

Article

TransGoT: Structured Graph-of-Thoughts Reasoning for Machine Translation with Large Language Models

Danying Zhang ¹, Yixin Liu ¹, Jie Zhao ^{1,*} and Cai Xu ² 

¹ School of Data Science and Artificial Intelligence, Chang'an University, Xi'an 710064, China; danyingz614@163.com (D.Z.); yixin.liu@chd.edu.cn (Y.L.)

² Guangzhou Institute of Technology, Xidian University, Guangzhou 510555, China; cxu@xidian.edu.cn

* Correspondence: jiezhaoh@chd.edu.cn

Abstract

Machine translation with large language models has recently attracted growing attention due to its flexibility and strong zero-shot and few-shot capabilities. However, most prompt-based LLM translation methods rely on linear generation or shallow self-refinement, implicitly committing to a single reasoning path. Such designs are brittle when translating long and syntactically complex sources, where reliable translation often requires structured planning and hypothesis exploration. In this paper, we propose TransGoT, a novel machine translation framework inspired by the graph-of-thoughts paradigm, which formulates translation as a structured, multi-stage reasoning process over a graph of intermediate thoughts. TransGoT explicitly decomposes translation into constraint identification, draft generation, and culture- and style-aware refinement, enabling systematic exploration and aggregation of alternative translation hypotheses. To better adapt graph-based reasoning to translation, we design two key mechanisms: (1) Uncertainty-driven thought transformation. Unlike general reasoning tasks, translation uncertainty is often localized and unevenly distributed across tokens, making holistic regeneration inefficient. We therefore design uncertainty-driven thought transformation, which leverages model-internal confidence signals to guide targeted token-level revision; (2) Dispersion-adaptive thought scoring. It emphasizes evaluation criteria with stronger inter-candidate variance to enable robust multi-criteria thought selection. We evaluate TransGoT on the WMT22 benchmarks and experimental results demonstrate that TransGoT consistently outperforms strong LLM-based translation baselines, validating the effectiveness of structured graph-based reasoning for machine translation.

Keywords: machine translation; chain of thought; large language model



Academic Editors: Xingyi Song,
Ye Jiang and Yunfei Long

Received: 5 January 2026

Revised: 25 February 2026

Accepted: 26 February 2026

Published: 27 February 2026

Copyright: © 2026 by the authors.
Licensee MDPI, Basel, Switzerland.
This article is an open access article
distributed under the terms and
conditions of the [Creative Commons
Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Machine translation (MT) is a long-standing task of automatically converting text from one language into another while preserving meaning and style. As a core application in natural language processing, machine translation enables cross-lingual communication and access to information [1,2].

Mainstream MT systems evolved from statistical paradigms to neural architectures. Early efforts rely on statistical methods, in particular phrase-based statistical MT [3]. These works decompose translation into local phrase correspondences with probabilistic models to align and translate phrases, relying heavily on feature engineering and phrase tables, and often struggle with complex linguistic context. In the last decade, neural MT has improved

end-to-end modeling through encoder–decoder architectures with recurrent networks and attention mechanisms, achieving higher translation quality [2,4,5]. The Transformer architecture further advanced quality by enabling fully attention-driven sequence modeling with strong scalability [6]. Transformers eliminate recurrence and enable highly parallel training, leading to improved translation quality. These advances establish powerful neural MT systems that underpin most contemporary translation approaches.

With the advent of pre-trained large language models (LLMs), MT research has entered a new era [7]. Modern LLMs can be applied to translation either by fine-tuning or by prompting. In the fine-tuning based approach, a pre-trained language model is adapted with parallel data [8–10]. Examples include mBART [11] and mT5 [12], which are sequence-to-sequence models pre-trained on large multilingual corpora and then fine-tuned on translation tasks. Such methods leverage massive monolingual data and often attain strong performance across many language pairs. However, fine-tuning LLMs for translation can be computationally intensive and may over-specialize the model, potentially weakening its in-context generalization ability [13].

In contrast, prompt-based approaches cast MT as an inference-time task for a fixed LLM, offering high flexibility without requiring task-specific training. For example, GPT-3 and subsequent models can perform translation via few-shot prompting without gradient updates [14]. A growing body of work has examined how prompt templates, in-context demonstrations, and decoding strategies influence MT performance [15–19]. However, current prompt-based MT methods typically follow a linear reasoning trajectory, producing a translation in a single pass or through a small number of sequential refinements. This design effectively commits to a single generation trajectory, which may be insufficient for inputs requiring non-local planning (e.g., long-distance reordering, discourse-level coherence, or ambiguity resolution that depends on multiple interacting cues), where exploring alternative hypotheses can be beneficial. For instance, in news-style documents, the antecedent of a pronoun or an implicit subject may only become unambiguous several sentences later; a linear pipeline that resolves the ambiguity locally can propagate the mistake throughout the paragraph, yielding globally inconsistent entity chains. This limitation is especially evident for long documents and syntactically complex sentences, where reliable translation often benefits from exploring alternative hypotheses and aggregating intermediate decisions rather than committing to a single path.

To this end, we aim to develop a more structured reasoning framework for MT. Recent work on LLM reasoning has shown the power of multi-step planning (chain-of-thought) [20], and even branching reasoning strategies (tree-of-thought) [21], to improve complex problem-solving. More recently, Graph of Thoughts (GoT) [22] generalizes these ideas by representing intermediate “thoughts” as vertices in a graph and allowing them to be combined and refined along different paths, showing strong reasoning ability. In GoT, vertices correspond to units of generated text (LLM “thoughts”) and edges capture logical or semantic dependencies. This enables the model to explore multiple hypotheses in parallel and to revisit or merge them as needed. By analogy to GoT, we posit that MT can benefit from an explicit multi-stage planning approach that generates and evaluates translation hypotheses in a structured way. However, the standard GoT does not prescribe MT-specific intermediate thoughts, where the search space should be structured around translation constraints (e.g., terminology and forbidden patterns), cultural appropriateness, and style control. Moreover, directly applying GoT to MT, however, can lead to error accumulation across thoughts and can not effectively filter low-quality translation thoughts.

In this work, we propose TransGoT, a novel translation framework inspired by graph-of-thoughts. TransGoT organizes the translation process into multiple and structured reasoning stages that build on one another. First, the model identifies key lexical or

structural constraints from the source (e.g., named entities and technical terms) to construct constraint thoughts. Conditioned on these constraints, it then generates initial translation thoughts as editable drafts. Finally, Considering the cultural [23] and stylistic [24] attributes are non-negligible factors that substantially affect translation adequacy and acceptability, TransGoT performs culture-aware and style-aware refinement, incorporating relevant background knowledge and style guidelines to revise the drafts and improving cultural appropriateness and maintaining consistency with the desired style.

To better harness GoT's structured thought search for the MT task, we design uncertainty-driven thought transformation and dispersion-weighted thought scoring, which enable uncertainty-aware revision and dispersion-adaptive multi-criteria scoring over intermediate thoughts. More specifically: (1) Predictive uncertainty is indicative of unreliable generation and is correlated with a higher likelihood of errors [25,26]. Compared with the general reasoning tasks, translation confidence often manifests locally at the token or phrase level rather than the holistic regeneration, so revising low-confidence spans can be more efficient than refining the entire draft to address the issue of error accumulation. Therefore, we detect parts of the draft with low confidence and explicitly mark them as revisable targets, so that TransGoT can re-attend to these spans and perform targeted rewriting in subsequent reasoning stages rather than blindly propagating early mistakes. Compared with GoT, TransGoT additionally uses token-level probabilities generated by the LLM (or truncated top-(k) logprobs when full access is unavailable) to quantify uncertainty and mask high-uncertainty spans for targeted rewriting. This type of probability-based uncertainty estimation is not part of standard GoT, and it is introduced in this work specifically for machine translation, where errors often concentrate in localized spans. This mechanism enables TransGoT to revise low-confidence regions while reducing unnecessary global rewrites, thereby improving adequacy and stability. (2) In addition, translation quality is inherently multi-dimensional and different criteria may vary in how well they discriminate among candidates. We then design a dispersion-adaptive scoring mechanism that assigns larger weights to criteria with higher dispersion across candidate thoughts, thereby enabling robust pruning and selection of high-quality thoughts.

The key contributions of this paper are summarized as follows:

- We propose a new MT paradigm TransGoT, a graph-of-thoughts-based translation framework that applies structured, multi-stage reasoning to translation.
- We deliberately design task-specific thought types in the translation graph, and propose uncertainty-driven masking and dispersion-adaptive scoring to enable targeted revision of low-confidence segments and robust multi-criteria selection of intermediate thoughts.
- Extensive experiments are conducted on multilingual translation benchmarks (WMT22) to verify the effectiveness of the proposed TransGoT. The results demonstrate significant improvements over competitive baselines.

2. Related Work

2.1. LLM-Based Translation

Building on the strong text understanding and generation capabilities of LLMs, recent work has increasingly investigated applying them to MT. These studies primarily fall into two categories: fine-tuning-based and prompt-based approaches. For the former, ref. [8] showed that LLMs achieve strong translation capability when fine-tuned with only a few dozen sentence pairs; ref. [9] applied a two-stage recipe (monolingual continued pretraining then parallel fine-tuning) to substantially boost translation quality; ref. [10] incorporated quality-aware strategies, such as weighting more difficult sentences higher during fine-tuning. Moreover, specialized translation LLMs have also been developed. For example,

the open Tower model was trained on multilingual translation data to excel in translation-related tasks [27]. However, several studies note that aggressive fine-tuning can harm some LLM capabilities. For example, ref. [28] found that full fine-tuning improves BLEU and COMET scores but can degrade abilities like formality control and few-shot generalization. Meanwhile, many closed-LLMs outperform their open-source counterparts [29], which makes fine-tuning these models to acquire strong translation capabilities infeasible.

Instead of updating model weights, other works use few-shot or zero-shot prompting to exploit LLM translation capabilities through in-context learning. Ref. [7] systematically studied different prompting strategies for MT and showed that example formatting and selection significantly affect quality. Ref. [30] evaluated GPT-3.5/4 and found that these models achieve competitive quality on high-resource languages with only prompt-based use. Indeed, ChatGPT (GPT-4) prompts can yield translations approaching the quality of commercial systems [31]. Other works explored advanced prompt techniques: ref. [32] showed that careful selection of few-shot exemplars further improves in-context MT. Ref. [33] prompted LLMs to self-correct their translations across multiple rounds, which enhances fluency and adequacy. Ref. [18] employed an automatic speech recognition transcript plus encoded speech as a “chain-of-thought” prompt to guide a speech translation LLM, achieving notable BLEU gains. Ref. [17] proposed MAPS to mimic the human translation process: it first analyzes the source text to extract relevant knowledge, and then integrates and selects the useful information for generating the final translation. Overall, prompt-based methods follow a linear reasoning trajectory, and can not fully explore the reasoning ability of LLMs.

2.2. Chain-of-Thought Prompting

Chain-of-thought (CoT) prompting was introduced by [20] as a way to improve LLM reasoning by generating intermediate reasoning steps. The key insight is that providing exemplar solutions with explicit step-by-step rationales enables models to decompose complex tasks. This approach can also be applied in a zero-shot style by simply appending phrases like “Let’s think step by step” to the prompt [34]. Ref. [35] further investigated why CoT is effective through ablation studies and found that it mainly helps the model structure its task understanding within the prompt. Numerous studies have been proposed to further enhance the CoT’s abilities to handle complex tasks. Representative work, such as [36], proposed self-consistency, where multiple CoT samples are generated and then aggregated to select the most consistent answer. However, these extensions typically generate or aggregate linear chains, without explicitly organizing the reasoning process into a structured search space. The Tree-of-thought (ToT) framework generalizes CoT by allowing an LLM to explore structured reasoning branches [21], highlighting the benefit of structured search in LLM prompting. In ToT, a model generates several “thought” vertices at each step and selectively expands a search tree. Graph-of-thought (GoT) further extends this idea by modeling LLM reasoning steps as a directed acyclic graph [22]. In GoT, each vertex represents an “LLM thought”, a coherent piece of generated content, and the edges encode logical or procedural dependencies among vertices. By searching over graph structures, GoT can integrate thoughts from different chains and apply feedback loops, yielding synergistic outcomes.

In MT, CoT-style prompting has been explored recently. As mentioned above, ref. [33] used a CoT strategy to iterate on translations. Ref. [37] prompted LLMs to explain translation errors in natural language, showing improved translation quality. Ref. [38] included ToT in their study of translation prompts and found that ToT can sometimes yield higher-quality translations or better error-detection compared with CoT. Our work focuses on formulating the GoT to MT.

3. Preliminary

GoT [22] is a general-purpose inference framework that organizes an LLM's intermediate reasoning states as a directed graph, enhancing LLMs' capabilities through networked reasoning. GoT enables the combination of multiple intermediate thoughts by allowing a vertex to depend on several predecessors. More broadly, its graph-based formulation subsumes both CoT [20] and ToT [21], supporting richer reasoning structures and flexible information aggregation without modifying model parameters. Through this, GoT can implement structured exploration, selective expansion, and controlled combination of partial solutions (thoughts).

3.1. Thought Graph

Given an input instance x (e.g., a prompt or a general problem statement), GoT maintains a thought graph

$$\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$$

at a reasoning step t , where each vertex $v \in \mathcal{V}_t$ corresponds to a thought (e.g., an initial solution, a partial solution, and an intermediate hypothesis), and each directed edge $(v_i \rightarrow v_j) \in \mathcal{E}_t$ indicates that v_j was produced by transforming v_i (predecessor vertex) under a specific operation. In contrast to linear reasoning, the graph representation allows multiple hypotheses to coexist and evolve, supporting constructed branching.

3.2. Thought Transformations

Common transformations of GoT include: (1) Aggregation Transformations, which merge multiple intermediate thoughts into a new thought, combining their complementary strengths while mitigating their individual weaknesses. More broadly, this mechanism naturally extends from aggregating individual thoughts to aggregating reasoning paths, i.e., longer chains of thoughts. Under the graph abstraction, this is achieved by connecting the vertices $v_1, v_2, \dots, v_k$ which represent the terminal thoughts of several chains, via outgoing edges into a single aggregation vertex v^+ ; (2) Refining Transformations, which refine an existing thought v by updating its content. In this case, no new vertices are added and a self-loop is introduced, i.e., the new edge inserted into $\mathcal{G}$ is $\mathcal{E}^+ = v \rightarrow v$. This loop represents an iterative revision of the same thought while keeping the original connectivity in the graph; (3) Generation Transformations, which generate k new thoughts from a single existing thought v . This operation is closely related to branching-style reasoning used in earlier schemes (e.g., ToT). Formally, we add k new vertices $\mathcal{V}^+ = \{v_1, v_2, \dots, v_k\}$ and connect them from v via new edges $\mathcal{E}^+ = \{(v \rightarrow v_1), (v \rightarrow v_2), \dots, (v \rightarrow v_k)\}$.

In GoT, transformations that generate or update textual thoughts invoke an underlying generator (typically an LLM), which defines a conditional distribution $p_\theta(\cdot|\text{context})$ over outputs, where θ is the fully pre-trained parameters of LLM. GoT does not estimate p_θ beyond using the generator for decoding. Instead, it relies on an external scoring function (Equation (1)) to rank and prune thoughts. Therefore, explicitly designed probability-based uncertainty estimation in TransGoT is not a required component of GoT.

3.3. Thoughts Pruning

To control the thought search, GoT assigns each thought a scalar utility via a task-specific evaluation function:

$$s(v) = \text{Score}(x, z_v) \in \mathbb{R}, \quad (1)$$

where the score function might be computed by an LLM-based critic, a learned metric, or heuristics. Typically, at each iteration (reasoning step), top- K vertices ranked by their scores are retained.

3.4. Graph of Operations and Graph Reasoning State

The controller of GoT comprises two key components: the Graph of Operations (GoO) and the Graph Reasoning State (GRS). The GoO is a static blueprint that encodes how a task is decomposed into a graph of thought operations. It specifies which transformations are applied to LLM-generated thoughts, as well as their execution order and dependency structure. In contrast, the GRS is a dynamic data structure that tracks the evolving reasoning process. In practice, the user instantiates a GoO to define the execution plan of thought operations, where each operation explicitly records its predecessors and successors. During execution, the GRS is updated online to reflect the current status of the reasoning process, such as which operations have been completed and the set of generated thoughts.

4. Framework

4.1. Problem Definition

Let x denote a source segment, which can be a sentence or a document chunk, and let y be its target translation. We assume access to an LLM parameterized by θ (fully pre-trained), which could be open-weight model (e.g., Llama (<https://www.llama.com/> (accessed on 25 February 2026))) or closed-weight model (e.g., ChatGPT (<https://chatgpt.com/> (accessed on 25 February 2026))). Our goal is to prompt the LLM to reason a translation y that preserves the source meaning, aligns with the target culture, follows the desired stylistic conventions, and satisfies other requirements.

4.2. TransGoT

TransGoT formulates translation as a staged search over a directed graph of intermediate thoughts, where multiple thought candidates are generated and pruned via a scoring function. In both GoT and TransGoT, intermediate thoughts are produced by a pre-trained LLM via conditional generation, i.e., decoding from an implicit conditional distribution $p_\theta(\cdot | \text{context})$, and neither of them introduces a new trained probabilistic model. Following the core idea of GoT, an LLM-generated thought in TransGoT is a vertex in a graph (e.g., constraint specification and draft translation). We define a directed acyclic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where each vertex $v \in \mathcal{V}$ corresponds to a translation thought z_v , and each edge $(v_i \rightarrow v_j) \in \mathcal{E}$ indicates z_{v_j} is generated conditioned on z_{v_i} . The final output y is generated by transforming and aggregating the information from a selected set of thought vertices.

Figure 1 illustrates the overall pipeline. We model the translation reasoning process as a directed heterogeneous graph. Specifically, we design TransGoT that performs staged transformations as follows (This resembles chain-of-thought reasoning like CoT, but its core lies in retaining multiple thoughts at each step and performing thought transformations, thus forming graph-based reasoning.):

$$\mathcal{Z}^{(c)} \sim p_\theta(\cdot | x), \quad (\text{Constraint Thoughts}) \quad (2)$$

$$\mathcal{Z}^{(d)} \sim p_\theta(\cdot | x, \mathcal{Z}^{(c)}), \quad (\text{Draft Thoughts}) \quad (3)$$

$$\mathcal{Z}^{(k)} \sim p_\theta(\cdot | x, \tilde{\mathcal{Z}}^{(d)}), \quad (\text{Culture-aware Thoughts}) \quad (4)$$

$$\mathcal{Z}^{(s)} \sim p_\theta(\cdot | x, \mathcal{Z}^{(k)}), \quad (\text{Style-aware Thoughts}) \quad (5)$$

$$y^* \sim p_\theta(\cdot | x, \mathcal{Z}^{(s)}), \quad (\text{Final Translation}) \quad (6)$$

Throughout this paper, $p_\theta(\cdot|\cdot)$ is the base LLM's conditional generation distribution under the given prompt/context. In TransGoT, we quantify token-level uncertainty for masking the draft thoughts (Section 4.2.2). For open-weight models like Llama, token probabilities are directly available. For closed-weight models where θ is inaccessible like ChatGPT, token logprobs may be truncated and we use the binned entropy approximation defined in Equation (9). We should note that the graph in TransGoT is not a syntactic/semantic parse graph of the sentence. Instead, it is a reasoning graph (i.e., thought graph) where nodes are intermediate textual states such as constraints, drafts, culture-aware rewrites, and style-aware rewrites. And edges denote conditioning dependencies induced by the controller (e.g., $\mathcal{Z}^{(s)}$ is generated conditioned on $\mathcal{Z}^{(k)}$). The generation is performed by the LLM: the graph provides multi-hypothesis exploration and structured pruning and enables multi-hypothesis exploration and structured pruning.

Following GoT, we generate multiple candidate thoughts at each stage (Equations (4)~(6)) and retain the top- K thoughts according to a multi-criteria scoring function, yielding a compact set of high-quality thoughts for subsequent stages. For simplicity, we omit the prompts for specific transformations in the above equations, as well as the scoring and ranking stages [22]. Particularly, $\tilde{\mathcal{Z}}^{(d)}$ is an uncertainty-masked version of the initial translation thought (Section Uncertainty-Driven Masking for Draft Revision), which explicitly flags low-confidence regions for targeted language. Particularly, considering that translation quality involves multiple metrics such as accuracy and fluency, we score and retain translation thoughts through a carefully designed dispersion-weighted multi-dimensional score (Section Dispersion-Adaptive Multi-Dimensional Scoring and Selection). This yields a compact set of high-quality thoughts that propagate to later stages, akin to graph search with pruning. Next, we will introduce each type of thought.

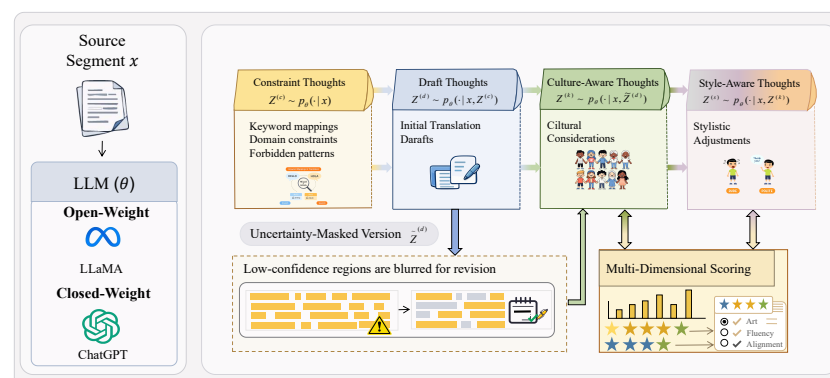


Figure 1. The overview of TransGoT.

4.2.1. Constraint Thoughts

TransGoT first prompts the LLM to produce a set of constraint thoughts $\mathcal{Z}^{(c)}$ that normalizes the requirements into an executable checklist. Concretely, $\mathcal{Z}^{(c)}$ includes: (i) keyword mappings (source key-word $\rightarrow$ required target key-word); (ii) domain framing (e.g., biomedical/legal); and (iii) forbidden patterns (e.g., do not translate product names). This step externalizes constraints as a stable conditioning signal for draft thought generation.

4.2.2. Draft Thoughts

Conditioned on $(x, \mathcal{Z}^{(c)})$, the LLM generates initial translation thoughts $\mathcal{Z}^{(d)}$. Unlike the final output, $\mathcal{Z}^{(d)}$ is treated as an editable draft whose uncertain parts will be explicitly revised in later stages.

Uncertainty-Driven Masking for Draft Revision

A key challenge in prompting-based translation is that early-stage errors can persist through later refinement. TransGoT introduces an uncertainty-driven mechanism that forces the model to re-attend to unreliable regions of the draft thoughts. Let $z^{(d)} = (w_1, \dots, w_m)$ be the token sequence of the draft translation thought. For each token w_i , we compute an uncertainty score u_i from the LLM token probabilities:

$$p_i \triangleq p_\theta(w_i \mid x, z^{(c)}, w_{<i}), \quad (7)$$

$$u_i = - \sum_{j=1}^{|V|} p_{i,j} \log(p_{i,j}), \quad (8)$$

where $z^{(c)}$ is a constraint thought and $|V|$ is the vocabulary size. Here we define u_i as the entropy of token probabilities. Intuitively, if u_i is relatively small, it indicates that the current token generated by LLM exhibits higher confidence. For open-weight models, u_i can be computed directly from the model's token distribution during decoding. For closed-weight models, token-level probabilities may be truncated (e.g., the APIs of ChatGPT (<https://platform.openai.com/chat> (accessed on 25 February 2026))). When top- k probabilities (or logprobs) are provided, we approximate entropy using a binned estimate and define the following binned entropy to approximate u_i .

$$u_i = - \sum_{j=1}^k p_{i,j} \log(p_{i,j}) - r \log(r), \quad (9)$$

where $r = 1 - \sum_{j=1}^k p_{i,j}$. We define the low-confidence index set as the top ρ -quantile:

$$\mathcal{L}_\rho = \{i \in \{1, \dots, m\} : u_i \geq \text{Quantile}_\rho(\{u_j\}_{j=1}^m)\}, \quad (10)$$

with $\rho = 0.1$ in our method. We then randomly sample a subset $\mathcal{M} \subseteq \mathcal{L}_\rho$ (to avoid deterministic artifacts) and replace tokens $\{w_i\}_{i \in \mathcal{M}}$ with a special marker [MASK]:

$$\tilde{z}^{(d)} = \text{Mask}(z^{(d)}; \mathcal{M}). \quad (11)$$

When generating culture-aware thoughts $z^{(k)}$, we explicitly instruct the LLM that [MASK] regions are uncertain and might be rewritten with justification grounded in x . This transforms uncertainty into a controllable refinement signal.

4.2.3. Culture- and Style-Aware Thoughts

Culture-Aware Thoughts

Considering culture-specific expressions and culturally nuanced references are often poorly served by literal translation [23], especially when no direct target-language equivalent exists, we explicitly incorporate cultural reasoning into TransGoT to guide the LLM toward culturally appropriate translations. Given $(x, \tilde{z}^{(t)})$, we prompt the LLM to generate cultural thoughts $z^{(k)}$ that resolve idioms, allusions, honorifics, and other culture-specific content. This step aims to improve culture appropriateness while maintaining semantic fidelity and constraint compliance.

Style-Aware Thoughts

Refs. [24,39] have pointed that stylistic attributes (e.g., formality and politeness) are crucial for producing natural and audience-appropriate translations. Therefore, we further model the desired style to steer the LLM toward stylistically consistent outputs. Conditioned on $(x, z^{(k)})$, the model generates $z^{(s)}$ to match target-side style requirements. Importantly, style adaptation is applied after cultural adaptation to reduce conflicts between pragmatic rewriting and stylistic polishing.

Dispersion-Adaptive Multi-Dimensional Scoring and Selection

Since multiple candidate culture and style thoughts can be generated with varying qualities, filtering for high-quality thoughts is necessary so that the LLM is guided by relevant signals rather than noisy or conflicting cues. Translation quality is inherently multi-dimensional, and collapsing multiple criteria into a single score by simply averaging criteria can dilute dimensions that better discriminate among candidates and may lead to suboptimal rankings in practice. TransGoT performs candidate selection under a dispersion-weighted multi-dimensional scoring mechanism.

Let $\mathcal{Q}$ be a set of quality dimensions (e.g., accuracy, fluency, cultural appropriateness, formatting correctness, and stylistic consistency, among others). For a candidate c (a culture or style thought), we assign a score $q_d(c)$ for each criteria $q \in \mathcal{Q}$, which can be instantiated by LLM-based judges [40,41]. TransGoT requires only that higher is better and scores are comparable within each dimension. Rather than simply averaging these scores for c , we propose dispersion-weighted mechanism to integrate them. We build this on a simple yet effective intuition: criteria that better discriminate among candidates (higher dispersion) should contribute more to the integrated score. Given a candidate set $\mathcal{C} = \{c_1, \dots, c_n\}$ where n is the number of thoughts, we compute a dispersion statistic for each criterion:

$$\Delta_q(\mathcal{C}) \triangleq \text{Dispersion}(\{q(c_j)\}_{j=1}^n), \quad (12)$$

where $\text{Dispersion}(\cdot)$ is a generic operator (variance in our experiments). We then normalize to obtain weights:

$$w_q(\mathcal{C}) = \frac{\Delta_q(\mathcal{C}) + \epsilon}{\sum_{q' \in \mathcal{Q}} (\Delta_{q'}(\mathcal{C}) + \epsilon)}, \quad (13)$$

with a small $\epsilon > 0$ for numerical stability. The final score for candidate c_j is

$$s(c_j; \mathcal{C}) = \sum_{q \in \mathcal{Q}} w_q(\mathcal{C}) \cdot q(c_j), \quad (14)$$

and we select the top- K candidates for reasoning via This mechanism is applied consistently to culture-aware thoughts and $\mathcal{Z}^{(k)}$ style-aware thoughts $\mathcal{Z}^{(s)}$. Finally, we further refine the translation by conditioning the LLM on the retained style-aware thought. Algorithm 1 summarizes TransGoT, which is prompt-only and compatible with both sentence-level and document-level translation.

Algorithm 1 TransGoT.

Require: source x , LLM.

- 1: Generate constraint thoughts $\mathcal{Z}^{(c)}$
 - 2: Generate initial translation candidates $\mathcal{Z}^{(d)}$ based $(x, \mathcal{Z}^{(c)})$
 - 3: Compute token confidences $\{u_i\}$ for $\mathcal{Z}^{(d)}$; build $\tilde{\mathcal{Z}}^{(d)}$ by masking top- ρ quantile tokens (highest uncertainty)
 - 4: Generate culture-aware thoughts $\mathcal{Z}^{(k)}$ from $(x, \tilde{\mathcal{Z}}^{(d)})$; select via dispersion-weighted scoring
 - 5: Generate style-aware thoughts $\mathcal{Z}^{(s)}$ from $(x, \mathcal{Z}^{(k)})$; select via dispersion-weighted scoring
 - 6: Generate the final translation y^* from $(x, \mathcal{Z}^{(s)})$
 - 7: Return y^*
-

5. Experiments

5.1. Datasets

We conduct our experiments on the WMT22 [42] benchmark, which extends the former WMT News Translation task to evaluate MT systems under a multi-domain setting. Each example in the dataset contains a varying number of sentences. Since our method and baselines do not require model training, we directly use the official test set. The test sets are conducted from four domains: news, social, e-commerce, and conversational. The organizers target 2000 source segments per language pair, aiming for 500 segments per domain. Following the work of [19], we use 8 translation directions across 5 languages (German (De), Czech (Cs), Chinese (Zh), Russian (Ru), and English (En)) in our experiments, using approximately 2,000 source segments per direction spanning four domains (news, social, e-commerce, and conversational). The detailed statistics are shown in Table 1.

Table 1. Statistics of the employed dataset. Specifically, when the first four lines are used as the source language, we use English as the target language; when English is the source language in the last row, the target can be any of the other four languages.

Source	Conv.	E-Com.	News	Social	Other	Total
German	462	501	506	515	-	1984
Czech	-	-	957	491	-	1448
Chinese	349	518	505	503	-	1875
Russian	-	508	1004	-	504	2016
English	484	530	511	512	-	2037

5.2. Baselines

We employ TEaR [19], which improves LLM-based MT by prompting the model to iteratively critique and revise an initial translation through a systematic self-refinement loop, as the strongest baseline in our experiments. Additionally, we compare TransGoT with the following baselines:

- IT (Initial Translation): which directly prompts the LLM to translate the sources.
- SCoT [37]: which formulate automatic translation post-editing as a direct LLM prompting task, where the LLM takes the source sentence and a neural MT output and is prompted to produce an improved translation by applying targeted edits.
- CT [33]: which improves MT by iteratively prompting an LLM to self-correct and rewrite an initial translation in multiple rounds, conditioning each refinement on the source sentence and the current translation to enhance fluency and overall quality progressively.

5.3. Metrics

Following the work of [19], we adopt two learned neural metrics as our evaluation criteria: the reference-free metric COMET-Kiwi [43] and the reference-based metric XCOMET-XL [44]. COMET-Kiwi is a quality estimation (QE) variant in the COMET family: given a source sentence x and a hypothesis y , it produces a scalar quality score without requiring a human reference. Complementarily, we use XCOMET-XL as a strong reference-based learned metric. XCOMET-XL takes (x, y, r) (source, hypothesis, and reference) as input and outputs a sentence-level quality score. For both metrics, higher values indicate better translations. We report the macro-averages over segments for each translation direction, i.e., the arithmetic mean of sentence-level metric outputs across all test segments in that direction.

Reliability and Limitation of Learned Metric. The two neural semantic evaluation models, i.e., the two learned metrics employed in our work, are trained to approximate human

judgment of translation quality and therefore are typically robust to benign paraphrase and word-order variation that preserves meaning. They also generally capture synonymy (e.g., “lunch” vs “luncheon”) reasonably well since the reliance on contextual representations rather than relying on surface overlap. As a result, they are typically more tolerant to synonymy and meaning-preserving paraphrase than string-based metrics. In contrast, we explicitly note a limitation that the sentence-level learned metrics (XCOMET-XL and c COMET-Kiwi) can still be imperfect in edge cases (rare entities, heavy ambiguity, or subtle scope effects). For logical scope, COMET-style evaluators are trained to detect major adequacy errors, and therefore they are typically expected to assign substantially lower scores when a hypothesis (target) contradicts the source/reference via explicit negation or other logical operators (e.g., “I did not eat at lunch” vs. “I ate at lunch”). We also acknowledge that fine-grained logical entailment remains challenging for any sentence-level learned metric, especially in cases of ambiguity or long-range dependencies. For this reason, we report results with two complementary learned metrics (reference-free COMET-Kiwi and reference-based XCOMET-XL), reducing the risk that conclusions are driven by a single metric’s blind spots.

5.4. Implementation Details

To ensure a fair comparison, all thought generation and aggregation operations are performed with GPT-4o mini accessed through the OpenAI API. GPT-4o mini is a lightweight, cost-efficient model designed for fast, focused generation tasks and supports standard text-based prompting. Note that TransGoT is a prompt-based methodological framework based on a fully pre-trained LLM. This means that there is no loss function, optimizer, or training hyperparameters to report for TransGoT. Considering we focus on the MT task, we use text inputs/outputs only.

5.5. Prompting Protocol

In this section, we present the prompts used for each type of thought.

5.5.1. Constraint Thoughts $\mathcal{Z}^{(c)}$

We prompt the LLM to canonicalize all requirements into an executable checklist. The output schema is

$$\mathcal{Z}^{(c)} = \{\mathcal{T}, \mathcal{D}, \mathcal{F}\}, \quad (15)$$

where $\mathcal{T}$ is a terminology map (source term $\rightarrow$ required target term), $\mathcal{D}$ provides domain grounding, and $\mathcal{F}$ lists forbidden operations.

Template:

System: You are a professional translator. Follow instructions strictly.

User:

[Task] Extract constraints for translating the SOURCE into TARGET.

Return a checklist with the following sections exactly:

[TERMS] source_term $\rightarrow$ required_target_term (one per line; empty if none)

[DOMAIN] domain + intended audience + key context assumptions

[DO-NOT] forbidden actions (do not add/delete info; do not translate product names)

SOURCE:

{x}

5.5.2. Draft Thoughts $\mathcal{Z}^{(d)}$

Conditioned on $(x, \mathcal{Z}^{(c)})$, we generate an initial translation draft.

Template:

System: You are a professional translator. Follow instructions strictly.

User:

Translate the SOURCE into TARGET. You SHOULD comply with the checklist.

[CHECKS]

- terms satisfied: {T}
- domain assumptions: {D}
- forbidden patterns: {F}

Output format:

[TRANSLATION]

<your translation>

SOURCE:

{x}

5.5.3. Culture-Aware Thoughts $\mathcal{Z}^{(k)}$

We construct an uncertainty-masked draft $\tilde{\mathcal{Z}}^{(d)}$ (Section Uncertainty-Driven Masking for Draft Revision). The culture-aware thought refines idioms, pragmatic intent, and culturally loaded references.

Template:

System: You are a professional translator. Be faithful and culturally appropriate.

User:

You are given:

(1) SOURCE

(2) a DRAFT with [MASK] markers indicating uncertain tokens/spans

Task:

- Rewrite the DRAFT into a culturally appropriate translation.
- You MUST rewrite every [MASK] region (no [MASK] remains).
- Preserve all FORMAT invariants and required TERMS.

Output format:

[TRANSLATION]

<revised translation>

SOURCE:

{x}

DRAFT:

{Masked_Draft}

5.5.4. Style-Aware Thought $\mathcal{Z}^{(s)}$

Given $(x, \mathcal{Z}^{(s)})$, we enforce style goals after cultural resolution.

Template:

System: You are a professional translator. Follow instructions strictly.

User:

Rewrite the TRANSLATION to match the [STYLE-NOTES].

This is an optional override when the TARGET and SOURCE styles are inconsistent.

[STYLE-NOTES]

- tone
- register

- formality
- politeness

Output format:
[TRANSLATION]
<styled translation>

SOURCE:
{x}
CURRENT TRANSLATION:
{Z(k)}

5.6. Results

Table 2 reports the evaluation on WMT22 using a reference-free metric and a reference-based metric. We have the following observations: (1) TransGoT achieves the best overall performance across all language directions on both metrics, outperforming the strongest prior baseline TEaR by +0.24/+0.21 on average for EN→X (COMET-Kiwi/XCOMET-XL) and +0.33/+0.27 for X→EN. These results suggest that structured graph-based reasoning can provide benefits beyond linear self-refinement in our setting; (2) The gains over TEaR are especially pronounced for CS→EN on COMET-Kiwi (+0.56) and for EN→ZH / EN→CS on XCOMET-XL (+0.37 / 0.29), suggesting that our uncertainty-guided revisions and multi-criteria selection are particularly effective; (3) In several directions, CT yields marginal or negative changes e.g., for DE→EN, COMET-Kiwi: −0.17, and for CS→EN, XCOMET-XL: −0.20, suggesting that iterative revision without targeted guidance may sometimes introduce regressions. In contrast, TransGoT's structured search and targeted editing mitigate such drift; (4) We perform significance testing using a paired *t*-test with $\alpha = 0.05$ on segment-level metric scores. For each translation direction and metric, we compute scores for the TransGoT and baseline on the same test segments and test the per-segment differences. Results show that TransGoT significantly outperforms all the baselines in most cases.

Table 2. The experimental results on WMT22. The up and down arrows indicate whether the method improves or deteriorates on the corresponding metrics.

Settings	Metrics	IT	CT	SCoT	TEaR	TransGoT
Zh→En	COMET-Kiwi	79.29	79.25 (↓0.04)	79.38 (↑0.09)	79.46 (↑0.17)	79.83 (↑0.37)
	XCOMET-XL	87.22	87.12 (↓0.10)	87.07 (↓0.15)	87.47 (↑0.25)	87.71 (↑0.24)
De→En	COMET-Kiwi	80.29	80.12 (↓0.17)	80.44 (↑0.15)	81.09 (↑0.80)	81.27 (↑0.18)
	XCOMET-XL	95.00	95.02 (↑0.02)	95.38 (↑0.38)	95.66 (↑0.66)	95.92 (↑0.26)
Ru→En	COMET-Kiwi	81.29	81.30 (↑0.01)	81.35 (↑0.06)	81.49 (↑0.20)	81.68 (↑0.19)
	XCOMET-XL	95.08	95.04 (↓0.04)	95.22 (↑0.14)	95.25 (↑0.17)	95.60 (↑0.35)
Cs→En	COMET-Kiwi	82.09	82.18 (↑0.09)	82.26 (↑0.17)	82.46 (↑0.37)	83.02 (↑0.56)
	XCOMET-XL	94.34	94.14 (↓0.20)	94.30 (↓0.04)	94.94 (↑0.60)	95.17 (↑0.23)
Avg.	COMET-Kiwi	80.74	80.71 (↓0.03)	80.86 (↑0.12)	81.12 (↑0.38)	81.45 (↑0.33)
	XCOMET-XL	92.91	92.83 (↓0.08)	92.99 (↑0.08)	93.33 (↑0.42)	93.60 (↑0.27)
En→Zh	COMET-Kiwi	82.07	82.08 (↑0.01)	82.13 (↑0.06)	82.33 (↑0.26)	82.60 (↑0.27)
	XCOMET-XL	87.75	87.72 (↓0.03)	87.81 (↑0.06)	87.99 (↑0.24)	88.36 (↑0.37)
En→De	COMET-Kiwi	83.48	83.33 (↓0.15)	83.56 (↑0.08)	83.55 (↑0.07)	83.71 (↑0.16)
	XCOMET-XL	96.72	96.68 (↓0.04)	96.78 (↑0.06)	96.76 (↑0.04)	96.87 (↑0.11)
En→Ru	COMET-Kiwi	83.62	83.56 (↓0.06)	83.66 (↑0.04)	83.80 (↑0.18)	84.24 (↑0.44)
	XCOMET-XL	93.03	92.99 (↓0.04)	93.10 (↑0.07)	93.16 (↑0.13)	93.21 (↑0.05)
En→Cs	COMET-Kiwi	84.56	84.58 (↑0.02)	84.63 (↑0.07)	84.82 (↑0.26)	84.93 (↑0.11)
	XCOMET-XL	93.08	93.08 (↑0.00)	93.12 (↑0.04)	93.41 (↑0.33)	93.70 (↑0.29)
Avg.	COMET-Kiwi	83.43	83.39 (↓0.04)	83.50 (↑0.07)	83.63 (↑0.20)	83.87 (↑0.24)
	XCOMET-XL	92.65	92.62 (↓0.03)	92.70 (↑0.05)	92.83 (↑0.18)	93.04 (↑0.21)

5.7. Ablation Analysis

Table 3 presents an ablation study on two translation directions (Zh→En and En→Zh), aiming to quantify the contributions of the two key components in TransGoT.

Table 3. Ablation results. The up and down arrows indicate whether the method improves or deteriorates on the corresponding metrics.

Method	Zh→En		En→Zh	
	COMET-Kiwi	XCOMET-XL	COMET-Kiwi	XCOMET-XL
TransGoT	79.83	87.71	82.60	88.36
- <i>w/o</i> Uncertainty	79.65 (↓0.18)	87.64 (↓0.07)	82.44 (↓0.16)	88.28 (↓0.08)
- Top-10 logprob	79.82 (↓0.01)	87.69 (↓0.02)	82.63 (↑0.03)	88.35 (↓0.01)
- Top-5 logprob	79.80 (↓0.03)	87.68 (↓0.03)	82.54 (↓0.06)	88.31 (↓0.05)
- <i>w/o</i> Weighted Score	79.71 (↓0.12)	87.53 (↓0.18)	82.41 (↓0.19)	88.20 (↓0.16)

5.7.1. Impact of Uncertainty-Driven Thought Transformation

Removing the uncertainty component (- *w/o* Uncertainty) consistently degrades performance across directions and metrics. In particular, COMET-Kiwi drops by 0.18 (Zh→En) and 0.16 (En→Zh), while XCOMET-XL decreases by 0.07 and 0.08, respectively. These results indicate that uncertainty-guided thoughts transformation is effective in identifying low-confidence segments for targeted refinement, thereby improving overall translation quality. Moreover, we can deduce that the uncertainty-based masking version of TransGoT is preferred when the logprobs of tokens are available, but our method remains usable when logprobs are unavailable.

Additionally, we examine portability under degraded probability access by restricting uncertainty estimation to truncated top-k logprobs. As shown in Table 3, using Top-10 logprob yields almost identical performance to the full setting, with changes within 0.01–0.03 on COMET-Kiwi and 0.01–0.02 on XCOMET-XL across both directions. When the exposure is further reduced to Top-5 logprob, performance degrades slightly more (most notably on En→Zh (up to 0.06 on COMET-Kiwi and 0.05 on XCOMET-XL)). But the overall drop remains modest. This indicates that the binned-entropy approximation retains stable behavior under truncated logprob access, and TransGoT's uncertainty-guided masking remains effective even when probability information is partial (a common case for closed APIs).

5.7.2. Effectiveness of Dispersion-Adaptive Scoring

Replacing our dispersion-adaptive scoring with a simple unweighted average over criteria (- *w/o* Weighted Score) leads to larger regressions: −0.12/−0.18 on Zh→En (COMET/XCOMET) and −0.19/−0.16 on En→Zh. This suggests that equal-weight aggregation under-utilizes discriminative criteria and can be overly influenced by less informative dimensions, whereas weighting criteria according to their dispersion better reflects their selection utility and yields more reliable intermediate-thought ranking.

Overall, both modules contribute meaningfully to TransGoT. Uncertainty-driven transformation primarily improves the editing stage by focusing revisions on error-prone spans, while dispersion-adaptive scoring strengthens the thought selection stage by stabilizing multi-criteria decision-making. The full model achieves the best results on all settings.

5.8. Human Evaluation

While COMET-Kiwi and XCOMET-XL are strong indicators of overall translation quality, claims about culture-aware and style-aware stages benefit from direct human assessment. Therefore, we add a human evaluation in this section. Table 4 reports the average ratings (0–4 for very bad to very good) from three bilingual annotators on 100 randomly selected En→Zh samples. TransGoT consistently outperforms the strongest baseline TEaR on both dimensions, improving cultural appropriateness from 2.81 to 3.37 (+0.56) and style adherence from 3.04 to 3.46 (+0.42). This indicates that the explicit culture-aware and style-aware refinement stages lead to translations that better follow Chinese cultural conventions and maintain a more consistent target style.

Table 4. Human evaluation results.

Settings	Methods	Cultural Appropriateness	Style Adherence
En → Zh	TEaR	2.81	3.04
	TransGoT	3.37	3.46

5.9. Inference Cost and Practical Considerations

TransGoT is a multi-call inference pipeline with branching, pruning, masking, and re-scoring, and thus may incur higher inference compute than single-pass methods. We report per-segment token usage for transparency. On average, TransGoT consumes 1.18k prompt tokens and 1.05k completion tokens per segment. For the strongest baseline TEaR, the corresponding numbers are 1.17k prompt tokens and 0.98k completion tokens. While TEaR does not adopt iterative refinement, it includes MQM-style annotation/instruction prompts that already introduce a substantial prompt-token overhead. Consequently, TransGoT does not substantially increase total token consumption relative to TEaR. However, for calls, our method invokes the LLM more than a dozen times, whereas TEaR requires only three calls. TransGoT typically requires a larger number of LLM calls, which might increase inference latency in practice. Reducing the number of calls is an important direction for future work, e.g., via early stopping, adaptive branching/pruning conditioned on segment difficulty, or stage consolidation to compress multiple refinement steps into fewer calls. We believe these directions can significantly improve the latency-quality trade-off while preserving the controllability benefits of TransGoT.

6. Discussion and Future Work

In this paper, we introduce TransGoT, a graph-of-thoughts-inspired prompting framework for machine translation. TransGoT formulates translation as a structured, multi-stage reasoning process over a directed graph of translation thoughts. A key strength of TransGoT lies in its task-specific adaptation of the GoT paradigm. Starting from constraint thoughts, TransGoT generates editable draft thoughts and then performs targeted revision through uncertainty-driven masking, followed by culture-aware and style-aware transformations. To robustly select intermediate thoughts under multi-dimensional quality criteria, we further propose a dispersion-adaptive scoring mechanism that emphasizes criteria that better discriminate among candidates. Together, these components turn translation into an explicit search-and-refine process over a compact thought graph, yielding stronger and more controllable translations. Empirical results on WMT22 benchmarks demonstrate the effectiveness of structured graph-based reasoning for translation.

Despite its effectiveness, TransGoT also has several limitations. First, the framework requires multiple LLM calls (generation, transformation, and scoring), which increases inference cost and latency; reducing the number of vertices via adaptive branching/pruning

policies is an important next step. Second, uncertainty-driven masking relies on token-level probabilities: while straightforward for open-weight models, closed-weight APIs may expose only partial probability information (or none), and the binned-entropy approximation may be noisy for certain languages or decoding settings. These limitations point to several promising directions for future work. One avenue is to develop more efficient graph control and pruning strategies to reduce computational overhead. Another is to explore richer uncertainty signals beyond token entropy, such as semantic or discourse-level uncertainty. Additionally, extending the current graph to a graph neural network (GNN)-based learnable module, i.e., encoding intermediate thoughts and performing information passing for more systematic aggregation, is also a promising direction.

Author Contributions: Conceptualization, D.Z. and J.Z.; methodology, J.Z., D.Z., Y.L. and C.X.; formal analysis, J.Z., D.Z. and C.X.; data curation, Y.L. and J.Z.; software, Y.L. and J.Z.; writing—original draft preparation, D.Z. and J.Z.; writing—review and editing, D.Z., J.Z. and C.X.; supervision, J.Z. and C.X.; funding acquisition, J.Z. and C.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. 62502051), Postdoctoral Fellowship Program of CPSF under Grant Number GZC20251101, Guangdong Basic and Applied Basic Research Foundation (Program No. 2025A1515010267), Postdoctoral Research Project Sustentation Funds of Shaanxi (No. 2024BSHSDZZ112), and the Fundamental Research Funds for the Central Universities CHD (No. 300102405103).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used in this work is publicly available and can be found in the original paper that we have cited.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lopez, A. Statistical machine translation. *ACM Comput. Surv. (CSUR)* **2008**, *40*, 1–49.
2. Stahlberg, F. Neural machine translation: A review. *J. Artif. Intell. Res.* **2020**, *69*, 343–418.
3. Koehn, P.; Och, F.J.; Marcu, D. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2003; pp. 127–133.
4. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 3104–3112.
5. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 7–9 May 2015.
6. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
7. Zhang, B.; Haddow, B.; Birch, A. Prompting large language model for machine translation: A case study. In *Proceedings of the International Conference on Machine Learning*, PMLR, Honolulu, HI, USA, 23–29 July 2023; pp. 41092–41110.
8. Zhu, D.; Chen, P.; Zhang, M.; Haddow, B.; Shen, X.; Klakow, D. Fine-tuning large language models to translate: Will a touch of noisy data in misaligned languages suffice? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 388–409.
9. Xu, H.; Kim, Y.J.; Sharaf, A.; Awadalla, H.H. A paradigm shift in machine translation: Boosting translation performance of large language models. In *Proceedings of the Twelfth International Conference on Learning Representations (ICLR)*, Vienna, Austria, 7–11 May 2024.
10. Dong, Y.; Lyu, X.; Li, J.; Wei, D.; Zhang, M.; Tao, S.; Yang, H. Two intermediate translations are better than one: Fine-tuning LLMs for document-level translation refinement. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL 2025, Volume 1: Long Papers)*, Vienna, Austria, 27 July–1 August 2025; pp. 14917–14933.

11. Liu, Y.; Gu, J.; Goyal, N.; Li, X.; Edunov, S.; Ghazvininejad, M.; Lewis, M.; Zettlemoyer, L. Multilingual denoising pre-training for neural machine translation. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 726–742.
12. Xue, L.; Constant, N.; Roberts, A.; Kale, M.; Al-Rfou, R.; Siddhant, A.; Barua, A.; Raffel, C. mT5: A massively multilingual pre-trained text-to-text transformer. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 6–11 June 2021; pp. 483–498.
13. Alves, D.; Guerreiro, N.; Alves, J.; Pombal, J.; Rei, R.; de Souza, J.; Colombo, P.; Martins, A.F. Steering large language models for machine translation with finetuning and in-context learning. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, 6–10 December 2023; pp. 11127–11148.
14. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
15. Vilar, D.; Freitag, M.; Cherry, C.; Luo, J.; Ratnakar, V.; Foster, G. Prompting palm for translation: Assessing strategies and performance. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Toronto, ON, Canada, 9–14 July 2023; pp. 15406–15427.
16. Karpinska, M.; Iyyer, M. Large language models effectively leverage document-level context for literary translation, but critical errors persist. In Proceedings of the Eighth Conference on Machine Translation (WMT 2023), Singapore, 6–7 December 2023; pp. 419–451.
17. He, Z.; Liang, T.; Jiao, W.; Zhang, Z.; Yang, Y.; Wang, R.; Tu, Z.; Shi, S.; Wang, X. Exploring human-like translation strategy with large language models. *Trans. Assoc. Comput. Linguist.* **2024**, *12*, 229–246.
18. Hu, K.; Chen, Z.; Yang, C.H.H.; Żelasko, P.; Hrinchuk, O.; Lavrukhin, V.; Balam, J.; Ginsburg, B. Chain-of-thought prompting for speech translation. In *Proceedings of the ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; IEEE: New York, NY, USA, 2025; pp. 1–5.
19. Feng, Z.; Zhang, Y.; Li, H.; Wu, B.; Liao, J.; Liu, W.; Lang, J.; Feng, Y.; Wu, J.; Liu, Z. Tear: Improving llm-based machine translation with systematic self-refinement. In *Proceedings of the Findings of the Association for Computational Linguistics: NAACL 2025*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2025; pp. 3922–3938.
20. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q.V.; Zhou, D.; et al. Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 24824–24837.
21. Yao, S.; Yu, D.; Zhao, J.; Shafran, I.; Griffiths, T.; Cao, Y.; Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 11809–11822.
22. Besta, M.; Blach, N.; Kubicek, A.; Gerstenberger, R.; Podstawski, M.; Gianinazzi, L.; Gajda, J.; Lehmann, T.; Niewiadomski, H.; Nyczyk, P.; et al. Graph of thoughts: Solving elaborate problems with large language models. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024; Volume 38, pp. 17682–17690.
23. Yao, B.; Jiang, M.; Bobinac, T.; Yang, D.; Hu, J. Benchmarking machine translation with cultural awareness. In *Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2024*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 13078–13096.
24. Kim, D.; Baek, Y.; Yang, S.; Choo, J. Towards Formality-Aware Neural Machine Translation by Leveraging Context Information. In *Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2023; pp. 7384–7392.
25. Xiao, Y.; Wang, W.Y. On hallucination and predictive uncertainty in conditional language generation. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2021): Main Volume, Online, 19–23 April 2021; pp. 2734–2744.
26. Geng, J.; Cai, F.; Wang, Y.; Koepl, H.; Nakov, P.; Gurevych, I. A survey of confidence estimation and calibration in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 6577–6595.
27. Alves, D.M.; Pombal, J.; Guerreiro, N.M.; Martins, P.H.; Alves, J.; Farajian, A.; Peters, B.; Rei, R.; Fernandes, P.; Agrawal, S.; et al. Tower: An open multilingual large language model for translation-related tasks. In Proceedings of the First Conference on Language Modeling, Philadelphia, PA, USA, 7–9 October 2024.
28. Stap, D.; Hasler, E.; Byrne, B.; Monz, C.; Tran, K.M. The fine-tuning paradox: Boosting translation quality without sacrificing LLM abilities. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers); Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 6189–6206.
29. Maslej, N.; Fattorini, L.; Perrault, R.; Gil, Y.; Parli, V.; Kariuki, N.; Capstick, E.; Reuel, A.; Brynjolfsson, E.; Etchemendy, J.; et al. Artificial Intelligence Index Report 2025. *arXiv* **2025**, arXiv:2504.07139.
30. Awadalla, H.H. How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation. *Microsoft Research*, February 2023. Available online: <https://www.microsoft.com/en-us/research/publication/how-good-are-gpt-models-at-machine-translation-a-comprehensive-evaluation/> (accessed on 25 February 2026).

31. Robinson, N.; Ogayo, P.; Mortensen, D.R.; Neubig, G. ChatGPT MT: Competitive for high- (but not low-) resource languages. In *Proceedings of the Eighth Conference on Machine Translation (WMT 2023)*, Singapore, 6–7 December 2023; pp. 392–418.
32. Agrawal, S.; Zhou, C.; Lewis, M.; Zettlemoyer, L.; Ghazvininejad, M. In-context examples selection for machine translation. In *Proceedings of the Findings of the Association for Computational Linguistics: ACL 2023*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2023; pp. 8857–8873.
33. Chen, P.; Guo, Z.; Haddow, B.; Heafield, K. Iterative translation refinement with large language models. In *Proceedings of the 25th Annual Conference of the European Association for Machine Translation (Volume 1)*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 181–190.
34. Kojima, T.; Gu, S.S.; Reid, M.; Matsuo, Y.; Iwasawa, Y. Large language models are zero-shot reasoners. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 22199–22213.
35. Madaan, A.; Hermann, K.; Yazdanbakhsh, A. What makes chain-of-thought prompting effective? a counterfactual study. In *Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2023; pp. 1448–1535.
36. Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.V.; Chi, E.H.; Narang, S.; Chowdhery, A.; Zhou, D. Self-consistency improves chain of thought reasoning in language models. In *Proceedings of the Eleventh International Conference on Learning Representations (ICLR)*, Kigali, Rwanda, 1–5 May 2023.
37. Raunak, V.; Sharaf, A.; Wang, Y.; Awadalla, H.; Menezes, A. Leveraging GPT-4 for automatic translation post-editing. In *Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2023; pp. 12009–12024.
38. Nguyen, L.; Xu, Y. Reasoning for Translation: Comparative Analysis of Chain-of-Thought and Tree-of-Thought Prompting for LLM Translation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2025; pp. 259–275.
39. Sennrich, R.; Haddow, B.; Birch, A. Controlling politeness in neural machine translation via side constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2016; pp. 35–40.
40. Kocmi, T.; Federmann, C. GEMBA-MQM: Detecting Translation Quality Error Spans with GPT-4. In *Proceedings of the Eighth Conference on Machine Translation, WMT 2023, Singapore, 6–7 December 2023*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2023; pp. 768–775.
41. Fernandes, P.; Deutsch, D.; Finkelstein, M.; Riley, P.; Martins, A.F.T.; Neubig, G.; Garg, A.; Clark, J.H.; Freitag, M.; Firat, O. The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation. In *Proceedings of the Eighth Conference on Machine Translation (WMT 2023)*, Singapore, 6–7 December 2023; pp. 1066–1083.
42. Kocmi, T.; Bawden, R.; Bojar, O.; Dvorkovich, A.; Federmann, C.; Fishel, M.; Gowda, T.; Graham, Y.; Grundkiewicz, R.; Haddow, B.; et al. Findings of the 2022 conference on machine translation (WMT22). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, Abu Dhabi, United Arab Emirates, 7–8 December 2022; pp. 1–45.
43. Rei, R.; Treviso, M.; Guerreiro, N.M.; Zerva, C.; Farinha, A.C.; Maroti, C.; De Souza, J.G.; Glushkova, T.; Alves, D.; Coheur, L.; et al. CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2022; pp. 634–645.
44. Guerreiro, N.M.; Rei, R.; Stigt, D.v.; Coheur, L.; Colombo, P.; Martins, A.F. xcomet: Transparent machine translation evaluation through fine-grained error detection. *Trans. Assoc. Comput. Linguist.* **2024**, *12*, 979–995.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.