# Agentic AI Translate: An Agentic Translator Prototype for Translation as Communication Design

Masaru Yamada
Rikkyo University; Translation Lab Inc.

May 2026

**Abstract**

We present **Agentic AI Translate**, an agentic translator prototype that operationalises the thesis of Yamada (forthcoming) — that the metalanguage of Translation Studies has become an instruction code for generative AI. The system replaces the dominant *text-in / text-out* paradigm of machine translation with a four-stage agentic cycle (**Identify** → **Prompt** → **Generate** → **Verify**), preceded by an **interactive specification phase** in which the user composes — through model-assisted dialogue — a structured translation brief grounded in skopos theory, register, audience, and genre conventions. The verification stage adopts the GEMBA-MQM error-span protocol [Kocmi & Federmann, 2023] for evidence-grounded scoring, and document-level coherence is preserved through a *DelTA-lite* memory of proper nouns and a running bilingual summary, after [Wang et al., 2025]. We describe the philosophical motivation, the architectural commitments, the four reference-material categories the system consumes, and the principal design tensions the architecture makes explicit. Empirical validation is left for future work; the contribution here is conceptual and architectural — an executable embodiment of the position that *translation in the GenAI era is communication design, not text conversion.*

**Keywords:** agentic translation, translation studies metalanguage, skopos, MQM, document-level translation, large language models, translation specifications.

## 1 Introduction

For four decades, machine translation research has been organised around a single optimisation target: lexical and grammatical fidelity between a source string and a target string. Statistical and neural systems progressively closed the accuracy gap to professional human output across high-resource pairs, and large language models (LLMs) have now made fluency, idiomaticity, and basic register-matching nearly free at the segment level [Kocmi et al., 2024; Karpinska & Iyyer, 2023]. In Kano-model terms [Kano, 1984], **accuracy has saturated as a Must-Be quality**: its presence no longer differentiates translations, only its absence is noticed.

The frontier of translation value has therefore moved to what Tannen (1986) called the *how* rather than the *what* — register, audience design, voice, stance, cultural framing, genre convention — the dimensions that have always mattered to professional translators but which computational research has historically left implicit. Yamada (forthcoming), in *Metalanguage and GenAI: Empowering Language Learners and Translators in Training* (forthcoming in *The Routledge Handbook of Translation and Technology*, 2nd ed.), argues that this is not merely a shift in evaluation criteria but a fundamental **reconfiguration of the translator's role**: from the manual drafter of target text toward the **designer of conditions** under which a generative system produces text — and the **verifier** of whether that text fulfils its communicative purpose. Crucially, Yamada observes:

*"The easier it becomes to generate text, the harder it becomes to ensure that text fulfils a specific communicative purpose."*

This *automation paradox* dissolves once we recognise that the vocabulary of Translation Studies (TS) — *skopos, register, audience, equivalence, foreignization, domestication, genre, stance, loyalty* — provides exactly the descriptive precision an LLM needs as instruction. **Theory becomes operational.** What was previously studied to think about practice is now spoken to instruct the machine.

This paper presents an executable embodiment of that argument. **Agentic AI Translate** is a research prototype and an agentic translator, openly released, that takes a translation request and walks the user through a structured authoring of a translation specification before any generation occurs. It then runs an agentic four-stage pipeline (*Identification* → *Prompting* → *Generation* → *Verification*) that uses the specification end-to-end, with document-level state to preserve terminological consistency across long inputs. The contribution is not empirical — we have not yet conducted the comparative MQM study against unstructured prompting that would validate the hypothesis — but architectural: an executable description of *what such a system must contain* if the position outlined above is to be realised in code.

The remainder of the paper is organised as follows. Section 2 develops the philosophical motivation. Section 3 specifies the architecture. Section 4 details the implementation. Section 5 positions the system relative to recent work in agentic LLM translation, document-level MT, and translation evaluation. Section 6 discusses limitations and the main design tensions. Section 7 outlines the validation plan and the structured-spec extension that constitutes the project's main research direction.

## 2 Philosophical Motivation: Translation as Communication Design

### 2.1 The two layers, restated for the GenAI era

Translation has always operated on two layers: the propositional content (the *what*) and the realisation that content takes in the target language (the *how*) — including register, sentence rhythm, sociolectal markers, footnoting practice, the management of culturally bound items, and the addressivity that positions the implied reader. House's (2015) overt/covert distinction, Reiss's (1971/2000) text-typology, Nord's (1997) functionalist framing, and Vermeer's (1978) skopos all foreground the priority of communicative purpose over surface equivalence; this is not a new observation but a consensus in TS [see Munday, 2016, for the canonical synthesis].

The *new* observation is that, until recently, encoding such constraints into a translation system meant either training a domain-specific model or post-editing the output of a generic one. Both approaches treated communicative design as something *applied to* translation rather than something *constitutive of* it. Generative LLMs change this: they accept long, structured natural-language instructions at inference time and condition their generation on those instructions to a degree that is qualitatively different from prior systems [Vilar et al., 2023; Karpinska & Iyyer, 2023]. **Communicative design becomes a first-class input.**

### 2.2 Voice as the unit of translation

Consider Murakami Haruki's translation of Salinger's *The Catcher in the Rye* into Japanese. The opening — "If you really want to hear about it..." — has been rendered by multiple Japanese translators with varying degrees of literal fidelity, but Murakami's choice deliberately preserves not

the surface lexicon of Salinger but the *voice* of Holden Caulfield: a particular cadence, a particular relation to the reader. A few-shot prompt to a current LLM, given a short Murakami sample, can reproduce that voice for adjacent passages with striking fidelity — not because the LLM has read Salinger or Murakami, but because the voice has been **specified** as a constraint that the model can honour.

This is the operational core of the *translation-as-design* claim: voice — the most apparently artisanal aspect of literary translation — turns out to be **specifiable**, and once specified, it is reproducible at scale. The translator's contribution is no longer manual drafting of every sentence, but the **design of voice as a constraint**.

### 2.3 The translator's reconfiguration

Yamada (forthcoming) frames the translator's emerging role as **designer + verifier**:

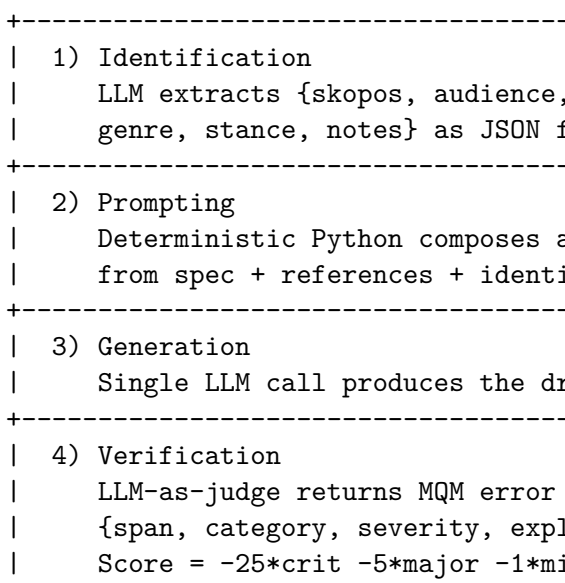
- **Designer**: composes, with metalinguistic precision, the situational analysis (skopos, audience, register, genre) and the operational artefacts (glossaries, paired examples, parallel-text exemplars) that condition the generative system.
- **Verifier**: judges output not as a post-edit task — surface error correction — but as a *functional* and *epistemic* judgement: does the output land for the audience? does it preserve factual structure? does it match the spec?

The critical pedagogical implication is that the **vocabulary** of TS — what Gambier (2009) called the *meta-language* of the discipline — is no longer studied to *think about* practice but to *instruct* the machine. The discipline's theoretical apparatus becomes operational infrastructure. The system described here is built on that recognition.

## 3 Architecture

The system comprises three concentric layers: the **four-stage cycle** (the pipeline), the **interactive specification** that conditions every stage, and the **persistent state** that preserves document-level coherence.

### 3.1 The four-stage cycle



```
|     Verdict computed deterministically vs. threshold.   |
|     If revise: errors fed back as Stage 2 refinement;   |
|     up to two iterations.                               |
+---------------------------------------------------------+
```

**Why four stages, not one?** A single end-to-end prompt forces the model to perform situational analysis, prompt assembly, generation, and self-evaluation in a single forward pass — producing fluent but largely unanalysable output. Decomposition exposes each commitment as an inspectable artefact. The Identification JSON, the assembled Stage 2 prompt, and the Verification error spans are all logged and visualised in the user interface; this is by design, since the pedagogical and research value of the system depends on each stage being legible.

**Why Stage 1 is a separate LLM call.** Situational analysis can in principle be folded into a single generation prompt. We separate it for two reasons. First, the JSON it produces — `{skopos, audience, register, genre, stance, notes}` — is the *most direct embodiment* of the metalanguage thesis: TS categories appear as structured fields, not as prose. Second, separating it allows the user to *see* the situational analysis the model has performed and challenge it before generation. In current practice this remains a read-only artefact; making it user-editable is a planned extension.

### 3.2 Interactive specification

The most distinctive element of the system is the layer that precedes the pipeline. After source text entry, the user clicks **Propose spec**; the model returns a structured markdown document with ten canonical sections — *Skopos, Audience, Register & Voice, Genre, Terminology guidance, Style decisions, Things to preserve, Things to localise, Things to avoid, Open questions* — drafted from the source and any uploaded references. The user may:

1. Edit the markdown directly in the UI;
2. Refine via chat ("audience is academic peer reviewers", "use plain da/dearu style throughout", "preserve emoji and source-language fan vocabulary");
3. Iterate until satisfied, then **lock** the spec, after which translation may run.

The lock step is intentional. It enforces an **architectural commitment** that no translation can be produced without an explicit, user-endorsed specification. The system therefore cannot be used as a generic MT tool; it can only be used as a *spec-driven translation tool*. This is the philosophical position made operational.

The specification is consumed identically by Stage 2 (Prompting) and Stage 4 (Verification): the verifier judges the translation against the same spec the generator was conditioned on. This closes a common evaluation loophole in which the verifier and the generator implicitly disagree about what counts as good output.

### 3.3 Reference-material layer

Four orthogonal categories of reference materials may be uploaded:

| Category | Format | Functional role |
|---|---|---|
| Glossary | TSV/CSV (source ↔ target) | Mandatory terminology |
| Paired examples | TSV/CSV (source ↔ target) | Translation-judgement few-shot |
| Parallel target-language texts | TXT/MD | Genre-voice exemplars |
| Style guide | MD/TXT | Free-form narrative constraint |

These categories follow the pragmatic taxonomy used in professional CAT/TMS workflows and partially align with the ASTM F2575 standard for translation specifications. The system injects all four into the spec proposal, the generation prompt, and the verifier — each consumer can decide how to weigh them. The current implementation injects all paired examples; selective retrieval (R-BM25 or embedding similarity, after Agrawal et al., 2023) is a planned upgrade.

### 3.4 Document-level memory (DelTA-lite)

For multi-paragraph inputs, the chunker splits the document at blank-line paragraph boundaries (with sentence-boundary fallback for over-long paragraphs). Each chunk is translated independently, but between chunks a **persistent memory** is updated by an auxiliary LLM call, after Wang et al. (2025):

- **Proper-noun ledger**: a running source-to-target dictionary of terms whose translations should remain stable (people, places, organisations, products, technical terms).
- **Bilingual running summary**: 50–150 words in the target language, capturing the document's progression for tonal continuity.
- **Immediate-window context**: the previous chunk's source and target text.

These three artefacts are injected into the next chunk's Stage 2 prompt under explicit headings (*Established terminology*, *Document summary so far*, *Immediately preceding chunk*) and the model is instructed to honour them. In informal observation on multi-paragraph literary and journalistic test inputs, the ledger correctly captures named entities that recur across chunks (e.g., *Natsume Soseki* → *Natsume Soseki*, *Kushami-sensei* → *Kushami*) without further intervention, mirroring the consistency improvements reported by Wang et al. (2025) at scale.

### 3.5 MQM-grounded verification

Stage 4 follows the **GEMBA-MQM** protocol of Kocmi & Federmann (2023): the verifier prompt is language-agnostic, instructs the model to identify error spans and assign each one an MQM category and severity, and returns a structured JSON list. The category set is the canonical inventory of Freitag et al. (2021): *Accuracy* (mistranslation, addition, omission, untranslated, do-not-translate), *Fluency* (grammar, punctuation, spelling, register, inconsistency, character encoding), *Terminology, Style, Locale convention, Other*. Severity is one of *critical, major, minor*. From the error list a deterministic score is computed:

$$\text{score} = -25 \cdot n_{\text{critical}} - 5 \cdot n_{\text{major}} - 1 \cdot n_{\text{minor}}$$

The verdict is *accept* if the score meets a configurable threshold (default -2, i.e. up to two minor issues are tolerated; any major or critical triggers revision), otherwise *revise*. On revision, the typed error list is appended verbatim to the Stage 2 prompt as actionable instructions, and Stage 3 re-runs.

The loop is bounded at two iterations — both Huang et al. (2024) and Stechly et al. (2024) show that intrinsic LLM self-correction yields rapidly diminishing returns and can degrade output.

We follow Fernandes et al. (2023) and Wang et al. (2024) in requiring the verifier to **emit evidence (error spans) before the score**, which empirically reduces verbosity and self-preference biases in LLM-as-judge configurations.

## 4 Implementation

The system is implemented in approximately 1200 lines of Python (excluding prompts and tests). The runtime stack is:

- **Anthropic SDK** with Claude Sonnet 4.6 as the default model (configurable);
- **Streamlit** for the UI;
- **python-dotenv** for local development; in deployment the API key is supplied per-session by the user via the sidebar (no shared key);
- No vector database, no GPU, no fine-tuning.

The minimal stack is intentional: every commitment in the system is in *prompts* and *Python flow control*, not in trained weights. This makes the system fully inspectable and reproducible, and keeps the cost of experimenting with alternative spec structures or verifier prompts at the level of editing a text file.

The repository is publicly available on GitHub under MIT licence (© Translation Lab Inc.) at https://github.com/chuckmy/agentic-translator, and a live demo is deployed on Streamlit Community Cloud at https://agentic-translator-chuckmy.streamlit.app, with bring-your-own API key. A bilingual test set covering three genres (news, literary description, academic abstract) in both translation directions is included to support reproducible exploration.

## 5 Related Work

**Spec-aware MT.** The closest precursor is Kayano & Sugawara (2025), who demonstrate that prompting with an explicit translation specification — purpose, audience, register — significantly improves preference scores on intent-rich texts, sometimes exceeding human reference translations. Their specification is presented as flat free-form text; we extend the idea by making the specification an interactively *authored* artefact with a stable structural template, and by carrying it through both generation and verification rather than only generation.

**Multi-agent translation.** Wu et al. (2024/2025) introduce **TransAgents**, a six-agent simulation (CEO, editor, translator, localizer, proofreader, QA) for ultra-long literary texts; the system is preferred by both expert and crowd judges over GPT-4 single-call output and over reference human translations on book-length input, despite *lower* d-BLEU. This is the strongest available evidence that pipeline decomposition is qualitatively beneficial for the *attractive-quality* dimensions identified in §2. Briakou et al. (2024)'s **Translating Step-by-Step** demonstrates the same principle within a single model — pre-translation research → drafting → refining → proofreading — establishing WMT24 SOTA. Our prototype is closer in shape to the latter than to TransAgents; planned extensions (R5, §7) move toward role decomposition.

**Document-level translation.** Karpinska & Iyyer (2023) document the strong human-evaluation preference for paragraph-level over sentence-level LLM translation, especially in literary registers. **DelTA** (Wang et al., 2025) introduces explicit four-tier memory — proper nouns, bilingual summary, long-term, short-term — with measurable consistency gains. Our DelTA-lite implements the first two tiers.

**LLM-as-judge for translation.** Kocmi & Federmann (2023) establish that GPT-4 with a fixed three-shot MQM prompt produces scores correlating with expert MQM well enough to win the WMT23 metrics task. xCOMET (Guerreiro et al., 2024) and MetricX (Juraska et al., 2023) demonstrate that *learned* metrics still outperform LLM-judge prompts at the segment level on WMT24/25; we accept this and treat the LLM-judge as the system's first line of evaluation while documenting (§6) the planned augmentation by xCOMET as an external signal.

**Self-correction in MT.** Madaan et al. (2023) introduce Self-Refine; Feng et al. (2025)'s **TEaR** shows MQM-typed feedback improves refinement quality. Huang et al. (2024) and Stechly et al. (2024) document the limits of intrinsic self-correction — apparent improvements are often artefacts of sampling diversity rather than genuine self-critique. We bound our revise loop accordingly.

**Translation Studies frameworks operationalised.** Singh et al. (2024) explore cultural and register-specific adaptation with LLMs, and a growing body of work investigates honorific and politeness-marker handling in LLM-mediated translation. The systematic encoding of TS frameworks (Reiss, Nord, House) as machine-readable schema remains an open area, identified in §7 as our primary research direction.

# 6 Discussion and Limitations

## 6.1 What the prototype claims, and what it does not

This is an **architectural** contribution. We claim that the architecture coherently embodies the *translation-as-design* position; we do not yet claim that it produces measurably better translations than alternatives. A controlled comparison — same source, same target, same model, with vs. without spec — across multiple genres and language pairs, evaluated by professional translators using full MQM, is the necessary next step.

## 6.2 Single-model verification is the weakest link

The verifier currently runs on the same model family as the generator. This exposes the system to **self-preference bias** [Zheng et al., 2023; Wang et al., 2024] and to the broader limits of intrinsic self-correction [Huang et al., 2024; Stechly et al., 2024]. The bounded loop (two iterations) and the evidence-first prompt structure (errors before scores) mitigate but do not resolve this. The planned augmentation (R2, §7) introduces a cross-model judge and an external learned QE signal (xCOMET-XL or MetricX).

## 6.3 The spec is currently free-form markdown

The interactive specification is a markdown document with ten canonical headings but otherwise unconstrained content. This permits expressive richness but limits machine readability and forecloses systematic A/B experimentation on individual fields (e.g., does specifying *loyalty target* produce measurable behavioural change?). The planned extension (R6, §7) replaces the markdown with a structured JSON schema operationalising Reiss's text typology, Nord's loyalty, House's overt/covert mode, and a domestication–foreignization continuum, with the markdown view derived from the schema. This is the project's primary research direction.

## 6.4 Reference materials are injected without selection

All paired examples are currently injected into every prompt. Agrawal et al. (2023) show that even one mismatched in-context example can degrade translation quality more than no examples at all;

Vilar et al. (2023) show that example *quality* dominates over similarity at large model scale. Both effects motivate retrieval-based selection (R-BM25 plus embedding similarity over a quality-tagged TU store), which is currently absent.

### 6.5 No empirical evaluation

The most important limitation. The validation plan is sketched in §7.1.

## 7 Future Work

### 7.1 Empirical validation

A factorial study comparing (a) generic-prompt translation, (b) free-form spec, (c) structured-schema spec, across (i) literary, (ii) journalistic, (iii) academic genres, in ($\alpha$) JA→EN and ($\beta$) EN→JA, evaluated by professional translators using Freitag-2021 MQM with ESA severity protocol, would establish whether and where the spec-driven architecture yields measurable gains. Inter-rater agreement and per-axis breakdown (accuracy vs. style vs. terminology) would identify which dimensions of *attractive quality* are most spec-sensitive.

### 7.2 Structured specification schema (R6)

Replace the markdown spec with a JSON schema:

```
{
  "skopos": "...",
  "text_type": "informative | expressive | operative | audiomedial",
  "house_mode": "overt | covert",
  "loyalty": { "author_intention": 0.7,
               "ST_culture_fidelity": 0.5,
               "TT_reader_orientation": 0.9,
               "commissioner_brief": 0.6 },
  "domestication_axis": 0.7,
  "audience": { ... },
  "register": { ... },
  "preserve": [...], "localize": [...], "avoid": [...]
}
```

Reiss's text typology, Nord's loyalty principle, House's overt/covert distinction, and the Schleiermacher–Venuti domestication–foreignization axis become **fields the user fills in**. The same schema underwrites the generation prompt, the verification prompt, and (critically) the experimental design — fields can be ablated, swapped, or held constant across runs to support A/B research that is structurally impossible with free-form spec.

### 7.3 Multi-agent decomposition

Following TransAgents and Briakou et al., split Stage 3 into **research** → **draft** → **localise** → **proofread**, with each pass governed by the same locked specification. Particular value is expected in the *localise* pass (cultural rendering, idiom handling), which TransAgents identified as the largest contributor to expert preference.

### 7.4 External quality signals

Add **xCOMET-XL** or **MetricX** (Juraska et al., 2023) as a parallel verifier whose score gates acceptance alongside the LLM judge. Cross-model judging (Sonnet generator + Gemini or GPT-5 judge) addresses self-preference bias.

### 7.5 Hallucination check

Augment Stage 4 with an entity-preservation check (named entities and numerals in source must appear or have explicit equivalents in target), per Guerreiro et al. (2023). Yamada's *factual* axis is currently the weakest detector for fluent-but-fabricated output.

## 8 Conclusion

We have described a research prototype that embodies, in executable form, the position that translation in the GenAI era is *communication design*. The core architectural commitments — interactive specification, the four-stage cycle, document-level memory, MQM-grounded evidence-first verification — are not arbitrary engineering choices but direct operationalisations of the metalanguage thesis advanced by Yamada (forthcoming). The system is openly released so that colleagues, students, and researchers can interrogate, extend, and contest the position it makes operational.

What remains is the empirical work that this paper has explicitly not undertaken: the controlled study that would tell us whether *making the specification explicit* produces, in measurable MQM terms, the qualitative shift in *attractive quality* that the lecture-platform argument predicts. That study, and the structured-schema extension that it requires, constitutes the project's next phase.

## References


[1] Agrawal, S., Zhou, C., Lewis, M., Zettlemoyer, L., & Ghazvininejad, M. (2023). In-context examples selection for machine translation. In *Findings of ACL 2023* (pp. 8857–8873).

[2] Briakou, E., Luo, J., Cherry, C., & Freitag, M. (2024). Translating step-by-step: Decomposing the translation process for improved translation quality of long-form texts. In *Proceedings of WMT 2024*. arXiv:2409.06790.

[3] Feng, Z., Zhang, Y., Li, H., Liu, W., Lang, J., Feng, Y., Wu, J., & Liu, Z. (2025). TEaR: Improving LLM-based machine translation with systematic self-refinement. In *Proceedings of NAACL 2025*. arXiv:2402.16379.

[4] Fernandes, P., Yin, K., Liu, E., Martins, A. F. T., & Neubig, G. (2023). The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation. arXiv:2308.07286.

[5] Freitag, M., Foster, G., Grangier, D., Ratnakar, V., Tan, Q., & Macherey, W. (2021). Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Transactions of the Association for Computational Linguistics, 9*, 1460–1474.

[6] Freitag, M., et al. (2024). Are LLMs breaking MT metrics? Results of the WMT24 metrics shared task. In *Proceedings of WMT 2024*.

[7] Gambier, Y. (2009). *Stratégies et tactiques en traduction et interprétation.* In Gambier, Y., & Doorslaer, L. van (Eds.), *Handbook of Translation Studies* (Vol. 1). Amsterdam: John Benjamins.

[8] Guerreiro, N. M., Voita, E., & Martins, A. F. T. (2023). Looking for a needle in a haystack: A comprehensive study of hallucinations in neural machine translation. In *Proceedings of EACL 2023* (pp. 1059–1075).

[9] Guerreiro, N. M., Rei, R., van Stigt, D., Coheur, L., Colombo, P., & Martins, A. F. T. (2024). xCOMET: Transparent machine translation evaluation through fine-grained error detection. *Transactions of the Association for Computational Linguistics, 12*, 979–995.

[10] House, J. (2015). *Translation Quality Assessment: Past and Present.* London: Routledge.

[11] Huang, J., Chen, X., Mishra, S., Zheng, H. S., Yu, A. W., Song, X., & Zhou, D. (2024). Large language models cannot self-correct reasoning yet. In *Proceedings of ICLR 2024.* arXiv:2310.01798.

[12] Juraska, J., Finkelstein, M., Deutsch, D., Siddhant, A., Tran, M., & Freitag, M. (2023). MetricX-23: The Google submission to the WMT 2023 metrics shared task. In *Proceedings of WMT 2023.*

[13] Kano, N., Seraku, N., Takahashi, F., & Tsuji, S. (1984). Attractive quality and must-be quality. *Journal of the Japanese Society for Quality Control, 14*(2), 39–48.

[14] Karpinska, M., & Iyyer, M. (2023). Large language models effectively leverage document-level context for literary translation, but critical errors persist. In *Proceedings of WMT 2023* (pp. 419–451).

[15] Kayano, S., & Sugawara, Y. (2025). Specification-aware machine translation and evaluation for purpose alignment. In *Proceedings of WMT 2025.* arXiv:2509.17559.

[16] Kocmi, T., & Federmann, C. (2023). Large language models are state-of-the-art evaluators of translation quality. In *Proceedings of EAMT 2023.* arXiv:2302.14520.

[17] Kocmi, T., & Federmann, C. (2023). GEMBA-MQM: Detecting translation quality error spans with GPT-4. In *Proceedings of WMT 2023.* arXiv:2310.13988.

[18] Kocmi, T., et al. (2024). Findings of the 2024 Conference on Machine Translation (WMT24). In *Proceedings of WMT 2024.*

[19] Madaan, A., et al. (2023). Self-Refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems 36* (NeurIPS 2023). arXiv:2303.17651.

[20] Munday, J. (2016). *Introducing Translation Studies: Theories and Applications* (4th ed.). London: Routledge.

[21] Nord, C. (1997). *Translating as a Purposeful Activity: Functionalist Approaches Explained.* Manchester: St. Jerome.

[22] Reiss, K. (1971/2000). *Translation Criticism: The Potentials and Limitations* (E. Rhodes, Trans.). Manchester: St. Jerome.

[23] Singh, P., Jangra, A., et al. (2024). Translating across cultures: LLMs for intralingual cultural adaptation. In *Proceedings of CoNLL 2024.*

[24] Stechly, K., Valmeekam, K., & Kambhampati, S. (2024). On the self-verification limitations of large language models on reasoning and planning tasks. In *Proceedings of ICML 2024.* arXiv:2402.08115.

[25] Tannen, D. (1986). *That's Not What I Meant! How Conversational Style Makes or Breaks Relationships.* New York: William Morrow.

[26] Vermeer, H. J. (1978). Ein Rahmen für eine allgemeine Translationstheorie. *Lebende Sprachen, 23*, 99–102.

[27] Vilar, D., Freitag, M., Cherry, C., Luo, J., Ratnakar, V., & Foster, G. (2023). Prompting PaLM for translation: Assessing strategies and performance. In *Proceedings of ACL 2023* (pp. 15406–15427).

[28] Wang, P., Li, L., Chen, L., Cai, Z., Zhu, D., Lin, B., Cao, Y., Liu, Q., Liu, T., & Sui, Z. (2024). Large language models are not fair evaluators. In *Proceedings of ACL 2024.* arXiv:2305.17926.

[29] Wang, Y., Zeng, J., Liu, X., Wong, D. F., Meng, F., Zhou, J., & Zhang, M. (2025). DelTA: An online document-level translation agent based on multi-level memory. In *Proceedings of ICLR 2025.* arXiv:2410.08143.

[30] Wu, M., Yuan, Y., Haffari, G., & Wang, L. (2024). (Perhaps) Beyond human translation: Harnessing multi-agent collaboration for translating ultra-long literary texts. *Transactions of the Association for Computational Linguistics* (2025). arXiv:2405.11804.

[31] Yamada, M. (forthcoming). Metalanguage and GenAI: Empowering language learners and translators in training. In M. A. Jiménez-Crespo & V. Enríquez-Raido (Eds.), *The Routledge Handbook of Translation and Technology* (2nd ed.). London: Routledge.

[32] Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., & Stoica, I. (2023). Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems 36* (NeurIPS 2023). arXiv:2306.05685.