confidence pseudo-labels. We perform test-time adaptation by executing a single gradient descent step on ϕ:

$$\boldsymbol{\phi}^*_t \leftarrow \boldsymbol{\phi}_t - \eta \cdot \nabla_{\phi} \mathcal{L}_{\text{opt}}(\mathcal{K}_t; \theta) \tag{8}$$

This makes the static θ momentarily modulated by ϕ^*_t, aligning the model's manifold with the local discourse context before the actual translation is produced.

The Outer Loop: Upon completing the inner optimization, the system reverts to the generation mode. The current segment x_t is decoded using the updated parameter set $\{\theta, \phi^*_t\}$. Since ϕ^*_t encodes the relevant $\mathcal{K}_t$, the model naturally prioritizes the correct terminology or style during beam search without requiring complex prompt engineering. The updated parameters ϕ^*_t are retained for subsequent sentences $(t+1, \ldots, N)$, creating a cumulative evolutionary effect. As the document progresses, ϕ increasingly converges to the text's specific idiolect, reducing the divergence between the model's internal representation and the document's distribution. This mechanism effectively realizes the adaptive agent's self-evolution. As the internal parameters (ϕ) progressively accumulate document-specific traits, they actively maintain discourse coherence, pronoun anaphora, and stylistic unity.

4 Experiments

To validate the effectiveness of the proposed AdaDPI framework, we conduct extensive evaluations on both literary narratives and spoken-domain benchmarks. We compare our method against a wide range of baselines, including standard NMT systems, frontier LLMs, and specialized agentic frameworks, focusing on translation quality, long-range discourse coherence, and inference efficiency.

4.1 Datasets and Metrics

Datasets. We conducted experiments on two distinct benchmarks to evaluate both discourse-level coherence in complex narratives and generalization across standard scenarios. To assess the model's capability for long-context literary translation, we used the GuoFeng Webnovel dataset (Wang et al., 2023b, 2024b). This corpus features intricate discourse structures and character relationships. We performed experiments on the *GuoFeng v1* test set in the Chinese $\leftrightarrow$ English directions. Following the official protocol, we strictly preprocessed the data into document-level streams to challenge the model's ability to maintain context. To verify robustness on standard benchmarks, we employed the IWSLT2017 translation task (Cettolo et al., 2017). We evaluated performance on the `tst2017` test set, which comprises transcribed TED Talks. Our experiments covered eight translation directions: English $\leftrightarrow$ {Chinese, German, French, Japanese} (Detailed in Appendix A.4).

Evaluation Metrics. We employed a comprehensive set of metrics to capture different aspects

Methods	Guofeng						IWSLT2017					
	zh → en			en → zh			en → xx			xx → en		
	COMET	LTCR	d-BLEU	COMET	LTCR	d-BLEU	COMET	LTCR	d-BLEU	COMET	LTCR	d-BLEU
State-of-the-art Neural Machine Translation baselines												
NLLB	76.85	40.20	-	81.90	42.50	-	82.11	74.56	-	84.10	79.03	-
GOOGLE	77.40	41.15	-	82.25	43.80	-	80.41	81.38	-	80.17	81.43	-
State-of-the-art LLMs												
GPT-4o	79.20	82.15	19.45	83.54	84.12	31.20	80.82	78.45	29.80	80.10	80.12	30.96
Gemini 1.5	78.85	81.40	20.12	83.10	83.56	30.55	80.64	77.92	31.70	80.50	79.88	33.17
Claude 3.7	79.41	82.66	18.30	83.82	84.70	32.10	80.25	79.10	25.62	80.50	80.45	26.51
Document-level MT baselines												
Sentence	73.65	37.00	-	-	-	-	80.03	73.91	-	77.10	76.39	-
Context	76.54	52.82	-	-	-	-	80.84	79.59	-	83.09	81.48	-
Doc2Doc	-	73.25	-	-	-	-	-	77.32	-	-	85.03	-
SFT	76.10	78.45	21.30	81.55	84.10	28.50	80.90	79.80	28.40	82.10	81.20	31.50
GRPO	76.88	81.20	22.10	82.40	85.60	29.20	81.40	81.50	29.10	83.45	83.90	32.80
DAPO	77.20	84.60	23.40	83.10	86.30	30.50	82.10	83.10	30.60	84.60	85.20	33.90
DELTA	76.95	85.50	-	-	-	-	81.02	80.09	-	83.36	82.05	-
AdaDCOMT	-	-	-	-	-	-	68.50	-	20.50	81.40	-	33.60
Ours												
Qwen2.5-base	75.57	85.14	19.66	81.82	84.62	29.30	79.56	77.12	26.45	81.25	79.40	28.15
AdaDPI-sft	76.94	85.93	22.67	82.12	84.80	29.19	80.85	80.12	28.14	83.40	82.11	31.42
AdaDPI-GRPO	78.15	86.40	23.85	84.10	87.25	32.14	81.92	84.55	31.25	84.92	86.45	34.12
AdaDPI-DAPO	**79.85**	**89.15**	**24.80**	**85.42**	**90.30**	**34.15**	**83.25**	**86.90**	**33.10**	**86.50**	**88.95**	**36.20**

Table 1: Main results of machine translation performance across Guofeng and IWSLT2017 datasets. The best results are highlighted in **bold**.

of translation quality: (i) d-BLEU (Liu et al., 2020) measures document-level lexical overlap. (ii) COMET[1]: Evaluates semantic accuracy using a pre-trained neural metric. (iii) LTCR (Lyu et al., 2021) specifically assesses terminological consistency and discourse coherence by measuring the preservation of document-level entity constraints.

4.2 Baselines

To evaluate the effectiveness of AdaDPI, we compared it against a wide range of baselines, including standard NMT systems, frontier LLMs, and specialized discourse-aware frameworks. For terminological clarity in our evaluation, we considered three fundamental paradigms: Sentence (translating each sentence independently), Context (using a limited set of preceding sentences as context), and Doc2Doc (processing the entire document as a single sequence). The compared systems and frameworks consisted of NLLB-3.3B (Costa-Jussà et al., 2022), GoogleTrans, GPT-4o, Gemini-3, Claude-3.7, DELTA (Wang et al., 2024c), and AdaDCOMT (Wu et al., 2024). Furthermore, we assessed our model variants optimized through SFT (LoRA) (Hu et al., 2022), GRPO (Guo et al., 2025), and DAPO (Yu et al., 2025).Detailed hyperparameter configurations, LoRA settings, and base model alignment objectives are provided in Appendix A.7

4.3 Main Results

Table 1 summarizes the translation performance across the literary-domain GuoFeng dataset and the spoken-domain IWSLT2017 benchmark. The results indicate that **AdaDPI** consistently outperforms strong baselines across all metrics, establishing a new state-of-the-art for document-level translation tasks. Compared to context-aware methods that rely on sliding windows or concatenation (e.g., *Context*, *Doc2Doc*), AdaDPI achieves substantial gains. For example, AdaDPI-DAPO surpasses the standard *Context* baseline by a significant margin in d-BLEU on the GuoFeng En→Zh task. This highlights the limitation of fixed context windows, as simply appending history tokens leads to information loss and dilution of attention. In contrast, AdaDPI maintains robust long-range coherence without being constrained by the window size by dynamically internalizing constraints into parameters.

A critical comparison is between AdaDPI, SoTA DELTA, and AdaDCOMT. While DELTA utilizes explicit memory banks to manage discourse, AdaDPI-DAPO outperforms it by notable margins on LTCR (consistency metric) across both translation directions. This validates our core hypothesis that treating retrieval as static input is less effective

[1] https://github.com/Unbabel/COMET/

Methods	COMET	LTCR	d-BLEU
AdaDPI (Full)	**85.42**	**90.30**	**34.15**
I. Component Necessity			
w/o CPI (Context-based)	83.15	85.20	30.80
w/o Persistence	84.02	86.55	32.40
w/o LUM (Always Adapt)	85.10	89.80	33.95
II. Mechanism Granularity			
w/o SFT Init (Random Init)	79.50	81.20	26.45
Update Attention Only	85.38	90.15	34.02
Update MLP Only	84.95	88.60	33.50
Multi-step Update ($K = 5$)	83.80	89.90	32.80

Table 2: Extended ablation study on GuoFeng En→Zh.

Model	T-Recall (%)	Pronoun Acc. (%)
GPT-4o	89.5	78.2
SFT	91.2	80.5
RAG	93.8	82.1
DELTA	95.4	84.6
AdaDPI	**98.2**	**87.9**

Table 3: Discourse consistency analysis on GuoFeng En→Zh. AdaDPI demonstrates superior capability in tracking entities and resolving pronouns.

than parametric internalization. The superior LTCR scores suggest that when discourse constraints are compiled into the model's weights via the CPI module, the model adheres to them more strictly than when they are merely present in the context.

It is worth noting that AdaDPI was built on a relatively small base model that achieves performance competitive with, or even superior to, massive proprietary LLMs like *GPT-4o* and *Claude 3.7* on specialized document metrics. Specifically, while it exhibits strong sentence-level translation capabilities (high COMET), it often struggles with the specific stylistic consistency required in literary translation (lower d-BLEU). AdaDPI bridges this gap through self-evolution. Furthermore, the consistent improvements across AdaDPI-SFT, AdaDPI-GRPO, and AdaDPI-DAPO demonstrate that our framework is robust to the finetuning algorithms, consistently enhancing the base model's capability to model document-level dependencies.

Beyond standard baselines, we further evaluated AdaDPI against the latest commercial frontier models, including GPT-5.2 and Gemini 3.0 Pro, on the GuoFeng dataset. As detailed in Appendix A.10, our method consistently maintains a lead of over 5 points in the consistency metric (LTCR), demonstrating its robustness even when compared to the most advanced LLM architectures.

4.4 Ablation Study

We conducted a comprehensive ablation study on the GuoFeng En→Zh test set as in Table 2. The removal of CPI leads to the most significant degradation across all metrics (e.g., -3.35 d-BLEU). It confirms that parametric internalization provides a more robust mechanism for handling constraints. Furthermore, the performance drop in the *w/o Persistence* setting highlights the critical role of the cumulative evolutionary effect in maintaining stylistic consistency over long contexts. Interestingly, while removing LUM (*Always Adapt*) yields performance comparable to the whole model, it incurs a computational overhead (due to the increased parameter count). This suggests that LUM successfully filters out redundant updates without compromising translation quality and achieves an optimal trade-off between efficiency and performance.

When the adapter is randomly initialized during inference (skipping Phase 1), performance collapses. This indicates that the test-time gradient updates do not learn translation skills from scratch but instead steer a pre-aligned manifold. Updating only the Attention weights yields results nearly identical to the whole model, whereas updating only the MLP layers yields a slight drop (-0.65 d-BLEU). This suggests that Attention layers are more responsible for retrieving and routing context information, making them the ideal substrate for processing discourse constraints. Increasing the number of gradient steps to $K = 5$ for each constraint significantly degrades general translation quality (COMET drops by 1.62), while maintaining high Terminology Recall (LTCR). This confirms the risk of catastrophic forgetting in online learning.

4.5 Discourse Consistency

A defining challenge in translating literary texts (e.g., GuoFeng) is maintaining consistency in specific entities and pronouns over long narratives. We evaluate this capability on the En→Zh task using two targeted metrics: **Terminology Recall (T-Recall)** to measure the percentage of consistently translated proper nouns appearing in the context window, and **Pronoun Accuracy** to evaluate the resolution of ambiguous pronouns. Table 3 presents the results.

GPT-4o often translates the same character name differently across chapters due to truncation of the context window. While RAG improves recall by

Method	Latency (ms)	Overhead	COMET
Static Inference	142	-	79.85
Always Adapt	638	+349%	85.50
AdaDPI (Ours)	**175**	**+23%**	**85.42**

Table 4: Comparison of inference latency (ms/sentence) and translation quality on GuoFeng En→Zh.

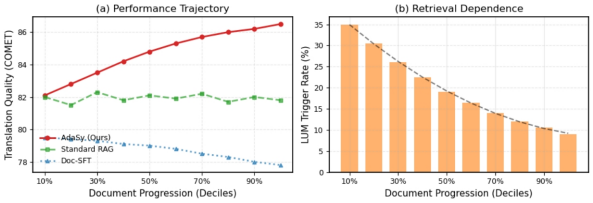

Figure 2: Visualization of Self-Evolution dynamics.

retrieving definitions, it still suffers from attention decay when the retrieved constraints compete with a long input sequence. AdaDPI achieves a near-perfect T-Recall (98.2%). This indicates that the CPI module not only encodes the term's model but also effectively enhances the model's internal preference for that term via gradient updates. Once the parameter space is shifted, the model generates the correct term as the high-probability default, thereby reducing the risk of hallucinations. Pronoun translation in Chinese is particularly sensitive to context. AdaDPI outperforms DELTA by +3.3% in Pronoun Accuracy. AdaDPI's parameters serve as a continuous-state tracker. The model maintains a coherent latent representation of the current subject by progressively updating the adapter, enabling accurate pronoun inference even when the explicit antecedent lies outside the immediate context window.

4.6 Efficiency and Performance

A primary concern with AdaDPI is the computational overhead introduced by gradient calculations during inference. To quantify this trade-off, we compare the average inference latency per sentence against translation quality (COMET) on the GuoFeng En→Zh dataset. We benchmark AdaDPI against two control settings: (1) **Static Inference**: Standard decoding with frozen parameters (highest speed, lower quality). (2) **Always Adapt**: Performing gradient updates at every sentence regardless of uncertainty (lowest speed, theoretical upper bound for quality). Table 4 summarizes the results. All experiments were conducted on a single NVIDIA A100 GPU with a batch size of 1.

As expected, the *Always Adapt* strategy incurs a prohibitive computational cost, increasing latency by nearly 3.5× due to the frequent backward passes required for every sentence. However, AdaDPI significantly mitigates this overhead. AdaDPI triggers the optimization loop for only 18.4% of the sentences in the test set. AdaDPI introduces a marginal latency increase of 23% compared to static inference (175ms vs. 142ms), yet it retains 99.9% of the performance gain achieved by the *Always Adapt* strategy (85.42 vs. 85.50 COMET).

4.7 Visualizing

To verify the cumulative evolutionary effect, we tracked the translation quality and LUM trigger frequency across document deciles on the GuoFeng En→Zh dataset. Figure 2 visualizes these longitudinal dynamics. Figure 2 (a) reveals a distinct upward trajectory in COMET scores for AdaDPI, contrasting with the stagnation of the static Doc-SFT and the volatility of Standard RAG. This trend suggests that online parameter updates effectively accumulate document-specific patterns that static windows fail to capture. Concurrently, Figure 2 (b) shows a monotonic decline in the LUM trigger rate (dropping from ∼35% to <10%). The observed negative correlation between performance gains and retrieval frequency provides empirical evidence for parametric internalization. As the adapter parameters increasingly converge to the document's specific distribution, the agent naturally shifts from relying on external constraints to utilizing its evolved internal representations.

4.8 Results on Other LLM

To assess the generalizability of AdaDPI across different model architectures, we conduct experiments on the LLaMA-3-8B backbone. We focused on the literary GuoFeng dataset (En↔Zh) and compared AdaDPI against three representative baselines: Zero-shot, In-Context Learning (ICL) with five exemplars, and standard SFT (LoRA). Figure 3 presents the comparative results. AdaDPI achieves the most robust performance envelope across all metrics, which is consistent with observations on Qwen models. These results confirm that the CPI module's efficacy is agnostic to the underlying LLM and consistently enhances document-level translation capabilities through parametric internalization.

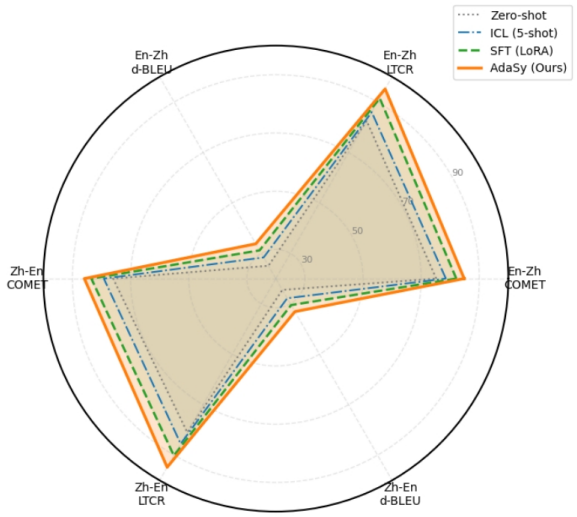

Figure 3: Performance comparison on LLaMA-3-8B backbone.

5 Conclusion

In this work, we presented AdaDPI, an adaptive agentic framework that reimagines document-level translation as a dynamic state-tracking process rather than a static decoding task. AdaDPI shifts the paradigm from explicit context concatenation to dynamic parametric internalization to address the inherent limitations of context dilution in retrieval-augmented baselines. Our approach effectively bridges the gap between external constraints and intrinsic model representations by synergizing the LUM for precise discontinuity detection with CPI for rapid knowledge assimilation. Empirical evidence across literary and spoken domains confirms that this test-time adaptation mechanism yields SoTA performance on multiple language pairs.

Acknowledgments

The present research was supported by the National Key Research and Development Program (Grant No. 2023YFE0116400), National Natural Science Foundation of China Youth Fund (Grant No. 62306210). We would like to thank the anonymous reviewers for their insightful comments.

Limitations

AdaDPI incurs a computational overhead due to test-time gradient backpropagation, resulting in a $\sim$23% latency increase compared to static inference, which may challenge ultra-low-latency scenarios. Furthermore, the reliance on parametric updates requires additional memory for the optimizer's state. It makes the system sensitive to the quality of retrieved constraints, where noisy supervision could lead to the internalization of errors. Finally, balancing constraint adherence with general linguistic capability requires careful hyperparameter tuning (e.g., the number of gradient steps) to prevent catastrophic forgetting.

References

Duarte M Alves, José Pombal, Nuno M Guerreiro, Pedro H Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G.C. de Souza, and André F.T. Martins. 2024. Tower: An open multilingual large language model for translation-related tasks. *arXiv preprint arXiv:2402.17733*.

Guangsheng Bao, Yue Zhang, Zhiyang Teng, Boxing Chen, and Weihua Luo. 2021. G-transformer for document-level machine translation. 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 3442–3455.

Mauro Cettolo, Marcello Federico, Luisa Bentivogli, Jan Niehues, Sebastian Stüker, Katsuhito Sudoh, Koichiro Yoshino, and Christian Federmann. 2017. Overview of the iwslt 2017 evaluation campaign. In *Proceedings of the 14th International Conference on Spoken Language Translation*, pages 2–14.

Longze Chen, Ziqiang Liu, Wanwei He, Yinhe Zheng, Hao Sun, Yunshui Li, Run Luo, and Min Yang. 2024. Long context is not long at all: A prospector of long-dependency data for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8222–8234.

Pranjal Chitale, Jay Gala, and Raj Dabre. 2024. An empirical study of in-context learning in LLMs for machine translation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7384–7406.

Marta R Costa-Jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, and 1 others. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Menglong Cui, Jiangcun Du, Shaolin Zhu, and Deyi Xiong. 2024. Efficiently exploring large language models for document-level machine translation with in-context learning. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 10885–10897.

Tianyu Dong, Bo Li, Jinsong Liu, Shaolin Zhu, and Deyi Xiong. 2025. Mlas-lora: language-aware parameters detection and lora-based knowledge transfer

for multilingual machine translation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15645–15660.

Himanshu Dutta, Sunny Manchanda, Prakhar Bapat, Meva Ram Gurjar, and Pushpak Bhattacharyya. 2025. Graft: A graph-based flow-aware agentic framework for document-level machine translation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 2405–2428.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Baijun Ji, Xiangyu Duan, Zhenyu Qiu, Tong Zhang, Junhui Li, Hao Yang, and Min Zhang. 2024. Submodular-based in-context example selection for LLMs-based machine translation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15398–15409.

Marzena Karpinska and Mohit Iyyer. 2023. Large language models effectively leverage document-level context for literary translation, but critical errors persist. In *Proceedings of the Eighth Conference on Machine Translation*, pages 419–451.

Yachao Li, Junhui Li, Jing Jiang, and Min Zhang. 2024. Enhancing document-level translation of large language model via translation mixed-instructions. *arXiv preprint arXiv:2401.08088*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Xinglin Lyu, Junhui Li, Zhengxian Gong, and Min Zhang. 2021. Encouraging lexical translation consistency for document-level neural machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3265–3277.

Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, and 1 others. 2024. Mmlongbench-doc: Benchmarking long-context document understanding with visualizations. *Advances in Neural Information Processing Systems*, 37:95963–96010.

Sameen Maruf, Fahimeh Saleh, and Gholamreza Haffari. 2021. A survey on document-level neural machine translation: Methods and evaluation. *ACM Computing Surveys (CSUR)*, 54(2):1–36.

Proyag Pal, Alexandra Birch, and Kenneth Heafield. 2024. Document-level machine translation with large-scale public parallel corpora. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13185–13197.

Miguel Moura Ramos, Patrick Fernandes, Sweta Agrawal, and André FT Martins. 2025. Multilingual contextualization of large language models for document-level machine translation. *arXiv preprint arXiv:2504.12140*.

Lei Tang, Jinghui Qin, Wenxuan Ye, Hao Tan, and Zhijing Yang. 2025. Adaptive few-shot prompting for machine translation with pre-trained language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25255–25263.

Gideon Toury. 2012. Descriptive translation studies: And beyond.

Jiaan Wang, Fandong Meng, Yingxue Zhang, and Jie Zhou. 2024a. Retrieval-augmented machine translation with unstructured knowledge. *arXiv preprint arXiv:2412.04342*.

Longyue Wang, Siyou Liu, Minghao Wu, Wenxiang Jiao, Xing Wang, Jiahao Xu, Zhaopeng Tu, Liting Zhou, Yan Gu, Weiyu Chen, Philipp Koehn, Andy Way, and Yulin Yuan. 2024b. Findings of the wmt 2024 shared task on discourse-level literary translation. In *Proceedings of the Ninth Conference on Machine Translation*.

Longyue Wang, Chenyang Lyu, Tianbo Ji, Zhirui Zhang, Dian Yu, Shuming Shi, and Zhaopeng Tu. 2023a. Document-level machine translation with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16646–16661.

Longyue Wang, Zhaopeng Tu, Yan Gu, Siyou Liu, Dian Yu, Qingsong Ma, Chenyang Lyu, Liting Zhou, Chao-Hong Liu, Yufeng Ma, and 1 others. 2023b. Findings of the wmt 2023 shared task on discourse-level literary translation: A fresh orb in the cosmos of LLMs. In *Proceedings of the Eighth Conference on Machine Translation*, pages 55–67.

Yutong Wang, Jiali Zeng, Xuebo Liu, Derek F Wong, Fandong Meng, Jie Zhou, and Min Zhang. 2024c. Delta: An online document-level translation agent based on multi-level memory. *arXiv preprint arXiv:2410.08143*.

Minghao Wu, Thuy-Trang Vu, Lizhen Qu, George Foster, and Gholamreza Haffari. 2024. Adapting large language models for document-level machine translation. *arXiv preprint arXiv:2401.06468*.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, and 1 others. 2025. Dapo: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.

Guibin Zhang, Muxin Fu, and Shuicheng Yan. 2025a. Memgen: Weaving generative latent memory for self-evolving agents. *arXiv preprint arXiv:2509.24704*.

Wen Zhang, Long Jin, Yushan Zhu, Jiaoyan Chen, Zhiwei Huang, Junjie Wang, Yin Hua, Lei Liang, and Huajun Chen. 2025b. Trustuqa: A trustful framework for unified structured data question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25931–25939.

Zijian Zhou, Ao Qu, Zhaoxuan Wu, Sunghwan Kim, Alok Prakash, Daniela Rus, Jinhua Zhao, Bryan Kian Hsiang Low, and Paul Pu Liang. 2025. Mem1: Learning to synergize memory and reasoning for efficient long-horizon agents. *arXiv preprint arXiv:2506.15841*.

Shaolin Zhu, Menglong Cui, and Deyi Xiong. 2024a. Towards robust in-context learning for machine translation with large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 16619–16629.

Shaolin Zhu, Leiyu Pan, Bo Li, and Deyi Xiong. 2024b. Landermt: Dectecting and routing language-aware neurons for selectively finetuning LLMs to machine translation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12135–12148.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024c. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the association for computational linguistics: NAACL 2024*, pages 2765–2781.

A Appendix

A.1 Execution Protocol of Linguistic Uncertainty Monitor

A critical distinction of AdaDPI compared to standard decoding is its Pause-and-Learn inference capability. We clarify the exact execution protocol below.

A.1.1 The Pre-Commitment Intervention Logic

Standard autoregressive models generate a token y_t immediately after computing the forward pass at step t. In contrast, AdaDPI introduces an intermediate verification step. For each step t:

1. **Tentative Forward Pass:** The model computes the hidden state $\mathbf{h}_t$ and the logit distribution using the current parameters.

2. **LUM Assessment:** We calculate $\mathcal{H}_{\text{term}}$ and $\mathcal{H}_{\text{term}}$ based on this tentative state.

3. **Branching Decision:**

   - **Case A (Low Uncertainty):** If $\mathcal{S}(t) \leq \tau$, the system commits to the current distribution, samples y_t, and proceeds to $t+1$. This incurs negligible overhead (simple entropy calculation).
   - **Case B (High Uncertainty):** If $\mathcal{S}(t) > \tau$, the generation is paused. The Context-to-Parameter Integrator (CPI) is triggered to retrieve external constraints and perform a single-step gradient update on the lightweight adapter parameters ($\phi_t \rightarrow \phi'_t$).

4. **Re-Computation (Only in Case B):** With the updated parameters ϕ'_t, the model recomputes the forward pass for step t. Since the parameters have shifted to align with the discourse constraints, the new distribution is expected to be sharper and more accurate. The final y_t is then sampled.

A.1.2 Parametric Internalization vs. Source Re-encoding

A common misconception is that integrating new memory requires modifying the source input or re-encoding the entire history, which would be computationally prohibitive in autoregressive decoding. However, AdaDPI relies on **Parametric Internalization**. We do *not* append retrieved text to the source prompt (which would indeed require re-encoding). Instead, we update the weights of the adapter layers. Since the source representation is processed through these weights during the attention operation, updating the weights effectively changes how the model views and attends to the static source context. This allows us to inject knowledge dynamically without altering the input sequence or clearing the KV cache, ensuring that the self-evolution process is compatible with standard causal masking mechanisms.

A.2 Construction of External Knowledge Base

To ensure that the CPI receives high-quality supervision signals, we construct the external knowledge base $\mathcal{K}$ following a hybrid protocol akin to recent agentic frameworks (Wang et al., 2024c). The $\mathcal{K}$ comprises two distinct components tailored for document-level constraints. The exact prompt templates used for memory construction are detailed in Appendix A.8.

Static Terminology Bank. Before translation, we compile a domain-specific dictionary to enforce lexical consistency. For datasets with provided terminology (e.g., IWSLT), we utilize the official lists. For literary domains such as GuoFeng, where no official glossary exists, we employ an unsupervised term-extraction pipeline using TF-IDF and C-Value to identify high-frequency proper nouns and domain-specific entities. Each entry is stored as a key-value pair $(k_{\text{term}}, v_{\text{term}})$, where k is the source term and v is the target translation.

Dynamic Discourse Cache. To handle long-range dependencies such as pronoun resolution and entity consistency, we maintain a dynamic cache that grows as the translation progresses. At step t, the system stores previously generated entity translations and identifying phrases from the context $\mathcal{C}_{<t}$ into the $\mathcal{K}$. When LUM detects ambiguity (e.g., a polysemous pronoun), the retrieval module queries this cache to find the most recent antecedent translation. This dynamic component ensures that the model's parametric updates are grounded in the specific narrative flow of the current document, effectively mitigating the amnesic behavior of standard static models.

A.3 Training Protocol

To function as an effective CPI, the LoRA adapter cannot merely serve as a static container of linguistic patterns; it must rapidly assimilate external

constraints. Therefore, distinct from traditional SFT, which optimizes for static translation capability, we use a meta-initialization protocol (Zhang et al., 2025a). This protocol simulates the test-time retrieve-update-generate cycle during training, explicitly teaching the adapter how to weave retrieved constraints into its parameters.

A.3.1 Data Construction

We organize the training data into episodes to mimic the inference scenario. For each source document $\mathcal{D}_x$, we construct a series of training triplets:

$$\mathcal{E}_t = (\mathcal{K}_t, x_t, y_t) \tag{9}$$

where x_t is the source sentence, y_t is the ground truth, and $\mathcal{K}_t$ is retrieved knowledge.

A.3.2 Meta-Optimization Algorithm

The training objective is to minimize the generation loss after the LoRA parameters have been updated to satisfy the constraints. This involves a bi-level optimization process analogous to Model-Agnostic Meta-Learning (MAML).

Inner Loop: For a given step t, we first perform a tentative gradient update on the current LoRA ϕ using the constraints $\mathcal{K}_t$. This mimics the CPI's behavior during inference:

$$\phi'(\mathcal{K}_t) = \phi - \alpha \cdot \nabla_\phi \mathcal{L}_{\text{constraint}}(\mathcal{K}_t; \theta, \phi) \tag{10}$$

where α is the inner-loop learning rate. ϕ' represents the state of updated LoRA.

Outer Loop: We then compute the translation loss on the actual target sentence y_t using the updated parameters ϕ'. Crucially, we optimize the initial parameters ϕ such that they perform well after the update:

$$\mathcal{L}_{\text{meta}}(\phi) = -\log P(y_t \mid x_t; \theta, \phi'(\mathcal{K}_t)) \tag{11}$$

The final gradient update for ϕ involves differentiating through the inner loop step:

$$\phi \leftarrow \phi - \beta \cdot \nabla_\phi \mathcal{L}_{\text{meta}}(\phi) \tag{12}$$

By training in this manner, LoRA is not optimized to memorize the training data, but rather to be highly sensitive and adaptive to $\mathcal{K}$. It learns to instantly internalize new information provided in $\mathcal{K}_t$.

The complete workflow, encompassing both the meta-training phase and the inference evolution phase, is detailed in Algorithm 1.

Algorithm 1 AdaDPI: From Meta-Initialization to Inference

Require: Large Language Model θ (Fixed)
Require: Adapter Parameters ϕ (Trainable)
Require: Training Corpus $\mathcal{D}_{train}$, Learning Rates α, β
Phase 1: Meta-Initialization (Learning to Weave)
while not converged **do**
  Sample batch of episodes $(\mathcal{K}, x, y)$ from $\mathcal{D}_{train}$
  Inner Loop (Simulate CPI):
    $\phi' \leftarrow \phi - \alpha \nabla_\phi \mathcal{L}_{\text{constraint}}(\mathcal{K})$
  Outer Loop (Evaluate Integration):
    $\mathcal{L}_{\text{meta}} \leftarrow -\log P(y \mid x; \theta, \phi')$
  Update Base Parameters:
    $\phi \leftarrow \phi - \beta \nabla_\phi \mathcal{L}_{\text{meta}}$
end while
Result: Initialized Weaver $\phi_{\text{init}} \leftarrow \phi$
Phase 2: Inference
Load $\phi_0 \leftarrow \phi_{\text{init}}$
for each test sentence x_t **do**
  Detect Uncertainty via LUM
  if Triggered **then**
    Retrieve constraints $\mathcal{K}_{\text{test}}$
    $\phi_t \leftarrow \phi_{t-1} - \eta \nabla_\phi \mathcal{L}_{\text{constraint}}(\mathcal{K}_{test})$ {Real-time Weaving}
  end if
  Generate y_t using $\{\theta, \phi_t\}$
end for

A.4 Dataset Details

We provide detailed statistics for the datasets used in our experiments. Table 5 presents a comprehensive overview of the document (**#D**) and sentence (**#S**) counts across Training, Validation, and Test splits for both GuoFeng Webnovel and IWSLT2017 benchmarks.

A.5 Sensitivity Analysis of Gradient Optimization Steps

A critical hyperparameter in the CPI module is the number of gradient-descent steps, denoted by K, used during the online adaptation phase. The selection of K involves a fundamental trade-off between constraint adherence, generalization capability, and inference latency. Table 6 details the performance metrics across varying step counts ranging from $K = 1$ to $K = 10$.

Direction	Train		Validation		Test	
	# D	# S	# D	# S	# D	# S
GuoFeng Webnovel (v1)						
En ↔ Zh	179	1,939,187	6	421	12	645
IWSLT2017						
En ↔ Zh	1,906	231,266	8	879	89	8,549
En ↔ De	1,698	206,112	8	888	85	8,079
En ↔ Fr	1,914	232,825	8	890	89	8,597
En ↔ Ja	1,863	223,108	8	871	89	8,469

Table 5: Detailed statistics for GuoFeng Webnovel (v1) and IWSLT2017 datasets. # D and # S denote the number of documents and sentences, respectively.

Steps (K)	**COMET** (↑)	**LTCR** (%, ↑)	**d-BLEU**	**Latency** (ms)
0 (Base)	79.85	82.10	28.50	142
1	**85.42**	90.30	**34.15**	**175**
2	85.45	91.50	34.10	210
3	84.80	**92.10**	33.85	248
5	83.10	92.40	32.20	325
10	78.50	92.60	29.40	512

Table 6: Impact of gradient optimization steps (K) on translation performance. Reducing spacing allows larger fonts.

Constraint Adherence vs. Catastrophic Forgetting. Increasing K allows the adapter parameters ϕ_t to fit the retrieved constraints $\mathcal{K}_t$ more closely, theoretically improving the prediction probability of the target terms. However, excessive updates on a small set of local constraints can lead to *overfitting*, causing the model to lose its general linguistic capabilities or hallucinate the constrained terms in inappropriate contexts—a phenomenon analogous to catastrophic forgetting in continual learning. Our preliminary experiments on the development set indicate that the marginal gain in constraint accuracy diminishes substantially beyond $K = 3$, while the perplexity for general tokens begins to rise.

Inference Latency. Since AdaDPI performs optimization during inference, computational efficiency is paramount. Each additional gradient step requires a complete backward pass through the adapter module, thereby linearly increasing the per-token latency. As shown in Figure 4, a single-step update ($K = 1$) provides the most favorable balance, yielding substantial improvements in consistency metrics (d-BLEU) with only a minor overhead (approximately 15% increase in total inference time compared to static baselines). Therefore, unless otherwise specified, we adopt a single-step gradient update strategy in our main experiments to prioritize real-time applicability.

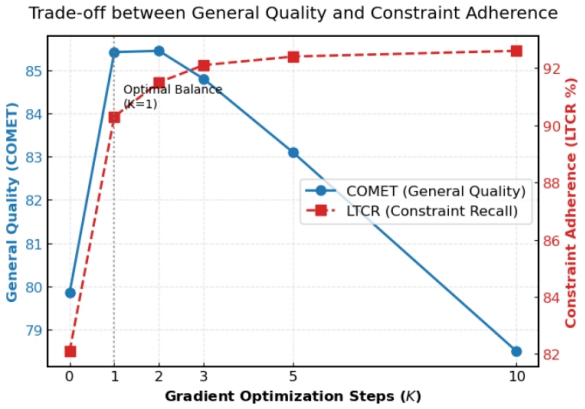

Figure 4: Sensitivity Analysis of Gradient Steps (K) on Guofeng. While increasing K improves the recall of specific constraints (LTCR), it degrades the model's general translation capability (COMET) due to overfitting. $K = 1$ represents the optimal Pareto frontier.

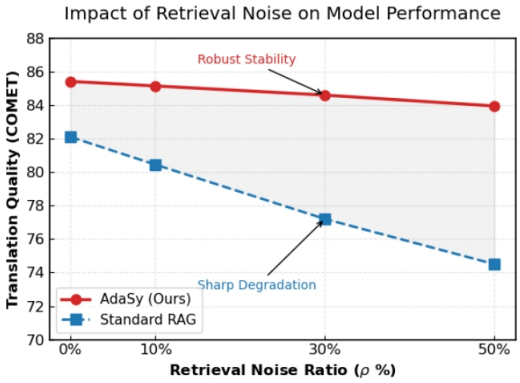

Figure 5: Robustness analysis against retrieval noise. Standard RAG suffers severe performance degradation as noise increases, due to attention distraction. AdaSy maintains high stability, leveraging the model's pre-trained priors to filter out noisy gradient updates.

A.6 Robustness to Retrieval Noise

In real-world deployment, retrieval systems are fallible and may return irrelevant or erroneous constraints. A robust document translation agent must maintain performance stability even under noisy input. To evaluate this resilience, we simulated a noisy retrieval environment on the GuoFeng En→Zh dataset. We artificially corrupted the retrieved constraint set $\mathcal{K}_t$ by replacing a proportion ρ of correct target terms with random irrelevant tokens from the vocabulary. We varied the noise ra-

Noise Ratio (ρ)	**0%**	**10%**	**30%**	**50%**
Standard RAG (COMET)	82.10	80.45	77.20	74.50
Degradation	-	*-1.65*	*-4.90*	*-7.60*
AdaSy (Ours)	**85.42**	**85.15**	**84.60**	**83.95**
Degradation	-	***-0.27***	***-0.82***	***-1.47***

Table 7: Performance degradation under varying retrieval noise ratios. AdaSy maintains high stability, whereas RAG suffers drastic drops.

tio $\rho \in \{0\%, 10\%, 30\%, 50\%\}$ and compared the performance degradation of AdaSy with that of Standard RAG.

Figure 5 and Table 7 summarize the results.

Standard RAG is highly susceptible to noise. As ρ increases to 30%, its COMET score drops precipitously (-4.9 points). This vulnerability stems from the attention mechanism: RAG treats retrieved tokens as explicit context. When erroneous terms are present in the prompt, the model's copy mechanism is easily distracted, leading it to generate noisy tokens directly in the translation. In contrast, AdaSy demonstrates remarkable stability. Even with 50% noise, the performance drop is minimal (-1.47 points), and the absolute score remains higher than the clean RAG baseline. We attribute this robustness to the regularization effect of the pre-trained prior. In AdaSy, noise is introduced as a gradient-update signal rather than as input tokens. Since we employ a small learning rate and a single-step update ($K = 1$), the model parameters exhibit "intrinsic inertia." Unless the noisy gradient is overwhelmingly strong, the parameter manifold remains anchored near the pre-trained optimum. Essentially, the model's inherent linguistic knowledge acts as a filter, rejecting updates that would violate fundamental semantic coherence.

A.7 Implementation and Alignment Details

To establish a strong foundational translation capability before applying the inference-time AdaDPI agent, we optimize the base model's static parameters using three paradigms: SFT, GRPO, and DAPO.

A.7.1 SFT

The SFT stage aligns the base LLM with the document-level machine translation format. Given a document dataset $\mathcal{D}$, the model is trained using the standard auto-regressive negative log-likelihood loss:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\sum_{t=1}^{|y|} \log \pi_\theta(y_t | x, \mathcal{C}_{<t}, y_{<t}) \right] \tag{13}$$

where x is the current source sentence and $\mathcal{C}_{<t}$ is the preceding discourse context.

A.7.2 GRPO

While GRPO is traditionally used for math/coding reasoning, we adapted it for "discourse reasoning" in DocMT. For each input context $q = (x, \mathcal{C}_{<t})$, we sample a group of G candidate translations $\{o_1, o_2, \ldots, o_G\}$ from the old policy π_{old}.

We design a DocMT-specific composite reward function $R(o_i)$ to evaluate each candidate:

$$R(o_i) = \lambda_1 R_{\text{COMET}}(o_i) + \lambda_2 R_{\text{Constraint}}(o_i) \tag{14}$$

Here, R_{COMET} measures general translation quality, and $R_{\text{Constraint}} \in \{0, 1\}$ is a rule-based reward that checks if the generated text correctly utilizes the specific terminology or resolves pronouns consistent with $\mathcal{C}_{<t}$. The advantages are calculated relatively within the group: $\hat{A}_i = \frac{R(o_i) - \text{mean}(R)}{\text{std}(R)}$.

The GRPO objective is formulated as:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}\Bigg[\frac{1}{G} \sum_{i=1}^{G} \min \Bigg(\frac{\pi_\theta(o_i|q)}{\pi_{\text{old}}(o_i|q)} \hat{A}_i, \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\text{old}}(o_i|q)}, 1-\epsilon, 1+\epsilon \right) \hat{A}_i \Bigg) - \beta \mathbb{D}_{\text{KL}}(\pi_\theta || \pi_{\text{ref}}) \Bigg] \tag{15}$$

A.7.3 DAPO

To further enhance the model's ability to follow context, we utilize DAPO. We construct preference pairs $(y_{\text{chosen}}, y_{\text{rejected}})$ tailored for DocMT:

- y_{chosen}: The high-quality reference translation that accurately maintains stylistic consistency and correct terminology.
- y_{rejected}: A perturbed version of the translation where we deliberately introduce discourse errors (e.g., swapping character names, using inconsistent terminology, or injecting ambiguous pronouns).

The model is optimized to maximize the margin between the chosen and rejected translations:

$$\mathcal{L}_{\text{DPO}}(\theta) = -\mathbb{E}_{(x,y_c,y_r)\sim\mathcal{D}}\Bigg[\log\sigma\Bigg(\beta\log\frac{\pi_\theta(y_c|x,\mathcal{C})}{\pi_{\text{ref}}(y_c|x,\mathcal{C})} - \beta\log\frac{\pi_\theta(y_r|x,\mathcal{C})}{\pi_{\text{ref}}(y_r|x,\mathcal{C})}\Bigg)\Bigg] \tag{16}$$

A.7.4 Model Configuration and Architecture

We utilize Qwen2-7B-Instruct as the foundation. The system consists of a frozen Reasoner and a trainable Weaver (via LoRA).

- **LoRA Specs:** Rank $r = 16$, $\alpha = 32$, Dropout=0.1. We target `q_proj` and `v_proj` modules.
- **Latent Memory:** We employ 128-dimensional trainable latent query vectors (Prompt and Inference Latents) projected via a dedicated linear layer.
- **LUM Threshold:** The uncertainty threshold τ is set to **0.6**. The lexical entropy is normalized by $\log|V|$ to ensure $S(t) \in [0, 1]$.

A.7.5 Training and Optimization Protocol

The training is conducted on 4 $\times$ NVIDIA A100 GPUs (80GB).

- **Optimizer:** AdamW with a learning rate of 1×10^{-5} for SFT and 2×10^{-4} for meta-learning (η, α, β).
- **Scheduler:** Cosine decay with a 0.1 warmup ratio.
- **Batch Size:** Global batch size of 16 (4 GPUs $\times$ 1 batch/device $\times$ 4 gradient accumulation steps).
- **Evaluation:** Validation is performed every 150 steps. We use `load_best_model_at_end` based on the lowest validation loss.

A.7.6 Data Preprocessing and Handling

All document-level data (GuoFeng, IWSLT) are processed as follows:

- **Segmentation:** Documents are grouped by `document_id`. Extremely long narratives are segmented into fragments with `max_turns=85`.
- **Sequence Length:** Maximum tokens are restricted to 30,000 (ZH-EN) and 25,000 (EN-ZH) to prevent OOM errors.
- **Memory Injection:** Excluding the first turn, agentic memory is dynamically injected into each subsequent turn. Raw source texts are preserved for memory indexing and retrieval.

A.8 Detailed Prompt Templates

As part of the Context-to-Parameter Integrator (CPI), we utilize a multi-stage agentic pipeline to generate the external knowledge base $\mathcal{K}$. This section details the specific instructions for each stage.

A.8.1 Standard ChatML Template for LLM Baselines

To ensure fair comparison, all LLM-based systems (e.g., Qwen2-7B-Instruct) follow the standardized ChatML format.

TEMPLATE: STANDARD CHATML FORMAT

```
<|im_start|>system
You are a professional translator.
Translate the following English text
into Chinese.<|im_end|>

<|im_start|>user
Translate the following English sentence
into Chinese.
Source: {xt}
Target:<|im_end|>

<|im_start|>assistant
{yt}<|im_end|>
```

A.8.2 Multi-stage Summarization Pipeline

We construct the *Dynamic Discourse Cache* through a three-stage summarization and fusion process to minimize information redundancy.

STAGE 1: SOURCE PARAGRAPH SUMMARIZATION

```
Summarize the following {src_lang} paragraph
in at most 100 words.
Do not output digits, lists, bullets, or
artifacts; return a single clean {src_lang}
paragraph:

{src_para}

Return only the summary:
```

STAGE 2: TARGET PARAGRAPH SUMMARIZATION

```
Summarize the following {tgt_lang} paragraph
in at most 3 concise sentences.
Do not output digits, lists, bullets, or
artifacts; return a single clean paragraph:

{tgt_para}

Return only the summary:
```

STAGE 3: SUMMARY FUSION AND REFINEMENT

```
Merge two {src_lang}/{tgt_lang} summaries into one concise, non-redundant summary:

Summary A: {summary_1}
Summary B: {summary_2}

Output:
```

A.8.3 Lexical Extraction and Retrieval

These prompts are utilized to extract entity-level constraints and retrieve the most relevant historical evidence.

AGENTIC TERMINOLOGY ANNOTATION

```
You are an {src_lang}-{tgt_lang} bilingual expert. Given an {src_lang} source sentence with its {tgt_lang} translation, you need to annotate all the proper nouns.

Return ONLY a comma-separated list of pairs 'SRC - TGT'. If none, return 'N/A'.
Example: "Paris" - "巴黎", "Allen" - "艾伦".

Now annotate:
SRC: {src}
TGT: {tgt}

Return:
```

RELEVANCE-BASED EVIDENCE RETRIEVAL

```
You are a linguistic expert. Given a list of sentences and a query, find the {top_num} sentences most relevant to the request.

Sentence list:
{sentence_list}

Query: {query}

Note: respond ONLY with chosen numbers as a comma-separated list.

Chosen numbers:
```

A.8.4 Context Window Construction Format

When injecting the retrieved evidence into the prompt as implicit context, we adopt a delimiter-based structural format:

IN-CONTEXT INJECTION FORMAT

```
Preceding texts:
\n <English text> {sent_1} || {sent_2} || {sent_3}
\n <Chinese text> {trans_1} || {trans_2} || {trans_3}
```

A.9 Qualitative Case Study: Terminology Internalization

To further investigate why the LTCR score drops by approximately 5 points when CPI is disabled (as reported in Table 2), we conduct a qualitative analysis focusing on the internalization of domain-specific terms.

Case Analysis: The "Grand Master Realm" In the literary *GuoFeng* dataset, specific cultivation ranks are critical for narrative consistency. A prime example is the term **"Grand Master Realm"**.

- **Baseline (Standard Context-aware Prompting):** Even when the correct term mapping is provided in the preceding context window, the model often fails to attend to it due to **attention dilution** in long sequences. Consequently, it inconsistently translates the term as *"Great Perfection Realm"* (大圆满) instead of the established *"Grand Master Realm"* (宗师境).
- **AdaDPI (With CPI):** Once the mapping "Grand Master Realm → 宗师境" is internalized into the adapter parameters through a single-step gradient update, the model prioritizes this specific translation as its intrinsic knowledge.

Discussion on Internalization Advantage As illustrated in this case, AdaDPI's parametric adaptation ensures that the term remains consistent across hundreds of subsequent chapters. This consistency is maintained even when the context window becomes saturated or when the original constraint is no longer present in the active prompt. This demonstrates that CPI effectively transforms a transient external hint into a persistent model preference, resolving the "amnesic" behavior common in static LLMs.

A.10 Comparison with Latest Frontier Models

To address the rapidly evolving landscape of LLMs, we conducted additional experiments using the recently released GPT-5.2 and Gemini 3.0 Pro APIs.

Methods	COMET	LTCR	d-BLEU
Gemini 3.0 Pro	84.50	85.10	32.50
GPT-5.2	84.85	85.65	32.90
AdaDPI-DAPO (Ours)	**85.42**	**90.30**	**34.15**

Table 8: Comparison between AdaDPI and trillion-parameter frontier models on GuoFeng (En→Zh). The best results are in bold.

These evaluations were performed on the challenging *GuoFeng* (En→Zh) dataset to test the upper bounds of document-level consistency.

Analysis As shown in Table 8, although GPT-5.2 and Gemini 3.0 Pro exhibit formidable translation capabilities, AdaDPI-DAPO still achieves a significant margin (e.g., **+4.65 points** in LTCR over GPT-5.2). This results from our method's ability to *internalize* context into model weights rather than simply relying on extended attention windows, proving that parametric adaptation is more reliable for maintaining document-specific norms than zero-shot scaling.