Evaluating Terminology Translation Methods

Théo Salmenkivi-Friberg[†§*], **Iikka Hauhio**[‡§*], and **Tommi Nieminen**[†]
[*] equal contribution
[†] Department of Digital Humanities, University of Helsinki, Finland
[‡] Department of Computer Science, University of Helsinki, Finland
[§] Hipsu Oy, Helsinki, Finland
{theo.friberg,iikka.hauhio,tommi.nieminen}@helsinki.fi

Abstract

We present an evaluation of several state-of-the-art machine translation systems supporting terminology constraints in the English–Finnish translation direction. We first perform a meta-evaluation, in which we critically evaluate the evaluation metrics we use, including the questions asked of human evaluators and the automatic evaluation methods. We find that common metrics such as term accuracy and TERm do not agree with the human evaluators' judgement on the correctness of the terms, while LLM-as-a-judge shows promise even though it does not agree with the human evaluators on all questions. We then compare the evaluated systems based on the human evaluation results, LLM-as-a-judge, COMET, and chrF2. We find that of the systems considered, soft constraint methods, including a term-trained model and an LLM, perform better than hard constraints forced using a constrained beam search.

1 Introduction

Terminology translation refers to a group of methods that are used to guide a machine translator to use specific terms consistently in its output. In the past decade, multiple such methods have been introduced, but it is still unclear which of these methods are the best. We present a comparison of multiple state-of-the-art systems representing different approaches. Furthermore, we analyse the evaluation methods themselves and identify their strengths and pitfalls.

All methods considered in this paper are based on so-called *terminology constraints*, in which the input sentence is first analysed and the source language terms it contains are identified. Then the corresponding target language terms are inputted to the translation system as constraints that should be present in the output sentence (Hokamp and Liu, 2017; Dinu et al., 2019; Alam et al., 2021a; Semenov et al., 2023).

In the so-called *hard* methods (Hokamp and Liu, 2017; Post and Vilar, 2018; Hu et al., 2019; Hauhio and Friberg, 2024) a constrained beam search (CBS) algorithm is used to force the constraints to appear in the output, determining the most likely place for them using the beam search. In *soft* methods, on the other hand, the terms are annotated into the input sentence that is fed to the NMT model (Dinu et al., 2019; Bergmanis and Pinnis, 2021; Nieminen, 2024; Hauhio and Friberg, 2024). The model can then choose to use these annotations to construct the output sentence, but it might also alter or ignore them. This might be beneficial if the constraints are for some reason erroneous or inappropriate in the sentence's context, but it also lowers the controllability of the system. Recent soft methods also include using large language models with terms included in the prompt (Moslem et al., 2023; Kim et al., 2024).

We aim to answer two research questions: First, which approach to terminology translation performs the best? Second, which automatic evaluation methods correlate with the human evaluation? We use a single experiment to answer both of these questions. We evaluate five systems, some of them hard, some soft, and some combinations of the two. We perform several correlation and agreement tests to compare the human metrics to the different automatic evaluation methods, including

chrF2 (Popović, 2015), COMET (Rei et al., 2022), TERm (Alam et al., 2021a), term accuracy, and LLM-as-a-judge. We then use metrics we found reasonable to construct partial orderings of the evaluated systems, reporting which systems can be distinguished from each other.

In Sections 2 and 3, we give an overview of existing terminology translation approaches and evaluation methods. Then, in Section 4, we go through our experimental setup in detail. The results are presented in Section 5. Finally, we discuss the implications of our study in Section 6.

2 Terminology Translation

Terminology translation is an umbrella term for a large variety of methods that range from adding vocabularies to rule-based translator systems to including whole terminologies in large language model (LLM) prompts. In this paper, we focus on a subset of systems based on *terminology constraints* (or *lexical constraints*), since they perform a reasonably well-defined task, and thus can be directly compared with each other.

Constraint translation is a pipeline-based approach in which the translation system first performs *term recognition* on the input sentence. Depending on the input language, this may be a quite complex task. Hauhio and Friberg (2024), for example, perform both morphological and dependency parsing on Finnish to detect multi-word source terms. Constraint recognition is crucial as incorrectly detected input terms result in wrong output constraints. Often, however, this step is overlooked in studies and the constraints are merely assumed to be correct.[1]

After the constraints have been determined, either a soft or hard approach (or a combination thereof) is used to make the output contain the desired target language terms.

2.1 Soft Approaches

There are three main soft constraint approaches that we will cover: term-trained models, back-translation substitution, and prompted general-purpose LLMs.

[1]Some studies use synthetic term annotations based on word alignments (Bergmanis and Pinnis, 2021) while others use simple exact word matches with morphologically simple languages (Dinu et al., 2019). Others use preannotated datasets such as sentences from Wikipedia, in which each hyperlink to an article is considered to be a term annotation (Alam et al., 2021b).

Term-trained models. Machine translation models trained with soft lexical constraints were first introduced by Song et al. (2019), where source sentences in the training data were annotated by replacing selected source phrases with target phrases that occur in the corresponding target sentence. This causes the model to learn to copy target language phrases occurring in source sentences into translations. Dinu et al. (2019) independently applied soft lexical constraints specifically to terminology translation, and used two annotation methods: the **replace** method replaces the source term with the target term (as in Song et al. (2019)), while the **append** method appends the target term after the source term.

The two different annotation methods affect the strength of the soft constraint. When the target term is appended, the constraint is weaker, as the MT model can use both the source and target terms to generate the translation, and the appended target term may be overridden by a more likely translation for the target term. When the target term replaces the source term, the constraint is stronger, as the source term is not available for the MT model and it can therefore not produce any of the more likely translations for the term. Note that even in this case the constraint is not hard, as it may be overridden by other sentence context.

Term annotations in the input of the model are usually differentiated from the actual source text to help the model learn to process them differently. Dinu et al. (2019) indicate term annotations by using an additional input stream (factor) to indicate whether a token is part of the source text or a term annotation. Ailem et al. (2021) use a single input stream, with tags marking the start of the source term, start of the target term, and the end of the target term, as do most other soft constraint implementations (Alam et al., 2021b).

Terms are usually provided during inference in their lemma form, as that is the form in which terms are stored in termbases. This means that the models must be trained with data where the source sentences are annotated with lemma forms and the target sentences contain the grammatically appropriate surface forms of the term lemmas. For morphologically simple target languages this issue can be mostly ignored (as was done e.g. in Dinu et al. (2019)), as the surface forms and the lemma form are often identical. However, for morphologically complex target languages, explicit lemmatization is

necessary, and it was first introduced concurrently by Bergmanis and Pinnis (2021) (for Baltic languages) and Jon et al. (2021) (for Czech).

Back-translation substitution. While term-trained models require training new models on the term translation task, there is also at least one soft method that can be applied to any model without training it first on the term translation task: Hauhio and Friberg (2024) present a soft contraint method that they call back-translation substitution (BTS) that resembles the **replace** method (Dinu et al., 2019; Song et al., 2019), except that instead of replacing the source-sentence terms with the target terms, it uses back-translations of the target terms to the source language. The subtitution is only done if a language model deems the translation from the backtranslation to the target term more likely than the translation from the actual source term to the target term, a test which is done on terms alone without the context of the sentence. Hauhio and Friberg (2024) use architecturally identical OPUS models trained on the same data for the translation, the grading and the back-translation.

Hauhio and Friberg (2024) report the best results in human evaluation to have been achieved with a combination of constrained beam search (CBS) and back-translation substitution. They compare the constrained beam search alone to the combination of BTS and constrained beam search. However, they do not perform the inverse ablation with only the BTS without CBS, so it is unclear if BTS alone could work as a soft method.

Prompted LLMs. A third type of soft constraints is using large language models (LLMs) by inserting the terms into the prompt (Kim et al., 2024). Pre-trained LLMs can perform well in this task even though they are not specifically trained for it (although examples of this task might still appear in the training set), unlike specialized term-translation models (Dinu et al., 2019; Bergmanis and Pinnis, 2021; Nieminen, 2024). It is also possible to use an LLM to post-edit a translation so that it uses the correct terms (Moslem et al., 2023).

2.2 Hard Approaches

Hard methods use constrained decoding techniques to force the target language terms to appear in the output sentence. These methods construct a grammar that accepts only the outputs that contain the desired constraint words. The complexity of this grammar ranges from a trie containing all valid output sequences (Cao et al., 2021; Salmenkivi-Friberg and Hauhio, 2025) to regular languages (Hauhio and Friberg, 2024) and context-free grammars (Geng et al., 2023; Tam et al., 2024).

The simplest constrained decoding method sets the probability of tokens not allowed by the grammar to zero during decoding (Salmenkivi-Friberg and Hauhio, 2025), a method which is also used in the context of LLMs to force the output to, e.g., conform to JSON grammar (Tam et al., 2024). This, however, might result in endless generation when the model does not consider the constraint word probable. For example, if the constraint word is `cat` and the corresponding regular grammar is `.*cat.*`, the model gets stuck in the first wildcard which, due to being a wildcard, does not set any token probability to zero, with the exception of the end token.

To solve this issue, multiple beam search-based methods have been suggested. Hokamp and Liu (2017) introduce a method called Grid Beam Search in which the beam is divided into several "banks" which only allow hypotheses that contain a specific number of output constraint tokens. The algorithm attempts to insert each constraint at every position and compares the resulting hypotheses with the hypotheses in the banks, thus finding the place that the model considers most likely for each constraint. This algorithm has been further developed by Post and Vilar (2018) who consider an alternative beam allocation strategy to calculate the sizes of the banks, Hu et al. (2019) who use tries instead of a table to track which constraints are fulfilled, and Hauhio and Friberg (2024) who use a regular grammar (finite-state machine).

Constrained beam search is an appealing paradigm, as it is model agnostic, thus requiring no re-training, although it also comes with an additional computational cost, requiring the use of a large beam size for storing all of the banks. In contrast, soft methods can be used with greedy decoding or sampling (equivalent to a beam size of 1 in terms of computational cost).

3 Evaluation Methods

We divide evaluation methods relevant for terminology translation into human evaluation and automatic evaluation methods.

3.1 Automatic Evaluation

There is no standard method for evaluating terminology translation. Common machine translation metrics such as BLEU (Papineni et al., 2002), chrF (Popović, 2015), and COMET (Rei et al., 2022) are very popular, and while they are not specific to terminology translation, if the reference translation uses the correct terms, these metrics will indirectly measure terminology translation quality.

Alam et al. (2021a) introduce three terminology translation metrics: exact-match term accuracy, window overlap accuracy, and TERm. Bergmanis and Pinnis (2021) use a variant of exact-match term accuracy called lemmatized term accuracy. We also consider LLM-as-a-judge as a possible metric for terminology translation. These metrics are described below.

Term accuracy is likely the most common metric specific for terminology translation. There are two main variants of it: Exact-match term accuracy is the number of target language terms contained within the output sentence divided by the number of constraints (Alam et al., 2021a). Lemmatized term accuracy is the same, but all words are lemmatized before checking for the presence of the constraints (Bergmanis and Pinnis, 2021). While intuitive, we believe that these metrics are often misleading. They do not work at all for hard methods, as they always give the result 100% even if the term is inserted in the wrong position. For soft methods, they fail to recognize the situation in which the system ignored an erroneous constraint. Our hypothesis for this paper is that term accuracy does not correlate well with the human judgement of whether the terms are correctly applied or not.

Window overlap is a metric proposed by Alam et al. (2021a) that compares the words surrounding the terms in the output sentence to the words surrounding them in the reference sentence. As the implementation by Alam et al. (2021a) does not lemmatize the words, this metric does not, at least without modification, work well for highly agglutinative languages, and for this reason we do not consider it in this paper.

TERm is a variation of the TER metric, which measures the number of editing operations required to transform the output sentence into the reference sentence (Alam et al., 2021a). TERm differs from TER by giving the penalty of 2 (instead of 1) to operations that affect the terms in the sentence.

LLM-as-a-judge. Large language models have been found to be good evaluators of translation quality in general machine translation research (Kocmi and Federmann, 2023; Kim, 2025). Although they have not been used for terminology translation evaluation to our knowledge, we believe they are a valid evaluation method.

3.2 Human Evaluation

Human evaluation of terminology translation tasks often consists of direct assessment or ranking, combined with yes/no questions corresponding to error types or their inverses. While existing error hierarchies such as MQM (Lommel et al., 2013) contain some terminology-related error types, papers using human evaluation such as (Alam et al., 2021a; Bergmanis and Pinnis, 2021; Hauhio and Friberg, 2024) rarely use them and opt to use simple, ad-hoc error schemata tailored towards their specific research questions. Alam et al. (2021a) present their human evaluators two questions: "Is the source term translated correctly?" and "Is the term's surrounding context correct in the translation?". Bergmanis and Pinnis (2021) used an error typology with three error types: "Wrong lexeme", "Wrong inflection", and "Other". Hauhio and Friberg (2024) used a typology of two error types: "incorrectly applied constraint" and "erroneous due to other cause".

4 Experimental Setup

We conducted an experiment using two state-of-the-art terminology translation systems: the CBS and BTS-based method presented in (Hauhio and Friberg, 2024) representing hard approaches, and a term-trained model by Nieminen (2024) representing soft approaches[2]. We further included a large language model as another soft method, an NMT model baseline, and a hybrid hard-soft method using the term-trained model and constrained beam search at the same time.

We used two Finnish terminologies with English term translations as our evaluation datasets, as well as one dataset constructed by us. We only evaluate the English–Finnish translation direction due to our limited budget. Since Finnish is more agglutinative, we believe it provides a more interesting testbed for both the systems and the evaluation methods analysed in this study.

[2]We thank Kielikone Oy for the access to their proprietary implementation of the CBS/BTS method (*Mitra*) and the code used to call the NMT models.

4.1 Datasets

We collected or created three test datasets in different domains to probe model performance in different contexts and slightly different task formulations.

***Brands*.** We wrote 30 sentences with the intent of creating an intentionally hard dataset. Each source sentence mentions a fictional brand. Part of the brands were simply invented on the spot and part of them were sampled from the new Finnish surname proposals and new Swedish surname proposals provided by the Institute for the Languages of Finland. These lists of currently unused – partly computer-generated, partly just currently unused – surnames exist to offer inspiration to people wishing to take a new surname and are thus unlikely to appear in existing text. An unrelated fictional brand is required as a constraint, simulating the use case of a brand that is localized in an un-knowable way. This dataset intends to test copy-and-inflect behaviour in a context where the required constraint is surprising. This was intended as a purposefully difficult test where we hypothesized that soft methods would misbehave by refusing to obey the constraints.

***Rakennettu ympäristö* and *Valtioneuvosto*.** These two large datasets are constructed from the example sentences, notes and definitions of two real-life termbanks: *Rakennetun ympäristön sanasto* (Terminology of Built Environment) made available by the Finnish Ministry of the Environment and *Valtioneuvostosanasto* (Finnish Government Glossary) by the Prime Minister's Office.

An automated term-detection pipeline was run on the example sentences of the termbanks, using the termbanks themselves as the glossary. The constraints were then constructed according to the termbanks. Multiple factors make these noisy datasets for machine translation.

Firstly, there are factors that have to do with the termbank itself. The intended audience is a human translator seeking guidance on the proper usage of terms. As such the headwords tend to be explicative and not just contain the core of a term. For instance, the term "presenting official of the Government" might reasonably appear as just "presenting official" in a sentence context. As such, forcing "of the Government" to appear in the translation especially if it has been already otherwise conveyed leads to repetitive and awkward translations. The English terms also have a tendency to be longer, as both termbanks pertain to phenomena – Finnish legislative processes and building regulation – that occur in Finnish national languages: the English terms are varyingly post-hoc translations. The fact that some sentences are from the usage notes of terms also plays a part in the noisiness: the same usage notes might not apply to the different-language terms denoting the same concept and thus the source and gold sentence may not be a matching pair.

Secondly, there are factors that affect the term-detection pipeline. As the terms have a tendency to be long and explicative, they do not always appear in full in example sentences and usage notes – particularly those of other terms. These elliptical terms do not get detected by our pipeline. A failure in term detection may result in a missing constraint or a part of a constraint being detected instead of the whole. The translation given to this partial term may not even appear in the correct translation of the whole term.

These forms of noisiness are typical of real-world translation as opposed to our other test datasets which are laboratory examples of what we expect to be difficult corner cases. Here the task is not to copy or to copy and inflect but one where rejecting obviously incorrect terms can significantly improve translations and having the ability to ellipsise repetitive language is useful.

Due to our limited budget, we conducted our evaluation on a subset of 100 sentences. 18 of them are from the *Brands* dataset and 41 from the *Valtioneuvosto* and *Rakennettu ympäristö* each.

4.2 Evaluated Systems

We evaluate five system configurations that are presented in Table 1 chosen to represent the state-of-the-art methods in hard and soft term translation. The table also includes three configurations that we ultimately did not include in the human evaluation, which we describe in Appendix A.

As a baseline, we consider an Opus-MT English–Finnish translation model with a regular beam search of width 4 without any term constraints. As a hard method, we combine this model with both constrained beam search and back-translations, as described by Hauhio and Friberg (2024).

For soft methods, we include both a model fine-

[3] `opusTCv20210807+bt-2021-08-25`, which is specifically the version from which the term-trained model was fine-tuned from.

[4] `https://object.pouta.csc.fi/niemine1/opus_finetuned_term_model.eng-fin.zip`

Model	Soft method	CBS	BS	TS
opus-mt-tc-big-en-fi[3] (Tiedemann, 2020)	None	A (C)	H (Ba)	
	Back-translation substitution	H (CB)	A (Bt)	
Term-trained en-fi model[4] (Nieminen, 2024)	Append	H (CT)	H (T)	
Gemma 3 12B (Kamath et al., 2025)	Prompting			H (G)
EuroLLM 9B (Martins et al., 2024)	Prompting			A (E)

Table 1: Evaluated systems. CBS=Constrained beam search. BS=Beam search (width=4). TS=Temperature sampling ($T = 0.1$). Systems marked with H were included in the human evaluation and featured in the main portion of the paper, while systems marked with A were evaluated with automatic methods only and presented in Appendix A. Shortened form of name in parentheses

tuned from the aforementioned OPUS-MT model to support soft constraints, trained by Nieminen (2024), and a large language model, Gemma 3 12B (Kamath et al., 2025). We also evaluate the combination of the soft constraint model and constrained beam search as a hybrid hard-soft method.

To ensure a fair comparison of the different methods, we strived to use them in a manner that matches their real-world usage. This introduced differences between the systems that might advantage some of them, which we consider fair due to it matching their real advantages in production use. First, we use temperature sampling for the large language model and beam search for the Opus-MT-based models. Second, the LLM (Gemma 3) was given a list of synonyms from the termbank while a single randomly chosen target term was sampled for the other systems, as back-translation substitution and the term-trained model we use do not support synonyms.

4.3 Automatic Evaluation Methods

We conducted evaluation on a subset of 100 sentences per system, same as in the human evaluation (additional experiments with the whole dataset are presented in Appendix A). Following the best practices set out by Kocmi et al. (2021), we do not use BLEU; instead, we selected COMET (Unbabel/wmt22-comet-da), chrF2[5], TERm, term accuracy, and LLM-as-a-judge as our automatic evaluation metrics. Term accuracy was based on the finite-state machine of the *Mitra* system (due to this, CBS-based systems are ensured to get a 100% term accuracy unless they crash or timeout). We used the GLM-5 language model without reasoning through OpenRouter as our LLM-as-a-judge, and it was given as prompt written instructions corresponding to the oral instructions given to the human annotators[6] (see Appendix C). See the subsection below for the description of the human metrics.

4.4 Human Evaluation

We conducted the human evaluation with 100 sentences sampled from our evaluation data, which were translated by our five systems. All three evaluators were presented with the full 500 sentences in a shuffled order on the RedCap platform.

We ran a small pilot study to evaluate effect sizes and decide how many sentences we would likely need evaluated. This lead us to estimate that a hundred translations would likely let us draw statistically significant conclusions.

We recruited three human experts to evaluate our translations. One of the evaluators was selected for translation work experience, another for bilingual proficiency in the English-Finnish language pair and a third for experience working in the Finnish administration and familiarity with the jargon that it involves[7]. We conducted in-person interviews and a paid training task on a separate 30-sentence (six sentences translated by five systems) split which we (quite accurately according to our annotators) estimated to represent an hour of annotation work. We first presented the training split to our annotators, inspected their responses and had a conversation (in-person with one, by videoconference with another and over a textual medium with the third) about their responses, focusing on the ones that we found to

[5]Following the recommendations from Marie et al. (2021), we compute chrF using sacrebleu with the following signature: `nrefs:1|bs:1|seed:12345|case:mixed|eff:yes|nc:6|nw:0|space:no|version:2.6.0`.

[6]We attempted to enable constrained decoding to force the model to generate valid JSON according to our schema, but we found that the outputs still did not conform to our schema. We suspect that the inference provider might have a "soft" approach behind their JSON schema support such as appending the JSON schema to the prompt. This causes the caveat that we cannot guarantee that the model actually received our prompt unchanged.

[7]After the annotation job, they stated that their background in Finnish administration was useful, but that they did not feel that they had domain expertise in the particular jargon of the translations.

be inconsistent with our expectations and what we were asking. This concretely uncovered misunderstandings of what the questions meant and allowed us to validate both the annotators' understanding of the task and their willingness to commit to the remaining 500-sentence (100 sentences translated by five systems) split. We then discarded the data from the training task. Two of our annotators were paid a lump sum that we estimate to be 15€/h gross. The third was an employee of our university and the task was conducted during work hours and was compensated as part of their normal salary.

The human evaluation consisted of the following axes: general `fluency` as a target-language text (subjective analog scale), general translation `quality` focusing on the preservation of meaning (subjective analog scale), `nonsense`: sentence too corrupted for the evaluation of terms (per sentence boolean), `correct`: terms correctly applied (per sentence boolean), `missing`: missing terms (per sentence boolean), `untranslated`: terms appearing in the translation in their untranslated source-language form (per sentence boolean), `misspelling`: misspelling in terms (per sentence boolean), `misuse`: terms misplaced, misused, or misinflected such that the meaning of the sentence is corrupted (per sentence boolean).

The main difference between `fluency` and `quality` is that the former does not take into account the source sentence or terms, as it only measures the fluency in the target language. `quality` takes terms into account only insofar as it penalizes for a change in the meaning of the sentence, but it should not penalize for reasonable synonyms even if not present in the termbank.

We have included notes about the human evaluation process in Appendix B.

4.5 Meta-Evaluation Methods

To test the performance of our automatic evaluation metrics, we run correlation and agreement analysis. We also strive to implement rigorous statistical testing on our human evaluations. We report Cohen's κ for our human evaluators on the discrete metrics, and use $\kappa \geq 0.6$ as our threshold for whether the metric is meaningful or not[8]. For comparing continuous metrics, we use the Pearson correlation coefficient.[9] To compare continuous metrics such as COMET and TERm to the binary error typology of the human annotators, we convert the binary metric to continuous scale by taking the sum over the human annotators, and then calculate Kendall's τ-b.

5 Results

We first report the findings of our meta-evaluation, in which we calculate correlations between the human metrics and the automatic metrics, and the agreement for the humans and the LLM-as-a-judge. We then report the results for the evaluation of the systems, only including those metrics that we found meaningful in the meta-evaluation.

We publish our full system outputs to aid in replicability and – following Marie et al. (2021) – strongly encourage future publications comparing to our work to recompute metrics on our data as opposed to copying our computed results.[10]

5.1 Meta-Evaluation Results

Human correlation and agreement. We calculated the Pearson correlation coefficients (r) between the continuous human metrics, which are presented in Table 2. The lowest r between any pair of humans was 0.53, while the highest r was 0.65. The individual experts correlate with the mean of them all with $0.81 \leq r \leq 0.86$. We consider all these correlations high enough that the metric is meaningful and we can use it for ranking the systems.

In the boolean metrics (Table 3), we find that our annotators had the highest agreement on the question of terms left `untranslated` per Cohen's κ (0.94, 0.94, and 1). The second-best agreement was for the question of `missing` terms. The question of terms being `correct` has agreement in the range $0.58 < \kappa < 0.61$, so it sits on our threshold of 0.6, which we use for determining if the metric is meaningful. All other questions have a $\kappa << 0.6$, and thus we will discard them from the systems evaluation.

We calculated inter-class agreement using Cohen's κ and present it in Appendix D.

LLM-as-a-judge. The Pearson correlation coefficients for LLM–human expert pairs are in the range

[8]This arbitrary threshold is often used to indicate "substantial" agreement (Landis and Koch, 1977).

[9]We measured correlation instead of agreement as the analogue scale used for these metrics is unanchored and its interpretation might vary between annotators.

[10]The system outputs and the human evaluation dataset are available at `https://github.com/nomelif/eamt2026-term-translation`.

	Expert 1	Expert 2	Expert 3	Human mean	LLM
Expert 1		0.62	0.58	0.86	0.70
Expert 2	0.59		0.56	0.86	0.68
Expert 3	0.65	0.53		0.83	0.71
Human mean	0.82	0.81	0.85		0.82
LLM	0.74	0.69	0.73	0.83	

Table 2: Pearson correlation coefficient of the continuous metrics for the human annotators and the LLM-as-a-judge. Top right corner: general `fluency`. Bottom left corner: translation `quality`.

	1 vs. 2	1 vs. 3	2 vs. 3	1 vs. LLM	2 vs. LLM	3 vs. LLM	count (1, 2, 3)		
nonsense	0.51 (±0.11)	0.43 (±0.17)	0.36 (±0.12)	0.48 (±0.16)	0.41 (±0.12)	0.50 (±0.18)	35	22	22
correct	0.59 (±0.07)	0.58 (±0.07)	0.61 (±0.07)	0.35 (±0.06)	0.49 (±0.07)	0.40 (±0.07)	293	224	269
missing	0.65 (±0.10)	0.69 (±0.10)	0.62 (±0.11)	0.17 (±0.12)	0.12 (±0.11)	0.13 (±0.11)	63	65	59
untranslated	0.94 (±0.06)	0.94 (±0.06)	1 (±0)	0.93 (±0.06)	0.93 (±0.06)	0.93 (±0.06)	35	33	33
misspelling	0.18 (±0.17)	0.14 (±0.13)	0.39 (±0.15)	0.17 (±0.20)	0.29 (±0.19)	0.22 (±0.14)	10	28	47
misuse	0.36 (±0.09)	0.27 (±0.10)	0.38 (±0.09)	0.24 (±0.06)	0.29 (±0.07)	0.19 (±0.05)	129	162	91

Table 3: Cohen's κ for the binary metrics per annotator and the Human-LLM agreement. Means and confidence intervals were calculated using bootstrapping (BCa, size 100000). The count column has the number of positives for each category for the human annotators (total $n = 500$ sentences).

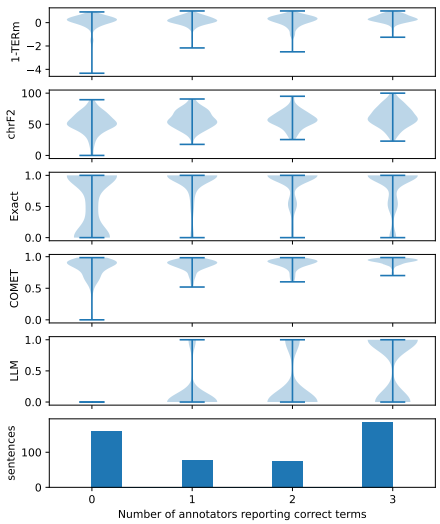

Figure 1: The automatic term metrics compared to human term accuracy in ascending order of Kendall's τ-b (see Table 4). Note that exact term accuracy is in practice quite discrete and LLM-as-a-judge is binary valued. Whereas a clear upwards trend can be seen in COMET, if such a trend is present for TERm (or more properly, 1-TERm), it is far subtler. Notice that we ran per-sentence chrF2 and that SacreBLEU's multiple sentence chrF2 seems to not be the mean of individual sentences.

Automated metric	Kendall's τ-b
1-TERm	0.143
chrF2	0.177
Term accuracy	0.195
COMET	0.322
LLM-as-a-judge `correct`	0.514

Table 4: Kendall's τ-b values for automated metrics against the mean of the human `correct` metric over the annotators. The value for chrF2 should be taken with a grain of salt, as SacreBLEU's implementation seems to yield different values depending on other sentences in the corpus and here we computed individual chrF2-scores for sentences.

$0.68 \leq r \leq 0.74$, while the coefficients for LLM–human mean are 0.82 and 0.83 for general `fluency` and translation `quality`, respectively. The LLM correlates with the experts more than the experts correlate with each other, while the correlation with the human mean is over 0.8. We consider the continuous LLM-as-a-judge metrics meaningful.

The Cohen's κ values for LLM-as-a-judge are presented in Table 3. Of these, only `untranslated` has $\kappa > 0.6$, similar to the humans. Surprisingly, while there was agreement between the humans for the question of `missing` terms, the κ for LLM–human pairs was in the range $0.12 \leq \kappa \leq 0.17$, implying low agreement.

Somewhat interestingly, when none of the humans evaluated a sentence to be `correct`, the LLM did not evaluate that sentence as `correct` either. See Figure 1 and the text below.

Other automatic metrics. We use Kendall's τ-b to assess if COMET, chrF2, term accuracy, and 1-TERm could be used as proxies for the output sentence having `correct` terms (Table 4). We compare these against the mean of the `correct` values by humans (we interpreted the booleans as 1 and 0 for the purpose of calculating the mean). We also include LLM-as-a-judge's `correct` as a binary value.

Term accuracy and 1-TERm have τ values 0.195 and 0.143, which are lower than the τ of COMET, 0.322, although all these values are quite low. LLM-as-a-judge has $\tau = 0.514$, far higher than any of the other automatic metrics, even though we note that it is also a fairly weak measure, having Cohen's $\kappa \leq 0.49$.

Note that since the inter-annotator agreement for

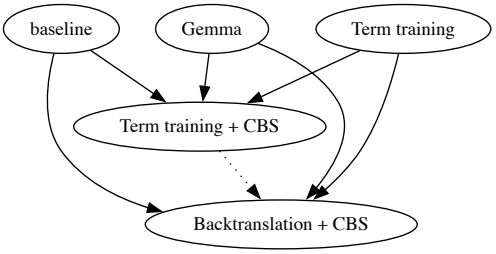

Figure 2: Target language `fluency`. All human annotators agree on the solid edges. The edge between Term training + CBS and Backtranslation + CBS can be concluded for only one human annotator. LLM-as-a-judge agrees on all edges, including the dotted edge.

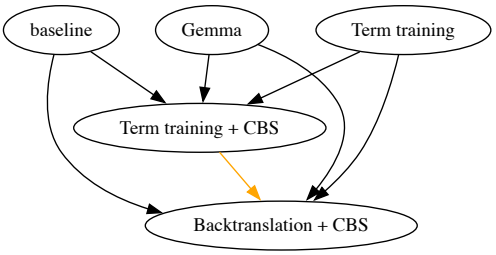

Figure 3: General translation `quality`. Human annotators agree on all black edges. LLM-as-a-judge agrees on all edges. To save space, this graph is also COMET (only black edges).

this human metric is low ($\kappa < 0.6$ for two of the human–human pairs), these results should be taken with a grain of salt. The distributions of the automatic metrics for cases in which all human experts agree and where they disagree are presented in Figure 1.

We also calculate Cohen's κ between term accuracy and `missing`, yielding 0.32, 0.36, and 0.29 for each of the human annotators, implying low agreement.

5.2 Systems Evaluation Results

Based on the meta-evaluation, we decided to use the human metrics `fluency`, `quality`, `correct`, `missing`, and `untranslated`, as well as the LLM-as-a-judge metrics `fluency`, `quality`, and

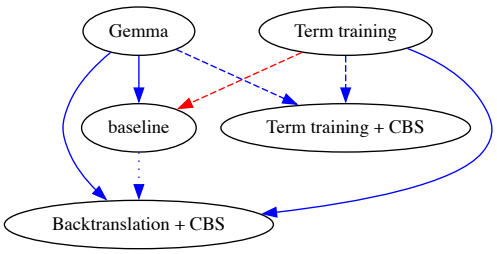

Figure 4: `correct` terms per human evaluation and LLM-as-a-judge. Solid edges have three human annotators agree, the dashed edges two and dotted edges one. Blue edges can be concluded from LLM-as-a-judge and the red edges cannot be.

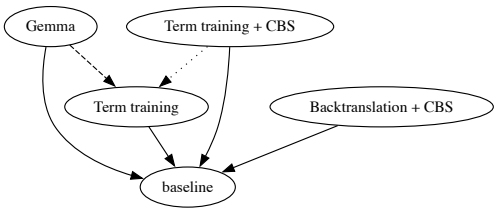

Figure 5: ¬`missing` terms per human evaluation. Solid edges have all three annotators agree, the dashed line has two and the dotted line only one.

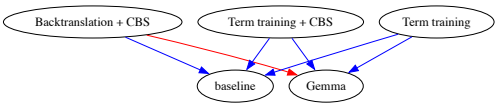

Figure 6: ¬`untranslated`. Human annotators agree on all edges, while red edge cannot be concluded from LLM-as-a-judge.

`untranslated`. We also report the COMET and chrF2 results, even though we stress that they can only be taken as general translation quality metrics, and not metrics of terminology translation quality.

We take an unconventional approach and will not list numbers in our results section. Instead, we present our results as directed graphs based on the binomial sign test. These graphs represent a partial ordering of the evaluated systems in which each arrow points from the better system towards the worse system, i.e., the target side of an arrow is statistically significantly weaker than the source. Controlling for multiple comparisons is a well-known issue in statistics (Bennett et al., 2009) and consequently we apply Bonferroni correction to the graphs when determining statistical significance, setting $\alpha = \frac{0.05}{N}$, where N is the number of comparisons made. Lack of an arrow does not mean that the systems are of the same quality, only that we could not refute the null hypothesis. We report the p-values in Appendix E.

This method was chosen as a response to the criticism towards common machine translation research methodology in which results are presented as tables of numbers. This leads to situations in which there is a temptation to compare two numbers without conducting a proper statistical significance test

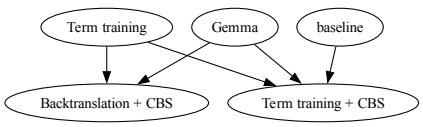

Figure 7: chrF2.

or even ensuring that the numbers are based on the same evaluation set (Marie et al., 2021). We want to discourage this behavior by instead presenting our results graphically. We release full system outputs to allow people to replicate our study.

The partial orderings produced by our chosen metrics are presented in Figures 2, 3, 4, 5, 6, and 7. All of these orderings are calculated based on the same subset of the sentences as used for human evaluation. Based on the results, the term-trained model and Gemma 3 outperform the CBS-based systems in `fluency`, `quality` and `correctness`, while, on the other hand, the term-trained model and the CBS-based methods outperform Gemma 3 in `untranslated` terms. For the `missing` metric, we could not determine the ordering between the hard and soft methods.

While the main portion of this paper considers only a subset of the data, we include automatic evaluation of the entire dataset in Appendix A, including results for specific subsets.

6 Discussion and Conclusions

We have presented a comprehensive evaluation of terminology translation systems as well as a meta-evaluation of evaluation metrics. According to our results, soft constraint methods outperform hard methods in the fluency of the output, the preservation of meaning, and the correctness of terms.

Our most interesting finding is that high term accuracy does not imply that the human evaluators consider the terms to be correct, and neither does it match the human conception of missing terms. This implies that maximizing the number of constraints present in the output sentence is not necessarily a good objective for a system. Hard methods that do this perform consistently poorly across different metrics.[11]

This also prompts the discussion of how exactly the terminology translation task is defined and what its purpose is. Hard methods allow great controllability of the output, as the user will always "get what they ask for". If the output is bad, they can adjust the constraints to get a better result. However, if the use case is domain adaptation and the system is just given a large termbank not tailored towards the specific sentences (such as *Rakennettu ympäristö* or *Valtioneuvosto*), a larger number of the constraints are just wrong and including them in the output sentence will just worsen the result. The developer of the machine translation system must consider which of these use cases is more probable.

Our study also reveals that the commonly used automatic evaluation methods for terminology translation such as term accuracy and TERm might be unsuitable for this task. LLM-as-a-judge shows promise as an automatic metric, but even there the agreement with humans was too low for measuring the correctness of terms. Although, as seen in Figure 1, the LLM never assessed that a sentence's terms are correct if the humans agreed that they were incorrect, meaning that LLM-as-a-judge correctness might possibly be used as a part of an ensemble evaluation metric.

This study used a relatively small dataset focused on a certain domain and based on a specific style in which the termbanks were constructed. Different termbanks might have yielded different results (cf. Hauhio and Friberg, 2024, for differences between the *Finnish Parliament* and *Forest Fires* datasets). The chosen datasets in this study were intentionally difficult, real-world termbanks, and vocabularies constructed from the ground up to support terminology-constrained translation might improve performance significantly. Further research is required to explore how terminology constraint-optimal vocabularies could be constructed.

We emphasize the need for human evaluation, both in terminology translation research and machine translation research in general.

Carbon Impact Statement

We did not train any models, so the carbon footprint is caused entirely from running the translator systems and the inference used for performing the LLM-as-a-judge evaluation.

We do not have access to exact energy comsumption numbers, as the computation was not run in systems controlled by us (for LLM-as-a-judge, the provider used on OpenRouter does not disclose their datacenter location), but based on conservative estimates, we assess the total carbon emissions to be less than 1 kg CO_2.

We also analyzed the carbon impact of our travel to the conference. For two people flying a total of three flights between Amsterdam and Helsinki,

[11] In automatic evaluation of the entire dataset (see Appendix A), this holds even for the *Brands* subset that is defined so that each constraint really must be included in the output sentence for the translation to be correct, since it is also the subset that is most prone for term-related errors as the output, and the source and target language terms are not related to each other.

the carbon dioxide equivalent footprint amounts to about 800 kg CO_2, according to the GBT Neo flight booking system.

Author Contributions

The study was jointly planned and executed by TS and IH. The two existing datasets were acquired by IH, and *Brands* was written by both IH and TS. The automated evaluations were run by TS, with the exception of LLM-as-a-judge that was run by IH. The human evaluation was conducted by TS and IH, with both taking part in the design and evaluator briefing, and TS being responsible for configuring the RedCap platform. The meta-evaluation was lead by TS and assisted by IH. The paper was written by IH, with some sections contributed by TS and TN.

Acknowledgements

TS is funded by the Academy of Finland grant 365541 "AI for REINFORCING DEMOCRACY". IH is funded by the Doctoral Programme in Computer Science at the University of Helsinki. Both IH and TS have been employed by Kielikone Oy who supported their research as part of product development. TN is independently funded.

The authors wish to thank Kielikone Oy for access to the *Mitra* translation system (Hauhio and Friberg, 2024), and Netta Lagus and Laura Jalkanen for their data evaluation work.

References

Melissa Ailem, Jingshu Liu, and Raheel Qader. 2021. Encouraging neural machine translation to satisfy terminology constraints. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1450–1455, Online. Association for Computational Linguistics.

Md Mahfuz Ibn Alam, Antonios Anastasopoulos, Laurent Besacier, James Cross, Matthias Gallé, Philipp Koehn, and Vassilina Nikoulina. 2021a. On the evaluation of machine translation for terminology consistency.

Md Mahfuz Ibn Alam, Ivana Kvapilíková, Antonios Anastasopoulos, Laurent Besacier, Georgiana Dinu, Marcello Federico, Matthias Gallé, Kweonwoo Jung, Philipp Koehn, and Vassilina Nikoulina. 2021b. Findings of the WMT shared task on machine translation using terminologies. In *Proceedings of the Sixth Conference on Machine Translation*, pages 652–663, Online. Association for Computational Linguistics.

Craig M Bennett, Michael B Miller, and George L Wolford. 2009. Neural correlates of interspecies perspective taking in the post-mortem atlantic salmon: an argument for multiple comparisons correction. *Neuroimage*, 47(Suppl 1):S125.

Toms Bergmanis and Mārcis Pinnis. 2021. Facilitating terminology translation with target lemma annotations. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3105–3111, Online. Association for Computational Linguistics.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. In *International Conference on Learning Representations*.

Georgiana Dinu, Prashant Mathur, Marcello Federico, and Yaser Al-Onaizan. 2019. Training neural machine translation to apply terminology constraints. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3063–3068, Florence, Italy. Association for Computational Linguistics.

Saibo Geng, Martin Josifoski, Maxime Peyrard, and Robert West. 2023. Grammar-constrained decoding for structured NLP tasks without finetuning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10932–10952, Singapore. Association for Computational Linguistics.

Iikka Hauhio and Théo Friberg. 2024. Mitra: Improving terminologically constrained translation quality with backtranslations and flag diacritics. In *Proceedings of the 25th Annual Conference of the European Association for Machine Translation (Volume 1)*, pages 100–115, Sheffield, UK. European Association for Machine Translation (EAMT).

Chris Hokamp and Qun Liu. 2017. Lexically constrained decoding for sequence generation using grid beam search. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1535–1546, Vancouver, Canada. Association for Computational Linguistics.

J. Edward Hu, Huda Khayrallah, Ryan Culkin, Patrick Xia, Tongfei Chen, Matt Post, and Benjamin Van Durme. 2019. Improved lexically constrained decoding for translation and monolingual rewriting. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 839–850.

Josef Jon, João Paulo Aires, Dusan Varis, and Ondřej Bojar. 2021. End-to-end lexically constrained machine translation for morphologically rich languages. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4019–4033, Online. Association for Computational Linguistics.

Aishwarya Kamath et al. 2025. Gemma 3 technical report.

Ahrii Kim. 2025. RUBRIC-MQM : Span-level LLM-as-judge in machine translation for high-end models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pages 147–165, Vienna, Austria. Association for Computational Linguistics.

Sejoon Kim, Mingi Sung, Jeonghwan Lee, Hyunkuk Lim, and Jorge Gimenez Perez. 2024. Efficient terminology integration for LLM-based translation in specialized domains. In *Proceedings of the Ninth Conference on Machine Translation*, pages 636–642, Miami, Florida, USA. Association for Computational Linguistics.

Tom Kocmi and Christian Federmann. 2023. 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*, pages 193–203, Tampere, Finland. European Association for Machine Translation.

Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the sixth conference on machine translation*, pages 478–494.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Arle Richard Lommel, Aljoscha Burchardt, and Hans Uszkoreit. 2013. Multidimensional quality metrics: a flexible system for assessing translation quality. In *Proceedings of Translating and the Computer 35*, London, UK. Aslib.

Benjamin Marie, Atsushi Fujita, and Raphael Rubino. 2021. Scientific credibility of machine translation research: A meta-evaluation of 769 papers. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7297–7306.

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.

Yasmin Moslem, Gianfranco Romani, Mahdi Molaei, John D. Kelleher, Rejwanul Haque, and Andy Way. 2023. Domain terminology integration into machine translation: Leveraging large language models. In *Proceedings of the Eighth Conference on Machine Translation*, pages 902–911, Singapore. Association for Computational Linguistics.

Tommi Nieminen. 2024. Adding soft terminology constraints to pre-trained generic MT models by means of continued training. In *Proceedings of the First International Workshop on Knowledge-Enhanced Machine Translation*, pages 21–33, Sheffield, United Kingdom. European Association for Machine Translation (EAMT).

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

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

Matt Post and David Vilar. 2018. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1314–1324.

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. 2022. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Théo Salmenkivi-Friberg and Iikka Hauhio. 2025. Lingonberry giraffe: Lexically-sound beam search for explainable translation of compound words. In *Proceedings of Machine Translation Summit XX: Volume 1*, pages 173–189, Geneva, Switzerland. European Association for Machine Translation.

Kirill Semenov, Vilém Zouhar, Tom Kocmi, Dongdong Zhang, Wangchunshu Zhou, and Yuchen Eleanor Jiang. 2023. Updated findings of the wmt 2023 shared task on machine translation with terminologies. In *Proceedings of the Eight Conference on Machine Translation (WMT)*. Association for Computational Linguistics. The updated version available at `https://wmt-terminology-task.github.io/upd_wmt_terminology_2023.pdf`.

Kai Song, Yue Zhang, Heng Yu, Weihua Luo, Kun Wang, and Min Zhang. 2019. Code-switching for enhancing NMT with pre-specified translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 449–459, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhi Rui Tam, Cheng-Kuang Wu, Yi-Lin Tsai, Chieh-Yen Lin, Hung-yi Lee, and Yun-Nung Chen. 2024. Let me speak freely? a study on the impact of format restrictions on performance of large language models. *arXiv preprint arXiv:2408.02442*.

Jörg Tiedemann. 2020. The Tatoeba Translation Challenge – Realistic data sets for low resource and multilingual MT. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182, Online. Association for Computational Linguistics.

A Automatic Evaluation of the Entire Dataset, Including Ablations

Ablations. We evaluated eight systems in this study. Besides the five described in detail in section 4.2, we had three ablations that we did not have resources to conduct human evaluation on. (See Table 1 in Section 4.2 for a tabular overview of all the systems, including ablations.) These ablations were the finite-state machine-based CBS without any soft method, the back-translation based soft method without CBS, both proposed by Hauhio and Friberg (2024), and another prompted LLM (EuroLLM (Martins et al., 2024)).

The first two explore whether the improvements reported in (Hauhio and Friberg, 2024) come from the backtranslation substitution, the CBS formulation or a synergistic combination of the two. The third was included because we wanted to have a robust representative for the LLM technique and did not know ahead of time which LLM would perform best in this task. Our decision to prioritize Gemma over EuroLLM for the human evaluation was motivated by a near-complete failure to adhere to terms on the *Brands* dataset (per exact term accuracy): it completed zero constraints out of 26 in the fi-en direction and 2 out of 39 in the inverse direction. This was contrary to COMET, in which EuroLLM performed better (see below). Viewing COMET as a measure of general translation quality, we determined that having such a blatant failure case in the term task disqualified EuroLLM (at least with the prompt we used) from representing the state of the art in prompted LLM term translation, even if COMET pointed towards EuroLLM having better general translation performance. As we discuss below, COMET is a poor measure of the term correctness as interpreted by our human experts.

The automatic evaluation results for the specific subsets of our dataset are reported in Figures 8, 9, 10 (COMET), 13, 14, and 15 (chrF2). We also report the inverse language direction that was left out of the main study in Figures 11, 12 (COMET), 16, 17, and 18 (chrF2).

COMET. Figures 8, 9, and 10 report results on the *Brands*, *Valtioneuvosto* and *Rakennettu ympäristö* datasets in the English to Finnish direction. Figures 11 and 12 do so in the opposite direction. Notably, no statistically significant conclusions could be drawn from the *Valtioneuvosto* data in the Finnish to English direction. It is to be noted that COMET is not designed to be constraint-aware, though it does have access the reference translation and in that way how constraints appear in it. We expect it to penalize sentences where not using a term significantly changes meaning but be accepting of valid synonyms. This makes the *Brands* dataset particularly ill-suited for evaluation with COMET, as the constraints in it are purposefully arbitrary proper names. We believe that COMET should be considered a proxy metric for some mixture of our general translation quality and target language fluency measures in our human experiment.

While our datasets have significantly different composition, COMET is fairly consistent with both itself and humans in the English to Finnish direction. The common pattern in this language direction is how the CBS methods perform badly. In practice the English to Finnish figures' partial orders align very well with Figures 2 and 3. The Finnish to English direction is more complex to interpret. The *Brands* dataset only tells us how certain systems outperform our baseline (Figure 11) and the *Valtioneuvosto* dataset lets us draw no conclusions at all. On the *Rakennettu ympäristö* data, we have EuroLLM outperforming every other method. Pure constrained decoding, which we included as an ablation, perplexingly outperforms every other constrained method and the backtranslation baseline. Finally, the term-trained model can not be compared to anything but EuroLLM.

Our ablation of only using CBS without a soft method is rated lower than Term training + CBS on English to Finnish *Rakennettu ympäristö* and higher on Finnish to English *Rakennettu ympäristö*. Otherwise they are undistinguishable. This does not help us verify or refute the intuitive idea that the backtranslation method at least somewhat improves constraint decoding outputs from the perspectives

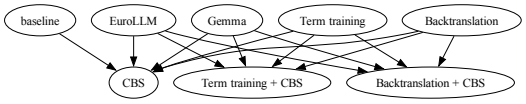

Figure 8: The partial order induced by COMET values for the *Brands* dataset in the English to Finnish direction. Target nodes have lower comet than source nodes. Only the edges for which the sign test gives a probability lower than $\frac{2 \cdot 0.05}{8(8-1)}$ to occur with the assumption of random flips are kept.

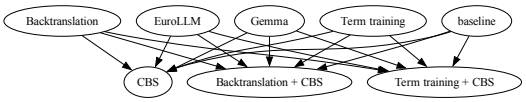

Figure 9: The partial order induced by COMET values for the *Valtioneuvosto* dataset in the English to Finnish direction. Target nodes have lower comet than source nodes. Only the edges for which the sign test gives a probability lower than $\frac{2 \cdot 0.05}{8(8-1)}$ to occur with the assumption of random flips are kept.

of meaning preservation and fluency. Based on COMET, the backtranslation-only method outperforms its combination with constraint decoding in the English to Finnish, giving some credibility to the hypothesis that the improvements seen in (Hauhio and Friberg, 2024) are due to back-translation independently from constraint decoding.

chrF2. We have included chrF2 for completeness and to follow the best practices set out by Kocmi et al. (2021). However, they recommend its inclusion primarily as a check for the (at the time) new COMET metric as a tried and true string metric. You can find the resulting partial order diagrams as Figures 13, 14, 15, 16, 17, and 18. When we attempted to compute chrF2 using SacreBLEU, we noticed that the corpus score is not the mean of sentence scores. To keep consistency with SacreBLEU's version, we implemented the bootstrap method described in (Kocmi et al., 2021). This method is much slower than the per-sentence bootstrap we could do for the COMET values, and thus due to time constraints, we could not run it with more than $n = 1000$.

B Details of the Human Evaluation

We decided on the prevalidation protocol involving a training task and a debriefing conversation based on a negative experience on a previous project. There, we discovered only after the annotation task was complete, that the annotator had misunderstood

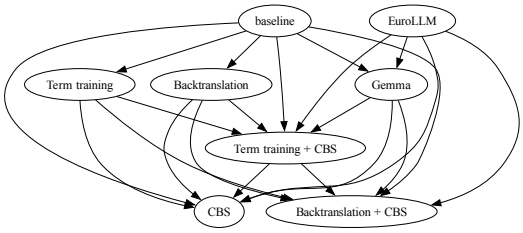

Figure 10: The partial order induced by COMET values for the *Rakennettu ympäristö* dataset in the English to Finnish direction. Target nodes have lower comet than source nodes. Only the edges for which the sign test gives a probability lower than $\frac{2 \cdot 0.05}{8(8-1)}$ to occur with the assumption of random flips are kept.

Figure 11: The partial order induced by COMET values for the *Brands* dataset in the Finnish to English direction. Target nodes have lower comet than source nodes. Only the edges for which the sign test gives a probability lower than $\frac{2 \cdot 0.05}{8(8-1)}$ to occur with the assumption of random flips are kept.

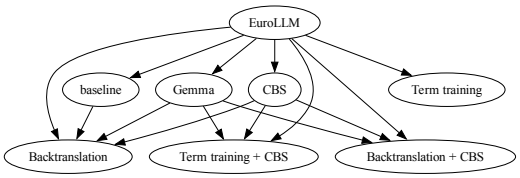

Figure 12: The partial order induced by COMET values for the *Rakennettu ympäristö* dataset in the Finnish to English direction. Target nodes have lower comet than source nodes. Only the edges for which the sign test gives a probability lower than $\frac{2 \cdot 0.05}{8(8-1)}$ to occur with the assumption of random flips are kept.

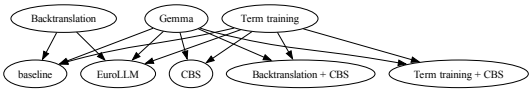

Figure 13: The partial order induced by chrF2 values for the *Brands* dataset in the English to Finnish direction. Target nodes have lower chrF2 than source nodes. Only the edges for which the sign test gives a probability lower than $\frac{2 \cdot 0.05}{8(8-1)}$ to occur with the assumption of random flips are kept.

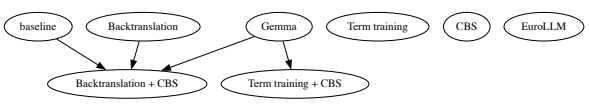

Figure 14: The partial order induced by chrF2 values for the *Rakennettu ympäristö* dataset in the English to Finnish direction. Target nodes have lower chrF2 than source nodes. Only the edges for which the sign test gives a probability lower than $\frac{2 \cdot 0.05}{8(8-1)}$ to occur with the assumption of random flips are kept.

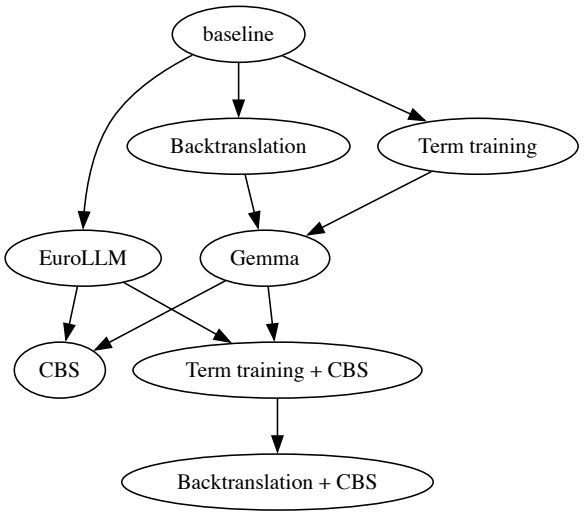

Figure 15: The partial order induced by chrF2 values for the *Rakennettu ympäristö* dataset in the English to Finnish direction. A true partial order is induced and to simplify the visualisation, transitive arrows are omitted. If there is a path from a source node to a destination node, the destination has lower chrF2 than the source. Only the edges for which the sign test gives a probability lower than $\frac{2 \cdot 0.05}{8(8-1)}$ to occur with the assumption of random flips are kept.

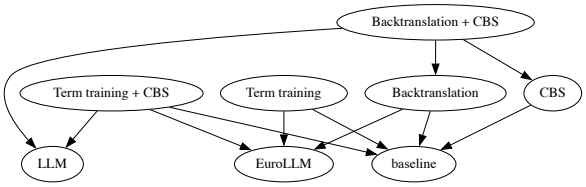

Figure 16: The partial order induced by chrF2 values for the *Brands* dataset in the Finnish to English direction. A true partial order is induced and to simplify the visualisation, transitive arrows are omitted. If there is a path from a source node to a destination node, the destination has lower chrF2 than the source. Only the edges for which the sign test gives a probability lower than $\frac{2 \cdot 0.05}{8(8-1)}$ to occur with the assumption of random flips are kept.

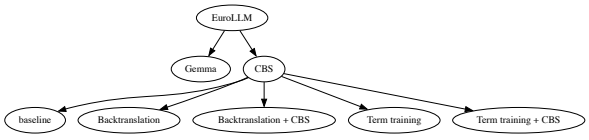

Figure 17: The partial order induced by chrF2 values for the *Rakennettu ympäristö* dataset in the Finnish to English direction. A true partial order is induced and to simplify the visualisation, transitive arrows are omitted. If there is a path from a source node to a destination node, the destination has lower chrF2 than the source. Only the edges for which the sign test gives a probability lower than $\frac{2 \cdot 0.05}{8(8-1)}$ to occur with the assumption of random flips are kept.

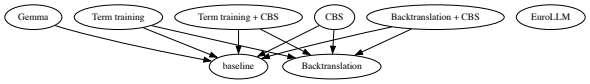

Figure 18: The partial order induced by chrF2 values for the *Valtioneuvosto* dataset in the Finnish to English direction. If there is a path from a source node to a destination node, the destination has lower chrF2 than the source. Only the edges for which the sign test gives a probability lower than $\frac{2 \cdot 0.05}{8(8-1)}$ to occur with the assumption of random flips are kept.

our instructions. We had also underestimated the amount of time required to reasonably complete the task. The combination of these factors lead to the annotations not being usable for us and the task undoubtedly being frustrating to the annotator. While our partially verbal instruction through the debriefing of the training task can be criticized for potentially giving different instructions to different annotators, we think that it results in fewer miscommunications and ultimately more usable data.

Colleagues in the speech synthesis domain told us that they tend to control for people with musical backgrounds in their annotator groups. The reason they gave was that having a musically trained ear notably changes a person's perception of speech. This made us think of our annotators' backgrounds and while we could not find such obvious confounding factors as musical background is in speech synthesis, we collected some rudimentary background information (see Table 5).

We firmly hold that qualitative feedback about the annotation process is an important part of a well-run human expert evaluation and highlights errata that, if ignored, could compromise the validity of the study. We solicited feedback from all three annotators after the main split of the data was evaluated. Two answered that they had little to add. The third was informally interviewed. The principal observation they wished to convey was how mental fatigue from a long session has an effect on the quality of work. A compounding factor to fatigue being the complexity of sentences: they noted that long and complex sentences (such as what they qualified as "legal domain sentences", which we take to mean *Valtioneuvosto* and *Rakennettu ympäristö*) were significantly more laborious to annotate than the sentences in more everyday language (which we take to refer to the sentences from the *Brands* dataset). From this, we gather that further human expert annotations should endeavour to produce shorter work sessions and that we should be aware of the complexity bias between our datasets. This casts doubt on the reliability of metrics such as chrF2 on the more complex sentences of *Rakennettu ympäristö* and *Valtioneuvosto*. To control for the sequential effects, we shuffled our data pre-annotation.

Our interviewee also noted that a significant amount of time was spent trying to classify errors that did not neatly fit our error taxonomy. This presents us with two tradeoffs: taxonomical complexity and overfitting error categories to the data. The more complex we make our taxonomy, the harder it will be to analyze the errors and more difficult it will be to align our annotators. The pilots and subsequent conversations (particularly the first one to be completed, which happened to be the interviewee's) informed our error categories for the full dataset. After the fact, the interviewee expressed regret at not having had the pilot be longer, so that the categories would have fit better. It is then an interesting question as to how much having had a more data-fitted error taxonomy would have shaped the conclusions drawn from the data.

Criticism of our instrument layout was raised by the interviewee. We created a form using the RedCap platform. For every sentence, we had the same list of elements from top to bottom: the original sentence, a table of terms and their expected translations, a reference translation and the system translation followed by our two subjective scales and our checkboxes. This layout was noted to be so tall that it required scrolling within a sentence and thus for the longer sentences it started to indirectly measure annotator working memory. After the annotation was well-underway, we noticed another issue: ideally, we would hope to measure fluency (and potentially even meaning preservation) blind to the wanted terms. Our instrument could have been designed such that the annotator locks in both subjective scales before seeing the terms.

We followed up on the observations about the error taxonomy by asking about particular cases where the error taxonomy did not fit. They had independently decided to collect screenshots of such cases during the annotation work, which they shared with us. For this, we are extremely grateful. The annotator stated that they would have wanted specific categories for:

- A term that is made up of multiple words where the first word is correct but subsequent words are reasonable synonyms not included in the target-language alternatives. The interviewee was not certain of the error category they se-

#	Native	MT Researcher	Translator	Language s.	Translation s.	NLP s.
1	Finnish	No	No	Non Fi-En	No	No
2	Finnish	No	Yes	English	Yes	Yes (not MT)
3	Finnish & English	No	No	No	No	No*

Table 5: Background information of our human experts. Columns ending in s. indicate higher education studies in a domain. (We discount obligatory English and Swedish courses as well as Finnish academic writing courses.) *Annotator 3 highlighted NLP-adjacent experience from their study focus in string processing algorithms, formal grammars and neural networks.

lected in these cases. We would expect these cases to be a missing error.

- A term is translated from the source version word by word (the example given was in *Brands* where this type of translation is particularly unacceptable). The interviewee argued that it is not truly missing, though that was the category they selected.
- The term is translated by a word of completely unrelated meaning that is a couple of character changes away from the right term, yet it is a real word and not truly a misspelling. Again, the interviewee argued that it is not truly missing, though that was the category they selected.
- The term is translated by another term in the same domain, which is actively misleading ("valtioneuvoston periaatepäätös" to "hallituksen päätöslauselma").

They also noted cases where term recogntition – that is, the step of the translation pipeline before the constraint translation system – had found phantom terms.

We include a RedCap item for reference as Figure 19.

C LLM-as-a-judge prompt

Below is an example prompt given to GLM-5 for automatic evaluation.

```
Arvioi seuraava konekäännös annetuilla
kriteereillä. Huomaa, että termeillä
viitataan spesifisti annetun taulukon
termeihin.

Alkukielinen virke:
decision which allows a municipality to
deviate for a special reason from a
provision, regulation, prohibition or
other restriction applying to
construction or other action.

Termit:
| Lähdekielinen termi | Halutut
käännökset |
|-|-|
| decision | ratkaisu *tai* päätös |
| construction | rakennettu kohde *tai*
rakennettava kohde |

Referenssikäännös:
päätös, jolla kunta voi erityisestä
syystä poiketa rakentamista tai muuta
toimenpidettä koskevasta säännöksestä,
määräyksestä, kiellosta tai muusta
rajoituksesta.

Järjestelmän tuottama käännös:
Ratkaisu/päätös, joka antaa kunnalle
mahdollisuuden poiketa erityisestä
syystä rakentamista tai muuta
toimenpidettä koskevasta säännöksestä,
määräyksestä, kiellosta tai muusta
rajoituksesta.

Vastaa seuraaviin kysymyksiin:

- Sujuvuus [1-10]: Onko käännös yhtä
luontevaa ja sujuvaa kohdekieltä kuin
referenssikäännös? (Sitä, vastaako
järjestelmän käännös lähtökielistä
tekstiä arvioidaan erikseen)
- Käännöslaatu [1-10]: Subjektiivinen
kokonaisarvio: Miten hyvä järjestelmän
käännös oli mielestäsi lähdetekstin
käännöksenä?
- Termeistä: Nämä kysymykset pohtivat
ainoastaan termien oikeaa käyttöä;
termeihin liittymättömien virheiden
tulisi heijastua yllä olevaan
yleisarvioon.
  - Järjetön [true/false]: Käännös on
niin järjetön, että termien
oikeellisuutta ei voi arvioida.
  - Oikein [true/false]: Termit täysin
oikein.
  - Puuttuu [true/false]: Vähintään yksi
 termi puuttuu kokonaan käännöksestä
  - Kääntämättä [true/false]: Vähintään
yksi termi jäi alkuperäisen kieliseksi (
ts. jäi kääntämättä).
  - Kirjoitusvirhe [true/false]:
Vähintään yhdessä termissä on
kirjoitusvirhe (pl. väärä sija). Jos
termi on osa yhdyssanaa, yhdyssanan
muussa osassa oleva virhe lasketaan
tähän kategoriaan.
  - Merkitysvirhe [true/false]:
Vähintään yhden termin väärä sijamuoto
tai vaihtunut paikka sanajärjestyksessä
muuttaa tekstin merkitystä.

Vastaa JSON-oliolla, joka on muotoa
```

Alkukielinen virke:

obligation of the actors of a construction project to cooperate in order to improve the quality of the construction work and to create the conditions for high quality implementation of the construction project.

Termit:

Lähdekielinen termi	Halutut käännökset
project	**projekti**
construction	**rakennettu kohde**
construction work	**rakennustyö**
construction	**rakennettu kohde**
project	**projekti**

Referenssikäännös:

rakentamishankkeen osapuolten velvollisuus tehdä yhteistyötä rakentamisen laadun parantamiseksi ja rakentamishankkeen laadukkaan toteuttamisen edellytyksien luomiseksi.

Järjestelmän tuottama käännös:

rakennetun kohteen projektin toimijoiden velvollisuus tehdä yhteistyötä rakennustyön laadun parantamiseksi ja edellytysten luomiseksi rakennetun kohteen projektin laadukkaalle toteuttamiselle.

Onko käännös yhtä luontevaa ja sujuvaa kohdekieltä kuin referenssikäännös? (Sitä, vastaako järjestelmän käännös lähtökielistä tekstiä arvioidaan erikseen)

* must provide value

Huono Hyvä

Change the slider above to set a response

reset

Subjektiivinen kokonaisarvio: Miten hyvä järjestelmän käännös oli mielestäsi lähdetekstin käännöksenä?

* must provide value

Huono Hyvä

Change the slider above to set a response

reset

Termeistä: Nämä kysymykset pohtivat ainoastaan termien oikeaa käyttöä; termeihin liittymättömien virheiden tulisi heijastua yllä olevaan yleisarvioon.

* must provide value

- ☐ Käännös on niin järjestön, että termien oikeellisuutta ei voi arvioida.
- ☐ Termit täysin oikein.
- ☐ Vähintään yksi termi puuttuu kokonaan käännöksestä.
- ☐ Vähintään yksi termi jäi alkuperäisen kieliseksi (ts. jäi kääntämättä).
- ☐ Vähintään yhdessä termissä on kirjoitusvirhe (pl. väärä sija). Jos termi on osa yhdyssanaa, yhdyssanan muussa osassa oleva virhe lasketaan tähän kategoriaan.
- ☐ Vähintään yhden termin väärä sijamuoto tai vaihtunut paikka sanajärjestyksessä muuttaa tekstin merkitystä.

Figure 19: The setup of our RedCap items.

```
{
    "Sujuvuus": 0,
    "Käännöslaatu": 0,
    "Järjetön": false,
    "Oikein": false,
    "Puuttuu": false,
    "Kääntämättä": false,
    "Kirjoitusvirhe": false,
    "Merkitysvirhe": false
}
```

D Inter-class agreement

We also computed κ values between the different metrics (Tables 6 and 7). This mostly goes to show that the annotators reported multiple error categories at less-than-chance probabilities. A particularly interesting element of the tables is the κ values (and counts) that intersect with the `correct` label. Annotator 3 never gave any negative features together with the `correct` label. Annotator 1 gave `misuse`, `nonsense`, `untranslated` and `missing` labels along with `correct` and annotator 2 gave `misuse` and `misspelling` labels. Our intention was that the `nonsense` and `correct` labels would never be used together with other labels, but annotator 3 was the only one to follow this intention.

E p-values for directed graphs

For the sake of completeness and reproducibility, we include tabular versions of the significance graphs for the human evaluations, COMET and chrF2 on the full human evaluation dataset. These include both a measure of magnitude of difference between the systems and the p-value computed with the binomial sign test and the H_0 of the comparisons between matching values coming from a fair (two-sided: we keep only discordant pairs) coin. We do not highlight any values as significant, as we draw no more conclusions from these tables than what we have drawn in the form of graphs in the main matter with our pre-set α threshold.

	misuse	nonsense	misspelling	untranslated	correct	missing
misuse		-0.08 (3)	-0.01 (2)	-0.07 (4)	-0.29 (37)	-0.08 (10)
nonsense	0.04 (31)		-0.03 (0)	-0.01 (2)	-0.11 (5)	-0.1 (0)
misspelling	-0.05 (5)	-0.01 (4)		0.01 (1)	-0.04 (0)	-0.04 (0)
untranslated	-0.04 (7)	0.01 (6)	-0.03 (1)		-0.13 (2)	-0.05 (2)
correct	-0.37 (28)	-0.32 (0)	-0.1 (1)	-0.13 (0)		-0.25 (1)
missing	-0.15 (7)	-0.11 (4)	0.01 (4)	-0.07 (1)	-0.25 (0)	

Table 6: Cohen's κ and number of coincident sentences of binary ratings for experts 1 and 2. Top right corner: expert 1; Bottom left corner: expert 2

	misuse	nonsense	misspelling	untranslated	correct	missing
misuse		-0.08 (0)	-0.04 (6)	-0.0 (6)	-0.37 (0)	-0.12 (3)
nonsense			-0.06 (0)	-0.06 (0)	-0.09 (0)	-0.07 (0)
misspelling				-0.06 (1)	-0.19 (0)	-0.03 (4)
untranslated					-0.13 (0)	-0.04 (2)
correct						-0.24 (0)
missing						

Table 7: Cohen's κ and number of coincident sentences of binary ratings for expert 3.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		0.2791	12.0037	-0.1042	8.8855
Gemma	0.3584		11.7246	-0.3833	8.6064
Backtranslation + CBS	0.0000	0.0000		-12.1079	-3.1182
Term training	0.4018	0.8392	0.0000		8.9897
Term training + CBS	0.0000	0.0000	0.1074	0.0000	

Table 8: COMET: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-98.5354	617.8049	-199.2881	593.9801
Gemma	0.8392		716.3404	-100.7527	692.5156
Backtranslation + CBS	0.0120	0.0009		-817.0931	-23.8248
Term training	0.1174	0.3634	0.0001		793.2683
Term training + CBS	0.0000	0.0002	0.9204	0.0000	

Table 9: chrF2: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		439.0000	3000.0000	183.0000	2349.0000
Gemma	0.3634		2561.0000	-256.0000	1910.0000
Backtranslation + CBS	0.0000	0.0000		-2817.0000	-651.0000
Term training	0.1293	0.5467	0.0000		2166.0000
Term training + CBS	0.0000	0.0000	0.1505	0.0000	

Table 10: `fluency`, human expert 1: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		1039.0000	2979.0000	155.0000	2908.0000
Gemma	0.0073		1940.0000	-884.0000	1869.0000
Backtranslation + CBS	0.0000	0.0000		-2824.0000	-71.0000
Term training	0.5296	0.0446	0.0000		2753.0000
Term training + CBS	0.0000	0.0001	0.2717	0.0000	

Table 11: `fluency`, human expert 2: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		340.0000	3906.0000	-2.0000	2471.0000
Gemma	0.5426		3566.0000	-342.0000	2131.0000
Backtranslation + CBS	0.0000	0.0000		-3908.0000	-1435.0000
Term training	0.4215	0.8392	0.0000		2473.0000
Term training + CBS	0.0000	0.0000	0.0000	0.0000	

Table 12: `fluency`, human expert 3: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		192.6000	3866.4000	181.5000	3117.1000
Gemma	0.5426		3673.8000	-11.1000	2924.5000
Backtranslation + CBS	0.0000	0.0000		-3684.9000	-749.3000
Term training	0.9196	0.9196	0.0000		2935.6000
Term training + CBS	0.0000	0.0000	0.0002	0.0000	

Table 13: `fluency`, LLM-as-a-judge: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-2.0000	-24.0000	-8.0000	-25.0000
Gemma	0.2060		-22.0000	-6.0000	-23.0000
Backtranslation + CBS	0.0000	0.0001		16.0000	-1.0000
Term training	0.0422	0.0904	0.0035		-17.0000
Term training + CBS	0.0000	0.0002	0.2500	0.0019	

Table 14: `misuse`, human expert 1: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-24.0000	-43.0000	-20.0000	-50.0000
Gemma	0.0000		-19.0000	4.0000	-26.0000
Backtranslation + CBS	0.0000	0.0014		23.0000	-7.0000
Term training	0.0000	0.1519	0.0005		-30.0000
Term training + CBS	0.0000	0.0001	0.1003	0.0000	

Table 15: `misuse`, human expert 2: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-4.0000	-27.0000	-6.0000	-29.0000
Gemma	0.1060		-23.0000	-2.0000	-25.0000
Backtranslation + CBS	0.0000	0.0000		21.0000	-2.0000
Term training	0.0525	0.2036	0.0001		-23.0000
Term training + CBS	0.0000	0.0000	0.2201	0.0001	

Table 16: `misuse`, human expert 3: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		4.0000	-29.0000	-6.0000	-19.0000
Gemma	0.1544		-33.0000	-10.0000	-23.0000
Backtranslation + CBS	0.0000	0.0000		23.0000	10.0000
Term training	0.1044	0.0331	0.0000		-13.0000
Term training + CBS	0.0027	0.0002	0.0331	0.0048	

Table 17: `misuse`, LLM-as-a-judge: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-1.0000	-20.0000	-1.0000	-13.0000
Gemma	0.2500		-19.0000	0.0000	-12.0000
Backtranslation + CBS	0.0000	0.0000		19.0000	7.0000
Term training	0.2500	0.3750	0.0000		-12.0000
Term training + CBS	0.0001	0.0005	0.0525	0.0001	

Table 18: `nonsense`, human expert 1: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		0.0000	-39.0000	-1.0000	-28.0000
Gemma	0.3281		-39.0000	-1.0000	-28.0000
Backtranslation + CBS	0.0000	0.0000		38.0000	11.0000
Term training	0.2500	0.2500	0.0000		-27.0000
Term training + CBS	0.0000	0.0000	0.0200	0.0000	

Table 19: `nonsense`, human expert 2: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		0.0000	-16.0000	0.0000	-6.0000
Gemma	∞		-16.0000	0.0000	-6.0000
Backtranslation + CBS	0.0000	0.0000		16.0000	10.0000
Term training	∞	∞	0.0000		-6.0000
Term training + CBS	0.0078	0.0078	0.0053	0.0078	

Table 20: `nonsense`, human expert 3: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		0.0000	-18.0000	0.0000	-9.0000
Gemma	∞		-18.0000	0.0000	-9.0000
Backtranslation + CBS	0.0000	0.0000		18.0000	9.0000
Term training	∞	∞	0.0000		-9.0000
Term training + CBS	0.0010	0.0010	0.0123	0.0010	

Table 21: `nonsense`, LLM-as-a-judge: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		7.0000	16.0000	16.0000	16.0000
Gemma	0.0098		9.0000	9.0000	9.0000
Backtranslation + CBS	0.0000	0.0029		0.0000	0.0000
Term training	0.0000	0.0010	0.3438		0.0000
Term training + CBS	0.0000	0.0029	0.3438	0.3438	

Table 22: `untranslated`, human expert 1: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		6.0000	15.0000	15.0000	16.0000
Gemma	0.0176		9.0000	9.0000	10.0000
Backtranslation + CBS	0.0000	0.0029		0.0000	1.0000
Term training	0.0000	0.0010	0.3438		1.0000
Term training + CBS	0.0000	0.0016	0.2500	0.2500	

Table 23: `untranslated`, human expert 2: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		6.0000	15.0000	15.0000	16.0000
Gemma	0.0176		9.0000	9.0000	10.0000
Backtranslation + CBS	0.0000	0.0029		0.0000	1.0000
Term training	0.0000	0.0010	0.3438		1.0000
Term training + CBS	0.0000	0.0016	0.2500	0.2500	

Table 24: `untranslated`, human expert 3: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		8.0000	16.0000	17.0000	18.0000
Gemma	0.0054		8.0000	9.0000	10.0000
Backtranslation + CBS	0.0000	0.0096		1.0000	2.0000
Term training	0.0000	0.0010	0.2500		1.0000
Term training + CBS	0.0000	0.0016	0.1562	0.2500	

Table 25: `untranslated`, LLM-as-a-judge: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		1.0000	0.0000	-1.0000	0.0000
Gemma	0.2500		-1.0000	-2.0000	-1.0000
Backtranslation + CBS	0.3438	0.2500		-1.0000	0.0000
Term training	0.2500	0.1562	0.2500		1.0000
Term training + CBS	0.3438	0.2500	0.3438	0.2500	

Table 26: `misspelling`, human expert 1: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-11.0000	-1.0000	-3.0000	-3.0000
Gemma	0.0018		10.0000	8.0000	8.0000
Backtranslation + CBS	0.2500	0.0032		-2.0000	-2.0000
Term training	0.1133	0.0241	0.1816		0.0000
Term training + CBS	0.1133	0.0192	0.1816	0.3281	

Table 27: `misspelling`, human expert 2: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-4.0000	5.0000	-3.0000	5.0000
Gemma	0.1202		9.0000	1.0000	9.0000
Backtranslation + CBS	0.0667	0.0123		-8.0000	0.0000
Term training	0.1619	0.2500	0.0192		8.0000
Term training + CBS	0.0754	0.0123	0.3184	0.0192	

Table 28: `misspelling`, human expert 3: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		1.0000	-2.0000	-3.0000	-4.0000
Gemma	0.2500		-3.0000	-4.0000	-5.0000
Backtranslation + CBS	0.1562	0.0625		-1.0000	-2.0000
Term training	0.0938	0.0312	0.2500		-1.0000
Term training + CBS	0.0547	0.0156	0.1719	0.2500	

Table 29: `misspelling`, LLM-as-a-judge: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-43.0000	-9.0000	-33.0000	-13.0000
Gemma	0.0000		34.0000	10.0000	30.0000
Backtranslation + CBS	0.0439	0.0000		-24.0000	-4.0000
Term training	0.0000	0.0275	0.0006		20.0000
Term training + CBS	0.0165	0.0000	0.1610	0.0006	

Table 30: `correct`, human expert 1: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-20.0000	13.0000	-26.0000	4.0000
Gemma	0.0004		33.0000	-6.0000	24.0000
Backtranslation + CBS	0.0088	0.0000		-39.0000	-9.0000
Term training	0.0000	0.1102	0.0000		30.0000
Term training + CBS	0.1610	0.0002	0.0469	0.0000	

Table 31: `correct`, human expert 2: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-26.0000	-3.0000	-29.0000	-16.0000
Gemma	0.0000		23.0000	-3.0000	10.0000
Backtranslation + CBS	0.1840	0.0002		-26.0000	-13.0000
Term training	0.0000	0.1857	0.0001		13.0000
Term training + CBS	0.0082	0.0484	0.0149	0.0181	

Table 32: `correct`, human expert 3: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-14.0000	16.0000	-10.0000	7.0000
Gemma	0.0040		30.0000	4.0000	21.0000
Backtranslation + CBS	0.0006	0.0000		-26.0000	-9.0000
Term training	0.0275	0.1519	0.0000		17.0000
Term training + CBS	0.0741	0.0001	0.0196	0.0008	

Table 33: `correct`, LLM-as-a-judge: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		33.0000	25.0000	23.0000	31.0000
Gemma	0.0000		-8.0000	-10.0000	-2.0000
Backtranslation + CBS	0.0000	0.0096		-2.0000	6.0000
Term training	0.0000	0.0032	0.2060		8.0000
Term training + CBS	0.0000	0.1719	0.0365	0.0143	

Table 34: `missing`, human expert 1: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		39.0000	34.0000	27.0000	40.0000
Gemma	0.0000		-5.0000	-12.0000	1.0000
Backtranslation + CBS	0.0000	0.0449		-7.0000	6.0000
Term training	0.0000	0.0010	0.0296		13.0000
Term training + CBS	0.0000	0.2500	0.0176	0.0002	

Table 35: `missing`, human expert 2: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		28.0000	27.0000	24.0000	32.0000
Gemma	0.0000		-1.0000	-4.0000	4.0000
Backtranslation + CBS	0.0000	0.2500		-3.0000	5.0000
Term training	0.0000	0.1060	0.1372		8.0000
Term training + CBS	0.0000	0.0547	0.0312	0.0096	

Table 36: `missing`, human expert 3: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		2.0000	-7.0000	1.0000	-4.0000
Gemma	0.1562		-9.0000	-1.0000	-6.0000
Backtranslation + CBS	0.0231	0.0029		8.0000	3.0000
Term training	0.2500	0.2500	0.0054		-5.0000
Term training + CBS	0.0859	0.0176	0.1372	0.0312	

Table 37: `missing`, LLM-as-a-judge: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-179.0000	2808.0000	-49.0000	2324.0000
Gemma	1.0000		2987.0000	130.0000	2503.0000
Backtranslation + CBS	0.0000	0.0000		-2857.0000	-484.0000
Term training	1.0000	0.3099	0.0000		2373.0000
Term training + CBS	0.0000	0.0000	0.2997	0.0000	

Table 38: `quality`, human expert 1: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		165.0000	1941.0000	-544.0000	1694.0000
Gemma	0.7310		1776.0000	-709.0000	1529.0000
Backtranslation + CBS	0.0000	0.0000		-2485.0000	-247.0000
Term training	0.1354	0.2604	0.0000		2238.0000
Term training + CBS	0.0000	0.0000	1.0000	0.0000	

Table 39: `quality`, human expert 2: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		156.0000	3353.0000	-253.0000	2754.0000
Gemma	0.4841		3197.0000	-409.0000	2598.0000
Backtranslation + CBS	0.0000	0.0000		-3606.0000	-599.0000
Term training	0.0854	0.4168	0.0000		3007.0000
Term training + CBS	0.0000	0.0000	0.1332	0.0000	

Table 40: `quality`, human expert 3: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.

	baseline	Gemma	Backtranslation + CBS	Term training	Term training + CBS
baseline		-210.1000	3192.8000	-110.5000	2522.7000
Gemma	0.6879		3402.9000	99.6000	2732.8000
Backtranslation + CBS	0.0000	0.0000		-3303.3000	-670.1000
Term training	1.0000	0.5467	0.0000		2633.2000
Term training + CBS	0.0000	0.0000	0.0024	0.0000	

Table 41: `quality`, LLM-as-a-judge: Top right corner: $\sum_i (y_i - x_i)$ where x_i and y_i are values of the column and row systems, respectively. Bottom left corner: p-value based on the sign test that x and y can be statistically distinguished.