

How Important is ‘Perfect’ English for Machine Translation Prompts?

Patrícia Schmidtová^{*,1} Niyati Bafna^{*,2} Seth Aycok^{*,3}
Gianluca Vico¹ Wiktor Kamzela⁴ Kathy Hämmerl^{†,6,7} Vilém Zouhar^{†,5}

¹Charles University, Faculty of Mathematics and Physics ²Johns Hopkins University
³University of Amsterdam ⁴Poznań University of Technology ⁵ETH Zurich
⁶Technical University of Munich ⁷Munich Center for Machine Learning

Abstract

Large language models (LLMs) show state-of-the-art performance in machine translation, but are also known to be sensitive to errors in user prompts. Given that these models are largely trained on and respond best to prompts in standard English, this may affect the quality of LLM outputs for second language English speakers as well as real-world lay users, with potentially disproportionate effects on the former. We explore this effect by modeling a range of error types exhibited by such users, motivated by studies of L2 English, and quantifying their impact on LLM performance. We work with two related tasks: machine translation and machine translation evaluation. We find that LLMs-as-MT are brittle to natural spelling errors but not to phrasal simplifications. However, the quality drop caused by these errors is lower than the variance over the initial prompt choice, suggesting that “perfect English” for a given prompt is less important than choosing a good prompt. Since lay users and L2 speakers may use non-optimal prompts as well as display imperfect language skills, our work calls for increasing the resilience of model performance to both these phenomena, in order to best serve a diverse user base, both from a robustness and fairness perspective.

1 Introduction

Large language models (LLMs) have recently dominated machine translation benchmarks (Kocmi et al., 2024a). These models are known to work best with English prompts, even for tasks in other languages (Dey et al., 2024), and are notoriously sensitive to errors in their prompts (Qiang et al., 2024, *inter alia*).

⁰We release a dataset of translations outputs in three language pairs from six state-of-the-art (closed & open) models and seven error augmenters to enable further research into the effects of prompt quality on LLM performance. We also release our code.

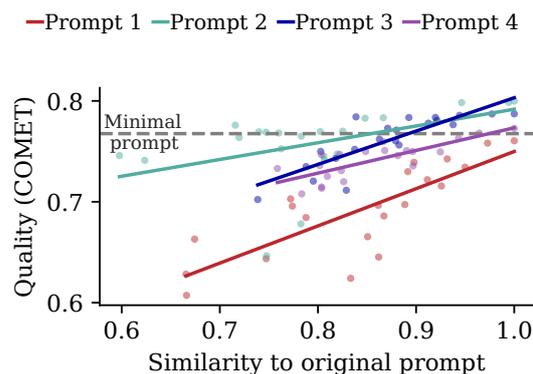


Figure 1: Model performance, as measured by the COMET score (y-axis), across all prompts augmented by orthographic errors and all models. The similarity of each error-augmented prompt to the original is measured by the inner product of their miniLM-L6-v2 sentence embeddings (x-axis). See the wording of the prompts in Table 1 in the appendix.

Research publications and model evaluation setups tend to use well-crafted and tuned prompts in correctly spelled, grammatical, and ‘standard’ language. However, the real-world audience of the models constitutes a diverse user base, including language learners and L2 speakers, as well as lay people. Users are not only unlikely to tune their prompts (Zamfirescu-Pereira et al., 2023), they may also include non-standard language in their prompts, stemming from reasons such as lack of proficiency in English, natural style variation, or convenience.¹ Thus, evaluations in research and industry that do not consider this variation may inflate the true model performance in the wild.

Our work aims to fill this practice-evaluation gap. While past works explore general prompt noise robustness, mostly of synthetic errors (Zhu et al., 2024), we model the effect of specific patterns of er-

¹For the sake of brevity and clarity, we use the word ‘error’ for deviations from standard English without any intended negative connotation.

rors from users on LLM performance. We evaluate LLM robustness to user errors in prompts on two machine translation-related tasks: machine translation (MT) itself, and LLM-based machine translation evaluation (Kocmi and Federmann, 2023b).

Given various error classes of interest, we simulate real-world LLM usage by synthesising these types of errors in the user prompts in a controlled manner with varying intensities. This allows us to quantify and compare the impact of different error types on task performance as well as on the ability of the model to generate outputs in the intended language (on-targetness). We also perform a small qualitative analysis of the resulting performance degradation and error modes. We focus specifically on errors in the *user prompt*, rather than other parts of the input, such as the system prompt. This mimics the typical user/researcher use-case, who would not usually have access to the entire system prompt of state-of-the-art LLMs.

Findings. Through a large-scale quantitative evaluation of the effect of seven error profiles across three language pairs and six state-of-the-art models, followed by qualitative analysis, we find that:

- **Error type matters:** Spelling errors have the greatest impact on LLM performance, while sentence-level simplifications and other phrase-level phenomena typical of L2 speakers or lazy users do not significantly degrade performance.
- **Prompt choice dominates:** The initial prompt choice has greater influence on performance than the vast majority of realistic user errors.
- **Errors reduce instruction-following but not translation quality:** Errors in prompts primarily affect models’ ability to follow instructions (e.g. to avoid redundant text alongside translations or produce off-target translations) rather than their core translation capabilities, with LLMs demonstrating surprising robustness to severe errors.

Similar findings also hold true for translation quality estimation. Lower-quality prompts show a weak detrimental effect on the automatic quality assessment, as meta-evaluated by system-level correlation with human judgments.

2 Related Work

LLMs for Machine Translation. General-purpose decoder-only LLMs have demonstrated state-of-the-art performance in machine translation

with zero- and few-shot prompts (Kocmi et al., 2024a). However, LLMs may refuse to answer or generate redundant text surrounding the translation, which adversely affects automatic evaluation (Briakou et al., 2024). Further, performance has been shown to vary considerably depending on the chosen prompt (Bawden and Yvon, 2023).

While LLMs show strong translation performance with zero-shot prompting (Hendy et al., 2023), this is particularly true for explicitly multilingual models such as EuroLLM (Martins et al., 2024). Both fine-tuning (Xu et al., 2024) and instruction-tuning on the translation task can further boost performance (Alves et al., 2023). For example, TowerLLM (Rei et al., 2024), which is instruction-tuned for multilingual translation and related tasks, achieved leading results on the WMT24 general translation task (Kocmi et al., 2024a).

Robustness of LLMs. Robustness of language models has been explored in the context of adaptation to low-resource settings and user-generated text. Srivastava and Chiang (2025) model multiple types of variation in input segments automatically and focus on variations in English, and Bafna et al. (2024) focus on dialectal variation.

Belinkov and Bisk (2018) diagnosed NMT models to be sensitive to both synthetic and natural errors in the input text. More recently, Peters and Martins (2025) found GPT-3.5 to be surprisingly resilient to synthetic errors. They also looked at the source sequence and not the user prompt, applying synthetic typos in 10-100% of input tokens.

Relatively little work has addressed errors in the prompt specifically. Zhu et al. (2024) generate synthetic adversarial prompts containing possible typos and semantic errors, introduced by several different tools. Additionally, while they cover a number of tasks, these are mostly classification tasks. Gonen et al. (2024) showed that prompt effectiveness is correlated with its perplexity under an LLM, implying that non-standard prompts are likely to do worse. This motivates our work, which quantifies the effect of natural variation and non-standardness as exhibited by real-word users.

In contrast to previous work, this paper examines the impact of *naturalistic* error types with *varying error intensities* on the *prompt* (rather than on the text to be translated).

LLM-as-a-judge for Translation. LLMs have been shown to be effective evaluators of models’

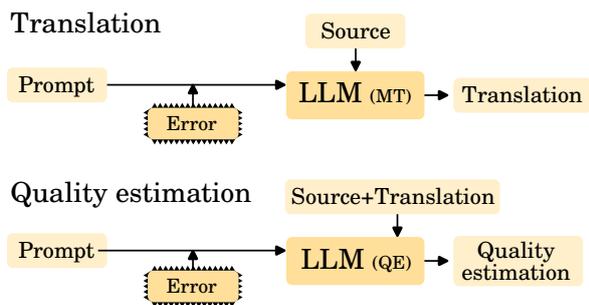


Figure 2: **Top:** Machine translation pipeline. The original prompt is augmented with an error, then filled with source language, target language, and source sentence, before being translated by an LLM. **Bottom:** Quality estimation pipeline. Translations are evaluated using the GPT Estimation Metric Based Assessment (GEMBA).

instruction-following abilities (Zheng et al., 2023), and have since been successfully applied to translation evaluation. Kocmi and Federmann (2023a,b) introduce GEMBA, a prompt-based metric using GPT-4, to produce direct assessments (DA), multidimensional quality metric (MQM) analysis, or error span annotations (ESA; Kocmi et al., 2024b).

In this work, we use the zero-shot GEMBA-DA prompt, because it achieves high system-level correlations with human judgments for both reference-based evaluation and reference-free quality estimation. It is competitive with fine-tuned metrics (Freitag et al., 2023, 2024) such as CometKiwi (Rei et al., 2022b) and XCOMET-QE (Guerreiro et al., 2024). Improvements to LLM-based quality estimation are observed with chain-of-thought prompting for error analysis (Lu et al., 2024) and fine-tuning on human judgments, which boosts poor segment-level correlations of LLMs-as-judges (Fernandes et al., 2023). Huang et al. (2024) investigate the effect of including source and references on evaluation performance. Closest to this part of our work, Qian et al. (2024) examine the effect of prompt formatting on LLM judges for translation.

3 Methods

Our aim is to model the effect of a range of *error types* made by users on LLM performance in a controlled setting. To this end, we design *error augmenters* to imitate each of several different error types of interest, motivated by real-world use. Each error augmenter allows us to model varying intensities of the respective error type. Users may also of course produce several error types simultaneously: therefore, we also study the effect of plausible compositions of particular error types that mimic users

with particular profiles of interest.

3.1 Modeling Error Profiles

Our error types model the following scenarios:

- **Natural orthographic errors** (spelling errors) are modeled by our orthographic error augmenter (A.2). This introduces character-level perturbations with a probability p , modeling documented errors from L1 and L2 speakers (Cook, 1997) (e.g. confusion between particular sets of letters and common vowel sequence transpositions) as well as random typos. We manually choose a range of $p \in [0, 0.4]$ as representing a natural spectrum for the intensity of this type of error, and generate error-augmented prompts for 10 uniformly spaced values of p within this range. We may imagine that the upper ends of the range represent less proficient L2 writers, including L1 and L2 children.
- **Phonetic errors** (A.3) supplement the above error augmenter by introducing spelling errors motivated by phonetic guesses.
- **Phrasal errors** (A.4) mimic phrasal substitutions and simplifications as made by beginner and intermediate speakers of English. We instructed an LLM to generate candidates for two intensity levels, and manually post-edited and modified the resulting prompts for naturalness.
- **Register errors** (A.5) also operate on a phrase level, and simulate the register, or the level of formality or casualness, that the user applies. We generated two intensity levels, similarly as with phrasal errors.
- **Low-proficiency writers, such as L2 learners** of English, presumably commit lexical/phrasal errors as well as spelling errors. We model this as a combined scenario by applying orthographic errors (with the same settings as above) over both levels of the phrasal error augmenter. This results in $10 \cdot 2 = 20$ error-augmented versions per prompt. Different compositions of the two error augmenters can be imagined to represent the diversity of proficiency in English, i.e. users with syntactic proficiency but imperfect spelling or vice versa.
- **Lazy users** use informal registers, and presumably also make spelling errors. As above, we compose the orthographic error augmenter in the selected range over both levels of the register error augmenter to generate prompts with varied error intensities.

Prompt 1: ### Instruction:\n Translate Input from {src_lang} to {tgt_lang} \n ### Input:\n {src_text}\n ### Response:\n
Prompt 2: Translate the following line from\n {src_lang} to {tgt_lang}.\n Be very literal, and only translate the content of the line, do not add any explanations: {src_text}
Prompt 3: Translate this from {src_lang} to {tgt_lang}:\n {src_lang}: {src_text}\n {tgt_lang}:
Prompt 4: Translate the following text from {src_lang} to {tgt_lang}.\n {src_text}
Prompt minimal: {src_lang}: {src_text}\n {tgt_lang}:

Table 1: Base forms for investigated machine translation prompts.

Orthographic (0.1) Trranslate ti from {src_lang} too {tgt_lang}; \n {src_lang}: {src_text} \n {tgt_lang}:
Orthographic (0.4): Tranzlate dhiss from {src_lang} to {tgt_lang}; \n {src_lang}: {src_text} \n {tgt_lang}:
Lexical/Phrasal (1): Make this text in {tgt_lang} from {src_lang}; \n {src_lang}: {src_text} \n {tgt_lang}:
Lexical/Phrasal (2): You translate this text to {tgt_lang} fromm {src_lang}; \n {src_lang}: {src_text} \n {tgt_lang}:
Phonetic: Tranzlate thees from {src_lang} to {tgt_lang}; \n {src_lang}: {src_text} \n {tgt_lang}:
Register (1): {tgt_lang} version of this pls: \n {src_lang}: {src_text} \n {tgt_lang}:
Register (2): change lang {src_lang} -> {tgt_lang}; \n {src_lang}: {src_text} \n {tgt_lang}:

Table 2: Various types and levels (in parentheses) of errors applied to Prompt 3.

- **Uniform errors.** We also apply a control character-level error augmenter that samples perturbations uniformly at random, rather than by user-inspired patterns such as the above (A.1). This helps us contrast model tolerance to the resulting “unnatural” perturbations against more natural error patterns as simulated above. This error augmenter creates random character substitutions with a probability p ; we use 10 choices of p uniformly spaced between $[0, 1]$.

For all scenarios involving the uniform and orthographic error augmenters, we generate 20 erroneous prompts per parametrization. More details on the above error augmenters are given in Appendix A. See examples of the error-augmented prompts in Table 2.

3.2 Error Intensity and Sampling

Recall that we want to obtain error-augmented prompts over a range of error intensities in order to measure the effect of errors on LLM performance. We would also like to compare the effect of different error *types* or profiles. However, the intensity of various error types scales differently. For example, different intensity of spelling errors do not correspond directly to different intensities of phrasal errors. To enable a consistent assessment of the impact of various error types on performance, and to permit cross-error-type comparisons, we measure LLM performance relative to the semantic distance of the error-augmented prompt to the original prompt. We use the following two measures to capture this distance:

- **chrF** between the base prompt and the error-augmented version gives a surface measure of

prompt similarity, where a lower chrF score is associated with a higher error intensity.

- **Inner product of embeddings** of the base prompt and the error-augmented prompts gives a more semantic measure of deviation of the error-augmented prompts from the original. The embeddings are derived from SentenceTransformers’ `all-MiniLM-L6-v2` (Reimers and Gurevych, 2019).

By using various parameterizations per each scenario of interest, we obtain error-augmented prompts that model our error profiles of interest over a range of error intensities as per the above measures. Given an error-augmented prompt, we can now observe the correlation between LLM task performance and the amount of deviation in that prompt from the original. Intuitively, this allows us to answer the question of which error profiles cause the most damage, given equivalent amounts of prompt perturbations.

In practice, we measure this correlation over discrete buckets of increasing error intensity. The prompt used for a particular input is sampled randomly over error-augmented prompts in a given bucket. This provides stable estimates of model performance by reducing vulnerability to outlier prompts.

4 Experimental Settings

Prompts. We choose four zero-shot prompts used by LLM-based systems at the WMT24 General Translation task (Kocmi et al., 2024a) as our base prompts. As a sanity check, we include results for an additional minimalistic baseline prompt that was shown to perform well by Zhang et al. (2023)

QE Prompt 1: Score the following translation from {src_lang} to {tgt_lang} on a continuous scale from 0 to 100, where a score of zero means ‘no meaning preserved’ and score of one hundred means ‘perfect meaning and grammar’.\n {src_lang} source: '{src_text}'\n {tgt_lang} translation: '{tgt_text}'\n Score:

QE Prompt 2: Please analyze the given source and translated sentences and output a translation quality score on a continuous scale ranging from 0 to 100. Translation quality should be evaluated based on both fluency and adequacy. A score close to 0 indicates a low quality translation, while a score close to 100 indicates a high quality translation. Do not provide any explanations or text apart from the score.\n {src_lang} Sentence: {src_text}\n {tgt_lang} Sentence: {tgt_text}\n Score:

Table 3: Base forms of the investigated quality estimation prompts.

and frequently occurs in LLM pre-training data (Briakou et al., 2023). We do not apply any errors to the minimal baseline prompt. See Table 1 for all base prompt forms and Table 2 for examples of perturbed prompts.

Setup. We select two closed-source API models and four open-weight models:

- GPT-4o-mini (OpenAI, 2024)
- Gemini-2.0-flash (Google, 2024)
- Llama-3.1-8B-Instruct (Dubey et al., 2024)
- Qwen2.5-7B-Instruct (Yang et al., 2025)
- EuroLLM-9B-Instruct (Martins et al., 2024)
- TowerInstruct-7B-v0.2 (Rei et al., 2024)

The models are selected to support the languages used for our experiments, and, for the open-weight models, so that we can run them on our infrastructure. See Appendix Table 7 for the list of models considered and their supported languages.

We use language pairs present in WMT 2024 (Kocmi et al., 2024a), specifically: Czech-Ukrainian, German-English, and English-Chinese. Qwen officially supports only English, while the other models either officially support or empirically show good performance on these languages. For each language pair, we randomly choose 500 segments, which is close to the total number in the WMT test set. For evaluation, we use ChrF (Popović, 2015) and COMET₂₂^{DA} (Rei et al., 2022a). We rely on both because COMET is known to struggle on out-of-distribution translations (Zouhar et al., 2024). See Appendix B for the evaluation settings.

Quality Estimation with GEMBA. We use GPT-4o-mini for consistency with our translation experiments. We use two base prompts: GEMBA-DA (quality estimation Prompt 1, Kocmi and Federmann, 2023b) and TMU-HIT’s WMT24 quality estimation prompt (QE Prompt 2, Sato et al., 2024). The full prompts are shown in Table 3. We test on Czech-Ukrainian, German-English, and English-Chinese, same as for translation. We meta-evaluate the quality estimation performance by computing

system and segment level Pearson correlations with human scores on submitted WMT24 systems. We follow a strict setup with no retries; when GEMBA fails to output a correctly formatted score, we set the score for that segment to 0. We limit experiments on the quality estimation task to orthographic errors on the two base prompts, both to maintain a realistic scenario and to limit costs.

5 Results and Discussion

5.1 Prompt Choice is Critical

First, we look at the effect of the prompt choice itself. Figure 1 shows the changes in translation quality depending on the semantic similarity to the original prompt (averaged across all models). Clearly, all four base prompts are affected by applying errors. There are also large differences in performance between the base prompts on their own, showing that prompt choice matters for state-of-the-art performance.

The ‘minimal’ prompt yields reasonable performance but stays behind the best base prompts. Its key benefit compared to the other prompts is the fact that the minimal prompt is essentially impossible to make errors with. Using an otherwise well-performing prompt with many errors leads to much worse performance, and may be worse than using a generally poorly-performing prompt, or the minimal prompt. At the same time, a generally poorly-performing prompt without errors can still perform worse than a generally better-performing prompt with a few errors. These observations reinforce the intuition that both prompt choice and language correctness matter for best results.

Our prompts also respond differently to various error types. Table 4 shows Pearson correlations of translation quality with the prompt similarity to the base prompt, split by error type and base prompt. We discuss the various error types further in Section 5.2. Prompts 3 and 4 appear more resilient to errors overall. For instance, for Prompt 3, there is no correlation between error intensity

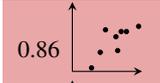
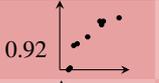
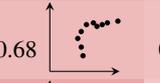
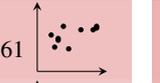
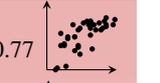
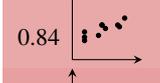
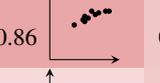
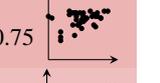
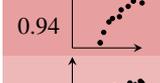
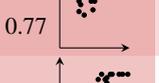
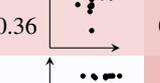
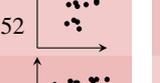
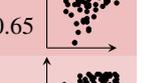
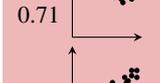
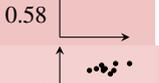
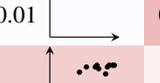
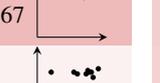
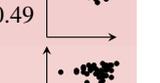
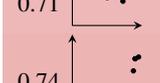
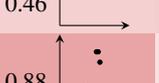
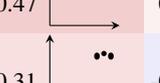
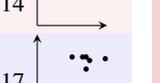
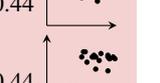
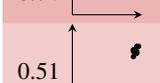
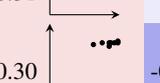
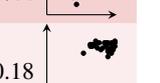
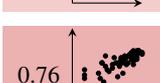
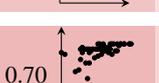
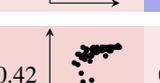
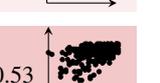
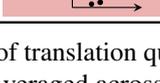
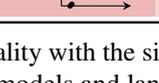
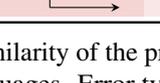
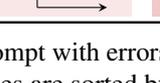
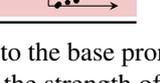
Error augmenter	Prompt 1	Prompt 2	Prompt 3	Prompt 4	All prompts
Uniform error	0.86 	0.92 	0.68 	0.61 	0.77 
Phonetic	0.84 	0.72 	0.86 	0.57 	0.75 
Lazy user	0.94 	0.77 	0.36 	0.52 	0.65 
Orthographic	0.71 	0.58 	-0.01 	0.67 	0.49 
L2	0.71 	0.46 	0.47 	0.14 	0.44 
Register	0.74 	0.88 	0.31 	-0.17 	0.44 
Phrasal	0.51 	0.59 	0.30 	-0.67 	0.18 
All error augmenters	0.76 	0.70 	0.42 	0.24 	0.53 

Table 4: Pearson correlation of translation quality with the similarity of the prompt with errors to the base prompt, per error type and prompt. Averaged across models and languages. Error types are sorted by the strength of the correlation averaged over all prompts. See Appendix Figure 5 for visualization in a single plot.

and translation quality when using realistic orthographic errors. This is likely due to the prompts’ short length. Because we preserve the critical variables of source and target language, as well as the input segment, introducing errors to the rest of these short prompts causes less confusion than in the more complex Prompts 1 and 2. Additionally, Prompt 4 even seems to benefit from phrasal errors, perhaps because its base form underperforms.

Further, the best-performing prompt for one model can be the worst-performing prompt for another, which is in line with prior findings (Voronov et al., 2024). We observe that GPT-4o tends to benefit from the “###” structure of Prompt 1 while the other models tend to copy parts of this prompt regardless of error type or intensity, leading to lower scores. Similarly, Prompt 4 works best of all prompts for EuroLLM and Qwen 2.5, but makes Gemini produce more off-target translations.

The results suggest that **users can benefit from choosing the ‘right’ prompt** for the model they are using, with shorter prompts being somewhat preferable. This seems to be **at least as important as avoiding certain error types** (see also Section 5.2).

5.2 Comparing Error Types

On average across all prompts, the baseline uniform error augmenter affects translation quality the most. This suggests that models are naturally more robust to realistic errors, since they were exposed to realistic errors during training, but not to the unrealistic uniform errors.

The simple spelling errors (common orthographic errors and phonetically inspired misspellings) capture a type of error commonly made by L2 speakers, and show a strong effect on the LLMs’ performance.

Phrase-level simplifications show the least overall effect. For Prompt 4, we even see a negative correlation, meaning that some substitutions performed better than the original prompt. Thus, while lack of fluency on the part of L2 users may introduce unnaturalness or awkwardness as perceived by native speakers, the models may actually respond well to this simplification. A similar weak effect also holds for the register simplifications.

The ‘L2’ and ‘Lazy User’ scenarios model natural compositions of error types that L2 users or lazy people respectively might display. Specifically, these scenarios combine the orthographic errors with phrasal and register errors, respectively. The ‘L2’ scenario shows similar rates of impact

Level	Prompt 1	Prompt 2	All prompts
System-	0.15 	0.71 	0.43 
Segment-	-0.38 	0.57 	0.09 

Table 5: Pearson correlation between system and segment-level evaluation performance and prompt similarity, for the orthographic error and quality estimation prompts (averaged across languages).

as the orthographic errors alone, suggesting that orthographic errors might have a weaker effect on simplified language. The ‘Lazy User’ scenario, however, shows more impact, i.e. rate of degradation per unit error, than its constituent error types, indicating that the register changes combined with orthographic errors has a stronger effect on the impact of each on model performance.

5.3 Frequency of Off-Target Translations

A common failure mode for LLMs-as-MT is responding in a non-target language. To measure this failure mode, we used the `langdetect` Python library. Figure 3 shows the proportion of outputs in the target language by model and language pair. **For all models and language pairs, high error intensities decrease the proportion of on-target outputs.** Of the target languages, German has the highest proportion of on-target outputs. For Czech-Ukrainian, TowerInstruct tends to output other languages even with the original prompt, because the two languages are not well-supported by the model.

Note also that COMET does not penalize wrong language output and may still score off-target outputs highly, for instance, if the model outputs a Russian translation when a Ukrainian translation was requested.

5.4 Transferability to Quality Estimation

Figure 4 shows that both quality estimation prompts are (weakly) affected by applying *realistic* orthographic errors. The two prompts differ both in their baseline performance and in the strength of the effect. This reinforces the idea that prompt choice matters as much for quality estimation as for translation.

Table 5 shows Pearson correlations of system-level correlations with with a prompt’s similarity to its base prompt. A higher correlation implies that decreasing prompt similarity also decreases qual-

ity estimation correlation with human judgments. We observe a similar effect of orthographic errors on the system-level correlation of GEMBA across languages, with an overall correlation of 0.43, compared to 0.49 for translation. This suggests the effect of orthographic errors is transferable and largely consistent across tasks. See Appendix Figure 9 for per-language results.

However, we observe a negative correlation for Quality Estimation Prompt 1 at the segment level. The original, error-free prompt already achieves a poor segment-level correlation of 0.16. This may be an artifact of our strict setting which prohibits retries and artificially sets the resulting score to 0 to elucidate the error impact. Further, the shorter prompt presents fewer possibilities to corrupt it. This may explain the reduced variance in outputs and therefore weaker correlations, though additional testing is required to elucidate this effect.

5.5 Qualitative Analysis

We performed qualitative analysis on the machine translation outputs by manually inspecting a sample of the lowest-scoring translations in each setting. We also sampled ten source segments per language pair with their translations from every setting, to understand how various error types and levels change the translation of a given sentence.

Models add supplementary information. In addition to providing the translation, Gemini frequently offers useful background information, such as multiple versions of the translation or the pronunciation of the Chinese translation explained in Latin script. This explains why Gemini produces longer outputs than any other model, as shown in Figure 8. The extra information appears in the outputs regardless of the error type or intensity. On the other hand, the model explains its choice of words more frequently with increasing error intensity. This behavior may be beneficial for a user, but is difficult to parse in an automated setting.

Lower scores are often due to worse instruction following. The translations by a single model tend to remain relatively stable across various error types and levels. The main explanation for the differing scores is the presence of redundant text alongside the translation, such as adding the name of the target language, saying “here is your translation,” repeating the source sentence, or paraphrasing the prompt. The base prompts in our experiments either explicitly state to only output the

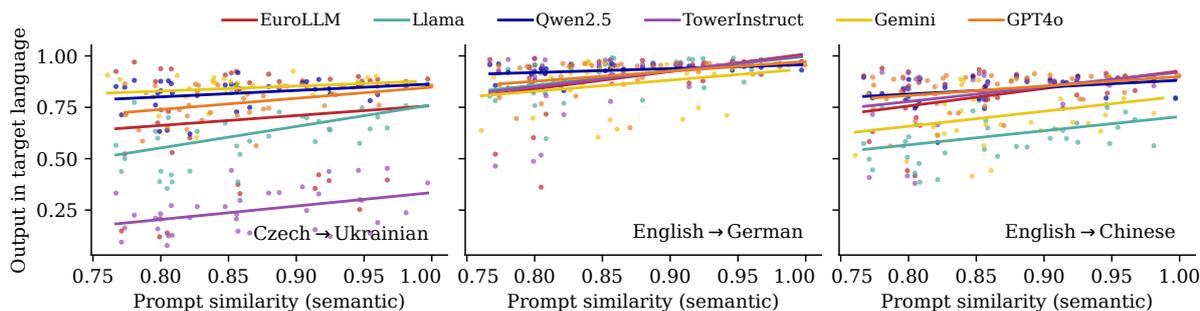


Figure 3: Percentage of outputs in the target language, by language pair and model. Note that TowerInstruct does not officially support Ukrainian or Czech.

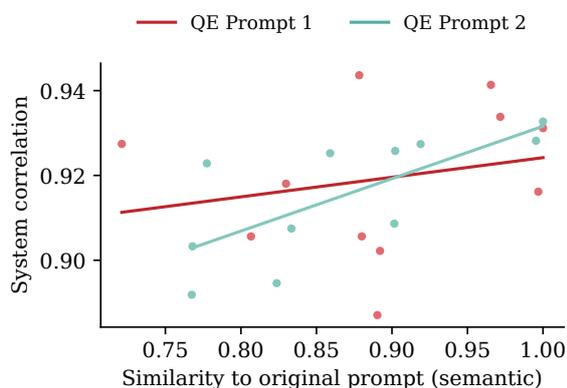


Figure 4: Changing QE model performance, in terms of system-level correlation (y-axis), across quality estimation prompts augmented with orthographic errors, against semantic similarity of the error-augmented prompt to its original (x-axis). The results suggest only a weak trend.

translation and nothing else, or strongly imply it by their structure. Therefore, when models output redundant text rather than only providing the translation as instructed, we consider it worse instruction following.

GPT-4o is a notable exception: It generates more diverse translations and less redundant text, unless subjected to a high level of uniform errors that render the text essentially unreadable to humans ($p > 0.5$). Up to that threshold, a lower metric score for GPT-4o is more likely to correspond to genuinely lower translation quality.

This finding is supported quantitatively by Appendix Figure 8. It shows that adding errors increases the average length of the LLM output and that there are frequently significant differences between target and reference length. GPT-4o produces the shortest texts while Gemini produces the longest. The average length of Qwen2.5 outputs is

the least affected as more errors are added.

LLMs can still translate with illegible prompts.

Realistic noising scenarios produce prompts that are mostly legible to humans. We also stress-tested the models by applying uniform errors with $p > 0.46$, exceeding the natural error range. This transformation makes the prompts largely illegible to humans (compare Table 2), for example: “*Reaajaky fgo trormm {src_lang} ttk {tgt_lang}:: \n {src_lang} {src_text} \n {tgt_lang}::*”

LLMs sometimes produce an error message or request clarification without providing a translation in response to an illegible prompt. However, they frequently produce valid translations even when given prompts with a high $p \geq 0.7$, which are nonsensical to the human eye. These translations often involve copying the prompt verbatim, attempting to translate or fix it, or treating the text as a cipher to decode. In rare cases, they ignore the errors altogether and only provide the expected translation. We show examples of responses to illegible prompts in Appendix Table 6.

The only parts of the prompt that remain legible in this stress-testing scenario are the unchanged source and target languages, as well as the source sentence to be translated. This suggests that if an LLM is capable of performing a task, it recognizes the task based on minimal information and performs it even with an objectively bad prompt. The models likely succeed because their training data contains abundant examples with just the language names and source text – the ‘Default’ prompt format described by Briakou et al. (2023). This finding may be helpful for future research, as it implies that if an LLM does not generalize to a task as demonstrated by a handful of prompts, further prompt engineering efforts are unlikely to change that outcome.

6 Conclusion

We investigated the impact of imperfect prompt construction on LLM performance. We model a range of user-inspired errors and apply them to prompts in a controlled manner, observing the resulting degradation in task performance over six models, two tasks, and four language pairs. We find that spelling errors have the most severe effect on performance, while phrase-level disfluencies and simplifications have lower impact and may even help in some cases. We also explore natural compositions of error types, finding that these compound the effects of their constituent error types. A qualitative analysis of the resulting outputs reveals that ‘imperfect’ English in prompts often does not lead to lower translation quality, but rather worse instruction following, such as failing to perform the task or providing extraneous output. This phenomenon is not always negative – some models include useful information for the user – however, it is penalized by automatic metrics.

Crucially, the effect of the initial prompt selection is greater than that of the majority of realistic user errors, emphasizing the importance of prompt choice. Yet, while practitioners may therefore benefit from optimizing prompt selection over a set of diverse and error-free prompts, lay users are unlikely to do so, and may further exhibit errors in their prompts due to imperfect language skills or other reasons. This work highlights the gap between LLM performance as evaluated in pristine conditions as opposed to real-world conditions, and calls for improving LLM resilience to prompt choice as well as user errors in prompts.

Limitations

This study only looks at errors in English prompts for MT-related tasks; error-classes as well as the nature of impact of typical user errors in different language prompts may naturally differ.

We used automatically generated errors rather than using error data from real learners. One important reason for this is the difficulty of sourcing real examples. While using generated errors may mean that some of the examples are less realistic, it allows for a broader statistical analysis and provides us with better control over our experimental variables.

Ethics Statement

We do not anticipate any negative ethical implications arising from this study. We took care to ensure realistic representations of errors without casting users in a negative light.

The total inference cost for the two proprietary models (GPT-4o-mini and Gemini-2.0-flash) is less than USD 100. While we did run the other models locally, the overall cost for all the models likely does not exceed USD 200.

The licenses for the open-weights models are: Llama 3.1 Community License for Llama 3.1; Apache 2.0 for EuroLLM and the Qwen model we used; and CC-BY-NC-4.0 for the Tower model we used (with its base model Llama 2 being licensed under the Llama 2 Community License). These licenses all permit our use of the model weights.

We used AI-assisted coding (i.e. Copilot) with the bulk of the code being human-written. For writing, AI was used to check grammar mistakes.

Acknowledgments

We are immensely grateful to EAMT for the grant which allowed us to meet in Prague, where the majority of discussions and experiments took place. VZ gratefully acknowledges the support of the Google PhD Fellowship. PS was supported by Charles University project GAUK 252986 and SVV project 260 821. GV was supported by Czech Science Foundation project 25-16242S. Last, but not least, we thank Ondřej Bojar, Ondřej Dušek, and Saad Mahamood for their feedback on this manuscript.

References

- Duarte Alves, Nuno Guerreiro, João Alves, José Pomal, Ricardo Rei, José de Souza, Pierre Colombo, and Andre Martins. 2023. [Steering large language models for machine translation with finetuning and in-context learning](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 11127–11148, Singapore. Association for Computational Linguistics.
- Niyati Bafna, Kenton Murray, and David Yarowsky. 2024. [Evaluating large language models along dimensions of language variation: A systematic investigation of cross-lingual generalization](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 18742–18762. Association for Computational Linguistics.
- Rachel Bawden and François Yvon. 2023. [Investigating the translation performance of a large multilingual language model: the case of BLOOM](#). In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, 157–170. European Association for Machine Translation.
- Yonatan Belinkov and Yonatan Bisk. 2018. [Synthetic and natural noise both break neural machine translation](#). In *International Conference on Learning Representations*.
- Eleftheria Briakou, Colin Cherry, and George Foster. 2023. [Searching for needles in a haystack: On the role of incidental bilingualism in PaLM’s translation capability](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 9432–9452. Association for Computational Linguistics.
- Eleftheria Briakou, Zhongtao Liu, Colin Cherry, and Markus Freitag. 2024. [On the implications of verbose LLM outputs: A case study in translation evaluation](#).
- Vivian J Cook. 1997. [L2 users and English spelling](#). *Journal of Multilingual and Multicultural Development*, 18(6):474–488.
- Krishno Dey, Prerona Tarannum, Md. Arif Hasan, Imran Razzak, and Usman Naseem. 2024. [Better to ask in english: Evaluation of large language models on english, low-resource and cross-lingual settings](#).
- Abhimanyu Dubey et al. 2024. [The Llama 3 herd of models](#).
- Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André Martins, Graham Neubig, Ankush Garg, Jonathan Clark, Markus Freitag, and Orhan Firat. 2023. [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*, 1066–1083, Singapore. Association for Computational Linguistics.
- Markus Freitag, Nitika Mathur, Daniel Deutsch, Chi-kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Frederic Blain, Tom Kocmi, Jiayi Wang, David Ifeoluwa Adelani, Marianna Buchicchio, Chrysoula Zerva, and Alon Lavie. 2024. [Are LLMs breaking MT metrics? results of the WMT24 metrics shared task](#). In *Proceedings of the Ninth Conference on Machine Translation*, 47–81. Association for Computational Linguistics.
- Markus Freitag, Nitika Mathur, Chi-kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Tom Kocmi, Frederic Blain, Daniel Deutsch, Craig Stewart, Chrysoula Zerva, Sheila Castilho, Alon Lavie, and George Foster. 2023. [Results of WMT23 metrics shared task: Metrics might be guilty but references are not innocent](#). In *Proceedings of the Eighth Conference on Machine Translation*, 578–628. Association for Computational Linguistics.
- Hila Gonen, Srini Iyer, Terra Blevins, Noah A. Smith, and Luke Zettlemoyer. 2024. [Demystifying prompts in language models via perplexity estimation](#).
- Google. 2024. [Introducing gemini 2.0: Our new ai model for the agentic era](#).
- Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André F. T. Martins. 2024. [xcomet: Transparent machine translation evaluation through fine-grained error detection](#). *Transactions of the Association for Computational Linguistics*, 12:979–995.
- Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. [How good are GPT models at machine translation? A comprehensive evaluation](#).
- Xu Huang, Zhirui Zhang, Xiang Geng, Yichao Du, Jijun Chen, and Shujian Huang. 2024. [Lost in the source language: How large language models evaluate the quality of machine translation](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, 3546–3562, Bangkok, Thailand. Association for Computational Linguistics.
- Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Benjamin Marie, Christof Monz, Kenton Murray, Masaaki Nagata, Martin Popel, Maja Popović, Mariya Shmatova, Steinhór Steingrímsson, and Vilém Zouhar. 2024a. [Findings of the WMT24 general machine translation shared task: The LLM era is here but MT is not solved yet](#). In *Proceedings of the Ninth Conference on Machine Translation*, 1–46. Association for Computational Linguistics.
- Tom Kocmi and Christian Federmann. 2023a. [GEMBA-MQM: Detecting translation quality error spans with GPT-4](#). In *Proceedings of the Eighth Conference*

- on Machine Translation, 768–775. Association for Computational Linguistics.
- Tom Kocmi and Christian Federmann. 2023b. [Large language models are state-of-the-art evaluators of translation quality](#). In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, 193–203, Tampere, Finland. European Association for Machine Translation.
- Tom Kocmi, Vilém Zouhar, Eleftherios Avramidis, Roman Grundkiewicz, Marzena Karpinska, Maja Popović, Mrinmaya Sachan, and Mariya Shmatova. 2024b. [Error span annotation: A balanced approach for human evaluation of machine translation](#). In *Proceedings of the Ninth Conference on Machine Translation*, 1440–1453. Association for Computational Linguistics.
- Qingyu Lu, Baopu Qiu, Liang Ding, Kanjian Zhang, Tom Kocmi, and Dacheng Tao. 2024. [Error analysis prompting enables human-like translation evaluation in large language models](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, 8801–8816, Bangkok, Thailand. Association for Computational Linguistics.
- Pedro Henrique Martins, Patrick Fernandes, João Alves, Nuno M. Guerreiro, Ricardo Rei, Duarte M. Alves, José Pombal, Amin Farajian, Manuel Faysse, Mateusz Klimaszewski, Pierre Colombo, Barry Haddow, José G. C. de Souza, Alexandra Birch, and André F. T. Martins. 2024. [Eurollm: Multilingual language models for europe](#).
- OpenAI. 2024. [Gpt-4o mini: Advancing cost-efficient intelligence](#).
- Ben Peters and Andre Martins. 2025. [Did translation models get more robust without anyone Even noticing?](#) In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2445–2458, Vienna, Austria. Association for Computational Linguistics.
- Maja Popović. 2015. [chrF: character n-gram F-score for automatic MT evaluation](#). In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, 392–395. Association for Computational Linguistics.
- Matt Post. 2018. [A call for clarity in reporting BLEU scores](#). In *Proceedings of the Third Conference on Machine Translation: Research Papers*, 186–191. Association for Computational Linguistics.
- Shenbin Qian, Archchana Sindhujan, Minnie Kabra, Diptesh Kanojia, Constantin Orasan, Tharindu Ranasinghe, and Fred Blain. 2024. [What do large language models need for machine translation evaluation?](#) In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 3660–3674. Association for Computational Linguistics.
- Yao Qiang, Subhrangshu Nandi, Ninareh Mehrabi, Greg Ver Steeg, Anoop Kumar, Anna Rumshisky, and Aram Galstyan. 2024. [Prompt perturbation consistency learning for robust language models](#). In *Findings of the Association for Computational Linguistics: EACL 2024*, 1357–1370. Association for Computational Linguistics.
- Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022a. [COMET-22: Unbabel-IST 2022 submission for the metrics shared task](#). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, 578–585. Association for Computational Linguistics.
- Ricardo Rei, Jose Pombal, Nuno M. Guerreiro, Jo textasciitilde ao Alves, Pedro Henrique Martins, Patrick Fernandes, Helena Wu, Tania Vaz, Duarte Alves, Amin Farajian, Sweta Agrawal, Antonio Farinhas, José G. C. De Souza, and André Martins. 2024. [Tower v2: Unbabel-IST 2024 submission for the general MT shared task](#). In *Proceedings of the Ninth Conference on Machine Translation*, 185–204. Association for Computational Linguistics.
- Ricardo Rei, Marcos Treviso, Nuno M. Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José G. C. de Souza, Taisiya Glushkova, Duarte Alves, Luisa Coheur, Alon Lavie, and André F. T. Martins. 2022b. [Cometkiwi: Ist-unbabel 2022 submission for the quality estimation shared task](#). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, 634–645, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-bert: Sentence embeddings using siamese bert-networks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.
- Ayako Sato, Kyotaro Nakajima, Hwihan Kim, Zhousi Chen, and Mamoru Komachi. 2024. [TMU-HIT’s submission for the WMT24 quality estimation shared task: Is GPT-4 a good evaluator for machine translation?](#) In *Proceedings of the Ninth Conference on Machine Translation*, 529–534. Association for Computational Linguistics.
- Aarohi Srivastava and David Chiang. 2025. [We’re calling an intervention: Exploring fundamental hurdles in adapting language models to nonstandard text](#). In *Proceedings of the Tenth Workshop on Noisy and User-generated Text*, 45–56, Albuquerque, New Mexico, USA. Association for Computational Linguistics.
- Anton Voronov, Lena Wolf, and Max Ryabinin. 2024. [Mind your format: Towards consistent evaluation of in-context learning improvements](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, 6287–6310. Association for Computational Linguistics.
- Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2024. [A paradigm shift in machine](#)

translation: Boosting translation performance of large language models. In *The Twelfth International Conference on Learning Representations*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiayi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. [Qwen2.5 technical report](#).

J.D. Zamfirescu-Pereira, Richmond Y. Wong, Bjoern Hartmann, and Qian Yang. 2023. [Why johnny can't prompt: How non-ai experts try \(and fail\) to design llm prompts](#). In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, New York, NY, USA. Association for Computing Machinery.

Biao Zhang, Barry Haddow, and Alexandra Birch. 2023. [Prompting large language model for machine translation: A case study](#). In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhonghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. [Judging llm-as-a-judge with mt-bench and chatbot arena](#). *Advances in Neural Information Processing Systems*, 36:46595–46623.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Yue Zhang, Neil Gong, and Xing Xie. 2024. [Promptrobust: Towards evaluating the robustness of large language models on adversarial prompts](#). In *Proceedings of the 1st ACM Workshop on Large AI Systems and Models with Privacy and Safety Analysis*, LAMPS '24, 57–68, New York, NY, USA. Association for Computing Machinery.

Vilém Zouhar, Pinzhen Chen, Tsz Kin Lam, Nikita Moghe, and Barry Haddow. 2024. [Pitfalls and outlooks in using COMET](#). In *Proceedings of the Ninth Conference on Machine Translation*, 1272–1288. Association for Computational Linguistics.

A Descriptions of Error Augmenters

A.1 Uniform errors

This error augmenter, parameterized by probability p , introduces random perturbations into the prompt, modeling natural typos. The perturbations include random character transposition, omission, doubling, and substitution for neighbouring letters on the keyboard. The character-wise frequency of error is controlled by p , and the type of error is sampled uniformly from the error types.

A.2 Orthographic error

This error augmenter models spelling errors, both due to imperfect proficiency in written English as well as random typos. Cook (1997) provide a classification of the types of spelling errors made by both L1 and L2 speakers, and report the relative frequency of these errors, finding higher error rates for L2 speakers, but similar distributions over error categories. Guided by this work, we define the follow classes of orthographic errors:

- **Natural typos:** We re-use our uniform error augmenter as described above. This corresponds to the category “other” as defined by Cook (1997).
- **Omission:** Omitting one of a non-word-initial consonant pair (e.g. $ck \rightarrow k$), dropping r before a consonant, dropping e if it is word-final, or before ly .
- **Insertion:** Doubling non-word-initial consonant.
- **Substitution:** Confusing specific sets of consonants (such as s, c, z), confusing vowels with each other. For the latter, we generate errors consistently with the finding that confusions between a, e, i constitute 60% of vowel substitutions.
- **Transposition:** Transposing consecutive vowels ($ie \rightarrow ei$), transposing certain bigrams (er, ng). Similarly to the uniform errors, the orthographic error augmenter is controlled by a parameter p , which corresponds to the probability of error on a given character. Varying p allows us therefore to generate prompts over different error intensities. Given a character to be perturbed, we sample a type of error from the above list, as per the natural distribution over these categories of error described in Cook (1997). Given a type of error (e.g. substitution), we uniformly sample a subtype of error from all subtypes applicable to the character and its context. For example, the “a-e”-confusion subtype is only relevant for “a’s”. Note that a character may have no relevant subtypes under a given type: In this case, we simply skip the character.

A.3 Phonetic LLM-generated errors

We also investigate the impact on LLM performance of errors made by non-native speakers writing English sentences based on phonetic transcriptions in their first languages. We prompt an LLM to mimic these errors in various languages (Arabic, Chinese, German, Polish and Spanish) spoken by beginner English learners. According to our tests, LLMs can simulate typical phonetic errors for a particular language, despite not being fully fluent in it. Example for a Polish person: *Translate the following line from English to Chinese.* \rightarrow *Translejt de following lajn from English tu Chinese.*

A.4 Phrasal simplification

We would like to study the effect of alternate lexical/phrasal simplification, as possibly committed by L2 speakers. Note that prompts generally use largely restricted vocabulary, and potential phrasal errors are therefore limited. We consider two levels of L2 proficiency: Beginner and intermediate, and prompt an LLM to mimic such errors made by L2 speakers of each level, generating $k = 10$ error-augmented candidates per prompt and level. We manually examine the generations and discard implausible options. We find that LLM-generated errors cover a reasonable range of plausible errors of this type.

A.5 Register changes

We are also interested in the effect of informal registers of users, who may query LLMs similarly to querying search engines, with non-standard casing, dropping of articles and function words, and reframing for conciseness. For example, *Translate from de to en* \rightarrow *translate de - en*. This type of errors also offers a limited number of possible transformations of a base prompt. Similarly to above, we prompt an LLM to generate $k = 10$ informal versions of each base prompt with the above changes, for two levels (medium and high) of informality, and manually discard unlikely candidates.

B Implementation Details

For evaluation we use the following settings:

- **ChrF:**
nrefs:1lcase:mixedleff:yeslnc:6lnw:0lspace:nolversion:2.3.1 (Post, 2018, sacrebleu)
- **COMET:** Python3.11.5lComet2.2.5lfp32lUnbabel/wmt22-comet-dalr1 (Rei et al., 2022a, sacrecomet Zouhar et al., 2024)

C Ablations

C.1 Impact of User Errors Per Language

We further examine how different language pairs and models respond to user errors. Appendix Figure 6 shows orthographic error sensitivity per language pair averaged across all models. We show six subplots, one for each combination of quality metric (ChrF, COMET) and measure of error level (noising probability p , semantic prompt similarity, and surface prompt similarity).

In general, all language pairs are affected to a similar degree. Czech-Ukrainian appears slightly more sensitive than the other two, possibly due to less robust support of the models for this language pair, while translation into Chinese scores lower on ChrF. Similarly, Appendix Figure 7 shows the sensitivity per language pair *and model*. All models are affected, for all languages, to a similar degree. Note that TowerInstruct does not support Czech or Ukrainian, and Llama-3.1 officially does not support Czech, Ukrainian, or Chinese.

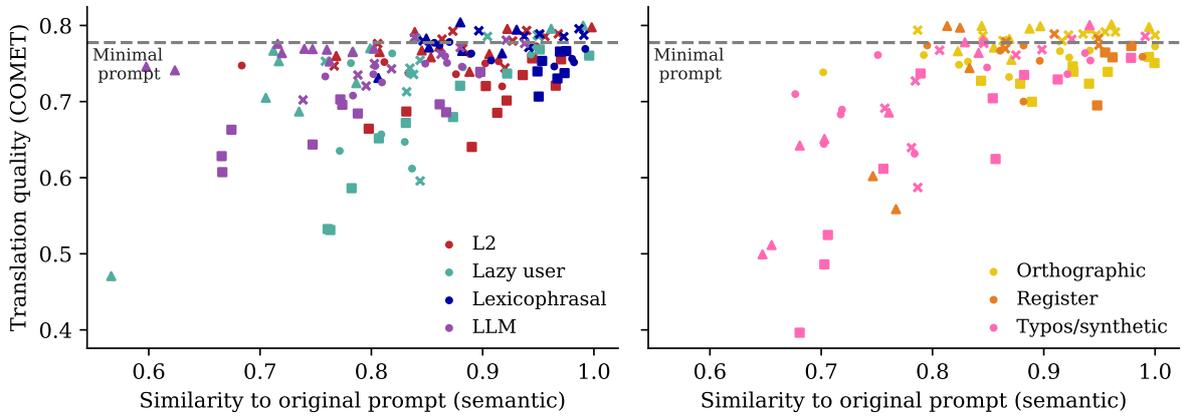


Figure 5: Average performance (across models and languages) with respect to individual prompts and error types. Each shape is one of four prompts. Visualizes Table 4 in a single plot.

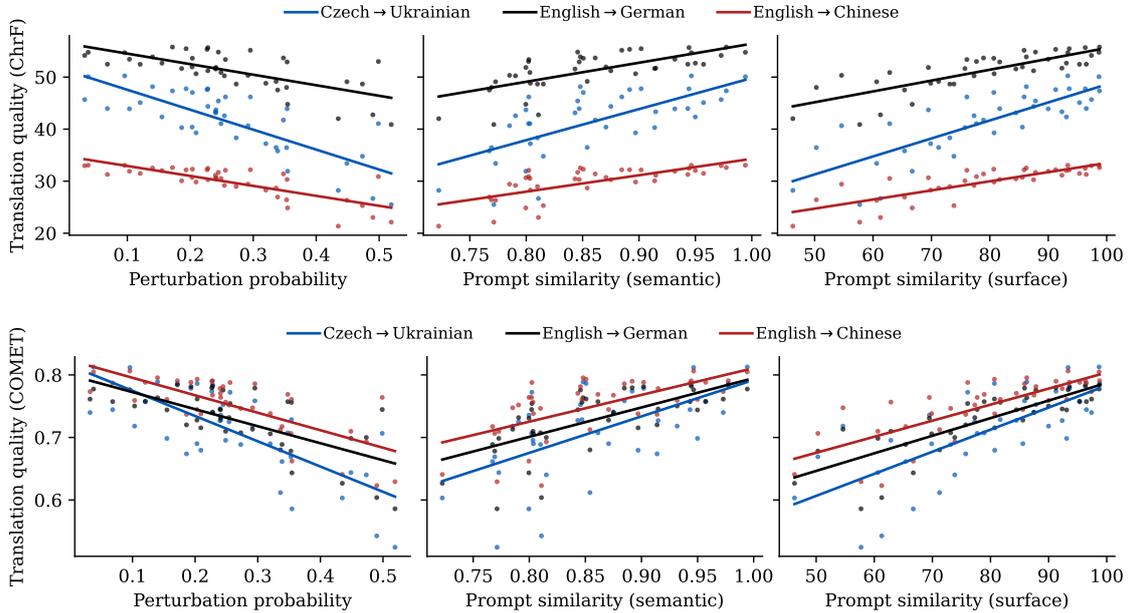


Figure 6: Sensitivity to error augmentation by language pair. Translation quality measured by ChrF (top) or COMET (bottom), given a certain amount of error augmentation (x-axes). The error augmentation probability refers to the probability p of applying orthographic errors. Prompt similarity (semantic) refers to the inner product of sentence embeddings. Prompt similarity (surface) refers to the chrF score of the error-augmented prompt against the base prompt.

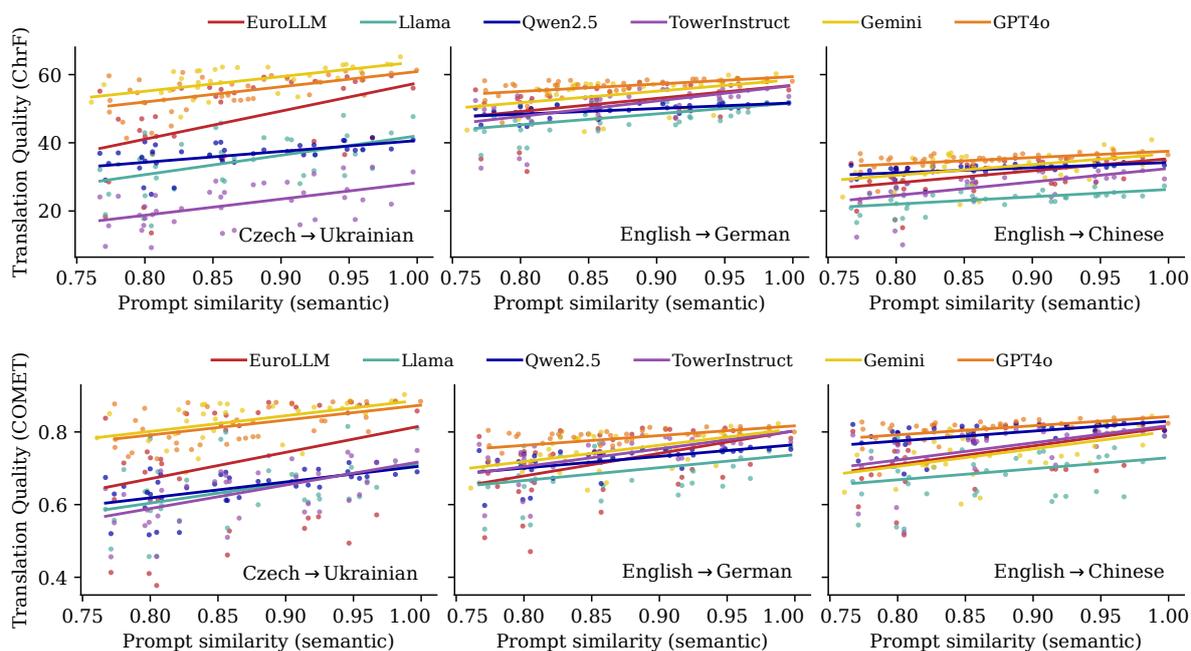


Figure 7: Sensitivity of individual models to prompt noising, for each language pair and by model. x-axis: Prompt similarity to base prompt (semantic). y-axes: Translation quality measured by ChrF (top) and COMET (bottom).

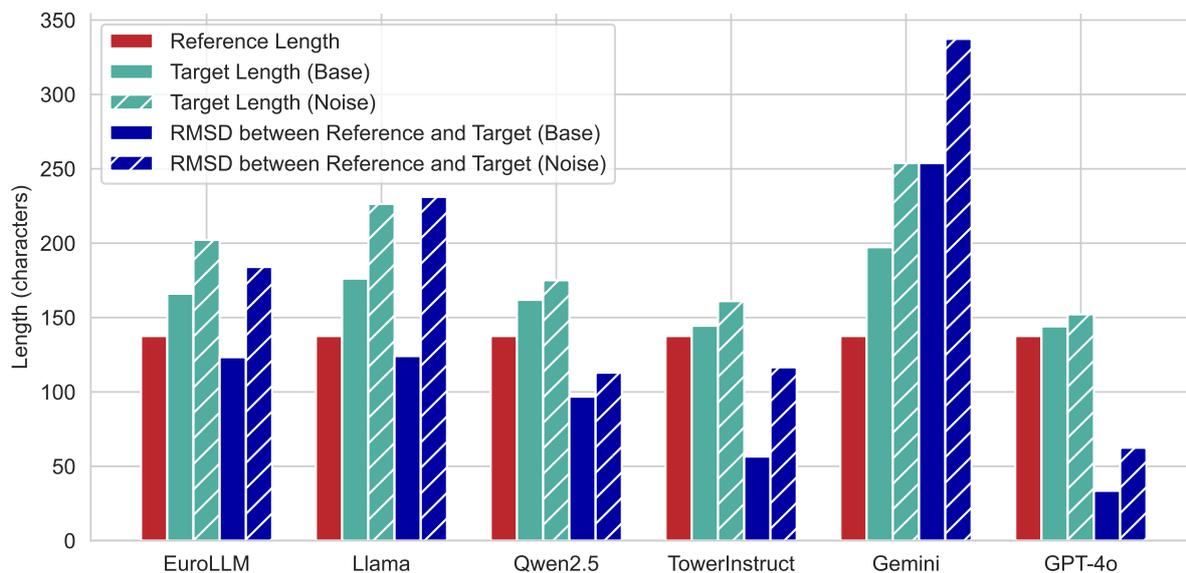


Figure 8: Average length of generated outputs compared to the reference length. We compare the output lengths given the base prompts to the aggregate of all error augmenter variants, and observe that error-augmented prompts lead to consistently longer texts.

EuroLLM, Orthographic (0.7): Rrwnslswfee tn gplowwgn linnee ffro \n English t German.\n Bee yg oiteerarl, ann olyn rnlqata thet cnotetn kff feg linns, ri ohtt adad anay exlqntaiiox:: *Ich finde es sehr lebensbejahend; aus so einfachen Zutaten etwas so Reichhaltiges und Komplexes zu machen. Wie bei der Geburt eines Kindes.*

Qwen2.5, Orthographic (0.8): Hier ist die Übersetzung von Englisch ins Deutsche: *Offensichtlich gibt es nur zwei Ebenen, die im Original-Super-Mario-Maker noch nicht abgeschlossen wurden.*

Gemini, Orthographic (0.8): Gerne, hier ist die Übersetzung des Satzes "I better get some motorbikes fixed, or else!" ins Deutsche:\n \n Ich sollte besser ein paar Motorräder reparieren lassen, oder aber!

Llama, Orthographic (0.8): The encoded message is a cipher. It appears to be a simple substitution cipher, where each letter has been shifted by a certain number of positions in the alphabet. After analyzing the text, I believe it is a Caesar cipher with a shift of 13 positions. Here's the decoded message: Original: Gafnsnlatae tet fflwlif llmie gon Decoded: "Fantastical life line gone" [...]

GPT-4o, Orthographic (0.9): It seems like there are some typos and formatting issues in your message. However, I can help you translate the provided English text into German. Here's the translation: ****English:**** "I find it very life affirming; transforming such simple ingredients into something so rich and complex. Like witnessing a birth." ****German:**** *"Ich finde es sehr lebensbejahend; so einfache Zutaten in etwas so Reichhaltiges und Komplexes zu verwandeln. Wie das Zeugnis einer Geburt."* If you need further assistance or corrections, feel free to ask!

TowerLLM, Orthographic (0.8): Gafnsnlatae tet fflwlif llmie gon \n Deutsch gehen Englisch.\n Sehr traurig, wenn man bedenkt, dass die Mieten in den letzten Jahren um durchschnittlich 6 Prozent pro Jahr gestiegen sind, aber diese Vorschläge würden die Mieten um bis zu 15 Prozent erhöhen, was ironischerweise höher ist als die historischen jährlichen Preissteigerungen. [...]

Table 6: Examples of LLM outputs when presented with perturbed prompts.

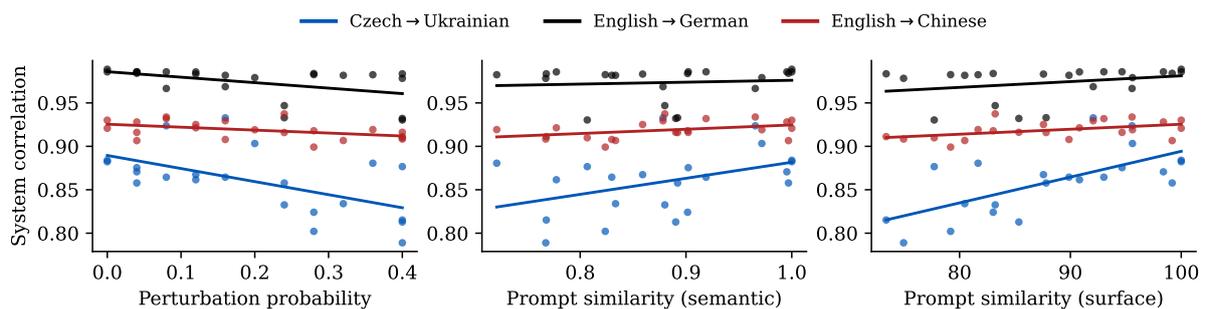


Figure 9: Sensitivity of QE outputs to perturbations by language pair. System-level correlation is measured against human evaluations on the test set, given a certain perturbation amount. Semantic prompt similarity measures the inner product of erroneous and base sentence embeddings, while surface similarity measures ChrF between erroneous and base prompts. The results show that effects are seen across language pairs, though the magnitude of the effect varies.

	English	Czech	Ukrainian	German	Chinese	Open
Claude-3.5 Sonnet ¹	✓	?	?	?	?	✗
CommandR+	✓	✓ ²	✓ ²	✓	✓	✗
GPT-4o	✓	✓	✓	✓	✓	✗
Gemini-2.0 Flash	✓	✓	✓	✓	✓	✗
Phi-3	✓	✓	✓	✓	✓	✓
Phi-4 14B	✓	✓	✓	✓	✓	✓
EuroLLM	✓	✓	✓	✓	✓	✓
Llama3	✓	✗	✗	✓	✗	✓
TowerInstruct	✓	✗ ³	✗ ³	✓	✓	✓
Aya23	✓	✓	✓	✓	✓	✓
DeepSeek-V3 ¹	✓	?	?	?	✓	✓
Qwen-2.5	✓	✗	✗	✗	✗	✓
Mistral	✓	✗	✗	✓	✓	✓

Table 7: List of models taken into consideration. The list of supported languages for the open-weight models is taken from their Hugging Face model cards.

¹: The model is multilingual but the list of supported languages is not available;

²: Languages included in the pre-training but not post-training ([Cohere documentation](#));

³: Tower70B took part in WMT2024 on the Czech→Ukrainian language pair ([Kocmi et al., 2024a](#)), but the model card for [Unbabel/TowerInstruct-7B-v0.2](#) does not include it.