Is Human Annotation Necessary? Iterative MBR Distillation for Error Span Detection in Machine Translation

Anonymous ACL submission

Abstract

Error Span Detection (ESD) is a crucial subtask in Machine Translation (MT) evaluation, aiming to identify the location and severity of translation errors. While fine-tuning models on human-annotated data improves ESD performance, acquiring such data is expensive and prone to inconsistencies among annotators. To address this, we propose a novel self-evolution framework based on Minimum Bayes Risk (MBR) decoding, named **Iterative MBR Distillation for ESD**, which eliminates the reliance on human annotations by leveraging an off-the-shelf LLM to generate pseudo-labels. Extensive experiments on the WMT Metrics Shared Task datasets demonstrate that models trained solely on these self-generated pseudo-labels outperform both unadapted base model and supervised baselines trained on human annotations at the system and span levels, while maintaining competitive sentence-level performance.

1 Introduction

In the landscape of Machine Translation (MT) evaluation, Error Span Detection (ESD) plays a pivotal role. By identifying not only the presence of errors but also their precise locations and severity, ESD provides the granular feedback essential for MT model correction (Blain et al., 2023; Freitag et al., 2024; Lavie et al., 2025).

However, despite its critical importance, the advancement of ESD models is severely constrained by major bottlenecks in supervision. Unlike sentence-level quality estimation, ESD demands fine-grained span-level annotation, which is a costly process that requires bilingual expertise (Freitag et al., 2021a; Kocmi et al., 2024). Furthermore, the gold-standard status of human labels is challenged by inherent subjectivity. As noted by Proietti et al. (2025), under existing annotation protocols, the agreement among human annotators is merely comparable to the agreement between automatic annotators and humans. Consequently, public ESD datasets remain limited in scale and consistency compared to the massive corpora available for general MT tasks. This data scarcity and inherent subjectivity limit the potential of supervised ESD models (Juraska et al., 2025) and prompt a fundamental question: *Is human annotation strictly necessary to train ESD models?*

Method	SPA	Acc^*_{eq}	SOFTF1
Base Model	.851	.518	.874
Gold-SFT	.826	.571	.915
MBR Distill	**.864**	**.573**	**.939**

Table 1: Our self-trained MBR Distillation framework surpasses both base model and human-annotated data trained baseline (Gold-SFT) at the system (SPA) and span (SOFTF1) levels.

In this work, we challenge the reliance on human annotation data by leveraging purely synthetic training data generated through **Iterative Minimum Bayes Risk (MBR) Distillation for ESD**, a novel framework for ESD model training. Inspired by the recent success of Large Language Models (LLMs) in self-evolution (Tao et al., 2024), we hypothesize that LLMs, possessing latent evaluation capabilities, can synthesize their own training signals. To construct pseudo-labels, we employ MBR decoding (Stolcke et al., 1997; Goel and Byrne, 2000). By selecting candidates that minimize the expected risk across a diverse sample set, this approach effectively leverages the model's internal "consensus" to filter out noise (Lyu et al., 2025).

Our approach operates as an iterative cycle. Starting with an off-the-shelf LLM, we generate diverse candidate error spans and select pseudo-labels via MBR decoding with the SOFTF1 utility function (Lyu et al., 2025). The model is then fine-tuned on these self-generated labels using diverse training objectives, including Supervised

Fine-Tuning (SFT), Direct Preference Optimization (DPO) (Rafailov et al., 2023), and Kahneman-Tversky Optimization (KTO) (Ethayarajh et al., 2024). Through iterations of this cycle, the model drives its own continuous improvement.

Our experiments yield a compelling and counter-intuitive finding: models trained exclusively on MBR-generated pseudo-labels outperform those fine-tuned on human annotations at both the system and span levels, while remaining competitive at the sentence level (as highlighted in Table 1).

The main contributions of this paper are summarized as follows:

- We propose Iterative MBR Distillation for ESD, a novel self-evolution framework that completely bypasses the need for human annotations.
- We provide a comprehensive evaluation across various training objectives (SFT, DPO, KTO). Empirical results demonstrate that our framework not only surpasses unadapted base model but also outperforms models trained on human annotations at the system and span levels, signaling a paradigm shift in ESD model training.

2 Related Work

2.1 MT Automatic Metrics

Early MT metrics such as BLEU (Papineni et al., 2002) and chrF (Popović, 2015) rely on surface-level matching and output a single scalar score per sentence. Neural metrics, including COMET (Rei et al., 2020) and MetricX (Juraska et al., 2025), show higher correlations with human judgments but typically still produce only sentence-level scores. As human evaluation has shifted toward multidimensional annotations, such as Multidimensional Quality Metrics (MQM) (Freitag et al., 2021a), automatic metrics have targeted finer-grained evaluation. Generative approaches, such as GEMBA-MQM (Kocmi and Federmann, 2023), prompt LLMs to output MQM-style error spans.

2.2 Self-Evolution of LLMs

Self-evolution empowers LLMs to autonomously acquire and refine knowledge by synthesizing training data and learning from their own generated feedback (Tao et al., 2024). In the context of machine translation, recent studies have explored self-evolutionary frameworks to continuously enhance translation capabilities. For example, methods like self-evolution knowledge distillation allow models to dynamically assess token-level learning difficulties and adjust prior knowledge during the training process (Song et al., 2025). Similarly, iterative self-correction frameworks leverage the LLM's own estimations to refine initial translations, demonstrating that models can bootstrap their performance without relying heavily on external human supervision (Feng et al., 2025).

While self-evolution has shown significant promise in text generation and translation, most existing works do not address the self-evolution of ESD models. The only closely related study is by Lyu et al. (2025) who introduced MBR decoding for generative ESD models. To alleviate computational bottlenecks, they investigated MBR distillation to transfer the superior performance of MBR decoding into the model weights. They employed DPO to train the model to approximate MBR outputs, confirming that the distilled model using greedy search can match the performance of full MBR decoding. However, their exploration has two key limitations: (1) it did not include a performance comparison against models trained on human annotations, and (2) it did not propose an iterative training mechanism. Our work bridges these gaps by introducing Iterative MBR Distillation for ESD, demonstrating that a self-evolving iterative framework can effectively surpass the performance of models fine-tuned on human-annotated data.

3 Preliminaries

3.1 Error Span Detection Model

Formally, let $x = (s, t)$ denote an input pair, where s is a source sentence and t is its corresponding translation. The ESD model, denoted as M_θ with parameters θ, generates an output E. Here, E represents a set of error annotations, specifically the character positions and severity values for each error span detected in t.

3.2 MBR Decoding for ESD

MBR decoding has been extensively utilized in sequence generation tasks, such as automatic speech recognition (Stolcke et al., 1997; Goel and Byrne, 2000) and MT (Eikema and Aziz, 2020; Freitag et al., 2022a) to address the shortcomings of Maximum a Posteriori (MAP) decoding. While MAP relies on the assumption that the model's probability perfectly aligns with generation quality, MBR

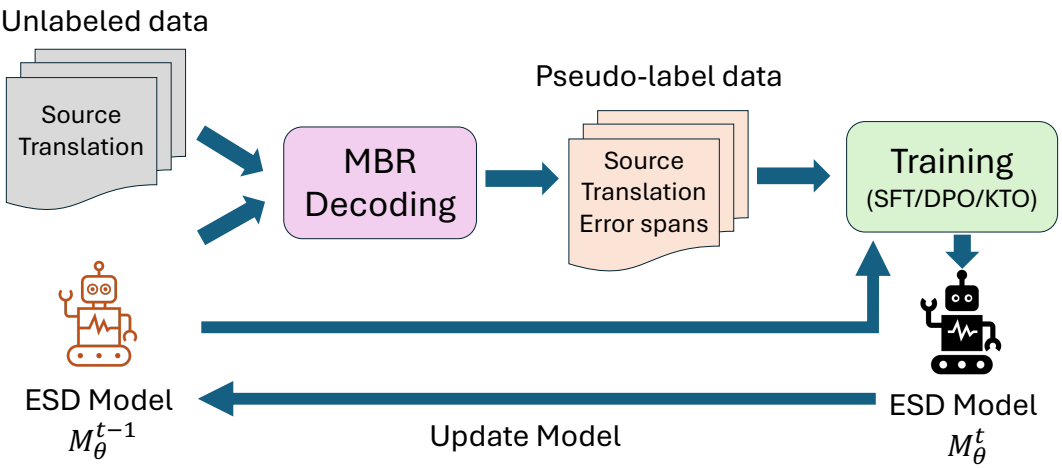

Figure 1: Overview of the Iterative MBR Distillation framework for ESD. Starting with unlabeled source-translation pairs, the model generates diverse candidate error spans. MBR decoding then evaluates these candidates to assign utility scores, identifying high-quality pseudo-labels (e.g., the best and worst hypotheses). Finally, the model is fine-tuned on these self-generated labels using SFT, DPO, or KTO. This cycle repeats iteratively, enabling the model to self-evolve and refine its ESD capabilities without relying on human annotations.

Algorithm 1 Iterative MBR Distillation for ESD

Require: Unlabeled data $\mathcal{D}_u = \{x_i\}_{i=1}^N$; base model $M_{\theta^{(0)}}$; iterations T; candidate count C; utility $u = \text{SOFTF1}$; loss function $\mathcal{L}$

for $t = 1$ to T **do**
  $\mathcal{D}^{(t)} \leftarrow \emptyset$
  for each $x_i \in \mathcal{D}_u$ **do**
    Sample $\mathcal{C}_i^{(t)}$ ($|\mathcal{C}_i^{(t)}| = C$) from $M_{\theta^{(t-1)}}(\cdot \mid x_i)$
    Set $\mathcal{S}_i^{(t)} \leftarrow \mathcal{C}_i^{(t)}$
    Calculate $score_E^{MBR}$ for all $E^c \in \mathcal{C}_i^{(t)}$ using Eq. 1
    $E_i^+ \leftarrow \text{argmax}_{E \in \mathcal{C}_i^{(t)}} score_E^{MBR}$
    $E_i^- \leftarrow \text{argmin}_{E \in \mathcal{C}_i^{(t)}} score_E^{MBR}$
    if $\mathcal{L}$ is $\mathcal{L}_{\text{SFT}}$ **then**
      Add (x_i, E_i^+) to $\mathcal{D}^{(t)}$
    else if $\mathcal{L}$ is $\mathcal{L}_{\text{DPO}}$ **and** $E_i^+ \neq E_i^-$ **then**
      Add (x_i, E_i^+, E_i^-) to $\mathcal{D}^{(t)}$
    else if $\mathcal{L}$ is $\mathcal{L}_{\text{KTO}}$ **then**
      Add (x_i, E_i^+) as E^+ to $\mathcal{D}^{(t)}$
      if $E_i^+ \neq E_i^-$ **then**
        Add (x_i, E_i^-) as E^- to $\mathcal{D}^{(t)}$
      end if
    end if
  end for
  Update $\theta^{(t)}$ by minimizing $\mathcal{L}(\theta^{(t)})$ on $\mathcal{D}^{(t)}$
end for
return $M_{\theta^{(T)}}$

aims to minimize the expected risk, or equivalently, maximize the expected utility, defined by a task-specific evaluation metric, e.g., BLEU (Papineni et al., 2002) for MT.

Recently, Lyu et al. (2025) proposed applying MBR decoding to generative ESD models. By selecting the hypothesis (error annotation) that maximizes utility against a set of samples drawn from the model, MBR allows ESD models to effectively aggregate consensus and mitigate prediction errors inherent in standard MAP decoding.

Formally, let $\mathcal{C} = \{E_1^c, \ldots, E_N^c\}$ be the set of candidate hypotheses. MBR decoding first assigns a $score_E^{MBR}$ to each candidate E^c, and then selects the hypothesis E^{MBR} with the highest score:

$$E^{MBR} = \text{argmax}\, score_E^{MBR},$$

where $score_E^{MBR}$ is defined as the expected utility with respect to the human annotation distribution $\mathcal{Y}$:

$$score_E^{MBR} = \mathbb{E}_{y \in \mathcal{Y}}\left[u(E^c, y) \mid x\right].$$

Since the human annotation is unavailable during inference, this expectation is approximated using a finite set of support hypotheses $\mathcal{S}$ sampled from the model. Formally, for a given input, we generate a support set $\mathcal{S}$ (where typically $\mathcal{S} = \mathcal{C}$) from the model $M_\theta(\cdot|x)$.

If we assume that the probability of each support hypothesis is equal, and do not use the model-estimated probabilities (which is consistent with

the fundamental assumption of MBR, that the model-estimated probabilities are not reliable), then the MBR score of the candidate E^c can be approximated by its average utility to the support hypotheses:

$$score_E^{MBR} \approx \frac{1}{|\mathcal{S}|} \sum_{E^s \in \mathcal{S}} u(E^c, E^s), \qquad (1)$$

where $u(\cdot, \cdot)$ is the utility function.

3.3 ESD Model Training

While LLM-based ESD models can perform inference without ESD-specific training (Kocmi and Federmann, 2023), fine-tuning on ESD datasets has proven to improve model performance (Juraska et al., 2025). The goal of training an ESD model is to tune its parameters to make its predictions more closely resemble the pre-defined target annotations (e.g., human annotations).

Formally, we assume access to a labeled dataset $\mathcal{D} = \{(x_i, E_i^t)\}_{i=1}^N$, where E_i^t denotes the target annotation. The general training objective is to find the optimal parameters θ^* that minimize a specific loss function $\mathcal{L}$ over the data distribution:

$$\theta^* = \arg\min_\theta \mathbb{E}_{(x,E^t)\sim\mathcal{D}}[\mathcal{L}(\theta; x, E^t)].$$

This general objective can be optimized through different paradigms. For instance, we can employ SFT to perform maximum likelihood estimation directly on the target annotation, or utilize preference learning, such as DPO and KTO.

Supervised Fine-Tuning (SFT) SFT is the standard approach for alignment which maximizes the conditional log-likelihood of E_i^t:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\sum_i \log M_\theta(E_i^t \mid x_i).$$

Direct Preference Optimization (DPO) DPO (Rafailov et al., 2023) is a pairwise preference optimization method designed to align language models with human feedback. Its primary optimization objective is to increase the margin of the estimated probabilities between the preferred and dispreferred outputs. Let M_θ denote the policy being optimized and M_{ref} the frozen reference policy. Given a preference pair (E_i^+, E_i^-) where $E_i^+ \neq E_i^-$ and the hypothesis E_i^+ is preferred over E_i^-, DPO optimizes:

$$\mathcal{L}_{\text{DPO}}(\theta) = -\sum_i \log \sigma(\lambda[\log \frac{M_\theta(E_i^+ \mid x_i)}{M_\theta(E_i^- \mid x_i)} - \log \frac{M_{\text{ref}}(E_i^+ \mid x_i)}{M_{\text{ref}}(E_i^- \mid x_i)}]),$$

where λ is the regularization parameter (referred to as β in standard DPO paper (Rafailov et al., 2023), but distinguished here to avoid confusion with KTO hyperparameters).

Kahneman-Tversky Optimization (KTO) KTO is a training objective designed to handle binary feedback signals, eliminating the strict requirement for paired preference data (Ethayarajh et al., 2024). While DPO dictates that each input x must be accompanied by a paired (E^+, E^-) such that $E^+ \neq E^-$, KTO evaluates examples individually. It requires only that at least one of E^+ (desirable) or E^- (undesirable) is present for a given x, although both can be utilized if available. Following Ethayarajh et al. (2024), the loss function is defined as:

$$\mathcal{L}_{\text{KTO}}(\theta) = \mathbb{E}_{(x,E)\sim\mathcal{D}}[w_E(1 - \sigma(v(x, E)))],$$

where $E \in \{E^+, E^-\}$ and w_E is the corresponding weight hyperparameter depending on whether E is desirable (E^+) or undesirable (E^-). The value function $v(x, E)$ is defined piecewise to reflect the asymmetry of the Kahneman-Tversky value model:

$$v(x, E) = \begin{cases} \beta(r_\theta(x, E) - z_{\text{ref}}) & \text{if } E = E^+ \\ \beta(z_{\text{ref}} - r_\theta(x, E)) & \text{if } E = E^- \end{cases}$$

Here, the implicit reward r_θ is the log-ratio between the policy and reference model:

$$r_\theta(x, E) = \log \frac{M_\theta(E \mid x)}{M_{\text{ref}}(E \mid x)},$$

and z_{ref} is a reference point (typically the KL divergence) used to control the saturation of the loss.

4 Proposed Method: Iterative MBR Distillation for ESD

To combine the performance benefits of MBR decoding with the inference efficiency of direct generation, we propose an iterative self-training framework, as illustrated in Figure 1. The core idea is to distill the knowledge from the MBR scoring process back into the model parameters. As shown

in Algorithm 1, the method iterates through cycles of data generation and model updates. In each iteration, the model generates candidates for unlabeled data, assigns MBR scores to identify the best (and potentially worst) hypotheses, and constructs a synthetic training set. The model is then updated using one of the training objectives described in Section 3.3 (SFT, DPO, or KTO), effectively amortizing the cost of MBR decoding.

5 Experiments

5.1 Experimental Setup

Datasets and Model We used all source and translation pairs (s, t) from the WMT20–23 Metrics Shared Task (Mathur et al., 2020; Freitag et al., 2021b, 2022b, 2023) datasets as $\mathcal{D}_u$. We used the MQM annotations from the WMT24 Metrics Shared Task (Freitag et al., 2024) as our test set, which covers three translation directions: English→German, English→Spanish, and Japanese→Chinese. We used *Qwen3-30B-A3B-Instruct-2507* (Team, 2025), a popular instruction-tuned LLM, as the base ESD model $M_{\theta^{(0)}}$. During testing, we employed a greedy search. We employed a GEMBA-MQM-style prompt (Kocmi and Federmann, 2023), supplemented by the disambiguation technique proposed by Juraska et al. (2025), which involves generating additional context for spans with ambiguous surface forms (2). We used guided generation (Willard and Louf, 2023) to ensure that each hypothesis adheres to the specified JSON format.

Iterative MBR Distillation Configuration Due to computational resource constraints, we set $T = 3$ and $C = 256$. Following Lyu et al. (2025), we used SOFTF1 as the MBR utility function for its robustness to empty annotations, and generated the candidate set via Top-K ($K = 10$) sampling at $temperature = 2.0$.

Baselines We compare against:

- **xCOMET:** The state-of-the-art evaluation metric (Guerreiro et al., 2024). We evaluate its largest *XXL* version in both reference-based (xCOMET) and reference-free (xCOMET-QE) settings. We report both its sentence-level regression scores (*-Reg) and token-level error span predictions (*-ESD).
- **Base Model:** direct prompting of $M_{\theta^{(0)}}$ with a GEMBA-MQM-style prompt.

```
Based on the source segment and machine
    translation surrounded with triple
    backticks, identify error types in
    the translation and classify them.
    The categories of errors are:
1. Accuracy (addition, mistranslation,
    omission, untranslated text,
    hallucination)
2. Fluency (character encoding,
    grammar, inconsistency,
    punctuation,
    whitespace,capitalization,
    register, spelling, unnatural flow)
3. Style (awkward)
4. Terminology (inappropriate for
    context, inconsistent use)
5. Locale convention (address,
    currency, date, measurement, name,
    number, telephone, time)
6. Non-translation (None sub-category)
7. Other (None sub-category)
8. No-error (None sub-category)
    Each error is classified as one of
        three categories: critical,
        major, and minor. Critical
        errors inhibit comprehension of
        the text. Major errors disrupt
        the flow, but what the text is
        trying to say is still
        understandable. Minor errors
        are technically errors, but do
        not disrupt the flow or hinder
        comprehension.

Please output in JSON format where each
    error span contains the following
    fields: span, severity, category,
    sub_category, and span_with_context
    (span_with_context is optional. The
    shortest context text that can
    reliably eliminate ambiguity. Only
    include if the error span is not
    unique.)
If no errors are found, output an empty
    list `[]`
Please do not output anything else.

{source_lang} source:
    ```{source_seg}```
{target_lang} translation:
    ```{target_seg}```
```

Figure 2: Prompt template for the ESD models.

- **Gold-*:** fine-tune $M_{\theta^{(0)}}$ on human annotations using SFT/DPO/KTO. The training data used was $\mathcal{D}_u$ with additional human annotations. For DPO and KTO, we use human annotations as E^+ and the output of $M_{\theta^{(0)}}$ with greedy decoding as E^-.

Training Details We randomly sampled a subset of data as the development set to select the checkpoint with the lowest training loss, while the remainder was used for model training, resulting in

Method	SPA	Acc^*_{eq}	SOFTF1	F1
xCOMET-ESD	.757	.553	.889	.302
xCOMET-Reg	.844	**.581**	-	-
xCOMET-QE-ESD	.688	.541	.879	.289
xCOMET-QE-Reg	.825	.549	-	-
Base Model	.851	.518	.874	.396
Gold-SFT	.826	.571	.915	.454
Gold-DPO	.673	.518	.909	.472
Gold-KTO	.689	.567	.910	.469
MBR Distill ($T = 1$, SFT)	.851	.578	.920	.497
MBR Distill ($T = 2$, SFT)	.862	.564	**.939**†	.560†
MBR Distill ($T = 3$, SFT)	.857	.530	**.939**†	**.577**†
MBR Distill ($T = 1$, DPO)	.843	.569	.928	.528†
MBR Distill ($T = 2$, DPO)	.860	.571	.932†	.539†
MBR Distill ($T = 3$, DPO)	.849	.572	.933†	.538†
MBR Distill ($T = 1$, KTO)	.842	.573	.925	.529
MBR Distill ($T = 2$, KTO)	**.864**†	.573	.933†	.535†
MBR Distill ($T = 3$, KTO)	.850	.571	**.939**†	.575†

Table 2: Evaluation results for the WMT 2024 Metrics Shared Task. We employ SPA, Acc^*_{eq}, and SOFTF1 as metrics at the system, sentence, and span levels, respectively. We report average scores across all translation directions. The best score is in bold. † indicates significantly better performance than the best baseline method within the baseline group on all translation directions ($p < 0.05$). xCOMET. *-Reg and *-ESD denote regression (score-only) and token-classification modes, respectively. xCOMET-QE-* indicates a reference-free variant; others are reference-based.

a development set of approximately 10k examples and a training set of 193k examples. We performed full-parameter fine-tuning on the model using the AdamW optimizer (Loshchilov and Hutter, 2019) with a learning rate of $1e-6$ and a warm-up step ratio of 0.05. We set $\lambda = 0.5$ for DPO and $\beta = 0.5$ for KTO. We set the batch size to 256 (accounting for gradient accumulation and multi-GPU parallelism) and the maximum training epoch to 3. During training, padding-free computations (Kundu et al., 2024) and Flash Attention (Dao, 2024) were used to accelerate training. Our training codebase is based on MS-Swift (Zhao et al., 2024), and training was completed on six machines, each equipped with 8× NVIDIA H200 GPUs. Training of all models ran almost continuously on these machines for 10 weeks, with approximately 70% of that time spent generating the large number of samples required for MBR decoding (we used vLLM (Kwon et al., 2023) to accelerate this process).

Evaluation Following prior work on ESD (Freitag et al., 2023, 2024; Lyu et al., 2025), we used Soft Pairwise Accuracy (SPA) (Thompson et al., 2024), Pairwise Accuracy with Tie Calibration (Acc^*_{eq}) (Deutsch et al., 2023) and SOFTF1 for system-, sentence- and span-level evaluation, respectively. For SPA and Acc^*_{eq}, we employed the PERM-BOTH test (Deutsch et al., 2021). For SOFTF1, we used paired bootstrap resampling (Koehn, 2004) for significance tests. For each evaluation metric, we adopted the best baseline method within the baseline group as the baseline for significance tests. For example, Base Model serves as the baseline for SPA, while Gold-KTO serves as the baseline for Acc^*_{eq}. For all ESD methods, system-level scores were calculated by averaging all sentence-level scores within that MT system.

5.2 Experimental Results

Main Result Table 2 summarizes the overall performance. In our reported results, the notation "MBR Distill (T, training objective)" indicates our proposed method trained for T iterations using the specific training objectives (SFT, DPO, KTO). MBR Distill ($T = 2$, KTO) consistently improves over the Base Model and Gold-* baselines on SPA and SOFTF1. Furthermore, when compared to the state-of-the-art xCOMET models, our MBR Distill framework significantly outperforms both xCOMET-ESD and xCOMET-QE-ESD at the system and span levels. Thus,

we conclude that our approach outperforms base model and human-annotated fine-tuned models at the system- and span-level. However, on Acc^*_{eq}, MBR Distill performed only as well as the baselines. Therefore, we conclude that our approach outperforms base model without ESD-specific and human-annotation-trained models at the system and span levels, while matching the performance of these baselines at the sentence-level. Given that our approach does not rely on any human-annotated ESD data, we believe it offers a novel paradigm for training ESD models.

Effect of Iterations T When T is in $[1, 2]$, there is a strong positive correlation between T and the scores on SPA, Acc^*_{eq}, and SOFTF1, indicating the effectiveness of the iteration. However, when further increasing to $T = 3$, the scores on these metrics actually decline compared to $T = 2$. We analyze the reasons for this in Section 6.1.

Effect of Different Training Objectives We did not observe any evidence that one training objective outperforms the others with Iterative MBR Distillation. Given that SFT is computationally cheaper (since it does not require a reference model M_{ref}), we recommend using SFT as the preferred training objective for Iterative MBR Distillation.

6 Analysis

6.1 Analysis of $T = 3$ Poor Performance

Method	Utility Var
Base Model	2.39×10^{-2}
MBR Distill ($T = 1$, SFT)	1.42×10^{-2}
MBR Distill ($T = 2$, SFT)	1.69×10^{-3}
MBR Distill ($T = 1$, DPO)	4.86×10^{-4}
MBR Distill ($T = 2$, DPO)	2.96×10^{-4}
MBR Distill ($T = 1$, KTO)	3.11×10^{-3}
MBR Distill ($T = 2$, KTO)	1.57×10^{-4}

Table 4: Estimated utility variance for different training configurations. The variance consistently decreases as the number of iterations (T) increases across all training objectives (SFT, DPO, KTO).

As mentioned in Section 5.2, when T increases from 2 to 3, the performance of MBR Distill stagnates or even declines. We hypothesize that this is because it becomes difficult to further decrease the MBR utility estimation error.

To verify this hypothesis, following the theory of Kamigaito et al. (2025), we indirectly measure the utility error by calculating the variance of the estimated utility. The results are shown in Table 4. We observe that for any variant of MBR Distill, as T increases, the utility variance decreases progressively. This indicates that the utility estimation error struggles to decrease further as the iterations scale up. As a potential direction for improvement, we will explore methods to maintain the surface diversity of candidates during iterative training to indirectly alleviate the issue of declining utility variance in future work.

6.2 Distribution and Severity of Error Spans

To better understand the behavior of the models, we investigate the distribution of error severities. Table 5 presents the total number of "Major" and "Minor" error spans, as well as their ratio, for human annotations, the untrained Base Model, the human-annotated Gold-SFT, and our best-performing iterative model, MBR Distill ($T = 2$, KTO).

The results reveal that the unadapted Base Model heavily over-predicts the number of Major errors, resulting in a Major-to-Minor ratio (1.02) that is drastically higher than the human reference (0.58). In contrast, both the supervised Gold-SFT and our self-trained MBR Distillation effectively calibrate this distribution. Remarkably, without ever explicitly training on human annotations, MBR Distill ($T = 2$, KTO) adjusts the model's severity recognition capabilities, achieving a Major-to-Minor ratio (0.63) that is highly aligned with human evaluation standards.

Counter-intuitively, this calibration occurs despite the fact that our chosen utility function (SOFTF1) does not explicitly penalize the generation of excessive errors, nor does it inherently favor Minor errors over Major ones. This observation is consistent with the findings of Lyu et al. (2025). It implies that the MBR distillation process intrinsically relies on the model's internal consensus to filter out overly harsh, hallucinatory evaluations (which are often idiosyncratic and lack consensus across the candidate set), thereby yielding a more natural and human-like severity distribution.

To provide an intuitive perspective on this phenomenon, Table 3 shows a qualitative example. When faced with a translation that is grammatically correct but stylistically awkward due to literal translation (e.g., translating "straightforward" to "直接的"), the Base Model tends to aggressively aggregate the text into a single "Major" error span. Conversely, MBR Distillation successfully learns

Source **Translation**	The user interface is straightforward and easy to navigate. 用户界面是直接的并且很容易去导航。
Human	Span: “直接的” (Minor) Span: “去导航” (Minor)
Base Model	Span: “是直接的并且很容易去导航” (**Major**) ← *Over-penalizes translation style.*
Gold-SFT	Span: “直接的” (Minor) Span: “去导航” (Minor)
MBR Distill ($T = 2$, KTO)	Span: “直接的” (Minor) Span: “去导航” (Minor) ← *Successfully calibrates severity without human data.*

Table 3: A qualitative example illustrating how the unadapted Base Model tends to over-predict error severity, grouping minor stylistic awkwardness (literal translation) into a single “Major” error. In contrast, both the human-trained Gold-SFT and our self-trained MBR Distill correctly identify the granular issues as “Minor” errors, aligning closely with human judgment.

Method	**Major**	**Minor**	$\frac{\textbf{Major}}{\textbf{Minor}}$
Human	9.0K	15.5K	0.58
Base Model	16.4K	16.1K	1.02
Gold-SFT	5.4K	11.8K	0.46
MBR Distill ($T = 2$, KTO)	13.7K	21.9K	0.63

Table 5: The total count and ratio of Major versus Minor error spans predicted by different models compared to human annotations on the WMT24 Metrics Shared Task test set.

to tone down this overly critical behavior, identifying the specific spans as “Minor” issues, mirroring the behavior of models trained strictly on human annotations.

7 Conclusions and Future Work

In this work, we challenged the traditional reliance on costly and subjective human annotations for training ESD models. We introduced Iterative MBR Distillation, a self-evolution framework that leverages MBR decoding to generate high-quality, synthetic training signals from an LLM’s own predictions. Through extensive experiments evaluating SFT, DPO, and KTO training objectives, we demonstrated that our self-trained models outperform both base model without ESD-specific and human-annotated models at the system and span levels, while successfully matching their performance at the sentence level. Ultimately, our findings signify a promising new paradigm for developing highly accurate and scalable ESD models free from the constraints of human supervision.

Finally, our analysis of iteration scaling revealed a performance bottleneck at higher iterations ($T = 3$), driven by a reduction in candidate diversity and utility variance. Addressing this bottleneck by maintaining candidate diversity during iterative training represents a key focus for our future work.

Limitations

While our proposed Iterative MBR Distillation framework demonstrates promising results for ESD, we acknowledge several limitations in our current study.

Firstly, there is a potential metric bias in our span-level evaluation. Specifically, we employ SOFTF1 as the utility function to drive the MBR decoding and distillation process, and subsequently use the exact same metric to evaluate the model’s span-level performance. This alignment between the optimization target and the evaluation metric may inherently favor models trained via our framework. Such metric-overfitting concerns have been widely discussed in prior literature regarding MBR decoding in machine translation (Kovacs et al., 2024). However, given the current lack of more suitable or orthogonal metrics for span-level ESD evaluation, we rely on SOFTF1 for this study and leave the exploration or development of independent evaluation metrics to future work.

Secondly, the empirical validation of our method is currently restricted to a single base Large Language Model (*Qwen3-30B-A3B-Instruct-2507*). This limitation is primarily dictated by the sheer computational expense associated with our framework. As detailed in Section 5.1, generating massive candidate sets for MBR scoring and conducting iterative full-parameter fine-tuning required extensive computational resources and time. Consequently, we were unable to scale our experiments to verify the generalizability of the proposed method

across models of varying sizes or different architectural families. Future research with access to broader computational resources could investigate how this self-evolution framework performs across a wider spectrum of LLMs.

Acknowledgments

References

Frederic Blain, Chrysoula Zerva, Ricardo Rei, Nuno M. Guerreiro, Diptesh Kanojia, José G. C. de Souza, Beatriz Silva, Tânia Vaz, Yan Jingxuan, Fatemeh Azadi, Constantin Orasan, and André Martins. 2023. Findings of the WMT 2023 shared task on quality estimation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 629–653, Singapore. Association for Computational Linguistics.

Tri Dao. 2024. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations (ICLR)*.

Daniel Deutsch, Rotem Dror, and Dan Roth. 2021. A statistical analysis of summarization evaluation metrics using resampling methods. *Transactions of the Association for Computational Linguistics*, 9:1132–1146.

Daniel Deutsch, George Foster, and Markus Freitag. 2023. Ties matter: Meta-evaluating modern metrics with pairwise accuracy and tie calibration. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12914–12929, Singapore. Association for Computational Linguistics.

Bryan Eikema and Wilker Aziz. 2020. Is MAP decoding all you need? the inadequacy of the mode in neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4506–4520, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. Model alignment as prospect theoretic optimization. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Zhaopeng Feng, Yan Zhang, Hao Li, Bei Wu, Jiayu Liao, Wenqiang Liu, Jun Lang, Yang Feng, Jian Wu, and Zuozhu Liu. 2025. TEaR: Improving LLM-based machine translation with systematic self-refinement. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 3922–3938, Albuquerque, New Mexico. Association for Computational Linguistics.

Markus Freitag, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021a. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474.

Markus Freitag, David Grangier, Qijun Tan, and Bowen Liang. 2022a. High quality rather than high model probability: Minimum Bayes risk decoding with neural metrics. *Transactions of the Association for Computational Linguistics*, 10:811–825.

Markus Freitag, Nitika Mathur, Daniel Deutsch, Chi-Kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Frederic Blain, Tom Kocmi, Jiayi Wang, David Ifeoluwa Adelani, Marianna Buchicchio, Chrysoula Zerva, and Alon Lavie. 2024. Are LLMs breaking MT metrics? results of the WMT24 metrics shared task. In *Proceedings of the Ninth Conference on Machine Translation*, pages 47–81, Miami, Florida, USA. Association for Computational Linguistics.

Markus Freitag, Nitika Mathur, Chi-kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Tom Kocmi, Frederic Blain, Daniel Deutsch, Craig Stewart, Chrysoula Zerva, Sheila Castilho, Alon Lavie, and George Foster. 2023. Results of WMT23 metrics shared task: Metrics might be guilty but references are not innocent. In *Proceedings of the Eighth Conference on Machine Translation*, pages 578–628, Singapore. Association for Computational Linguistics.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022b. Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, George Foster, Alon Lavie, and Ondřej Bojar. 2021b. Results of the WMT21 metrics shared task: Evaluating metrics with expert-based human evaluations on TED and news domain. In *Proceedings of the Sixth Conference on Machine Translation*, pages 733–774, Online. Association for Computational Linguistics.

Vaibhava Goel and William J Byrne. 2000. Minimum bayes-risk automatic speech recognition. *Computer Speech & Language*, 2(14):115–135.

Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André F. T. Martins. 2024. xcomet: Transparent machine translation evaluation through fine-grained error detection. *Transactions of the Association for Computational Linguistics*, 12:979–995.

Juraj Juraska, Tobias Domhan, Mara Finkelstein, Tetsuji Nakagawa, Geza Kovacs, Daniel Deutsch, Pidong Wang, and Markus Freitag. 2025. MetricX-25 and GemSpanEval: Google Translate submissions to the

WMT25 evaluation shared task. In *Proceedings of the Tenth Conference on Machine Translation*, pages 957–968, Suzhou, China. Association for Computational Linguistics.

Hidetaka Kamigaito, Hiroyuki Deguchi, Yusuke Sakai, Katsuhiko Hayashi, and Taro Watanabe. 2025. Diversity explains inference scaling laws: Through a case study of minimum Bayes risk decoding. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 29060–29094, Vienna, Austria. Association for Computational Linguistics.

Tom Kocmi and Christian Federmann. 2023. GEMBA-MQM: Detecting translation quality error spans with GPT-4. In *Proceedings of the Eighth Conference on Machine Translation*, pages 768–775, Singapore. Association for Computational Linguistics.

Tom Kocmi, Vilém Zouhar, Eleftherios Avramidis, Roman Grundkiewicz, Marzena Karpinska, Maja Popović, Mrinmaya Sachan, and Mariya Shmatova. 2024. Error span annotation: A balanced approach for human evaluation of machine translation. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1440–1453, Miami, Florida, USA. Association for Computational Linguistics.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

Geza Kovacs, Daniel Deutsch, and Markus Freitag. 2024. Mitigating metric bias in minimum bayes risk decoding. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1063–1094.

Achintya Kundu, Rhui Dih Lee, Laura Wynter, Raghu Kiran Ganti, and Mayank Mishra. 2024. Enhancing training efficiency using packing with flash attention. *Preprint*, arXiv:2407.09105.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Alon Lavie, Greg Hanneman, Sweta Agrawal, Diptesh Kanojia, Chi-Kiu Lo, Vilém Zouhar, Frederic Blain, Chrysoula Zerva, Eleftherios Avramidis, Sourabh Deoghare, Archchana Sindhujan, Jiayi Wang, David Ifeoluwa Adelani, Brian Thompson, Tom Kocmi, Markus Freitag, and Daniel Deutsch. 2025. Findings of the WMT25 shared task on automated translation evaluation systems: Linguistic diversity is challenging and references still help. In *Proceedings of the Tenth Conference on Machine Translation*, pages 436–483, Suzhou, China. Association for Computational Linguistics.

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

Boxuan Lyu, Haiyue Song, Hidetaka Kamigaito, Chenchen Ding, Hideki Tanaka, Masao Utiyama, Kotaro Funakoshi, and Manabu Okumura. 2025. Minimum bayes risk decoding for error span detection in reference-free automatic machine translation evaluation. *Preprint*, arXiv:2512.07540.

Nitika Mathur, Johnny Wei, Markus Freitag, Qingsong Ma, and Ondřej Bojar. 2020. Results of the WMT20 metrics shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 688–725, Online. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Lorenzo Proietti, Stefano Perrella, and Roberto Navigli. 2025. Has machine translation evaluation achieved human parity? the human reference and the limits of progress. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 790–813, Vienna, Austria. Association for Computational Linguistics.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Yuncheng Song, Liang Ding, Changtong Zan, and Shujian Huang. 2025. Self-evolution knowledge distillation for LLM-based machine translation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10298–10308, Abu Dhabi, UAE. Association for Computational Linguistics.

Andreas Stolcke, Yochai Konig, and Mitch Weintraub. 1997. Explicit word error minimization in n-best list rescoring. In *EUROSPEECH*.

Zhengwei Tao, Ting-En Lin, Xiancai Chen, Hangyu Li, Yuchuan Wu, Yongbin Li, Zhi Jin, Fei Huang,

Dacheng Tao, and Jingren Zhou. 2024. A survey on self-evolution of large language models. *Preprint*, arXiv:2404.14387.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Brian Thompson, Nitika Mathur, Daniel Deutsch, and Huda Khayrallah. 2024. Improving statistical significance in human evaluation of automatic metrics via soft pairwise accuracy. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1222–1234, Miami, Florida, USA. Association for Computational Linguistics.

Brandon T. Willard and Rémi Louf. 2023. Efficient guided generation for large language models. *Preprint*, arXiv:2307.09702.

Yuze Zhao, Jintao Huang, Jinghan Hu, Xingjun Wang, Yunlin Mao, Daoze Zhang, Zeyinzi Jiang, Zhikai Wu, Baole Ai, Ang Wang, Wenmeng Zhou, and Yingda Chen. 2024. Swift:a scalable lightweight infrastructure for fine-tuning. *Preprint*, arXiv:2408.05517.