Contrastive Perturbation Enhancement for LLM-Based Machine Translation *

Thai Nguyen-Quoc[1[0009−0006−3288−4455]]
Hoan Nguyen-Cong[2[0000−0001−7298−1598]]
Huong Le-Thanh[1[0000−0001−7298−1598]] **

[1] School of Information and Communication Technology, Hanoi University of Science and Technology, Vietnam
[2] Ho Chi Minh City University of Technology, Ho Chi Minh, Vietnam
thai.nq107@gmail.com, hoan.nc0506@gmail.com, huonglt@soict.hust.edu.vn

Abstract. Large language models have been increasingly effective in various NLP tasks, especially for machine translation task. However, these models require a lot of computational resources and need to be further fine-tuned on a specific training data to achieve better performance. Medium-sized language models often show significantly poor performance compared to the large language models. Therefore, it is necessary to study methods to solve this problem. In this paper, we propose a method to fine-tune a language model with a size of several billion parameters based on the instructions from a large language model such as GPT-4 through the contrastive learning technique, called **CoPE** - **Co**ntrastive **P**erturbation **E**nhancement for LLM-Based Machine Translation. Our proposal consists of three stages: fine-tuning the language model on a parallel dataset, generating entailment as positive and contradiction as negative examples from the training dataset based on a high-performance large language model such as GPT-4, and then using these examples to improve the model through the contrastive learning technique. These examples will be evaluated and ranked to increase the influence of quality examples. Experimental results show that our proposal with a base model of LLaMA-3.1 with 8B parameters achieves 35.99 bleu score, 85.28 COMET-22 score, and 88.90 XCOMET score on the WMT'21 and WMT'22 datasets. This result is competitive with models such as ALMA-13B-R trained based on the contrastive preference optimization technique and is higher than the GPT-3.5 model.

Keywords: Machine Translation · Contrastive Learning · Large Language Model.

1 Introduction

Machine translation (MT) has witnessed significant advancements with the advent of large language models (LLMs), particularly decoder-only architectures

* Supported by AI VIETNAM.
** Corresponding Author

like the GPT series Brown et al. (2020) [1], OpenAI (2023) [2], Mistral Jiang et al. (2023) [3], and LLaMA series Touvron et al. (2023) [4]. These models have demonstrated remarkable translation capabilities, surpassing conventional encoder-decoder architectures Vaswani et al. (2017) [5] in many cases. However, the performance of smaller LLMs (7B or 13B) still lags behind larger models (e.g., GPT-3.5, GPT-4; OpenAI (2023) [2]) and dedicated MT systems Zhu et al. (2023) [6].

One key challenge in improving smaller LLM-based MTs is the limited linguistic diversity of their pre-training data, often biased towards English. While fine-tuning with high-quality parallel data Xu et al. (2024) [7] can enhance translation abilities, it still falls short of achieving state-of-the-art performance. This gap highlights the need for novel training methods that can effectively leverage the potential of smaller LLMs for MT.

In this paper, we propose a novel training framework called **Contrastive Perturbation Enhancement** (**CoPE**) to address this challenge. Our main contributions are summarized as follows:

– We introduce **CoPE**, a training framework designed to improve the performance of medium-sized language models in machine translation. CoPE leverages contrastive learning by generating high-quality entailment and contradiction examples using a large model like GPT-4.
– CoPE enhances the model's ability to generalize by learning from diverse perturbed examples, helping the language model generalize better to unseen data and improve its translation quality.
– We validated CoPE on the WMT'21 and WMT'22 datasets, achieving improvements with an 8B-parameter model. This demonstrates that medium-sized models can approach the performance of larger ones with efficient training techniques.

2 Related Work

Machine Translation: Machine translation (MT) has significantly evolved with the advent of transformer encoder-decoder architectures Vaswani et al. (2017) [5], which are the backbone of prominent models like NLLB-200 (NLLB TEAM et al. (2022) [8]), M2M100 Fan et al. (2020) [9], BiBERT Xu et al. (2021) [10], and MT5 Xue et al. (2020) [11]. However, the emergence of decoder-only large language models (LLMs) such as the GPT series Brown et al. (2020) [1], OpenAI (2023) [2]) has introduced new paradigms in MT. Despite their potential, smaller LLMs often underperform due to their pre-training on English-centric datasets, which limits their linguistic diversity. Efforts to enhance these models, such as fine-tuning LLaMA-2 with non-English data and supervised fine-tuning (SFT) with high-quality parallel data, have shown promise Xu et al. (2024) [7]. The ALMA model, for instance, outperforms many moderated-size LLMs but still lags behind leading models like GPT-4. To bridge this gap, novel training methods like Contrastive Preference Optimization (CPO) have been introduced, significantly improving performance with minimal additional parameters Xu et al. (2024) [12]. For a given source sentence x, CPO uses GPT-4 and a model fine-tuned (called ALMA model) to generate respective translations, as y_{gpt-4} and y_{alma}. These translations are the paired with the target sentence of x to form a triplet $y = (y_{ref}, y_{gpt-4}, y_{alma})$, representing three different translation outputs for the input x. This triplet is evaluated using metrics such as XCOMET to select translations y_w as the preferred and y_l as the dis-preferred. A significant risk for CPO is that correct translations y_{ref} may be inadvertently removed in this process, which could adversely impact model training.

Large Language Model: Large language models (LLMs) have revolutionized natural language processing (NLP) tasks, including machine translation. The transformer architecture Vaswani et al. (2017) [5] is fundamental to these models. Research has shown that scaling up LLMs enhances their capabilities, leading to emergent abilities such as in-context learning (Wei et al., 2022). Notable examples include GPT-4 OpenAI (2023) [2] and ChatGPT (OpenAI, 2022), which have demonstrated impressive results across various NLP tasks. However, achieving such performance requires substantial computational resources and infrastructure. Translation-specific LLMs like ALMA Xu et al. (2024) [7] and Aya 23 Aryabumi et al. (2024) [14] have achieved top-tier performance through extensive pretraining and fine-tuning, albeit at high costs.

Contrastive Learning: Contrastive learning has been widely adopted in various domains, including NLP, to improve model performance by learning from positive and negative pairs. In the context of MT, contrastive learning has been used to address issues like exposure bias and adversarial perturbations. Previous works have employed techniques such as scheduled sampling Bengio et al. (2015) [15] and reinforcement learning Paulus et al. (2017) [16] to tackle exposure bias. Adversarial training has also been explored to enhance robustness Miyato et al. (2017) [17], Zhu et al. (2019) [18]. Contrastive learning methods, such as those used in Word2Vec Mikolov et al. (2013) [19] and sentence representation learning Logeswaran et al. (2018) [20], have shown significant improvements in various NLP tasks. Recent studies have leveraged LLMs for generating high-quality entailment and contradiction examples to train contrastive learning models, demonstrating superior performance Schick et al. (2020) [22], Zhang et al. (2023) [21]. However, the quality of generated content remains a challenge, necessitating efficient approaches to refine and utilize these examples effectively. CoNT (2022) [29] is a strong contrastive learning framework for neural text generation which outperforms the MLE based training method on five text generation tasks. Recently, Wang et al. (2024) [28] introduced a learning method based on LLMs to refine contrastive generation for better sentence representation, which demonstrated the high efficiency of the contrastive learning method.

3 Methodology

In this section, we will present CoPE, a framework designed to improve LLM-based machine learning models using contrastive learning methods. By generating high-quality entailment and contradiction samples, CoPE leverages these samples to further fine-tune the translation model using contrastive loss. The proposed framework consists of three main steps:

- **Instruction Tuning:** A medium-sized language model is fine-tuned on the training dataset.
- **Contrastive Perturbation Generation:** entailment and contradiction samples are generated based on the training data, and high-quality samples are filtered.
- **Contrastive Training:** The model is then further fine-tuned using contrastive learning, leveraging the high-quality contrastive samples.

3.1 Fine-Tuning Language Model for Machine Translation Task

We fine-tune a large language model on the original training dataset for machine translation task. Our goal is to train a many-to-many multilingual model of moderate size to perform as well as large language models, we will use a language model like LLaMA-3.1 with 8B parameters. Let x is a sentence in the source language and y its translation in the target language. Based on proposals of Xu et al. (2023) [7] and Hendy et al. (2023) [23], we use a fixed prompt for the sentence-level translation, denoted as $\mathcal{I}$, to fine tune a large language model parameterized by θ. The fixed prompt is used with the following structure:

```
Translate this from [source language] to [target language]:
[source language]: <source sentence>
[target language]:
```

The objective log-likelihood loss can be illustrated as follows:

$$\mathcal{L}_{NLL}(x, y, \theta) = -logP(y|x, \mathcal{I}; \theta) = -\sum_{t=1}^{T} logP(y_t|y_{<t}, x, \mathcal{I}; \theta)$$

where T is length of the target sentence, y_t is the t-th target token. During the training stage, the fixed prompt template and the source sentence is not computed in $\mathcal{L}_{NLL}(x, y, \theta)$ to improve the model, this method is described by Zhang et al. (2023) [21].

3.2 Contrastive Perturbation Generation

Contrastive learning technique shows effectiveness when obtaining quality positive and negative samples. So, we use a large language model C_ϕ (like GPT-4) and the original training dataset to generate contrastive perturbation samples in the target sentence.

The bilingual sentence pair (x, y) is combined with an instruction I to generate additional k samples $y_i)$ with $(i = 1, 2, 3, ..., k)$ in the target language. The results obtained from the model C_ϕ using this instruction are formatted as follows:

$$< y_i, sim(y_i, x), sim(y_i, y), p(y_i|I, x) >$$

where y_i is the i-th generated sample; $sim(y_i, x)$ is the cosine similarity of y_i and the source sentence x; $sim(y_i, y)$ is the cosine similarity of y_i and the target sentence y; $p(y_i|I, x)$ is the probability of the output sequence y_i with length T:

$$p(y_i|I, x) = \prod_{t=1}^{T} p(y_t|y_{<t}, x, I, \phi)$$

We use two different instructions to generate contrastive perturbation samples.

Firstly, an entailment instruction I^+ is used to generate entailment samples by providing an alternative expression with the same meaning. The entailment instruction is described in the section 5.1. These entailment samples are formatted as:

$$< y_i^+, sim(y_i^+, x), sim(y_i^+, y), p(y_i^+|I^+, x) >$$

We then normalize these measures to rank the k generated entailment samples based on the following formula:

$$c(y_i^+) = \frac{sim(y_i^+, x)}{\sum_{j=1}^{k} sim(y_j^+, x)} + \frac{sim(y_i^+, y)}{\sum_{j=1}^{k} sim(y_j^+, y)} + \frac{p(y_i|I, x)}{\sum_{j=1}^{k} p(y_j|I, x)}$$

With k generated entailment samples, we have a set of the metrics used to rank $C^+ = c(y_1^+), c(y_2^+), ..., c(y_k^+)$. C^+ is sorted in the descending order. $c(y_i^+)$ is higher when the similarity of the generated entailment sample to the sentence pair (x, y) is higher and the probability of generating that sentence is higher.

Secondly, a contradiction instruction I^- is used to generate contradiction samples by swapping, changing, or contradicting some details in order to express a different meaning. The contradiction instruction is also described in the section 5.1. These contradiction samples are fomartted as:

$$< y_i^-, sim(y_i^-, x), sim(y_i^-, y), p(y_i^-|I^-, x) >$$

We also normalize these measures to rank the k generated contradiction samples based on the following formula:

$$c(y_i^-) = \frac{sim(y_i^-, x)}{\sum_{j=1}^{k} sim(y_j^-, x)} + \frac{sim(y_i^-, y)}{\sum_{j=1}^{k} sim(y_j^-, y)} - \frac{p(y_i|I^-, x)}{\sum_{j=1}^{k} p(y_j|I^-, x)}$$

With k generated contradicting samples, we have a set of the metrics used to rank $C^- = c(y_1^-), c(y_2^-), ..., c(y_k^-)$ sorted in the ascending order. In this set,

y_i^- is lower when the similarity of the generated contradicting sample to the sentence pair (x, y) is lower and the probability of generating that sentence is higher. This helps ensure that the generated contradicting sentences will be true sound patterns but still have good semantics.

3.3 Contrastive Training

After generating sets C^+ and C^- in the stage 3.2, we use these samples to continue fine-tune the model θ from the stage 3.1 based on the contrastive learning technique.

Let z_x is the hidden representation of the source sentence x; z_y is the hidden representation of the target sentence y; z_{C^-} is a set of the hidden representations of the contradiction samples in C^-. We use the contradiction samples as the negative sample for contrastive learning. Then the objective loss is:

$$\mathcal{L}_{neg}(\theta) = \sum_{i=1}^{N} log \frac{exp(sim(z_x^{(i)}, z_y^{(i)})/\tau)}{\sum_{z_j^{(i)} \in z_{C^-}^{(i)}} w_j * exp(sim(z_x^{(i)}, z_j^{(i)})/\tau)}$$

Similarity, let z_{C^+} is a set of the hidden representations of the entailment samples in C^+. We use the entailment samples as additional positove example to augment contrastive learning. The objective loss is:

$$\mathcal{L}_{pos}(\theta) = \sum_{i=1}^{N} log \frac{exp(sim(z_x^{(i)}, z_y^{(i)})/\tau)}{\sum_{z_j^{(i)} \in z_{C^+}^{(i)}} w_j * exp(sim(z_x^{(i)}, z_j^{(i)})/\tau)}$$

where τ is the temperature; $sim(\cdot, \cdot)$ is the cosine similarity funtion and $w_j = 1/k$ is the weight that represents the importance of the j-th sample (k is the rank of z_j in C).

Finally, combining the loss of the entailment sample and the contradiction sample, we train the model θ by maximizing the following objective function:

$$\mathcal{L} = \max_{\theta} \mathcal{L}_{NLL}\theta + \alpha * \{\mathcal{L}_{neg}(\theta) - \mathcal{L}_{pos}(\theta)\}$$

where α is a hyperparameter used to consider the importance of the contrastive loss.

4 Experiment

4.1 Data

We use the same parallel training data with Xu et al. (2024) [7] with 58K training examples across many languages (cs, de, is, zh, ru, en), this dataset consists human-written datasets from WMT datasets. We use 5 translation directions in the experiment: $cs \rightarrow en$, $de \rightarrow en$, $is \rightarrow en$, $zh \rightarrow en$, $ru \rightarrow en$. To evaluate the performance of our proposal, we use a test dataset from WMT'21 and WMT'22, which consist 8K parallel sentences with 5 translation directions above, Freiteg et al. (2022) [30].

Model	**De→En**			**Cs→En**		
	BLEU	**COMET-22**	**XCOMET**	**BLEU**	**COMET-22**	**XCOMET**
WMT Winners	**33,34**	85,05	93,74	**64,14**	**89,00**	85,65
GPT-4	32,41	**85,35**	**94,47**	46,86	87,26	**88,48**
GPT-3.5-text-davinci-003	30,78	84,79	92,78	44,51	86,16	83,51
ALMA-7B-LoRA	29,56	83,95	92,93	43,49	85,93	81,34
ALMA-7B-R (CPO)	30,52	84,61	93,85	42,92	86,29	85,76
ALMA-13B-R (CPO)	30,89	84,95	94,20	**44,39**	86,85	**88,03**
LLaMA-3.1-8B Zero-shot	20,42	77,74	79,89	32,56	82,15	78,56
LLaMA-3.1-8B SFT	30,26	84,15	92,01	42,36	85,47	82,15
CoPE (w/ Pos.)	30,94	84,24	92,16	43,29	85,85	83,45
CoPE (w/ Neg.)	31,16	84,65	93,55	43,77	86,46	85,72
CoPE (w/ Pos.+Neg.)	**31,45**	**85,01**	**94,17**	43,15	**86,92**	88,34

Model	**Is→En**			**Zh→En**		
	BLEU	**COMET-22**	**XCOMET**	**BLEU**	**COMET-22**	**XCOMET**
WMT Winners	**41,60**	86,98	78,14	**33,49**	81,02	87,20
GPT-4	41,29	**87,21**	**81,11**	23,82	**82,46**	**92,06**
GPT-3.5-text-davinci-003	31,88	82,13	66,44	24,98	81,62	90,92
ALMA-7B-LoRA	35,64	86,09	75,02	23,64	79,78	83,94
ALMA-7B-R (CPO)	38,64	86,66	79,14	22,45	80,95	90,79
ALMA-13B-R (CPO)	39,67	**87,14**	**80,49**	23,23	81,64	**91,65**
LLaMA-3.1-8B Zero-shot	11,98	62,79	64,78	19,18	74,67	80,45
LLaMA-3.1-8B SFT	37,06	86,12	76,38	23,48	80,21	85,78
CoPE (w/ Pos.)	38,65	86,14	77,12	23,21	80,54	86,07
CoPE (w/ Neg.)	**40,24**	86,51	78,45	**23,80**	81,36	88,78
CoPE (w/ Pos.+Neg.)	39,66	86,98	80,31	23,51	**81,72**	90,45

Model	**Ru→En**			**Avg**		
	BLEU	**COMET-22**	**XCOMET**	**BLEU**	**COMET-22**	**XCOMET**
WMT Winners	**45,18**	**85,95**	90,91	**43,55**	85,60	87,13
GPT-4	41,09	85,87	**90,95**	37,09	**85,63**	**89,41**
GPT-3.5-text-davinci-003	38,52	84,80	89,29	34,13	83,90	84,59
ALMA-7B-LoRA	39,21	84,84	88,50	34,31	84,12	84,35
ALMA-7B-R (CPO)	38,42	85,11	90,10	34,59	84,72	87,93
ALMA-13B-R (CPO)	39,06	85,45	91,18	35,45	85,21	**89,11**
LLAMA-3.1-8B Zero-shot	30,45	72,84	83,67	22,92	74,04	77,47
LLAMA-3.1-8B SFT	37,26	85,17	89,17	34,08	84,22	85,10
CoPE (w/ Pos.)	39,95	85,25	90,25	35,21	84,40	85,87
CoPE (w/ Neg.)	**40,98**	85,67	90,89	**35,99**	84,93	87,48
CoPE (w/ Pos.+Neg.)	40,06	**85,75**	**91,24**	35,57	**85,28**	88,90

Table 1. The results table of different models across various metrics for 5 translation directions. The best results are in bold. The results in the first three rows represent the SOTA models such as GPT-4, WMT Winners. Results from the fourth onward compare to SOTA methods, including ALMA, CPO.

4.2 Training Setup

We use LLaMA-3.1-8B as a baseline model to continue fine tune for machine translation task. In the stage 3.2, GPT-4 is used to generate the entailment and contradiction samples. During the training phase in the stage 3.1 and 3.3, we apply LoRA technique with lora rank is 16. We set the temperature is 0.1 and α is 0.1 in the stage 3.3. In the stage 3.2, we choose $k = 5$ is the number of generated samples. We use the unsloth library [3] to help train faster and save memory.

4.3 Baselines

In this experimental section, we compare the results of our proposal with several SOTA models. First, the ALMA method proposed by Xu et al. (2024) [7] with three model: $ALMA-7B-LoRA$, $ALMA-7B-R$ and $ALMA-13B-R$ using the contrastive preference optimization method (CPO) Xu et al. (2024) [12]. Second, we evaluate based on the zero-shot translation results from the GPT-4 model. And finally, we compare with the results of the WMT competition winners.

4.4 Results

The experiment result is described in Table 1. We use sacreBLEU[4] [24] and COMET score [25] to evaluate the performance of models. Two versions of the COMET score are applied: COMET-22[5] [27] and XCOMET[6] [26].

In this experimental part, to evaluate the effectiveness of the proposal, we set up with different scenarios. We refined LLaMA-3.1-8B and used it as the base model for the scenarios, as 'LLaMA-3.1-8B SFT' in the results table. The first scenario, we only uses the entailment samples (positive samples) during training in the stage 3.3, as '**CoPE (w/ Pos)**' in the results table. The second scenario, we only uses the contradiction samples (negative samples) during training in the stage 3.3, as '**CoPE (w/ Neg)**' in the results table. The final scenario, we combine the entailment samples (positive samples) and the contradiction samples (negative samples) during training in the stage 3.3, this is the full version of **CoPE**, as '**CoPE (w/ Pos.+Neg)**' in the results table.

From the results table, it can be seen that our model uses 8B parameters but has competitive results with the large language models. Our result gives the best BLEU score in the negative-only scenario and is 0.42 BLEU score higher than the final model (CoPE). Moreover, it has a BLEU score higher than the GPT3.5-text-davinci-003 model by 1.86 and higher than the ALMA-13B model trained based on the CPO technique by 0.54.

[3] https://github.com/unslothai/unsloth
[4] https://github.com/mjpost/sacrebleu
[5] https://huggingface.co/Unbabel/wmt22-comet-da
[6] https://huggingface.co/Unbabel/XCOMET-XXL

Our experimental results are only behind GPT-4 and WMT Winners, achieving 85.28 score on the COMET-22 measure. From the table, this score is higher than the GPT3.5-text-davinci-003 model by 1.38 score and approximately the same as the ALMA-13B-R model.

When evaluating based on the XCOMET measure, our results are higher than those of WMT Winners and GPT3.5-text-davinci-003 model, only lower than that of GPT-4 by 0.51 score, and approximately equal to that of ALMA-13B-R, which are 88.90 and 89.11 score, respectively.

System

```
You are LANGUAGE EXPERT
### Your task is generated sentences that FOLLOWED defined Instruction
### Response Format
You must response as XML format with these variables: text, sim_tgt, sim_src
### Variable definitions:
  text: generated sentence that FOLLOWED pre-defined Instruction
  sim_tgt: the cosine similarity of the content with the {target_lang} sentence.
  sim_src: the cosine similarity of the content with the {source_lang} sentence.
### Sample output in XML format
<output>
  <s>
  <text>My family was never poor, and I have never gone hungry.</text>
  <sim_tgt>0.4</sim_tgt>
  <sim_src>0.5</sim_src>
  </s>
  <!-- Add more sentences as needed -->
  <s>
  <text>I have never experienced hunger, and my family was not impoverished.</text
      >
  <sim_tgt>0.3</sim_tgt>
  <sim_src>0.2</sim_src>
  </s>
</output>
```

Contradiction

```
###Instruction###
Given {source_lang} sentence: "{source_input}"
and its TRANSLATION into {target_lang}: "{target_input}"

Your task is write 5 sentences that are the OPPOSITE of the {target_lang} sentence
      by using antonyms, adjusting, altering, swapping, changing, or contradicting
      some details in order to express a different meaning.

## Only response in predefined form, DO NOT explain
```

Entailment

```
###Instruction###
Given {source_lang} sentence: "{source_input}"
and its TRANSLATION into {target_lang}: "{target_input}"

Your task is to write 5 sentences that logically follow from or are entailed by
     the {target_lang} sentence using synonyms. These entailments should be
     statements that must be true if the original sentence is true, without adding
      any new information not implied by the original.

## Only response in predefined form, DO NOT explain
```

Fig. 1. Prompts for the contrastive perturbation generation.

5 Analysis

5.1 Prompts for the contrastive generation

We use the following two instruction to generate entailment and contradiction samples, describe in Figure 1. The entailment samples is generate by paraphrasing using different words and sentence structures while preserving its original meaning. The contradiction samples is generate by using antonyms, adjusting, altering, swapping, changing, or contradicting some details in order to express a different meaning.

5.2 What is the best value for k generated samples?

In the stage 3.2, k is the number of samples generated to be negative samples or positive samples. In this section, we also perform different comparisons to choose the best k value. The average results for the corresponding measures with different k values are shown in Table 2. Based on this table of results, we can see that increasing the number of samples in contrastive learning helps to gradually increase the model evaluation metrics and reaches the highest value at $k = 5$ with BLEU=35.57, COMET-22=85.28 and XCOMET=88.90. Continuing to increase the value of k, the results obtained begin to decrease gradually.

	BLEU	COMET-22	XCOMET
$k = 1$	32,53	84,74	84,85
$k = 3$	32,95	84,92	86,17
$k = 5$	**35,57**	**85,28**	**88,90**
$k = 7$	34,68	84,56	86,85

Table 2. The table of results compares different values of k.

6 Conclusion

In this study, we proposed a method **CoPE** to improve the many-to-many multilingual machine translation model. Taking advantage of the advantages of the large language models such as ChatGPT, we provide guidelines to generate quality entailment as positive and contradiction as negative samples, along with evaluating these samples as the basis for the ranking process. Then, we fine-tune a medium-sized language model as LLaMA-3.1-8B and use it to train on the original training dataset. This model, after training, is good enough to represent these positive and negative samples. Finally, we continue train this model based on the contrastive learning technique on both positive and negative samples. The results show that the method achieves efficiency and has higher evaluation metrics than models such as GPT-3.5, approximately achieving the results on the 13B-parameter model as ALMA-13B-R trained based on the contrastive preference optimization technique. This method initially demonstrates the competitiveness of the contrastive learning techniques on models to achieve asymptotic performance with the large language models like GPT-4.

Acknowledgements We express our sincere gratitude to Mr. Vinh Dinh-Quang, Ms. Phuc Nguyen-Hong, and AI VIET NAM, AISOLUS for their invaluable insights and contributions to our research.

References

1. Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger et al. Language Models are Few-Shot Learners. https://doi.org/10.48550/arXiv.2005.14165
2. OpenAI. GPT-4 Technical Report. https://doi.org/10.48550/arXiv.2303.08774
3. Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, et al. Mistral 7B. https://doi.org/10.48550/arXiv.2310.06825
4. Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, Guillaume Lample. LLaMA: Open and Efficient Foundation Language Models. https://doi.org/10.48550/arXiv.2302.13971
5. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. Attention Is All You Need. https://doi.org/10.48550/arXiv.1706.03762
6. Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, Lei Li. Multilingual Machine Translation with Large Language Models: Empirical Results and Analysis. https://doi.org/10.48550/arXiv.2304.04675
7. Haoran Xu, Young Jin Kim, Amr Sharaf, Hany Hassan Awadalla. Llama-adapter: A Paradigm Shift in Machine Translation: Boosting Translation Performance of Large Language Models. The Twelfth International Conference on Learning Representations. 2024. https://doi.org/10.48550/arXiv.2309.11674
8. NLLB Team. No Language Left Behind: Scaling Human-Centered Machine Translation. https://doi.org/10.48550/arXiv.2207.04672
9. Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, et al. Beyond English-Centric Multilingual Machine Translation. https://doi.org/10.48550/arXiv.2010.11125
10. Haoran Xu, Benjamin Van Durme, Kenton Murray. BERT, mBERT, or BiBERT? A Study on Contextualized Embeddings for Neural Machine Translation https://doi.org/10.48550/arXiv.2109.04588
11. Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, Colin Raffel. mT5: A massively multilingual pre-trained text-to-text transformer. https://doi.org/10.48550/arXiv.2010.11934
12. Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, Young Jin Kim. Contrastive Preference Optimization: Pushing the Boundaries of LLM Performance in Machine Translation. 2024 https://doi.org/10.48550/arXiv.2401.08417
13. Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. https://doi.org/10.48550/arXiv.2201.11903
14. Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, et al. Aya 23: Open Weight Releases to Further Multilingual Progress. https://doi.org/10.48550/arXiv.2405.15032

15. Samy Bengio, Oriol Vinyals, Navdeep Jaitly, Noam Shazeer. Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks. https://doi.org/10.48550/arXiv.1506.03099
16. Romain Paulus, Caiming Xiong, Richard Socher. A Deep Reinforced Model for Abstractive Summarization. https://doi.org/10.48550/arXiv.1705.04304
17. Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, Shin Ishii. Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning. https://doi.org/10.48550/arXiv.1704.03976
18. Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, Jingjing Liu. FreeLB: Enhanced Adversarial Training for Natural Language Understanding. https://doi.org/10.48550/arXiv.1909.11764
19. Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. https://doi.org/10.48550/arXiv.1301.3781
20. Lajanugen Logeswaran, Honglak Lee. An efficient framework for learning sentence representations. https://doi.org/10.48550/arXiv.1803.02893
21. Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. https://doi.org/arXiv.2303.16199
22. Timo Schick, Hinrich Schütze. It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners. https://doi.org/10.48550/arXiv.2009.07118
23. Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. How good are gpt models at machine translation? a comprehensive evaluation. 2023. https://doi.org/arXiv:2302.09210
24. Matt Post. A Call for Clarity in Reporting BLEU Scores. 2018. https://doi.org/10.48550/arXiv.1804.08771
25. Ricardo Rei, Craig Stewart, Ana C Farinha, Alon Lavie. COMET: A Neural Framework for MT Evaluation. 2020. https://doi.org/10.48550/arXiv.2009.09025
26. Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, André F.T. Martins. xCOMET: Transparent Machine Translation Evaluation through Fine-grained Error Detection. https://doi.org/10.48550/arXiv.2310.10482
27. Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, André F. T. Martins. COMET-22: Unbabel-IST 2022 Submission for the Metrics Shared Task. https://aclanthology.org/2022.wmt-1.52
28. Wang, Huiming and Li, Zhaodonghui and Cheng, Liying and Soh, De Wen and Bing, Lidong. Large Language Models can Contrastively Refine their Generation for Better Sentence Representation Learning. Proceedings of North American Chapter of the Association for Computational Linguistics (NAACL). 2024 https://doi.org/10.48550/arXiv.2310.10962
29. Chenxin An, Jiangtao Feng, Kai Lv, Lingpeng Kong, Xipeng Qiu, Xuanjing Huang. CoNT: Contrastive Neural Text Generation. 36th Conference on Neural Information Processing Systems (NeurIPS 2022). https://doi.org/ 10.48550/arXiv.2205.14690
30. Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and Andre F. T. Martins. Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In Proceedings of the Seventh Conference on Machine Translation (WMT), pp. 46–68, Abu Dhabi, United Arab Emirates (Hybrid), December 2022. Association for Computational Linguistics. https://aclanthology.org/2022.wmt-1.2