Chain-of-Dictionary Prompting Elicits Translation in Large Language Models*

Hongyuan Lu[♡†], **Haoran Yang**[♡†], **Haoyang Huang**[♠]
Dongdong Zhang[♠], **Wai Lam**[♡], **Furu Wei**[♠]
[♡]The Chinese University of Hong Kong
[♠]Microsoft Corporation
{hylu,hryang,wlam}@se.cuhk.edu.hk
{haohua,dozhang,fuwei}@microsoft.com

Abstract

Large language models (LLMs) have shown surprisingly good performance in multilingual neural machine translation (MNMT) even if not being trained explicitly for translation. Yet, they still struggle with translating low-resource languages. As supported by our experiments, a bilingual dictionary between the source and the target language could help. Motivated by the fact that multilingual training effectively improves cross-lingual performance, we show that a chained multilingual dictionary with words expressed in more languages can provide more information to better enhance the LLM translation. To this end, we present a novel framework, COD, Chain-of-Dictionary Prompting, which augments LLMs with prior knowledge with the chains of multilingual dictionaries for a subset of input words to elicit translation abilities for LLMs. Experiments indicate that ChatGPT and InstructGPT still have room for improvement in translating many language pairs. And COD elicits large gains by up to 13x chrF++ points for MNMT (3.08 to 42.63 for English to Serbian written in Cyrillic script) on FLORES-200 full devtest set. We demonstrate the importance of chaining the multilingual dictionaries, as well as the superiority of COD to few-shot in-context learning for low-resource languages. Using COD helps ChatGPT to obviously surpass the SOTA translator NLLB 3.3B.[1]

1 Introduction

Large language models (LLMs) possess the ability to carry out high-quality machine translation tasks without specific training, as observed in previous studies (Brown et al., 2020; Lin et al., 2022; Le Scao et al., 2022; Zhang et al., 2022; Wang et al., 2023; Tang et al., 2024). LLMs can be prompted to do so by requesting them to complete a prompt, such as "Translate the following sentence to English from French:" followed by an input sentence written in French. However, despite their training on extensive datasets, these models may encounter difficulties in correctly translating rare words that frequently occur in low-resource situations.

Motivated by such a lexical-level problem, we seek how to incorporate dictionaries for improving MNMT. Further, motivated by the fact that multilingual training effectively improves cross-lingual performance (Liu et al., 2020; Lu et al., 2023, 2024), we use multilingual dictionaries to enhance the translation performance of LLM prompting.

To this end, we leverage the multilingual dictionaries as the prior knowledge, and we describe a method to prompt LLMs with hints that indicate a set of possible chained multilingual translations for specific words in the input. This method involves adding a string such as "'limit' means 'Grenze' means 'çäk'." to the start of the standard machine translation prompt as lexicon hints for MT. This approach is motivated by the fact that supervised machine translation models have effectively used dictionaries to enhance translation (Zhang and Zong, 2016; Arthur et al., 2016; Zheng et al., 2021). We also propose the method as a chain of dictionary in the light of Chain-of-Thought (CoT) reasoning (Wei et al., 2022) that represents the reasoning procedure as intermediate thinking steps. In our case, we show how to incorporate multilingual knowledge in a zero-shot manner by chaining the translations of words across various languages to improve LLM's MNMT capabilities. This allows us to specify the task in the prompt and provide background knowledge that is useful in completing the task of machine translation, without placing any strict constraints on how the model employs this knowledge, as demonstrated in Figure 1.

We conducted extensive experiments with the

*This research/paper was partially supported by the Center for Perceptual and Interactive Intelligence (CPII) Ltd. under the Innovation and Technology Commission's InnoHK scheme.

† Equal Contribution.

[1]Code and resources available at https://github.com/HongyuanLuke/Chain-of-Dictionary.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 958–976
November 12-16, 2024 ©2024 Association for Computational Linguistics

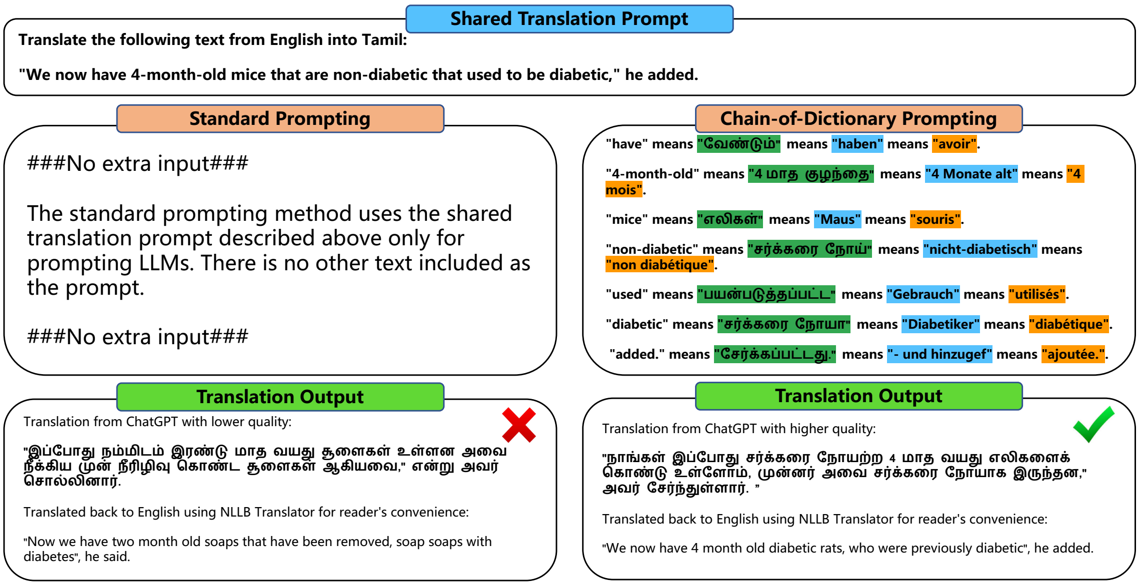

Figure 1: An illustration for CoD for English to Tamil translation. CoD consists of two sections: the standard translation prompt (the upper box) and the chained multilingual dictionaries. We highlight by languages the chained dictionary part for CoD, containing the words and their translations in different languages. CoD outperforms standard prompting in this example, and other methods such as the conventional Chain-of-Thought have been shown as less effective for MT (Peng et al., 2023). We bold the text for the actual inputs/outputs. Other non-bolded texts are placed for the explanation to the readers.

novel framework we propose, namely CoD (**C**hain-**o**f-**D**ictionary Prompting for Machine Translation), which achieved notable improvements in low-resource translation on FLORES-200 benchmarks (NLLB-Team, 2022) between English to almost all the other languages, using various language models. To gain a better understanding of CoD's capabilities, we analyzed and examined the model's behaviour by comparing it to both settings that incorporate bilingual dictionaries as well as separating the word mappings instead of chaining the multilingual dictionaries. CoD achieves the best empirical performance, which demonstrates its necessity in chaining the multilingual dictionary. Also, our experiments demonstrate that CoD achieves better performance than the standard few-shot demonstrations for low-resource languages. We speculate that the retrieved few-shot demonstrations are not relevant to the target translation, and therefore not particularly useful for low-resource translations. Our main contributions are three-fold:

- This paper proposes a novel framework called CoD (**C**hain-**o**f-**D**ictionary Prompting for Machine Translation) which adds chains of multilingual dictionaries to prompt LLMs that substantially improve machine translation.
- We conduct experiments on FLORES-200 for all translation directions between English and other languages. We observe that ChatGPT and InstructGPT still have room for improvement in translating many language pairs. We found that CoD can improve ChatGPT on a large portion of the languages, and can elicit translation in some languages that ChatGPT almost completely fails in translating.
- We observe that CoD can also be favourable to few-shot demonstrations, and CoD on ChatGPT can even surpass the SOTA translator NLLB 3.3B. We also verify that it is possible to save computation by truncating stopwords from the dictionary.

2 Chain-of-Dictionary Prompting for Neural Machine Translation

Large language models show their promising translation performance when sufficiently pre-trained (Lu et al., 2023; Wang et al., 2023). However, this is frequently not the case, especially for these low-resource languages. There are thousands of languages around the world, and current research on MT has scaled to at least 200 (NLLB-Team, 2022). It is an important research topic to explore the capabilities of LLMs to cover as many languages as possible. Despite the importance of covering low-resource languages in LLMs, we will report in this paper that the latest LLMs are still far from satisfying in covering these low-resource languages from FLORES-200 (NLLB-Team, 2022).

We propose a novel framework called CoD (**C**hain-**o**f-**D**ictionary Prompting) to address these difficulties by chaining multilingual dictionary knowledge into prompting-based machine translation. Compared to in-context learning that uses few-shot demonstrations to prompt the LLMs, dictionaries are comparatively easier to store and acquire than the demonstrations, particularly for low-resource languages (Zhang and Zong, 2016; Arthur et al., 2016; Hämäläinen and Alnajjar, 2020; Ghazvininejad et al., 2023). This makes CoD an attractive external resource for MT with LLMs.

Our novel approach, CoD, utilizes prompting-based translation and integrates chained multilingual dictionary information as prior knowledge directly into the prompt. When presented with a source sentence, we search for the multilingual dictionary entries for a subset of the words: before making the conventional translation request to LLMs, we append additional textual inputs to the prompt that outline possible chained multilingual translations for those specific words.

Therefore, the prompts for each sentence consist of two parts, as illustrated in Figure 1:

> (1) the translation prompt: *“Translate the following text from <source-language> into <target-language>: <source-sentence>”*.
> (2) the chained multilingual dictionaries: *“<word X in source-language> means <word X in target-language> means <word X in auxiliary-language 1> means <word X in auxiliary-language 2>.”*;

We do not include few-shot in-context learning in our methodology as we inspected that it is usually hard to retrieve relevant demonstrations for low-resource languages, which yields limited improvements. In the remaining sections, we will report relevant experimental results which indicate that few-shot demonstrations are less favourable to our methods for low-resource translations.

We also found that using non-chained decomposed multilingual dictionaries instead of CoD degrades the results:

“<word X in source-language> means <word X in target-language>. <word X in source-language> means <word X in auxiliary-language 1>. <word X in source-language> means <word X in auxiliary-language 2>.”[2]

We evaluate Machine Translation performance for all available languages using the LLM which we subsequently enhance with CoD. We then employ top languages that report the highest evaluation scores as our auxiliary languages to construct our multilingual dictionaries.

Multilingual Dictionary We propose to use the prompt “*Extract the words from the following texts: <input-sentence>*” to extract the keywords from the source language with LLMs such as ChatGPT. We then translate the extracted words into different languages with off-the-shelf MT models such as NLLB to create the dictionaries for CoD. During inference, the matched keywords and their translations are extracted from the dictionary to be appended to the translation prompt.

We use French (fra_Latn), German (deu_Latn), and Portuguese (por_Latn), three high-resource languages that our LLM performs well on, as our auxiliary languages for multilingual dictionaries. This means that we have a chain of 5 languages in the prompt, including the three auxiliary languages mentioned above and the source and the target language. We leave the exploration of further chaining to future work.

3 Experimental Setup

3.1 Baselines

We experiment with ChatGPT, a multilingual large language model that has shown strong abilities for the task of machine translation (Wang et al., 2023). At the time of writing, this LLM was widely popular. We experiment with ChatGPT to test CoD. We also conduct experiments on InstructGPT with the version of text-davinci-003 as well as BLOOM-7b (Le Scao et al., 2022):

- **GPT-3.5-TURBO** We use a ChatGPT model GPT-3.5-TURBO accessed via the official API through Python. All paired results are run within a week for fair comparison.
- **TEXT-DAVINCI-003** This is one of the InstructGPT models accessed via the official API provided by OpenAI through Python.

[2] We also attempted using different linking words such as “-” and “translates to” instead of “means”, where on-par performance is spotted. Also, note that keeping the dictionary word order to their order of appearance in the source sentence is important. Shuffling the word order can degrade the results.

- **BLOOM** BLOOM (Le Scao et al., 2022) is an open-sourced LLM trained in 46 natural languages. We use its 7B version as our baseline without any further tuning in this paper.
- **NLLB** NLLB (NLLB-Team, 2022) is an open-sourced SOTA translator. We use its 3.3B version as our baseline.

Based on the different versions of GPT models, we use the following prompting methods as the baselines to be compared:

- **Monolingual Dictionary**: This is a baseline that uses a monolingual dictionary that contains the words from the target language only.
- **Bilingual Dictionary**: This is a baseline that uses a bilingual dictionary for prompting large language models on the task of machine translation (Zhang and Zong, 2016; Arthur et al., 2016; Hämäläinen and Alnajjar, 2020; Ghazvininejad et al., 2023). It replaces the multilingual dictionaries in blue from Figure 1 with a bilingual dictionary built with the source language and the target language for the task of MT.
- **Decomposed Dictionary**: This is a baseline that removes the chaining of the dictionary and replaces the chained multilingual dictionaries in blue from Figure 1 with decomposed multilingual dictionaries. Refer to Section 2 for more details of this baseline model.
- **Few-shot Demonstration**: This is a baseline that does not use any dictionary. Instead, it retrieves from FLORES-200 devtest the top one/three translation pairs that are semantically similar to the current input translation, measured by BertScore (Zhang* et al., 2020) using the English sentences.

3.2 Datasets and Evaluation Metrics

For our evaluations on the task of machine translation for various languages including many low-resource languages, we use the dev-test division from FLORES-200 benchmarks (NLLB-Team, 2022), There are 1,012 sentences included in the dataset, which were extracted from English Wikipedia covering a variety of topics and domains. These sentences have been manually curated by professional translators into about 200 languages.

We report on all the languages in FLORES-200 for both directions from English and into English.

For the evaluation metrics, we report the chrF++ (Popović, 2015) and the BLEU (Papineni et al., 2002) evaluations provided by the sacreBLEU repository.[3] We use the model [eamt22-cometinho-da][4] for generating the COMET scores (Rei et al., 2020).

3.3 Dictionaries

To create the offline dictionaries used in our experiments, we first use the prompt "*Extract the words from the following texts: <input-sentence>*" to extract the keywords from the source language with LLMs such as ChatGPT. We then use the NLLB translator[5] to translate the monolingual English corpus from FLORES-200 into the remaining languages as our dictionaries. We excluded three languages which are not supported by the NLLB translator from our experiments. We use an off-the-shelf stopwords list for experiments on truncating stopwords to save computations with CoD.[6]

We use the English corpora from FLORES-200 to create our dictionary in this paper. For experiments on translating into English, we remove the English reference words from the dictionary to ensure there is no information leakage.

3.4 Dictionary Quality

With NLLB 3.3B, we translated the words into rare words with multiple attempts and translated them back into English. We then asked ChatGPT whether the translated-back version had the equivalent meaning to the original English. The process was done repeatedly until GPT reported that they were the same or the max tries (3 times) had been hit. In this manner, 71% of the words are successfully translated without hitting the max tries. For those failed translations, we exclude them from the dictionaries used by the bilingual chain or CoD.

3.5 Prompting Design

This section outlines the prompt design we opted for in creating the green text depicted in Figure 1.

Prior work compared various prompts for machine translation on LLM (Wang et al., 2023), and they have found similar performance of different prompts reported on a limited number of languages. They have opted for a basic prompt *"Translate the*

[3] https://github.com/mjpost/sacrebleu
[4] https://github.com/Unbabel/COMET
[5] https://huggingface.co/spaces/Narrativaai/NLLB-Translator
[6] https://gist.github.com/sebleier/554280

following text into <target-language>: <source-sentence>" as their best prompt. In contrast, our preliminary experiments show that removing the source language name can hurt the performance of translation. Therefore, we opted for *"Translate the following text from <source-language> into <target-language>: <source-sentence>"*.

Our preliminary experiments show that missing the keyword 'Tradition Script' for Chinese prompts the model to keep generating Simplified Chinese. Therefore, we specify the language script in our prompt when the languages can be written in different scripts and should be differentiated. For example, we write "Achinese with Arabic script" for the language "ace_Arab".

4 Results and Analysis

4.1 En-X Results

En-X: ChatGPT We firstly compare ChatGPT (GPT-3.5-TURBO) with the normal prompt in chrF++ on FLORES-200 with CoD. We plot the results in Figure 2 for better clarity. In Figure 2, we sort the chrF++ scores from ChatGPT in descending order, and we split the whole results into two figures. The upper figure represents the first half, and the bottom figure represents the second half. It can be observed in the bottom figure that ChatGPT does not handle the translation perfectly and it reports a score under 30 points in chrF++ for around 100 out of the 200 languages. The results indicate that CoD brings clear improvements. For space reasons, we leave Table 7 in the Appendix to present the detailed results for translating from English into the remaining languages. Table 11 in the Appendix also reports the detailed BLEU evaluations. Those results also indicate strong improvements with CoD. We speculate there are two reasons for improvement with CoD. Firstly, putting the desired translation target lexical shrinks the translation space and eases the translation. Secondly, using auxiliary languages in the chain gives better cross-lingual cues when there is no direct mapping between source and target lexical.

En-X: Languages Improved on ChatGPT Table 1 reports that more than 67% (135 out of 200) of the languages can be improved by CoD. For those languages that can be improved by CoD, more than 50% (71 out of 135) is improved by at least 5 points in chrF++. 13 languages can be improved by at least 10 points in chrF++ and 2 languages can be improved by at least 20 points in chrF++. We also observe quite strong results with CoD that bring 13x improvement (3.08 to 42.63) when translating from English into Serbian written in Cyrillic script. This leads to the conclusion that CoD gives promising results with good improvements in most languages and excellent improvements in several languages. CoD can even elicit translation in some languages that ChatGPT almost completely fails in translating, which is quite promising.

En-X: Languages Not Improved on ChatGPT As in Table 1, some languages are not benefited from CoD. We observe there are no languages with more than 20 points of decrease in chrF++ with CoD, and there are only 2 languages with more than 5 points of decrease in chrF++ with CoD. Compared to the languages with improvements reported above, the advantages of using CoD clearly outweigh the disadvantages when used indistinguishably regardless of the languages.

En-X: Languages Selection Though one could use CoD regardless of the languages, it will be better to use CoD only for those low-resource ones. This can be told visually from Figure 2 that CoD brings better improvements for the bottom figure that the baseline reports lower scores compared to the upper figure with higher baseline scores. The selection can be done with a threshold on the scores, and we observe that for those languages with a baseline score under 20 points in chrF++, CoD brings consistent improvements. We found using our universal list of high-resource auxiliary languages performs well and one can tune the list for specific languages for further improvements.[7]

En-X: COMET Scores We first obtain 99 languages out of the 200 languages from FLORES-200, which is supported by COMET (this list is obtained by matching the language names to the description in the official COMET repository)[8] Table 4 reports COMET scores, which aligns with our previous conclusion and indicates that CoD is effective. The average score of COMET is 0.325 for CoD, which is apparently higher than 0.277 from the baseline. We also found the same conclusion in the remaining 101 languages not perfectly

[7]We have found putting source and target language at the head of the chain empirically works well via early attempts. We empirically suggest to set the chain length as 5. Further increasing the length can further improve the information, while making the method less cost-effective.

[8]https://github.com/Unbabel/COMET

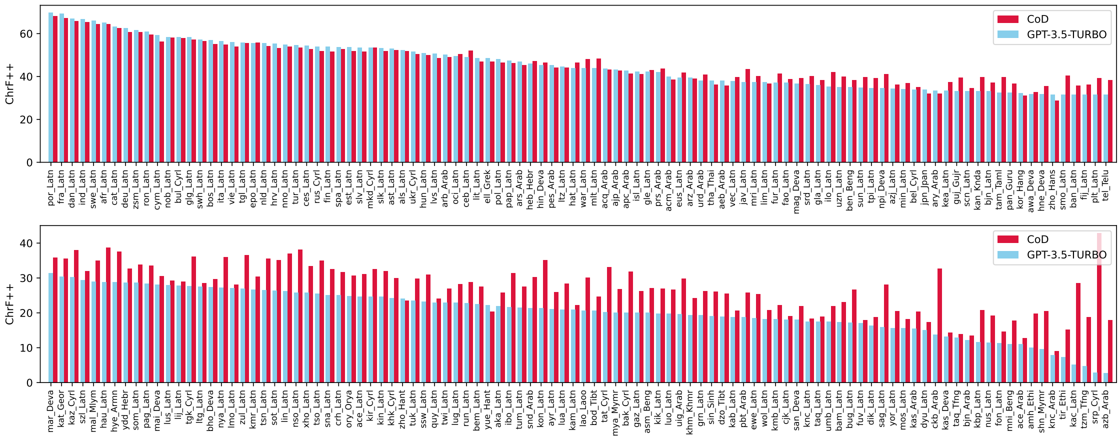

Figure 2: An illustrated comparison of 200 languages from English into the languages between the baseline ChatGPT (GPT-3.5-TURBO) and CoD. We sorted the language scores in chrF++ for ChatGPT in descending order, and we split the whole figure into two parts for clarity. We present the first half in the upper figure, and we present the second half in the bottom figure. CoD is effective for many languages, especially for low-resource ones.

Direction	# improved	> 5 points	> 10 points	> 20 points	# degraded	> 5 points	> 20 points
X-En	200/200	200/200	200/200	197/200	0/200	0/0	0/0
En-X	135/200	71/135	13/135	2/135	65/200	2/65	0/65

Table 1: Statistics of the changes in chrF++ with CoD on GPT-3.5-TURBO with 200 languages. 83.75% of the directions (335 out of 400) are improved. The advantage of CoD clearly outweighs the disadvantage.

supported by COMET. Since they are not perfectly supported, we do not report those languages here to avoid confusion.

4.2 X-En Results

X-En: ChatGPT In addition to the results for translation from English into other languages, we also use our multilingual dictionary for testing translation into English. Table 8 and Table 13 in the Appendix report the comparison between GPT-3.5-TURBO and CoD. We observe very good improvements in all languages when translating into English. We speculate that the underlying reason is that English is the major language used to pretrain GPT-3.5-TURBO. Dictionaries give hints to the model to produce better translation output by relying on the dictionary vocabulary and predicting the relationship between them. We also found that the translation capacity of ChatGPT can be non-symmetric, e.g., for umb_Latn, English translation reports a score of 17.41 in chrF++, while translating into English reports a score of 4.64 only.

X-En: BLOOM Table 3 reports results in chrF++ on BLOOM on 10 randomly selected low-resource languages translating into English. While the improvement is clear (e.g., from 7.05 to 12.50 on ckb_Arab), the improvement on BLOOM seems less significant than on ChatGPT. One reason could be that we are using a smaller model on BLOOM (7B). This can make the instruction less native to the LLMs as we do not do any instruction tuning or fine-tuning on BLOOM. We leave this to future work for further improvement.

Model	chrF++	BLEU
GPT-3.5	35.30	12.52
Monolingual Dictionary†	31.58	10.97
Bilingual Dictionary‡	36.37	12.63
Decomposed Dictionary	31.20	8.96
Few-shot ICL (1)	36.72	12.78
Few-shot ICL (3)	36.93	12.95
CoD (Partially Replaced I)	37.78	13.72
CoD (Partially Replaced II)	37.47	13.29
CoD (Chain 1)†	31.58	10.97
CoD (Chain 2)‡	36.37	11.06
CoD (Chain 3)	35.47	12.29
CoD (Chain 4)	37.90	13.90
CoD (Chain 5)	**38.27**	**13.90**

Table 2: Evaluations of CoD and various baselines on GPT-3.5 averaged from 200 languages. We report on translating from English into other languages. †,‡: the models are the same except for their different names.

Language	BLOOM	CoD	CoD w/o stopwords
srp_Cyrl	26.20	**39.26**	38.66
tzm_Ting	12.55	10.93	**13.12**
ckb_Arab	7.05	**12.50**	9.83
kon_Tatn	14.09	**17.03**	14.56
smo_Latn	13.80	15.09	**16.01**
uig_Arab	11.97	**14.86**	13.54
azb_Arab	12.42	**14.39**	12.50
amh_Ethi	13.12	**17.00**	16.82
nus_Latn	13.24	**14.70**	14.27
kac_Latn	13.25	**16.28**	14.73

Table 3: Evaluations in chrF++ of CoD on BLOOM in the direction of translating from other languages into English. We report results on 10 randomly selected low-resource languages on the FLORES-200 full devtest set.

Model	FLORES-200
GPT-3.5-TURBO	0.277
CoD	**0.325**

Table 4: Results of COMET scores for 99 supported languages on the FLORES-200 full devtest. We report on translating from English into other languages.

X-En on BLOOM: Save Computations via Removing Stopwords Table 3 truncate stopwords and reduces 4,978 dictionaries from the total of 15,074. The experiments are conducted on 10 randomly selected low-resource languages. The results in chrF++ indicate that such truncation can effectively save about 1/3 of the dictionary prompts, while still maintaining satisfying translation performance. While the original CoD shows better performance in most directions, removing stopwords can even occasionally surpass the original CoD, for example on tzm_Ting: CoD(10.93), removing stopwords (**13.12**). We postulate that it is hard for GPTs to translate even those stopwords for low-resource languages.

4.3 X-Y Results

X-Y: ChatGPT Table 10 compares CoD to GPT-3.5-TURBO on X-Y translations that we randomly select from the 30 languages as experiments with InstructGPT. The languages contain both higher-resourced and lower-resourced ones. CoD brings excellent improvements to 25/30 of the translations, by up to more than 10x improvements (1.33->14.48 in chrF++ scores for srp_Cyrl->kac_Latn).

Model	X-En	En-X
GPT-3.5-TURBO	44.98	33.22
NLLB	54.77	**43.39**
CoD	**66.12**	36.49

Table 5: Results of CoD (based on GPT-3.5-TURBO) compared to SOTA translator NLLB with chrF++ scores on 200 languages from FLORES-200 full devtest set.

Model	chrF++	BLEU
GPT-3.5	32.97	11.45
Ghazvininejad et al. (2023)	35.60	11.58
CoD	**36.30**	**12.01**

Table 6: Evaluations of CoD and various baselines on GPT-3.5 averaged from 200 languages. We report on translating from English into other languages.

4.4 Comparison to SOTA Translators

Table 5 reports the translation performance of CoD on both X-En and En-X directions. While NLLB surpasses CoD on EX, we observe that CoD can give a promising performance on X-En and even surpass the SOTA translator NLLB.[9]

4.5 Ablation Study

Table 2 reports the ablation study using GPT-3.5 that was accessed through the online GUI user interface. More details are in the Appendix A.

Multilingual Dictionary As in Table 2, using multilingual dictionaries from CoD instead of using a bilingual dictionary clearly improves the translation performance. Compared to using a bilingual dictionary that brings improvements of 1.07 chrF++ points to GPT-3.5, CoD brings a further improvement of 1.56 points in chrF++. This is more drastic on GPT-3.5-TURBO in Table 6, where bilingual dictionary (Ghazvininejad et al., 2023) clearly shows lower performance than CoD. In comparison, CoD effectively improves the BLEU score on the baseline from 11.45 to 12.01. Also as in Table 2, using a monolingual dictionary with target translation only can be harmful, and we suspect that it can confuse the model as there is no cross-lingual cue in the monolingual dictionary.

[9] We also found that using perfect English dictionaries on X-En improves CoD from 66.12 to 68.37. This means that our generated dictionaries are of good quality.

Source Sentence	With only eighteen medals available a day, a number of countries have failed to make the medal podium.
Standard GPT4 Prompt	Translate the following text from English into Kikongo with Latin script: {Source Sentence}
Bilingual Dictionary Prompt	"eighteen" means "kumi na nana". "medals" means "bamedayi". "available" means "kele na kati". "countries" means "bansi". "failed" means "me nunga ve". "podium" means "kisika ya lukumu". Translate the following text from English into Kikongo with Latin script: {Source Sentence}
CoD Prompt	"eighteen" means "kumi na nana" means "dix-huit" means "achtzehn" means "dezoito". "medals" means "bamedayi" means "médailles" means "Auszeichnungen" means "medalhas". "available" means "kele na kati" means "à disposition" means "verfügbar" means "disponíveis". "countries" means "bansi" means "pays" means "Länder" means "países". "failed" means "me nunga ve" means "échoué" means "Versagen" means "falhou". "podium" means "kisika ya lukumu" means "le podium" means "Podium" means "pódio". Translate the following text from English into Kikongo with Latin script: {Source Sentence}
Target Sentence	Na bidimbu kumi na nana yina ba ke lwatisaka muntu me nunga vandaka na kilumbu yina, bansi mingi lembanaka kuvanda na zulu ya nduatusulu ya bidimbu ya binungi...
Standard GPT4 Output BLEU=2.02, chrF++=40.59	Na pamba ve kuna banzété zómi na nane na mokolo moko, mabundu mingi mazangaka kokoma na podium ya banzété.
Bilingual Dictionary CoD Output BLEU=7.13, chrF++=55.01	Na me kele na "kumi na nana bamedayi" kele na kati ye kilumbu, diaka bansi ve me nunga ve kuyela kisika ya lukumu bamedayi.
CoD Output BLEU=10.62, chrF++=62.76	Na bamedayi kele na kati ya kumi na nana mosi kaka na kilumbu, bansi mingi me nunga ve kufika na kisika ya lukumu ya bamedayi.
Source Sentence	With only eighteen medals available a day, a number of countries have failed to make the medal podium.
Standard GPT4 Back BLEU=8.48, chrF++=48.84	With as many as eight trees in a day, many congregations missed the tree platform.
Bilingual CoD Back BLEU=13.34, chrF++=63.12	In the current "eighteen medals" there are and to date, no more countries have failed to progress to the medal rankings.
CoD Back BLEU=24.46, chrF++=69.80	With only 18 medals a day, most nations have failed to reach the medal podium.

Figure 3: A case study on translating from English into Kikongo with Latin script using GPT-4 throughout the cases. We evaluate the results on BLEU and chrF++. We highlight in green the words translated wrong by baselines but translated correctly by CoD, even if the words are not presented in the multilingual dictionary chains.

Chained Dictionary Removing chained dictionaries and using non-chained dictionaries that flatten all the dictionaries clearly deteriorates the translation results. We postulate that one reason is that a flattened dictionary introduces repeated source language text as redundant information, which can degrade the results. This claim aligns with the fact in Shi et al. (2023) that LLMs can be easily distracted by irrelevant context. Reducing the chaining length (CoD (Chain 1, 2, 3, 4)) also drops the performance. We kindly note that our goal is rather research-oriented. We leave longer chaining and more choices of chained languages to future work, which might yield better performance.

Few-shot In-context Learning (ICL) Retrieving few-shot demonstrations for in-context learning instead of CoD for languages in FLORES-200 brings minor improvement. We postulate that the reason is the difficulty in understanding low-resource languages, and therefore the retrieved demonstrations are still not very useful to the desired translation. While increasing the number of demonstrations in the prompt can further boost the performance, the results are still not very promising, below CoD.

Selection of Auxiliary Languages Partially replacing the auxiliary language (CoD (Partially Replaced I, II)) to arbitrary other languages (for example, Arabic (arb_Arab) instead of high-resource German (deu_Latn)) drops the performance.[10] We should use more high-resource languages in the chain for better performance. We suspect that such high-resource languages yield stronger cross-lingual hints to be used for the translations.

4.6 Case Study

Figure 3 presents a case study demonstrating the powerfulness of CoD. The baseline output from GPT4 is almost lost about which topics are discussed in the sentence. Using a bilingual dictionary is useful, but the bilingual baseline is still lost about the detailed semantics. In comparison, CoD successfully provides a high-quality translation, scoring the best in BLEU and chrF++. We also highlight in green where the translation is successfully elicited by CoD, even if the words are not provided in the multilingual dictionary. We hypothesise that CoD provides richer context to the LLMs to translate relevant words in the source sentences, even if they are not directly presented by CoD. Figure 4 and Figure 5 demonstrate cases that show a similar phenomenon, and they are available in the Appendix, at the end of this paper.

5 Related Work

Neural Machine Translation via Prompting Language Models Limited research has been conducted on effective methods for prompting large language models in machine translation. The majority of existing research has concentrated on evaluating the translation capabilities of large language models, utilizing uncomplicated prompts such as 'Translate to language_name: text' (Brown et al., 2020; Lin et al., 2022; Le Scao et al., 2022; Zhang et al., 2022). Various prompt formats have been explored by the scholars (Reynolds and McDonell, 2021; Wang et al., 2023), whereas Garcia and Firat (2022) have examined the potential use of prompts for regulating the formality or specific dialect of the output. Furthermore, Agrawal et al. (2022) and Vilar et al. (2022) have focused on identifying appropriate in-context examples to improve machine translation quality with LLMs.

[10]We also found that using other languages that are similar to the target language, such as the languages written in the same script, can lead to an obvious drop in performance. We suspect that putting a similar language to the target language tends to produce those languages in the output. However, using high-resource language in Latin script as the auxiliary language does not suffer from such a problem.

Lexical-based Neural Machine Translation Our research is connected to the concept of lexical restrictions in MT, which can be categorized into either hard constraints (Hokamp and Liu, 2017; Post and Vilar, 2018) or soft constraints (Song et al., 2019; Dinu et al., 2019; Chen et al., 2021).

Also, several works have explored the use of dictionaries in supervised MT. Zhang and Zong (2016) improves NMT with a bilingual dictionary that includes less common or unseen words present in the bilingual training data. Arthur et al. (2016) enhances the translation of infrequent words by supplementing the system with discrete translation lexicons and utilizing the attention vector to select the pertinent lexical probabilities. Hämäläinen and Alnajjar (2020) uses a dictionary to generate synthetic parallel data to better train the NMT models. A previous work uses bilingual dictionaries to improve MT (Ghazvininejad et al., 2023).

CoD is one of the first applications of applying dictionaries on Machine Translation on LLMs. Note that this paper focuses on proving the effectiveness of applying a dictionary to LLMs rather than providing an actual dictionary to be used.

6 Conclusions

CoD is a novel framework that uses chained multilingual dictionaries when prompting large language models (LLMs) for MNMT. We evaluate ChatGPT, InstructGPT, and BLOOM on the FLORES-200 dataset for MNMT. We found that ChatGPT and InstructGPT still have room for improvement in translating many language pairs. CoD elicits large gains by up to 13x chrF++ points for MNMT (3.08 to 42.63 for English to Serbian written in Cyrillic script) on FLORES-200 full devtest set. We also verified the necessity of the chained multilingual dictionaries, and we found that both of them are quite important to CoD. CoD also outperforms few-shot demonstrations which struggle to retrieve relevant demonstrations for low-resource settings. CoD can even surpass the strong SOTA NLLB translator in translation. Extensive case studies demonstrate that CoD elicits translation even if the words are not directly presented by CoD. There are over 7,000 languages around the world, and CoD is the first work that scales the translation capability of LLMs to over 200 languages. We hope that CoD can help researchers to improve cross-lingual performance on neural models further.

Limitations

This paper presents an analysis of 200 languages only. However, there are more than thousands of languages around the world.

Although CoD can lead to a very slight degradation in translation performance for a small subset of languages, our experiments have shown that the impact is typically insignificant and can be probably simply due to randomness. Therefore, the practical usage of CoD remains unaffected.

While CoD brings by up to 1.8x inference time as found in our implementation, the inference time for actual LLM APIs can be down to milliseconds, so this is realistic to apply CoD to real products.

While CoD brings by up to 3x prompt length, many LLMs support very long input lengths, for example, 32K for GPT4. So this is realistic to apply CoD to real products. One can also save the tokens by prompting rare words only with CoD.

This work also does not directly compare to those ones that require fine-tuning on LLMs (Jiao et al., 2023) which requires error-guided data. Nevertheless, CoD is easy to use and does not require additional data. It is comparatively easy to curate good-quality dictionaries with off-the-shelf tools.

We also consider and focus on the task of Machine Translation, as it is one of the most fundamental NLG tasks.

Ethical Statement

We honour and support the EMNLP Code of Ethics. There is no ethical issue known to us. A well-known and widely used LLM is used in our work, which is subjected to generating offensive context. However, the above-mentioned issues are widely known to commonly exist for LLMs. Any content generated does not reflect the view of the authors.

References

Sweta Agrawal, Chunting Zhou, Mike Lewis, Luke Zettlemoyer, and Marjan Ghazvininejad. 2022. In-context Examples Selection for Machine Translation. *arXiv e-prints*, page arXiv:2212.02437.

Philip Arthur, Graham Neubig, and Satoshi Nakamura. 2016. Incorporating discrete translation lexicons into neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1557–1567, Austin, Texas. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Guanhua Chen, Yun Chen, Yong Wang, and Victor O. K. Li. 2021. Lexical-constraint-aware neural machine translation via data augmentation. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20.

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.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Xavier Garcia and Orhan Firat. 2022. Using natural language prompts for machine translation. *arXiv e-prints*, page arXiv:2202.11822.

Marjan Ghazvininejad, Hila Gonen, and Luke Zettlemoyer. 2023. Dictionary-based Phrase-level Prompting of Large Language Models for Machine Translation. *arXiv e-prints*, page arXiv:2302.07856.

Mika Hämäläinen and Khalid Alnajjar. 2020. A template based approach for training nmt for low-resource uralic languages - a pilot with finnish. In *Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence*, ACAI '19, page 520–525, New York, NY, USA. Association for Computing Machinery.

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.

Wenxiang Jiao, Jen-tse Huang, Wenxuan Wang, Zhiwei He, Tian Liang, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2023. ParroT: Translating during chat using large language models tuned with human translation and feedback. In *Findings of the Association* *for Computational Linguistics: EMNLP 2023*, pages 15009–15020, Singapore. Association for Computational Linguistics.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. 2022. BLOOM: A 176B-Parameter Open-Access Multilingual Language Model. *arXiv e-prints*, page arXiv:2211.05100.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. 2022. Few-shot learning with multilingual generative language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9019–9052, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Hongyuan Lu, Haoyang Huang, Shuming Ma, Dongdong Zhang, Wai Lam, Zhaochuan Gao, Anthony Aue, Arul Menezes, and Furu Wei. 2023. TRIP: Accelerating document-level multilingual pre-training via triangular document-level pre-training on parallel data triplets. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7845–7858, Singapore. Association for Computational Linguistics.

Hongyuan Lu, Haoyang Huang, Dongdong Zhang, Furu Wei, and Wai Lam. 2024. Revamping multilingual agreement bidirectionally via switched back-translation for multilingual neural machine translation. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 264–275, St. Julian's, Malta. Association for Computational Linguistics.

NLLB-Team. 2022. No language left behind: Scaling human-centered machine translation.

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.

Keqin Peng, Liang Ding, Qihuang Zhong, Li Shen, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. 2023. Towards Making the Most of ChatGPT for Machine Translation. *arXiv e-prints*, page arXiv:2303.13780.

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, New Orleans, Louisiana. Association for Computational Linguistics.

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

Laria Reynolds and Kyle McDonell. 2021. Prompt programming for large language models: Beyond the few-shot paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI EA '21, New York, NY, USA. Association for Computing Machinery.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed Chi, Nathanael Schärli, and Denny Zhou. 2023. Large Language Models Can Be Easily Distracted by Irrelevant Context. *arXiv e-prints*, page arXiv:2302.00093.

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. ACL.

Tianyi Tang, Hongyuan Lu, Yuchen Jiang, Haoyang Huang, Dongdong Zhang, Xin Zhao, Tom Kocmi, and Furu Wei. 2024. Not all metrics are guilty: Improving NLG evaluation by diversifying references. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6596–6610, Mexico City, Mexico. Association for Computational Linguistics.

David Vilar, Markus Freitag, Colin Cherry, Jiaming Luo, Viresh Ratnakar, and George Foster. 2022. Prompting PaLM for Translation: Assessing Strategies and Performance. *arXiv e-prints*, page arXiv:2211.09102.

Jiaan Wang, Yunlong Liang, Fandong Meng, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. 2023. Is ChatGPT a Good NLG Evaluator? A Preliminary Study. *arXiv e-prints*, page arXiv:2303.04048.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Jiajun Zhang and Chengqing Zong. 2016. Bridging Neural Machine Translation and Bilingual Dictionaries. *arXiv e-prints*, page arXiv:1610.07272.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: Open Pre-trained Transformer Language Models. *arXiv e-prints*, page arXiv:2205.01068.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Xin Zheng, Zhirui Zhang, Shujian Huang, Boxing Chen, Jun Xie, Weihua Luo, and Jiajun Chen. 2021. Non-parametric unsupervised domain adaptation for neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4234–4241, Punta Cana, Dominican Republic. Association for Computational Linguistics.

A More Experimental Details

For the ablation study with GPT-3.5, We manually tested 800 instances from the FLORES-200 dataset that covers all the languages. For the ablation study with GPT-3.5-TURBO, we report the full devset evaluations.

B Creating the Dictionary

Other tools such as FastAlign (Dyer et al., 2013) can also be used for word alignment in creating dictionaries with bilingual corpora.

C InstructGPT

Table 12 and Table 14 compare CoD against TEXT-DAVINCI-003 on 30 languages that we found CoD works well on ChatGPT from FLORES-200 full devtest set. The results indicate that CoD improves all of them on InstructGPT as well, with an average boost of 12.02 in chrF++ (from 18.99 to 31.01) and 2.61 in BLEU (from 3.73 to 6.34).

Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD
ace_Arab	10.96	**12.87**	ace_Latn	24.38	**30.94**	acm_Arab	**40.16**	38.37	acq_Arab	**43.46**	40.02	aeb_Arab	**38.16**	36.19
afr_Latn	**65.25**	64.71	ajp_Arab	**43.38**	42.47	aka_Latn	22.01	**25.69**	als_Latn	**52.60**	51.64	amh_Ethi	10.05	**19.93**
apc_Arab	**42.60**	41.24	arb_Arab	**49.85**	49.08	ars_Arab	**46.68**	45.13	ary_Arab	**33.53**	32.04	arz_Arab	**39.25**	38.77
asm_Beng	19.83	**27.11**	ast_Latn	**52.81**	52.52	awa_Deva	32.16	**32.47**	ayr_Latn	21.05	**25.76**	azb_Arab	2.96	**18.11**
azj_Latn	34.17	**36.65**	bak_Cyrl	20.15	**31.90**	bam_Latn	17.43	**23.02**	ban_Latn	31.68	**35.63**	bel_Cyrl	33.90	**35.00**
bem_Latn	22.63	**27.46**	ben_Beng	35.29	**38.08**	bho_Deva	27.98	**29.43**	bjn_Arab	12.06	**13.35**	bjn_Latn	32.88	**36.87**
bod_Tibt	20.70	**24.37**	bos_Latn	**56.10**	55.38	bug_Latn	17.62	**26.56**	bul_Cyrl	**58.23**	57.73	cat_Latn	**63.29**	62.19
ceb_Latn	48.75	**52.04**	ces_Latn	**54.79**	52.88	cjk_Latn	17.89	**19.17**	ckb_Arab	13.23	**32.63**	crh_Latn	24.79	**31.68**
cym_Latn	**59.53**	56.03	dan_Latn	**67.01**	66.12	deu_Latn	**62.42**	61.04	dik_Latn	16.12	**18.74**	dyu_Latn	14.90	**17.30**
dzo_Tibt	18.82	**25.29**	ell_Grek	**48.01**	46.85	epo_Latn	**55.88**	55.76	est_Latn	**53.50**	51.53	eus_Latn	39.71	**42.16**
ewe_Latn	18.44	**25.22**	fao_Latn	37.13	**39.11**	fij_Latn	31.55	**36.21**	fin_Latn	**53.83**	51.55	fon_Latn	11.26	**14.49**
fra_Latn	**68.09**	67.02	fur_Latn	37.32	**41.31**	fuv_Latn	16.94	**17.84**	gaz_Latn	20.03	**26.24**	gla_Latn	35.68	**38.53**
gle_Latn	42.38	**42.69**	glg_Latn	**58.48**	57.36	grn_Latn	19.40	**26.26**	guj_Gujr	33.56	**39.56**	hat_Latn	43.68	**46.34**
hau_Latn	29.14	**38.57**	heb_Hebr	46.52	**47.42**	hin_Deva	44.88	**47.07**	hne_Deva	32.00	**35.74**	hrv_Latn	**55.58**	53.36
hun_Latn	**50.92**	50.40	hye_Armn	28.80	**37.34**	ibo_Latn	21.43	**31.37**	ilo_Latn	32.32	**42.18**	ind_Latn	**66.67**	65.31
isl_Latn	**42.28**	25.52	ita_Latn	**56.40**	55.15	jav_Latn	37.89	**43.37**	jpn_Jpan	**33.95**	31.96	kab_Latn	18.76	**20.54**
kac_Latn	3.59	**28.07**	kam_Latn	20.79	**22.29**	kan_Knda	33.02	**39.36**	kas_Arab	15.16	**20.52**	kas_Deva	13.01	**14.30**
kat_Geor	30.22	**35.66**	kaz_Cyrl	29.99	**37.51**	kbp_Latn	11.65	**20.71**	kea_Latn	33.64	**37.30**	khk_Cyrl	24.14	**30.22**
khm_Khmr	19.20	**24.44**	kik_Latn	19.66	**26.86**	kin_Latn	24.31	**32.01**	kir_Cyrl	24.42	**32.38**	kmb_Latn	17.84	**22.10**
kmr_Latn	26.38	**30.71**	knc_Arab	7.76	**9.09**	knc_Latn	17.55	**18.37**	kon_Latn	21.12	**34.89**	kor_Hang	**31.61**	30.85
lao_Laoo	21.01	**30.04**	lij_Latn	27.68	**29.03**	lim_Latn	**37.21**	36.56	lin_Latn	26.48	**37.02**	lit_Latn	**48.34**	46.75
lmo_Latn	26.79	**27.75**	ltg_Latn	27.68	**28.34**	ltz_Latn	**44.50**	44.11	lua_Latn	20.85	**28.46**	lug_Latn	22.85	**28.04**
luo_Latn	14.50	**15.52**	lus_Latn	27.98	**28.59**	lvs_Latn	**50.52**	48.10	mag_Deva	36.66	**38.99**	mai_Deva	28.12	**30.54**
mal_Mlym	28.95	**35.13**	mar_Deva	30.67	**35.65**	min_Latn	34.26	**36.70**	mkd_Cyrl	52.97	**53.62**	mlt_Latn	43.76	**48.23**
mni_Beng	11.22	**17.95**	mos_Latn	15.57	**18.17**	mri_Latn	37.47	**40.11**	mya_Mymr	20.06	**26.94**	nld_Latn	**55.52**	54.00
nno_Latn	**54.85**	53.96	nob_Latn	**58.27**	58.07	npi_Deva	34.20	**40.68**	nso_Latn	25.53	**37.80**	nus_Latn	11.50	**18.95**
nya_Latn	27.11	**35.98**	oci_Latn	49.07	**50.73**	ory_Orya	24.47	**30.76**	pag_Latn	28.51	**33.59**	pan_Guru	32.29	**36.83**
pap_Latn	**47.51**	46.27	pbt_Arab	18.95	**25.67**	pes_Arab	**44.75**	44.69	plt_Latn	31.58	**39.04**	pol_Latn	**48.30**	46.51
por_Latn	**69.87**	68.18	prs_Arab	41.71	**43.96**	quy_Latn	23.08	**24.09**	ron_Latn	**60.75**	59.49	run_Latn	22.93	**28.56**
rus_Cyrl	**53.38**	52.39	sag_Latn	15.67	**27.96**	san_Deva	17.64	**22.01**	scn_Latn	33.27	**34.70**	shn_Mymr	9.72	**20.18**
sin_Sinh	19.02	**26.30**	slk_Latn	**53.49**	51.99	slv_Latn	**53.02**	51.45	smo_Latn	31.80	**40.43**	sna_Latn	24.90	**32.49**
snd_Arab	21.45	**30.11**	som_Latn	28.75	**33.95**	sot_Latn	26.57	**35.02**	spa_Latn	**53.91**	52.89	srd_Latn	35.88	**40.48**
srp_Cyrl	3.08	**42.63**	ssw_Latn	23.12	**31.02**	sun_Latn	34.90	**39.13**	swe_Latn	**66.50**	64.92	swh_Latn	**56.93**	56.66
szl_Latn	29.02	**31.95**	tam_Taml	32.30	**39.80**	taq_Latn	17.50	**18.66**	taq_Tfng	12.65	**13.84**	tat_Cyrl	20.10	**33.33**
tel_Telu	30.85	**38.26**	tgk_Cyrl	28.03	**36.01**	tgl_Latn	**55.45**	55.13	tha_Thai	**38.46**	36.51	tir_Ethi	7.34	**15.15**
tpi_Latn	34.45	**39.29**	tsn_Latn	26.84	**35.68**	tso_Latn	25.68	**34.71**	tuk_Latn	23.33	**29.74**	tum_Latn	21.51	**27.43**
tur_Latn	**54.46**	53.42	twi_Latn	22.84	**26.77**	tzm_Tfng	7.14	**18.56**	uig_Arab	19.50	**29.53**	ukr_Cyrl	**51.65**	50.10
umb_Latn	17.41	**22.03**	urd_Arab	37.86	**41.17**	uzn_Latn	35.22	**39.93**	vec_Latn	37.49	**39.77**	vie_Latn	**55.81**	42.21
war_Latn	43.93	**48.31**	wol_Latn	18.30	**20.76**	xho_Latn	25.43	**33.32**	ydd_Hebr	28.88	**32.58**	yor_Latn	15.51	**20.60**
yue_Hant	**22.36**	17.41	zho_Hans	**30.99**	28.92	zho_Hant	**23.83**	23.80	zsm_Latn	**61.85**	58.52	zul_Latn	27.03	**36.29**

Table 7: Comparison between GPT-3.5-TURBO and CoD. Results in chrF++ for MT on the FLORES-200 dataset. The best results are bolded and highlighted. We report on translating from English into the languages.

Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD
ace_Arab	3.43	**44.72**	ace_Latn	10.20	**57.40**	acm_Arab	28.57	**59.86**	acq_Arab	29.49	**61.15**	aeb_Arab	24.18	**54.56**
afr_Latn	53.42	**73.29**	ajp_Arab	33.14	**63.78**	aka_Latn	7.33	**46.52**	als_Latn	33.69	**63.26**	amh_Ethi	3.84	**50.16**
apc_Arab	30.26	**61.71**	arb_Arab	33.75	**63.22**	ars_Arab	31.83	**62.53**	ary_Arab	21.72	**53.29**	arz_Arab	25.74	**55.55**
asm_Beng	12.10	**52.60**	ast_Latn	36.38	**60.59**	awa_Deva	19.87	**54.48**	ayr_Latn	4.44	**42.24**	azb_Arab	8.61	**49.86**
azj_Latn	17.48	**46.86**	bak_Cyrl	8.87	**47.07**	bam_Latn	4.95	**48.20**	ban_Latn	17.35	**58.12**	bel_Cyrl	17.16	**41.73**
bem_Latn	7.58	**47.98**	ben_Beng	20.56	**59.26**	bho_Deva	15.54	**49.91**	bjn_Arab	4.06	**41.10**	bjn_Latn	19.08	**60.84**
bod_Tibt	2.18	**43.64**	bos_Latn	37.91	**63.31**	bug_Latn	7.41	**48.21**	bul_Cyrl	35.93	**63.12**	cat_Latn	42.26	**65.33**
ceb_Latn	31.97	**65.14**	ces_Latn	35.64	**60.83**	cjk_Latn	4.32	**41.62**	ckb_Arab	8.81	**57.24**	crh_Latn	18.42	**52.10**
cym_Latn	45.87	**73.44**	dan_Latn	45.04	**65.50**	deu_Latn	41.01	**61.28**	dik_Latn	5.21	**46.62**	dyu_Latn	4.01	**41.79**
dzo_Tibt	1.78	**43.47**	ell_Grek	30.18	**60.42**	epo_Latn	37.90	**62.61**	est_Latn	33.51	**59.36**	eus_Latn	21.30	**50.40**
ewe_Latn	4.63	**45.04**	fao_Latn	29.36	**61.53**	fij_Latn	9.26	**44.69**	fin_Latn	31.06	**56.56**	fon_Latn	3.69	**43.84**
fra_Latn	42.07	**63.68**	fur_Latn	29.46	**60.09**	fuv_Latn	4.84	**42.54**	gaz_Latn	4.30	**43.33**	gla_Latn	21.07	**55.88**
gle_Latn	28.45	**59.61**	glg_Latn	37.44	**61.50**	grn_Latn	7.48	**47.28**	guj_Gujr	20.13	**60.41**	hat_Latn	28.32	**62.44**
hau_Latn	10.06	**58.24**	heb_Hebr	29.36	**67.53**	hin_Deva	27.99	**61.85**	hne_Deva	18.04	**58.22**	hrv_Latn	34.31	**58.49**
hun_Latn	30.15	**57.97**	hye_Armn	16.00	**59.32**	ibo_Latn	6.84	**54.52**	ilo_Latn	17.23	**58.31**	ind_Latn	38.00	**67.27**
isl_Latn	28.22	**57.93**	ita_Latn	29.95	**52.02**	jav_Latn	22.75	**64.47**	jpn_Jpan	22.62	**49.73**	kab_Latn	4.46	**48.52**
kac_Latn	3.53	**39.22**	kam_Latn	6.45	**48.81**	kan_Knda	17.92	**56.25**	kas_Arab	7.43	**50.76**	kas_Deva	7.11	**44.15**
kat_Geor	12.32	**49.73**	kaz_Cyrl	15.20	**52.77**	kbp_Latn	3.98	**44.44**	kea_Latn	34.65	**68.33**	khk_Cyrl	9.36	**46.79**
khm_Khmr	10.19	**59.19**	kik_Latn	6.78	**50.63**	kin_Latn	12.75	**55.58**	kir_Cyrl	9.61	**44.01**	kmb_Latn	5.22	**42.84**
kmr_Latn	15.22	**53.58**	knc_Arab	1.55	**28.22**	knc_Latn	4.80	**42.19**	kon_Latn	5.85	**47.39**	kor_Hang	23.97	**57.30**
lao_Laoo	7.35	**60.86**	lij_Latn	29.21	**61.76**	lim_Latn	35.69	**64.23**	lin_Latn	8.34	**51.59**	lit_Latn	28.29	**54.88**
lmo_Latn	2.18	**3.75**	ltg_Latn	12.80	**55.21**	ltz_Latn	35.92	**66.06**	lua_Latn	6.48	**49.75**	lug_Latn	7.82	**52.45**
luo_Latn	4.48	**49.09**	lus_Latn	7.14	**39.55**	lvs_Latn	30.01	**57.89**	mag_Deva	21.45	**58.77**	mai_Deva	15.28	**56.73**
mal_Mlym	16.42	**55.04**	mar_Deva	18.08	**56.50**	min_Latn	17.00	**62.12**	mkd_Cyrl	36.50	**65.19**	mlt_Latn	38.20	**70.00**
mni_Beng	3.29	**40.55**	mos_Latn	3.98	**41.18**	mri_Latn	15.94	**53.64**	mya_Mymr	3.51	**47.27**	nld_Latn	28.24	**47.58**
nno_Latn	42.33	**62.62**	nob_Latn	39.54	**60.44**	npi_Deva	20.98	**59.29**	nso_Latn	11.05	**56.51**	nus_Latn	3.54	**48.61**
nya_Latn	12.30	**53.52**	oci_Latn	43.66	**70.67**	ory_Orya	14.66	**52.97**	pag_Latn	14.73	**48.91**	pan_Guru	21.73	**59.52**
pap_Latn	38.24	**68.25**	pbt_Arab	8.99	**52.05**	pes_Arab	29.11	**63.37**	plt_Latn	12.71	**55.42**	pol_Latn	25.91	**49.40**
por_Latn	45.35	**67.57**	prs_Arab	29.22	**63.77**	quy_Latn	5.18	**37.49**	ron_Latn	38.71	**62.48**	run_Latn	8.56	**49.75**
rus_Cyrl	31.51	**59.16**	sag_Latn	4.28	**43.93**	san_Deva	10.07	**48.64**	scn_Latn	29.06	**61.36**	shn_Mymr	4.19	**46.06**
sin_Sinh	4.41	**50.02**	slk_Latn	34.41	**60.61**	slv_Latn	32.00	**57.15**	smo_Latn	12.54	**55.08**	sna_Latn	10.18	**52.33**
snd_Arab	11.23	**55.49**	som_Latn	11.93	**56.17**	sot_Latn	10.65	**57.30**	spa_Latn	27.07	**50.01**	srd_Latn	28.68	**62.98**
srp_Cyrl	38.43	**66.65**	ssw_Latn	9.28	**52.91**	sun_Latn	20.93	**61.45**	swe_Latn	44.56	**67.92**	swh_Latn	36.04	**70.62**
szl_Latn	31.06	**63.08**	tam_Taml	13.15	**55.50**	taq_Latn	4.74	**38.96**	taq_Tfng	2.44	**50.04**	tat_Cyrl	10.53	**48.99**
tel_Telu	16.44	**55.97**	tgk_Cyrl	14.12	**55.12**	tgl_Latn	37.32	**67.56**	tha_Thai	20.02	**60.53**	tir_Ethi	2.49	**46.58**
tpi_Latn	16.97	**44.33**	tsn_Latn	9.47	**49.83**	tso_Latn	10.07	**52.51**	tuk_Latn	13.71	**50.86**	tum_Latn	7.23	**43.80**
tur_Latn	32.87	**61.14**	twi_Latn	8.00	**47.02**	tzm_Tfng	2.56	**52.38**	uig_Arab	7.88	**46.95**	ukr_Cyrl	34.80	**63.45**
umb_Latn	4.64	**41.97**	urd_Arab	22.46	**57.77**	uzn_Latn	17.58	**51.81**	vec_Latn	35.77	**64.54**	vie_Latn	28.84	**64.69**
war_Latn	31.13	**66.47**	wol_Latn	6.01	**47.45**	xho_Latn	14.35	**59.45**	ydd_Hebr	20.51	**70.76**	yor_Latn	7.86	**49.83**
yue_Hant	25.13	**53.60**	zho_Hans	23.39	**55.13**	zho_Hant	22.97	**51.96**	zsm_Latn	37.48	**67.78**	zul_Latn	14.43	**60.28**

Table 8: Comparison of CoD against GPT-3.5-TURBO. Results in chrF++ for MT on the FLORES-200 dataset. The best results are bolded and highlighted. We report on translating from the languages into English.

Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD
ace_Arab	-0.41	**-0.22**	ace_Latn	-0.97	**-0.52**	acm_Arab	**0.72**	0.67	acq_Arab	**0.65**	0.58
aeb_Arab	**0.70**	0.63	afr_Latn	**0.48**	0.37	ajp_Arab	-0.34	**-0.13**	aka_Latn	-0.72	**-0.71**
als_Latn	-0.48	**-0.32**	amh_Ethi	**0.66**	0.59	apc_Arab	0.28	**0.40**	arb_Arab	0.33	**0.38**
ars_Arab	**0.71**	0.65	ary_Arab	**0.85**	0.81	arz_Arab	**0.42**	0.32	asm_Beng	0.01	**0.21**
ast_Latn	-0.03	**0.05**	awa_Deva	-0.24	**-0.23**	ayr_Latn	-0.89	**-0.74**	azb_Arab	-0.03	**0.22**
azj_Latn	**0.62**	0.54	bak_Cyrl	-0.46	**-0.08**	bam_Latn	-1.46	**0.68**	ban_Latn	-0.60	**-0.51**
bel_Cyrl	**0.84**	0.80	bem_Latn	-0.67	**-0.49**	ben_Beng	-0.13	**0.02**	bho_Deva	-0.46	**-0.31**
bjn_Arab	-0.59	**-0.34**	bjn_Latn	-0.62	**-0.39**	bod_Tibt	-0.55	**-0.43**	bos_Latn	0.58	**0.60**
bug_Latn	-0.76	**-0.35**	bul_Cyrl	**0.70**	0.66	cat_Latn	-0.24	**-0.14**	ceb_Latn	**0.47**	0.42
ces_Latn	-0.64	**-0.34**	cjk_Latn	-0.31	**-0.12**	ckb_Arab	**0.72**	0.65	crh_Latn	-0.42	**-0.13**
cym_Latn	**0.78**	0.71	dan_Latn	-0.68	**-0.39**	deu_Latn	-0.44	**-0.17**	dik_Latn	0.32	**0.43**
dyu_Latn	-0.88	**-0.84**	dzo_Tibt	-0.34	**-0.32**	ell_Grek	**0.75**	0.68	epo_Latn	**0.84**	0.79
est_Latn	**0.90**	0.86	eus_Latn	**0.65**	0.59	ewe_Latn	-0.62	**-0.42**	fao_Latn	-0.31	**-0.29**
fij_Latn	-0.60	**-0.39**	fin_Latn	-0.42	**-0.20**	fon_Latn	-0.50	**-0.33**	fra_Latn	**0.52**	0.49
fur_Latn	-0.48	**-0.17**	fuv_Latn	-1.39	**-0.27**	gaz_Latn	-0.62	**-0.38**	gla_Latn	-0.24	**-0.01**
gle_Latn	0.31	**0.44**	glg_Latn	**0.75**	0.72	grn_Latn	**-0.28**	-0.29	guj_Gujr	**0.65**	0.62
hat_Latn	-0.95	**-0.59**	hau_Latn	**0.60**	0.57	heb_Hebr	-0.75	**-0.39**	hin_Deva	-1.21	**-1.00**
hne_Deva	-0.53	**-0.43**	hrv_Latn	**0.65**	0.63	hun_Latn	0.17	**0.30**	hye_Armn	-0.01	**0.20**
ibo_Latn	**-0.30**	-0.55	ilo_Latn	0.42	**0.42**	ind_Latn	**0.63**	0.52	isl_Latn	**0.60**	0.50
ita_Latn	-0.95	**-0.52**	jav_Latn	**0.69**	0.64	jpn_Jpan	-0.08	**-0.03**	kab_Latn	**-0.08**	-0.13
kac_Latn	0.08	**0.25**	kam_Latn	-0.60	**-0.50**	kan_Knda	-0.76	**-0.43**	kas_Arab	**0.20**	0.20
kas_Deva	-0.49	**-0.31**	kat_Geor	-0.19	**0.02**	kaz_Cyrl	0.02	**0.30**	kbp_Latn	-1.06	**-0.48**
kea_Latn	-0.67	**-0.32**	khk_Cyrl	-0.20	**0.05**	khm_Khmr	0.21	**0.40**	kik_Latn	-0.25	**-0.17**
kin_Latn	**-0.86**	-0.91	kir_Cyrl	-0.17	**-0.05**	kmb_Latn	-0.43	**-0.28**	kmr_Latn	**0.72**	0.64
knc_Arab	-0.35	**-0.25**	knc_Latn	0.00	**0.02**	kon_Latn	**-0.49**	-0.55	kor_Hang	-0.15	**0.04**
lao_Laoo	**0.71**	0.68	lij_Latn	-0.71	**-0.62**	lim_Latn	-0.57	**-0.47**	lin_Latn	-0.68	**-0.45**
lit_Latn	**0.41**	0.33	lmo_Latn	**-0.19**	-0.23	ltg_Latn	-0.38	**-0.36**	ltz_Latn	**-0.58**	-0.59
lua_Latn	-0.53	**-0.29**	lug_Latn	**-0.26**	-0.26	luo_Latn	-0.43	**-0.41**	lus_Latn	-0.62	**-0.29**
lvs_Latn	**0.81**	0.76	mag_Deva	-0.95	**-0.95**	mai_Deva	-1.49	**-1.45**	mal_Mlym	-0.31	**0.08**
mar_Deva	**0.75**	0.69	min_Latn	-0.87	**-0.72**	mkd_Cyrl	**0.77**	0.70	mlt_Latn	-0.25	**0.08**
mni_Beng	**0.51**	0.51	mos_Latn	-0.51	**-0.37**	mri_Latn	-1.01	**-0.49**	mya_Mymr	0.55	**0.55**
nld_Latn	-0.41	**-0.11**	nno_Latn	0.15	**0.32**	nob_Latn	**0.73**	0.69	npi_Deva	**0.16**	0.12
nso_Latn	-0.36	**-0.30**	nus_Latn	-1.11	**-0.09**	nya_Latn	-0.97	**-0.91**	oci_Latn	-0.89	**-0.67**
ory_Orya	-0.35	**-0.33**	pag_Latn	**-0.82**	-0.90	pan_Guru	-1.62	**-0.73**	pap_Latn	-0.66	**-0.57**
pbt_Arab	-0.43	**-0.38**	pes_Arab	**0.84**	0.79	plt_Latn	**0.79**	0.72	pol_Latn	0.33	**0.38**
por_Latn	**0.76**	0.70	prs_Arab	-0.19	**-0.05**	quy_Latn	-0.53	**-0.29**	ron_Latn	**0.67**	0.63
run_Latn	-0.11	**-0.08**	rus_Cyrl	**0.60**	0.53	sag_Latn	-0.03	**0.06**	san_Deva	**0.60**	0.56
scn_Latn	-0.66	**-0.54**	shn_Mymr	-0.89	**-0.86**	sin_Sinh	**0.82**	0.79	slk_Latn	**0.54**	0.43
slv_Latn	-0.69	**0.05**	smo_Latn	-0.34	**-0.28**	sna_Latn	-0.99	**-0.65**	snd_Arab	**0.81**	0.75
som_Latn	**0.77**	0.73	sot_Latn	-0.35	**-0.23**	spa_Latn	**0.74**	0.70	srd_Latn	-0.01	**-0.00**
srp_Cyrl	**0.81**	0.74	ssw_Latn	-0.69	**-0.53**	sun_Latn	0.30	**0.37**	swe_Latn	0.24	**0.25**
swh_Latn	0.39	**0.40**	szl_Latn	0.26	**0.37**	tam_Taml	-0.97	**-0.87**	taq_Latn	-1.07	**-1.03**
taq_Tfng	-0.82	**-0.76**	tat_Cyrl	-0.06	**0.05**	tel_Telu	-0.05	**0.16**	tgk_Cyrl	-0.81	**-0.80**
tgl_Latn	0.01	**0.07**	tha_Thai	**0.35**	0.33	tir_Ethi	-1.17	**-0.71**	tpi_Latn	**0.60**	0.50
tsn_Latn	-0.36	**-0.23**	tso_Latn	-1.13	**-0.95**	tuk_Latn	-0.22	**-0.21**	tum_Latn	-0.62	**-0.50**
tur_Latn	**0.03**	-0.12	twi_Latn	-0.48	**-0.32**	tzm_Tfng	-0.74	**-0.60**	uig_Arab	-0.41	**-0.10**
ukr_Cyrl	**0.52**	0.45	umb_Latn	-0.52	**-0.52**	urd_Arab	**0.60**	0.56	uzn_Latn	**0.38**	0.32
vec_Latn	**0.04**	-0.04	vie_Latn	-0.67	**-0.36**	war_Latn	**0.32**	0.26	wol_Latn	-0.64	**-0.60**
xho_Latn	**0.61**	0.57	ydd_Hebr	**0.39**	0.31	yor_Latn	-0.73	**-0.57**	yue_Hant	**0.69**	0.65
zho_Hans	**0.25**	0.12	zho_Hant	**0.47**	0.44	zsm_Latn	**0.30**	0.19	zul_Latn	-1.15	**-0.89**

Table 9: Comparison between GPT-3.5-TURBO and CoD. Results in COMET for MT on the FLORES-200 dataset. The best results are bolded and highlighted. We report on translating from English into the languages.

Direction	GPT	CoD	Direction	GPT	CoD	Direction	GPT	CoD	Direction	GPT	CoD	Direction	GPT	CoD
amh_Ethi->lao_Laoo	15.43	**16.40**	azb_Arab->tsn_Latn	20.68	**24.81**	bak_Cyrl->amh_Ethi	7.68	**10.72**	bug_Latn->tgk_Cyrl	15.41	**16.16**	ckb_Arab->tzm_Tfng	**8.68**	7.72
hau_Latn->kac_Latn	4.51	**11.69**	hye_Armn->tsn_Latn	22.56	**24.00**	ibo_Latn->hye_Armn	**16.74**	16.47	kac_Latn->srp_Cyrl	6.93	**11.55**	kbp_Latn->shn_Mymr	4.73	**6.99**
kir_Cyrl->bug_Latn	10.17	**14.10**	kon_Latn->srp_Cyrl	**5.01**	3.72	lao_Laoo->snd_Arab	**12.61**	7.80	lin_Latn->zul_Latn	21.35	**23.00**	nso_Latn->bug_Latn	10.65	**16.40**
nya_Latn->sag_Latn	15.57	**18.13**	plt_Latn->nso_Latn	23.60	**28.42**	sag_Latn->lin_Latn	21.70	**24.23**	shn_Mymr->amh_Ethi	4.39	**5.92**	smo_Latn->lao_Laoo	19.36	**19.84**
snd_Arab->bug_Latn	8.26	**15.68**	sot_Latn->amh_Ethi	8.86	**10.83**	srp_Cyrl->kac_Latn	1.33	**14.48**	tat_Cyrl->hye_Armn	22.22	**23.51**	tgk_Cyrl->amh_Ethi	8.82	**11.30**
tsn_Latn->plt_Latn	23.99	**25.14**	tso_Latn->sot_Latn	**25.90**	25.77	tzm_Tfng->amh_Ethi	3.42	**3.43**	uig_Arab->tgk_Cyrl	14.94	**17.74**	zul_Latn->amh_Ethi	8.75	**11.19**

Table 10: Comparison of CoD against GPT-3.5-TURBO. Results in chrF++ for MT on the FLORES-200 dataset. The best results are bolded and highlighted. We report on translating from X into Y.

Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD
ace_Arab	1.88	**1.94**	ace_Latn	5.19	**5.95**	acm_Arab	**11.31**	10.77	acq_Arab	**13.81**	13.56
aeb_Arab	**10.61**	9.37	afr_Latn	**36.89**	36.22	ajp_Arab	13.07	**13.10**	aka_Latn	4.25	**4.54**
als_Latn	**23.99**	23.11	amh_Ethi	1.86	**3.38**	apc_Arab	**12.24**	11.27	arb_Arab	**20.11**	18.97
ars_Arab	**16.87**	16.50	ary_Arab	**7.84**	7.29	arz_Arab	**11.20**	10.73	asm_Beng	3.02	**4.75**
ast_Latn	**22.41**	21.97	awa_Deva	7.33	**7.36**	ayr_Latn	**3.69**	3.48	azb_Arab	2.23	**2.50**
azj_Latn	8.44	**8.86**	bak_Cyrl	3.48	**5.41**	bam_Latn	3.05	**3.58**	ban_Latn	7.59	**8.65**
bel_Cyrl	8.50	**8.73**	bem_Latn	4.40	**5.12**	ben_Beng	8.66	**9.37**	bho_Deva	6.26	**6.83**
bjn_Arab	**2.19**	2.06	bjn_Latn	8.04	**8.82**	bod_Tibt	**0.86**	0.77	bos_Latn	**27.80**	26.42
bug_Latn	4.01	**4.73**	bul_Cyrl	**29.56**	28.70	cat_Latn	**37.62**	36.93	ceb_Latn	20.32	**22.93**
ces_Latn	**26.91**	25.14	cjk_Latn	2.97	**2.99**	ckb_Arab	2.66	**5.00**	crh_Latn	4.54	**5.84**
cym_Latn	**33.71**	30.34	dan_Latn	**42.25**	40.69	deu_Latn	**35.46**	33.02	dik_Latn	2.95	**3.29**
dyu_Latn	2.48	**2.60**	dzo_Tibt	0.17	**0.28**	ell_Grek	**21.70**	20.34	epo_Latn	**25.05**	24.90
est_Latn	**21.50**	19.58	eus_Latn	8.63	**8.82**	ewe_Latn	3.09	**4.03**	fao_Latn	12.72	**13.05**
fij_Latn	5.81	**8.02**	fin_Latn	**21.18**	18.90	fon_Latn	2.07	**2.31**	fra_Latn	**47.09**	43.97
fur_Latn	10.85	**13.44**	fuv_Latn	2.87	**3.03**	gaz_Latn	2.60	**3.31**	gla_Latn	8.91	**9.70**
gle_Latn	15.77	**15.91**	glg_Latn	**31.05**	30.06	grn_Latn	3.99	**4.54**	guj_Gujr	8.94	**11.38**
hat_Latn	15.67	**17.23**	hau_Latn	6.06	**10.45**	heb_Hebr	17.79	**18.74**	hin_Deva	19.18	**19.73**
hne_Deva	7.58	**8.62**	hrv_Latn	**25.92**	23.75	hun_Latn	**19.17**	18.22	hye_Armn	5.61	**8.35**
ibo_Latn	4.52	**7.66**	ilo_Latn	8.97	**11.84**	ind_Latn	**39.70**	37.89	isl_Latn	**15.80**	14.82
ita_Latn	**28.28**	26.14	jav_Latn	11.51	**15.11**	jpn_Jpan	**30.09**	27.94	kab_Latn	3.21	**3.95**
kac_Latn	0.72	**3.68**	kam_Latn	4.06	**4.37**	kan_Knda	6.68	**9.04**	kas_Arab	1.93	**2.58**
kas_Deva	1.71	**1.75**	kat_Geor	5.53	**6.36**	kaz_Cyrl	6.32	**8.51**	kbp_Latn	2.54	**3.53**
kea_Latn	7.61	**9.40**	khk_Cyrl	4.21	**5.67**	khm_Khmr	1.94	**2.32**	kik_Latn	4.17	**5.15**
kin_Latn	4.51	**6.40**	kir_Cyrl	4.45	**5.88**	kmb_Latn	**3.32**	2.98	kmr_Latn	5.56	**6.48**
knc_Arab	**1.18**	1.14	knc_Latn	2.73	**2.89**	kon_Latn	3.43	**7.27**	kor_Hang	**12.30**	11.36
lao_Laoo	7.58	**8.79**	lij_Latn	4.84	**5.57**	lim_Latn	**8.77**	8.28	lin_Latn	4.94	**8.39**
lit_Latn	**17.64**	16.01	lmo_Latn	5.15	**5.51**	ltg_Latn	**5.46**	5.25	ltz_Latn	**13.87**	13.52
lua_Latn	3.77	**4.57**	lug_Latn	4.22	**5.28**	luo_Latn	3.41	**4.23**	lus_Latn	**6.00**	5.85
lvs_Latn	**20.99**	19.02	mag_Deva	10.42	**11.34**	mai_Deva	**5.10**	5.04	mal_Mlym	5.02	**6.16**
mar_Deva	6.81	**8.52**	min_Latn	8.58	**9.76**	mkd_Cyrl	**23.79**	23.26	mlt_Latn	13.70	**15.90**
mni_Beng	1.11	**1.42**	mos_Latn	2.63	**2.70**	mri_Latn	11.65	**13.07**	mya_Mymr	1.35	**1.95**
nld_Latn	**24.82**	23.32	nno_Latn	**25.90**	25.15	nob_Latn	**29.56**	29.05	npi_Deva	7.52	**9.97**
nso_Latn	5.30	**10.49**	nus_Latn	2.19	**3.05**	nya_Latn	5.42	**7.50**	oci_Latn	19.63	**20.34**
ory_Orya	4.27	**5.76**	pag_Latn	5.93	**7.08**	pan_Guru	9.82	**11.64**	pap_Latn	**18.91**	16.88
pbt_Arab	3.19	**4.73**	pes_Arab	**17.00**	15.83	plt_Latn	5.80	**8.55**	pol_Latn	**18.97**	17.24
por_Latn	**47.12**	44.80	prs_Arab	15.16	**16.08**	quy_Latn	**3.85**	3.58	ron_Latn	**34.84**	33.26
run_Latn	3.97	**5.27**	rus_Cyrl	**26.34**	23.90	sag_Latn	2.71	**4.89**	san_Deva	1.86	**2.44**
scn_Latn	7.30	**7.85**	shn_Mymr	1.69	**2.09**	sin_Sinh	2.96	**3.83**	slk_Latn	**25.38**	24.26
slv_Latn	**25.01**	23.27	smo_Latn	7.81	**13.30**	sna_Latn	4.50	**5.95**	snd_Arab	3.92	**6.47**
som_Latn	5.35	**6.52**	sot_Latn	5.35	**8.23**	spa_Latn	**26.00**	24.98	srd_Latn	9.74	**11.85**
srp_Cyrl	2.94	**18.64**	ssw_Latn	4.00	**4.89**	sun_Latn	8.97	**10.88**	swe_Latn	**40.67**	38.81
swh_Latn	**27.98**	26.57	szl_Latn	6.86	**7.30**	tam_Taml	5.67	**7.67**	taq_Latn	3.30	**3.49**
taq_Tfng	1.40	**1.56**	tat_Cyrl	3.74	**6.30**	tel_Telu	6.89	**8.67**	tgk_Cyrl	5.84	**8.71**
tgl_Latn	**27.30**	27.00	tha_Thai	**5.24**	4.79	tir_Ethi	0.97	**2.56**	tpi_Latn	9.22	**12.54**
tsn_Latn	4.91	**8.82**	tso_Latn	4.34	**7.57**	tuk_Latn	4.47	**5.72**	tum_Latn	3.99	**4.95**
tur_Latn	**22.49**	21.12	twi_Latn	4.39	**5.18**	tzm_Tfng	2.20	**2.74**	uig_Arab	3.25	**4.92**
ukr_Cyrl	**23.22**	21.95	umb_Latn	**3.15**	2.66	urd_Arab	12.68	**13.91**	uzn_Latn	7.21	**9.13**
vec_Latn	9.31	**10.25**	vie_Latn	**34.42**	32.06	war_Latn	15.38	**18.62**	wol_Latn	3.67	**4.28**
xho_Latn	4.53	**5.96**	ydd_Hebr	6.51	**7.96**	yor_Latn	3.27	**4.03**	yue_Hant	**26.40**	23.78
zho_Hans	**39.82**	36.30	zho_Hant	**29.30**	28.00	zsm_Latn	**31.51**	30.35	zul_Latn	4.49	**6.78**

Table 11: Comparison of CoD against GPT-3.5-TURBO. Results in BLEU for MT on the FLORES-200 dataset. The best results are bolded and highlighted. We report on translating from English into the languages.

Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD
srp_Cyrl	10.19	**22.18**	kac_Latn	9.27	**20.38**	ckb_Arab	18.73	**32.59**	azb_Arab	21.40	**25.44**	tzm_Tfng	16.87	**31.00**
kon_Latn	34.07	**40.00**	tat_Cyrl	26.62	**36.36**	nso_Latn	19.73	**30.46**	sag_Latn	13.05	**29.22**	bak_Cyrl	11.15	**20.55**
shn_Mymr	21.73	**31.61**	lin_Latn	21.79	**33.80**	uig_Arab	23.57	**32.35**	hau_Latn	25.01	**34.22**	ibo_Latn	23.59	**32.24**
amh_Ethi	23.27	**32.52**	zul_Latn	27.89	**36.19**	bug_Latn	16.46	**27.75**	lao_Laoo	5.66	**22.29**	tso_Latn	28.26	**37.79**
kbp_Latn	16.22	**28.17**	tsn_Latn	23.51	**32.12**	smo_Latn	5.28	**46.61**	snd_Arab	19.90	**33.19**	hye_Armn	22.44	**33.29**
nya_Latn	23.49	**31.70**	sot_Latn	25.32	**33.68**	tgk_Cyrl	2.02	**18.49**	plt_Latn	8.15	**29.03**	kir_Cyrl	25.33	**35.22**

Table 12: Comparison of CoD against TEXT-DAVINCI-003. Results in chrF++ for MT on the FLORES-200 dataset. The best results are bolded and highlighted. We report on translating from the languages into English.

Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD
ace_Arab	3.35	**45.43**	ace_Latn	10.12	**56.56**	acm_Arab	27.78	**59.66**	acq_Arab	29.45	**61.01**
aeb_Arab	24.38	**54.72**	afr_Latn	53.62	**72.57**	ajp_Arab	33.45	**62.98**	aka_Latn	7.87	**46.86**
als_Latn	33.73	**62.52**	amh_Ethi	4.10	**51.04**	apc_Arab	29.70	**61.44**	arb_Arab	33.30	**62.87**
ars_Arab	32.31	**61.93**	ary_Arab	21.67	**53.13**	arz_Arab	25.54	**55.65**	asm_Beng	12.09	**53.20**
ast_Latn	36.22	**60.15**	awa_Deva	19.51	**53.83**	ayr_Latn	4.49	**43.40**	azb_Arab	8.37	**49.53**
azj_Latn	16.96	**46.19**	bak_Cyrl	8.63	**47.04**	bam_Latn	4.85	**48.18**	ban_Latn	17.36	**59.05**
bel_Cyrl	17.05	**42.43**	bem_Latn	7.84	**47.90**	ben_Beng	20.66	**58.67**	bho_Deva	14.90	**50.09**
bjn_Arab	4.12	**40.50**	bjn_Latn	19.12	**61.20**	bod_Tibt	2.22	**43.97**	bos_Latn	38.22	**63.50**
bug_Latn	7.43	**48.05**	bul_Cyrl	35.84	**62.73**	cat_Latn	42.42	**65.37**	ceb_Latn	31.85	**65.36**
ces_Latn	36.18	**61.24**	cjk_Latn	4.81	**42.89**	ckb_Arab	8.98	**56.80**	crh_Latn	18.32	**52.26**
cym_Latn	45.90	**73.99**	dan_Latn	45.39	**65.68**	deu_Latn	40.51	**61.48**	dik_Latn	5.14	**48.32**
dyu_Latn	3.93	**42.68**	dzo_Tibt	1.79	**42.59**	ell_Grek	30.53	**60.12**	epo_Latn	37.60	**62.90**
est_Latn	33.66	**59.58**	eus_Latn	21.10	**50.68**	ewe_Latn	4.64	**45.93**	fao_Latn	29.33	**61.67**
fij_Latn	9.21	**44.86**	fin_Latn	31.07	**55.91**	fon_Latn	3.59	**43.36**	fra_Latn	42.02	**63.87**
fur_Latn	29.28	**60.58**	fuv_Latn	4.79	**43.43**	gaz_Latn	4.54	**43.91**	gla_Latn	21.09	**56.10**
gle_Latn	28.53	**59.30**	glg_Latn	37.42	**61.86**	grn_Latn	7.43	**48.15**	guj_Gujr	19.97	**59.71**
hat_Latn	28.12	**62.50**	hau_Latn	9.98	**57.93**	heb_Hebr	34.75	**67.09**	hin_Deva	27.76	**62.22**
hne_Deva	18.31	**58.17**	hrv_Latn	33.90	**59.12**	hun_Latn	31.08	**57.48**	hye_Armn	15.75	**59.69**
ibo_Latn	6.98	**54.39**	ilo_Latn	16.95	**58.19**	ind_Latn	37.62	**67.28**	isl_Latn	28.66	**58.33**
ita_Latn	30.12	**52.28**	jav_Latn	22.78	**63.84**	jpn_Jpan	22.87	**49.58**	kab_Latn	4.56	**49.96**
kac_Latn	3.78	**40.68**	kam_Latn	6.42	**48.78**	kan_Knda	18.13	**55.96**	kas_Arab	7.56	**50.38**
kas_Deva	7.18	**45.60**	kat_Geor	12.51	**50.18**	kaz_Cyrl	15.35	**52.41**	kbp_Latn	3.86	**44.19**
kea_Latn	35.17	**68.21**	khk_Cyrl	9.43	**46.67**	khm_Khmr	10.09	**59.33**	kik_Latn	6.66	**51.01**
kin_Latn	12.50	**56.50**	kir_Cyrl	9.53	**44.09**	kmb_Latn	5.24	**43.07**	kmr_Latn	14.87	**54.00**
knc_Arab	2.54	**27.88**	knc_Latn	5.04	**43.30**	kon_Latn	5.82	**47.48**	kor_Hang	23.65	**58.02**
lao_Laoo	7.64	**60.68**	lij_Latn	29.70	**61.27**	lim_Latn	35.97	**63.71**	lin_Latn	8.40	**51.53**
lit_Latn	28.36	**55.20**	lmo_Latn	28.16	**61.42**	ltg_Latn	12.63	**55.58**	ltz_Latn	35.99	**65.84**
lua_Latn	6.45	**49.93**	lug_Latn	7.92	**51.68**	luo_Latn	4.66	**48.09**	lus_Latn	7.74	**40.62**
lvs_Latn	30.24	**57.50**	mag_Deva	21.31	**59.37**	mai_Deva	15.98	**56.00**	mal_Mlym	16.31	**55.22**
mar_Deva	18.50	**56.44**	min_Latn	17.83	**61.81**	mkd_Cyrl	35.93	**65.21**	mlt_Latn	38.24	**69.79**
mni_Beng	3.35	**41.00**	mos_Latn	4.07	**41.80**	mri_Latn	16.36	**53.46**	mya_Mymr	3.52	**46.61**
nld_Latn	28.29	**48.10**	nno_Latn	42.43	**62.41**	nob_Latn	39.44	**60.62**	npi_Deva	20.99	**59.30**
nso_Latn	10.61	**56.78**	nus_Latn	3.61	**49.63**	nya_Latn	11.86	**53.30**	oci_Latn	45.60	**71.14**
ory_Orya	14.19	**53.04**	pag_Latn	14.93	**48.79**	pan_Guru	21.52	**59.82**	pap_Latn	39.13	**68.55**
pbt_Arab	9.16	**51.80**	pes_Arab	29.21	**63.56**	plt_Latn	13.40	**55.84**	pol_Latn	26.05	**50.42**
por_Latn	45.32	**67.64**	prs_Arab	29.57	**64.31**	quy_Latn	5.16	**38.41**	ron_Latn	38.90	**62.75**
run_Latn	8.75	**49.24**	rus_Cyrl	31.17	**58.97**	sag_Latn	4.27	**44.78**	san_Deva	10.26	**48.61**
scn_Latn	29.03	**61.42**	shn_Mymr	4.17	**46.02**	sin_Sinh	4.48	**49.37**	slk_Latn	34.61	**59.74**
slv_Latn	31.91	**56.46**	smo_Latn	12.90	**54.71**	sna_Latn	10.22	**52.77**	snd_Arab	11.40	**55.61**
som_Latn	11.78	**56.63**	sot_Latn	10.85	**56.56**	spa_Latn	27.10	**50.19**	srd_Latn	29.21	**63.24**
srp_Cyrl	38.67	**66.70**	ssw_Latn	9.08	**53.04**	sun_Latn	20.81	**61.58**	swe_Latn	44.43	**67.44**
swh_Latn	36.36	**70.61**	szl_Latn	30.86	**62.58**	tam_Taml	12.73	**54.97**	taq_Latn	5.11	**40.81**
taq_Tfng	2.42	**49.72**	tat_Cyrl	10.59	**49.60**	tel_Telu	15.88	**56.35**	tgk_Cyrl	14.10	**53.95**
tgl_Latn	37.25	**67.86**	tha_Thai	20.48	**59.97**	tir_Ethi	2.58	**45.77**	tpi_Latn	16.99	**44.01**
tsn_Latn	9.52	**49.81**	tso_Latn	10.03	**52.40**	tuk_Latn	13.67	**50.77**	tum_Latn	7.19	**44.40**
tur_Latn	33.03	**60.81**	twi_Latn	7.81	**46.98**	tzm_Tfng	2.52	**52.68**	uig_Arab	8.05	**46.64**
ukr_Cyrl	33.90	**63.83**	umb_Latn	4.78	**42.39**	urd_Arab	22.60	**57.89**	uzn_Latn	17.65	**51.93**
vec_Latn	35.76	**64.59**	vie_Latn	29.38	**64.75**	war_Latn	31.18	**65.74**	wol_Latn	6.09	**47.40**
xho_Latn	14.82	**59.65**	ydd_Hebr	20.34	**70.65**	yor_Latn	7.98	**50.36**	yue_Hant	24.66	**52.89**
zho_Hans	23.80	**54.52**	zho_Hant	22.75	**51.99**	zsm_Latn	37.47	**67.79**	zul_Latn	14.61	**60.45**

Table 13: Comparison of CoD against GPT-3.5-TURBO. Results in chrF++ for MT on the FLORES-200 dataset. The best results are bolded and highlighted. We report on translating from the languages into English.

Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD	Language	GPT	CoD
srp_Cyrl	2.08	**4.84**	kac_Latn	2.22	**2.73**	ckb_Arab	3.24	**5.42**	azb_Arab	3.92	**4.43**	tzm_Tfng	2.55	**4.54**
kon_Latn	8.33	**11.27**	tat_Cyrl	4.70	**6.96**	nso_Latn	3.86	**7.12**	sag_Latn	1.78	**4.75**	bak_Cyrl	2.32	**3.19**
shn_Mymr	3.49	**5.15**	lin_Latn	3.54	**7.22**	uig_Arab	7.80	**8.46**	hau_Latn	4.43	**7.28**	ibo_Latn	4.33	**7.50**
amh_Ethi	4.45	**5.75**	zul_Latn	4.73	**7.11**	bug_Latn	2.58	**5.01**	lao_Laoo	1.30	**1.38**	tso_Latn	5.83	**10.96**
kbp_Latn	2.41	**5.51**	tsn_Latn	4.15	**6.76**	smo_Latn	3.83	**17.43**	snd_Arab	3.55	**5.84**	hye_Armn	3.88	**6.69**
nya_Latn	3.75	**6.57**	sot_Latn	4.28	**6.72**	tgk_Cyrl	2.03	**3.00**	plt_Latn	2.35	**4.38**	kir_Cyrl	4.10	**6.18**

Table 14: Comparison of CoD against TEXT-DAVINCI-003. Results in BLEU for MT on the FLORES-200 dataset. The best results are bolded and highlighted. We report on translating from the languages into English.

Source Sentence	Using ships to transport goods is by far the most efficient way to move large amounts of people and goods across oceans.
Standard GPT4 Prompt	Translate the following text from English into Central Kurdish with Arabic script: {Source Sentence}
Bilingual Dictionary Prompt	"transport" means "گواستنەوە". "efficient" means "کارامە". "amounts" means "بڕەکان". Translate the following text from English into Central Kurdish with Arabic script: {Source Sentence}
CoD Prompt	"transport" means "گواستنەوە" means "le transport" means "Verkehr" means "transporte". "efficient" means "کارامە" means "efficaces" means "effizient" means "eficiente". "amounts" means "بڕەکان" means "Les montants" means "Beträge" means "Montantes". Translate the following text from English into Central Kurdish with Arabic script: {Source Sentence}
Target Sentence	بەکارهێنانی کەشتی بۆ گواستنەوەی کەلوپەل تا ئێستا چوستترین ڕێگایە بۆ گواستنەوەی ژمارەی زۆری خەڵک و کاڵا بە ناو ئۆقیانوسەکان
Standard GPT4 Output BLEU=4.81, chrF++=38.75	بەکارهێنانی کەشتی بۆ بەرهەمدان بە بەرز بوونی زۆر سەلماندن و دراوە لە دوورگایەکان
Bilingual Dictionary CoD Output BLEU=5.09, chrF++=48.52	بەکارهێنانی کۆپەکە بۆ گواستنەوەی کاڵا کارابەترین ڕێگاە بۆ جێبەجێ کردنی بڕە زۆرەکانی خەڵك و کاڵا لە سەربەری زەریاکان
CoD Output BLEU=27.85, chrF++=67.47	با بەکارهێنانی کەشتی بۆ گواستنەوەی کاڵا، باشترین ڕێگای کارامەیە بۆ گواستنەوەی بڕەکانەی زۆری خەڵک و کاڵا لەسەر دریاکان
Source Sentence	Using ships to transport goods is by far the most efficient way to move large amounts of people and goods across oceans.
Standard GPT4 Back BLEU=2.33, chrF++=40.33	The use of ships for production has been increasingly documented in the islands.
Bilingual CoD Back BLEU=27.79, chrF++=74.31	The use of the bubble to transport cargo is the most efficient way to implement the large amounts of people and cargo on the surface of the oceans.
CoD Back BLEU=49.73, chrF++=81.74	The use of ships to transport goods is the most efficient way to transport large amounts of people and goods on the seas.

Figure 4: A case study on translating from English into Central Kurdish with Latin script using GPT-4 throughout the cases. We evaluate the results on BLEU and chrF++. We highlight in green the words translated wrong by baselines but translated correctly by CoD, even if the words are not presented in the multilingual dictionary chains.

Source Sentence	There's a tradition to pass the Easter night awake at some exposed point to see the sunrise.
Standard GPT3.5 Prompt	Translate the following text from English into Central Kurdish with Arabic script: {Source Sentence}
Bilingual Dictionary Prompt	"tradition" means "نەریت" . "exposed" means "بەرکەوتوو" \n "sunrise" means "خۆر هەڵدێت". Translate the following text from English into Central Kurdish with Arabic script: {Source Sentence}
CoD Prompt	"tradition" means "نەریت" means "tradition" means "Tradition" means "tradição". "exposed" means "بەرکەوتوو" means "exposés" means "ausgesetzt" means "expostos". "sunrise" means "خۆر هەڵدێت" means "le lever du soleil" means "Sonnenaufgang" means "Sol nascente". Translate the following text from English into Central Kurdish with Arabic script: {Source Sentence}
Target Sentence	نەریتێک هەیە بۆ تێپەڕاندنی شەوی ئیستەر بە بەئاگاییەوە لە هەندێک خاڵی ئاشکرا بۆ بینینی خۆرهەڵاتن.
Standard GPT3.5 Output BLEU=2.35, chrF++=43.23	هاتوو دادگەیەک هەبووە بەرجەم بگرە تایبەتی بە ئامادەی هەندێک نووسراوی گورزی ژیاری خەڵکی تانی خوێندە خوێندنی خوابەی ڕۆژی یان هەمەووی.
Bilingual Dictionary CoD Output BLEU=2.91, chrF++=43.23	هەنگاوی تەمام کردنی شەو وەرەیی بە شەو خۆر هەڵدێت بەر کەوتەکانی تاقی کردنی نەریتییەک هەیە
CoD Output BLEU=7.66, chrF++=59.60	هەندێک نەریت هەیە بۆ گەڕانەوەی شەو تایبەتی خەڵکی تەقصیریتەوە بەر بەرکەوتوو بۆ بینینی خۆر هەڵدێت.
Source Sentence	There's a tradition to pass the Easter night awake at some exposed point to see the sunrise.
Standard GPT3.5 Back BLEU=1.13, chrF++=57.74	There was a court of law, and in particular the presence of some of the most learned writings of the people, which were read in the course of the day or night.
Bilingual CoD Back BLEU=2.18, chrF++=55.57	The step of the eclipse of the moon by night, the sun rises by night, has the connotations of a traditional experiment.
CoD Back BLEU=8.65, chrF++=65.67	There are some traditions for the return of the night special people from Taxairat exposed to see the sun rise.

Figure 5: A case study on translating from English into Central Kurdish with Latin script using GPT-3.5 throughout the cases. We evaluate the results on BLEU and chrF++. We highlight in green the words translated wrong by baselines but translated correctly by CoD, even if the words are not presented in the multilingual dictionary chains.