Generating Cross-lingual Summary based on Multilingual Pre-Trained Model

Mengyang Sun, Tianjian Li, Yifan Zhu, *Member, IEEE,* Peng Zhang, and Jie Tang *Fellow, IEEE,*

Abstract—Cross-lingual text summarization is known as generating a summary in one language given an article in another language. The task shows its significance in a number of scenes where gists are concentrated from multilingual documents, such as semantic retrieval, information extraction, and cross-lingual reading assistance. Current solutions can be categorized into three families: pre-translation, post-translation, and one-step methods. However, most open-sourced solutions mainly focus on limited resourceful languages, and there is also a lack of convincing open-sourced methods utilized in practical applications or services. For a better cross-lingual summarization, we provide a solution by first producing mGLM, an auto-regressive language model pretrained on our well-balanced multilingual corpora consisting of 101 languages. Benefiting from the strong generalization ability of mGLM, our cross-lingual summarizer is capable of covering multiple languages and domains by finetuning only on limited supervision upon few languages and domains. The well-balanced corpora bring mGLM stronger performance on low-resource languages while still keeping it competitive among resourceful languages. Efforts on finetuning and engineering are also made in this work to establish an efficient and robust summarizing service. Experiments on benchmarks and real scenarios show the comprehensive capacity of our system.

Index Terms—Language model, multilingual, cross-lingual, pretraining, finetuning, prompt learning, language balancing.

1 Introduction

The task of cross-lingual summarization (i.e. XLS) is known as summarizing articles from one language into gists in another language. There are a number of scenes where cross-lingual summaries show their significance, such as semantic retrieval, information extraction, and cross-lingual reading assistance. For example, news, reports, and other text materials in foreign languages can be auto-summarized into native sentences illustrating their TLDRs, which could help readers quickly capture their interests. More importantly, XLS provides significant support especially for quickly understanding materials in low-resource languages. Current solutions for cross-lingual summarizing can be categorized into three families: pre-translation, post-translation, and one-step methods. Pre-translation frameworks translate source articles first and then summarize the translated text through monolingual summarizers, such as Ouyang et al. [1]; Post-translation frameworks first utilize monolingual summarizers for inner-lingual summarizing and then translate the outputs into target languages, such as Wan et al. [2]; One-step frameworks directly summarize articles into a different language through a well-trained bilingual or multilingual language model, such as Xu et al. [3]. Recently, popular large language models such as ChatGPT [4], etc. also exhibit strong capabilities in XLS among multiple languages, as long as they are properly prompted via a form of user-bot dialogue.

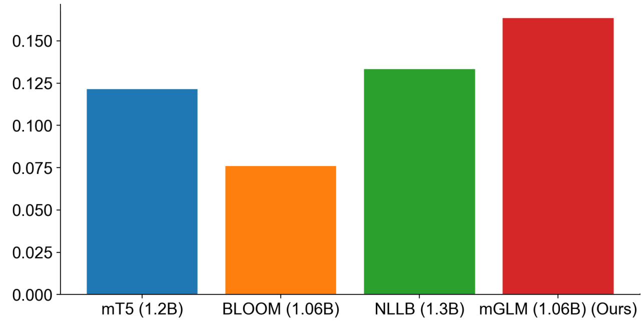


Fig. 1. A Comparison of Same-scaled Novel Multilingual Models on Minor Language Summarizing over ISummCorp

Basically, the reliability of pure post-translation frameworks is questionable since the translating procedure only focuses on concentrated sentences, disregarding the resourceful original context. Specifically, if there is an article about key economic factors for a company's recruitment, and the summary is generated as *"Options can be used to attract and retain talented employees."*. It would be possible for a pure post-translation framework to generate a Chinese summary as "选项(choice)可用于吸引和留住有才华的员工" instead of the correct translation "期权(right to hold share)可

- *M. Sun and J. Tang are with the Department of Computer Science and Technology, Tsinghua University, Beijing, China, 100084. Email: sunmy19@mails.tsinghua.edu.cn, jietang@tsinghua.edu.cn.*
- *T. Li is with the Center for Language and Speech Processing, Johns Hopkins University, MD, USA. Email: tli104@jhu.edu. The work was done when the author was an intern at Zhipu AI.*
- *Y. Zhu is with the School of Computer Science, Beijing University of Posts and Telecommunications, Beijing, China, 100876. Email: yifan_zhu@bupt.edu.cn. The work was done during the author's postdoctoral period at Tsinghua University.*
- *P. Zhang is with the Department of Computer Science and Technology, Tsinghua University, Beijing, China, 100084, and also with the Zhipu AI, Beijing, China, 100083. Email: peng.zhang@aminer.cn*
- *M. Sun and T. Li contributed equally to this work.*
- *Correspondence to J. Tang.*

Manuscript received XXX XX, XXXX; revised XXX XX, XXXX.

用于吸引和留住有才华的员工". Since the word *option* has two different translations in Chinese, and it depends on the context. Furthermore, pre-translation frameworks may not be as questionable as post-translation, but they still suffer the issue of information deviation, which is also known as the error propagation along the pipeline [5]. To overcome the weakness of pure pipeline methods like pre- or post-translation, some solutions focus on combining hints from the original text into the 2nd-step procedure in the pipeline to improve their reliability, such as Wan [6]. However, issues of error propagation still exist, and usually, these approaches also suffer from time consumption. Instead of using monolingual language models, bilingual or multilingual language models provide a one-step solution for XLS, so that they can escape the issue of information deviation. But on the other side, those models heavily rely on the training samples of cross-lingual summaries which are not common in nature, therefore they are still influenced by the quality of machine translation used in creating paired cross-lingual samples.

In recent years, the pretrain-finetune framework has been preeminent in the area of natural language processing (NLP). Massive unlabelled data is trained during the pretraining phase towards different self-supervised objectives. The use of pretrained checkpoints in downstream tasks shows considerable improvements compared to those end-to-end models. Basically, large language models (i.e. LLMs) are capable of containing huge amounts of common knowledge inside themselves after pretraining. As a result, pretrained language models are experts in generalization as well as transfer learning. This capability enables researchers to obtain a universal summarizer by finetuning pretrained models only with a limited set of samples. Moreover, large multilingual models, which are pretrained on a mixture of many languages, are also capable of transferring supervision across distinct languages, such as mBERT [7], XLM-R [8], mBART [9], mT5 [10], BLOOM [11] and NLLB [12] etc. For one-step XLS, this transferring ability is desirable because acquiring labeled data for low-resource languages is expensive. Moreover, based on supervised finetuning or reinforcement learning on instruction datasets, human-aligned LLMs like ChatGPT and ChatGLM [13] are also capable of XLS between distinct languages, as long as they are multilingual and we prompt them with an appropriate demonstration (For example, we can use "*Please summarize the above English article into Chinese:*" to instruct ChatGPT to perform an English-to-Chinese summarization).

However, most existing multilingual LLMs suffer from a lack of support for minor languages. For example, mBART and BLOOM only support 25 and 46 languages respectively; mBERT, XLM-R, mT5, and NLLB support more than 100 languages but still show unsatisfactory performances among low resource ones; LLaMA2 and LLaMA2-Chat [14] are capable of tokenizing multilingual text but most of their pretraining corpora are in English. Although there are variants of LLaMA in Chinese or other languages, the LLaMA family still lacks most languages, especially the minor ones; Some commercial AI assistants such as GPT4 [15], ChatGLM2 [13] and Bard [16], etc. are able to support multilinguality, but they are all resource-consuming and mostly not open-sourced. Specifically, pipelines inside these systems are not fully available. ChatGLM provides an open-sourced version called ChatGLM2-6B [13] but it is actually a bilingual model only supporting English and Chinese. As a result, it's not fair to compare the XLS capacities of those commercial systems with open-sourced multilingual models.

In this paper, we propose an open-sourced cross-lingual text summarizer based on pretraining a new large multilingual model which we refer to as mGLM. Unlike most open-sourced multilingual models which follow encoder-only or encoder-decoder frameworks, mGLM is constructed based on GLM architecture, which is an autoregressive blank infilling framework unifying both the tasks of natural language understanding and generation. We collect a massive amount of 25TB of unlabelled text and feed part of it into a GLM [17] model. Before the training step, we reconstruct the language distribution of our mixed corpora, to guarantee each language contributes ideal importance to the corpora as it should be in the real world. This enables our mGLM to achieve a better performance among low-resource languages, as shown in Figure 1, while still keeping competitive among mainstream languages. The utilization of mGLM promotes us to finetune an effective and efficient multilingual summarizer. We carefully construct the finetuning dataset of XLS over different domains. Experiments show that our model overall provides competitive solutions and especially shows superior performance in low-resource language modeling.

In addition, during the finetuning for cross-lingual generation, we find that multilingual models show great zero-shot abilities for understanding unexposed source languages, but very limited capacity of generating unexposed target languages when we use distinct prompt words to guide them. We refer to this phenomenon as **Lack of Prompt Transferring** in cross-lingual generation after finetuning. We consider it as a generative overfitting phenomenon, not for text understanding but only for text generation. This makes it hard to build cross-lingual generators by task-specific finetuning under limited samples. Moreover, related phenomenons also exist in few-shot in-context learning, but exhibit in a different manner. We will discuss possible ways to relieve this issue at the end of this paper.

The main contributions of this work are as follows:

- We collect and re-balance 25TB text corpora consisting of 101 languages. The re-balancing step for our corpora benefits the performance over low-resource languages.
- We pretrain a multilingual language model named mGLM, which follows an autoregressive blank infilling framework and learns from our well-balanced corpora. mGLM exhibits powerful zero-shot and cross-lingual capacity, as well as a better capacity for modeling low-resource languages.
- Based on mGLM, we conduct cross-lingual text summarization through finetuning mGLM on a carefully constructed dataset consisting of bilingual pairs of summarization. Our summarizer is capable of summarizing multilingual articles into English or Chinese summaries.

2 Background and Related Work

In this section, we involve some backgrounds and essential techniques for constructing a cross-lingual summarizer, including the fundamental pipelines, pretrained language models, multilingualism of language models, and prompt learning paradigm for finetuning, etc.

2.1 Pipelines of XLS

Early XLS work generally focuses on 2-phase pipeline methods whose main idea is decomposing XLS into subtasks of translation as well as monolingual summarization. Two separate categories are pre-translation framework and post-translation framework. Besides, thanks to the rapid development of multilingual neural networks, many one-step models are proposed. Those models markedly outperform 2-phase pipeline methods. In addition, they simplify the effort of training two language models (translator and summarizer) or paying for commercial machine translation services. This section briefly introduces those frameworks of XLS.

Pre-translation Pre-translation pipelines include extractive methods as well as generative methods. Extractive pre-translation methods translate original articles into target language and then select key sentences from the translated text, such as Leuski et al. [18] which summarizes Hindi articles into English sentences. Some of those methods follow a more complicated way of combining both original and translated text during summarizing. For example, Wan [6] computes a saliency score of each translated Chinese sentence with the help of original English text, then selects salient and top-ranked Chinese sentences as summaries; Generative pre-translation methods utilize a generative model for the 2nd-step monolingual summarization. Ouyang et al. [1] train an abstractive summarizer on English pairs of a noisy document and a clean summary, imitating the real scenarios where translators may generate noise in translated text.

Post-translation Post-translation pipelines follow a manner of summarizing and then translating. Orăsan et al. [19] firstly follow that manner to summarize Romanian news to English via a maximum marginal relevance summarizer as well as a translator service. To alleviate the unreliability of translation, Wan et al. [2] trains a model to predict the translating quality of each original sentence, and only selects high-quality sentences to compose summaries. However, as mentioned in the introduction of this paper, post-translation pipelines are often not reliable because of the unavailability of meaningful context. This weakness limits the acceptance of post-translation methods, no matter whether they utilize extractive or generative summarizers. Compared with pre-translation methods, post-translation typically achieves worse performance [5].

One-step One-step methods could be basically categorized into two groups: methods using end-to-end models, and methods using pretrained models. For end-to-end models, training a powerful cross-lingual summarizer is hard due to the lack of resourceful samples of cross-lingual summaries. Therefore, some works focus on multi-task training which integrates related tasks (translation and monolingual summarization) together during training, such as Zhu et al. [20]. Specifically, Zhu et al. [20] train an encoder-decoder framework whose encoder encodes all sequences from the both tasks and uses two independent decoders for each task. Other methods focus on training an end-to-end student model with the help of two models: a translator and a summarizer, such as Shen et al. [21]. However, the end-to-end student models are still not capable of breaking the ceiling of pipeline models; On the other side, the multilingual pretrained generative models recently have exhibited impressive power in many multilingual scenes including XLS. Models like mBART [9], mT5 [10], BLOOM [11], and NLLB [12], etc. all make considerable improvements, compared with previous methods. As a result, this framework absorbs huge amounts of knowledge from massive external corpora and could generally outperform other methods, according to Wang et al. [5]. More recently, instruction-tuned language models, also known as chat models which are refined on well-designed demonstration data via supervised finetuning or reinforcement learning, exhibit impressive performances across common downstream scenes including XLS. Specifically, by converting a XLS task to a form of question-answering or dialogue, we are able to utilize chat models to satisfy XLS requirements.

Pros and Cons Post-translation framework suffers an unreliability of translation. And both pre-translation and post-translation all suffer an issue of information deviation along the pipeline. Besides, 2-step pipelines are also time-consuming and generally require an extra translator to be trained. One-step methods alleviate the issues of pipeline methods and mostly outperform them, especially with the help of pretrained language models. Moreover, the need of universal XLS restricts the large-scale application of bilingual summarizers. As a result, a universal multilingual pretrained model is desirable for comprehensive downstream scenarios of XLS.

Chatting LLMs are experts in satisfying requirements of universal scenes. Certainly, they show their professionality in concentrating articles into summaries. However, current open-sourced chat models are mostly monolingual or bilingual; Although some commercial chatters are able to support many languages, their complete systems are not open-sourced yet. Besides, the outputs of chatting LLMs are usually uncontrollable. That means they may have uncertain lengths or formats, which would be a disadvantage of utilizing commercial chatters in real XLS applications. Additionally, running an effective chatting LLM is usually resource-consuming.

2.2 Pretrained Models and Multilinguality

Benefiting from the massive unlabelled data trained during the pretraining phase, pretrained language models are usually experts in generalization as well as transfer learning across different scenes. For example, some pretrained models are able to satisfy textual tasks in academic scenes only after they are finetuned on a few samples of news reports or novels. It is because pretrained models learn common knowledge across all the scenes during pretraining, so that they are capable of transferring new knowledge from one scene to other scenes with the help of those common knowledge. Existing pretrained language models can be roughly classified into three frameworks [17]: autoregressive framework that learns left-to-right language models, such as GPT [22]; autoencoding framework that learns bidirectional context encoders, such as BERT [23]; and encoder-decoder framework that adopts bidirectional attention for encoder and unidirectional attention for decoder, such as T5 [24].

Many of these language models are originally introduced for English-only text [10]. On the other side, there are some multilingual variants of these language models which are pretrained on a mixture of many languages. Moreover, language models trained on large-scale multilingual corpora exhibit strong cross-lingual transfer learning ability, which means they are able to perform downstream tasks on languages that are never been exposed during finetuning. This cross-lingual transferring ability is desirable because acquiring labeled data for low-resource languages is usually expensive. Popular multilingual models such as mBERT [7], XLM-R [8], mBART [9], mT5 [10], NLLB [12], and BLOOM [11] all make considerable contributions to the area of multilingual modeling.

Following the autoregressive framework of GPT, there is a multilingual language model proposed by BigScience, named **BLOOM**. BLOOM is trained to continue text from a prompt on vast amounts of text data using industrial-scale computational resources. [11] It is designed for outputting coherent text in 46 languages and 13 programming languages that is hardly distinguishable from text written by humans.

mBERT and XLM-R are representatives of autoencoding multilingual models. **mBERT** is the multilingual variant of BERT [23] pretrained on 104 top largest languages in Wikipedia. It is basically the same as BERT except it down-samples resourceful languages and up-samples low-resource languages [25]. It is observed that by multilingual pretraining, mBERT outperforms monolingual BERT when both are finetuned on low-resource languages, since it transfers knowledge from other resourceful languages. **XLM-R** is the multilingual variant of RoBERTa [26] pretrained on 2.5TB of clean CommonCrawl data in 100 languages. XLM-R specifically points out **the curse of multilinguality**: For a fixed-sized model, the per-language capacity decreases as we increase the number of supported languages.

mBART, mT5, and NLLB are representatives of encoder-decoder multilingual models. **mBART** is the multilingual variant of BART [27], which is an encoder-decoder model consisting of a denoising autoencoder that encodes the corrupted document then autoregressively maps the encoded document to its original uncorrupted counterpart. The model is first proposed for neural machine translation and is able to support 25 languages. **mT5** is the multilingual variant of T5 [24], which is also an encoder-decoder model using "text-to-text" format for all text-based NLP problems. Specifically, T5 is trained to output the literal text of the label instead of a class index, which is also regarded as a way of prompt learning. By verbalizing class indices into literal words, T5 can use the same training objectives for every task. mT5 is pretrained on mixed corpora consisting of 101 languages during the training process. **NLLB** is another multilingual encoder-decoder model proposed to support around 200 languages, most of which are low-resource ones. The aim of NLLB is to provide a comprehensive solution for low-resource language translation. They evaluate over 40,000 different translation directions to prove the effectiveness of NLLB. However, according to the curse of multilinguality proposed by [8], a model supporting more languages is more likely to suffer a decrease of per-language capacity. As a result, it is still questionable whether NLLB is really capable of handling massive languages as well as per-language generalization simultaneously.

It is worth noting that, most multilingual models sample pretrained languages by a sampling rate: $p(L) \propto |L|^{\alpha}$, where $|L|$ is the number of examples in a language, and α is a hyperparameter. While mBERT and mBART use $\alpha = 0.7$ as the exponent of sampling rate, XLM-R chooses $\alpha = 0.3$ as an optimal value via a trade-off between high and low resource languages. As a result, XLM-R performs better on low-resource languages. Inspired by XLM-R, mT5 also implements $\alpha = 0.3$ for a balanced performance across all languages. However, even mT5 is not able to provide a satisfactory performance on low-resource languages. That means the sampled language distribution might require a further balancing.

Those pretrained multilingual models already proved their effectiveness in multilingual modeling from various aspects. However, as a foundation model for XLS, those models suffer issues. For example, BLOOM follows a decoder-only architecture which makes it hard to model bidirectional context; mBERT and XLM-R follow an autoencoder framework so that they are not capable of directly generating long text; mBART and mT5 follow an encoder-decoder framework for language generation, but require more parameters to match the performance of BERT-based models [17]. Moreover, they all suffer a lack of minor language support because of their resourceful-language-centered pretraining corpora. As a result, none of the above multilingual models is effective enough for constructing a comprehensive cross-lingual summarizing service.

Additionally, there are other multilingual models proposed in open-sourced communities. However, they are even not regarded as comprehensive multilingual models since most of their pretraining corpora are still in English. For instance, LLaMA2 [14], a language model proposed by Meta, is pretrained on multiple languages but actually, 89.70% of its corpus is in English. A pretraining corpus with such a majority in English means that the model may not be suitable for use in other languages. [14]

2.3 Finetuning, Prompting and In-context Learning

Traditional pretrain-finetune framework trains a language model via some self-supervised objectives during the pretraining phase. However, in the finetuning phase it usually chooses to introduce additional parameters and finetune them using task-specific objective functions [28]. For example, a sentiment analysis or an opinion mining task may introduce a linear or non-linear classifier to conduct their classification jobs. This inconsistency of objectives leads pretrained parameters to adapt to downstream tasks.

As model sizes increase, large language models exhibit powerful capacities in in-context learning, which is also known as few-shot learning or emergence. LLMs are able to continue the input sentences with logically correct information. As a result, LLMs can solve some of downstream tasks as long as they are fed with several demonstrating examples. However, feeding LLMs with more examples extends their processing time, and it is also uncertain that which samples should be selected as demonstrations during the in-context learning.

The powerful in-context capacities only emerge in large language models. But we can still follow prompt learning paradigm during finetuning smaller models. Instead of directly updating pretrained parameters to adapt downstream tasks, those downstream tasks are reformulated as pretraining objectives (e.g. next token prediction, or masked token reconstruction) to close the gap between pretraining and finetuning, with the help of natural language prompts [29]. Basically, a generative task can be reconstructed by a prompt describing the aim of the task, such as ”Translate: _” for a translating task, and ”Answer: _” for a QA task; A classification task can also be reconstructed by a prompt and a verbalizer specifying the target description of a unique category. The predicted label can be obtained by $\hat{y} = argmax_{y \in \mathrm{Y}} \left(\frac{1}{|\mathrm{V}_y|} \sum_{v \in \mathrm{V}_y} P([MASK] = v|x) \right)$, where y denotes a specific class, and V_y represents all the verbalizers of class y.

In this paper, we focus more on prompt-based fintuning, since our efficient summarizing model is relatively small compared with those large language models like GPT3 [30]. To conduct a fair evaluation, we choose to compare our model with same-sized multilingual models.

3 Multilingual GLM

We design and construct the multilingual GLM (mGLM) as the foundation of our cross-lingual summarizer. mGLM follows the general model architecture as well as the training manner of GLM [17]. Figure 2 provides two examples illustrating how mGLM is pretrained on unlabeled corpora (such as English), and finetuned for Chinese title generation. The major difference between GLM and mGLM is the vocabulary. mGLM is constructed based on a multilingual vocabulary of 250k words. To balance the effectiveness and efficiency, we set the hidden size to 1536, the number of layers to 24, and the number of heads to 16. The maximum length of training sequences is 512 and we train 2,560 sequences for each batch. The models are trained on 64 NVIDIA Tesla A100 80GB GPUs for 600k iterations. As a result, the parameter size of our model is around 1.06B, and the number of tokens we pretrained from corpora is around 800 billion. Those are also at the same scale of $mT5_{Large}$, whose model size is around 1.2B and the number of pretrained tokens is around 1 trillion. Thus, we also refer to our model as $mGLM_{Large}$. Other same-scaled multilingual models are also mentioned in this paper, which we refer to as $BLOOM_{Large}$ (1.06B) and $NLLB_{Large}$ (1.3B).

3.1 GLM: General Language Model

In this section, we review a novel autoregressive model called General Language Model, or GLM for short [17]. GLM proposes an autoregressive blank infilling framework to address the challenge of unifying various self-supervised pretraining objectives. With the help of 2D positional encoding, GLM can formulate NLU tasks as cloze questions that contain task descriptions, and autoregressively answer them by generation.

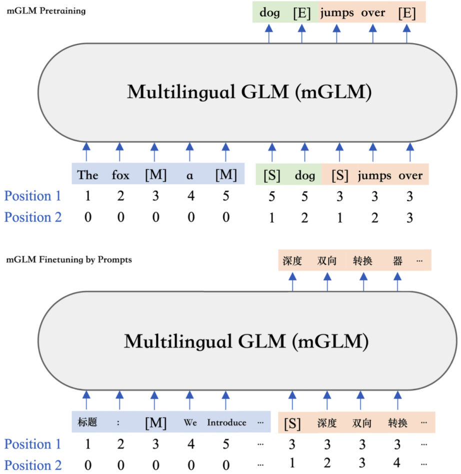


Fig. 2. Examples of Pretraining and Finetuning mGLM through Autoregressive Blank Infilling

Given an input sequence $\boldsymbol{x} = [x_1, x_2, ...x_n]$, GLM randomly samples multiple text spans $\boldsymbol{s}_1, \boldsymbol{s}_2, ...\boldsymbol{s}_m$ and each span corresponds to a consecutive list of tokens $[x_i, x_{i+1}, ...x_j]$. Those spans are corrupted in the original sequence and GLM is trained to predict the missing tokens in an autoregressive manner at the end of input corrupted sequences. Unlike BERT which assumes the independency between tokens, GLM takes inter-dependencies of masked spans into consideration and fully captures the dependencies by randomly permuting the generating order of spans. The definition of GLM pretraining objective can be formulated in Eq. 1,

$$\max_{\theta} \mathbb{E}_{z \sim Z_m} \left[\sum_{i=1}^{m} \log p_{\theta} \left(s_{z_i} \mid x_{corrupt}, s_{z_{<i}} \right) \right], \qquad (1)$$

where m is the number of corrupted text spans; Z_m denotes the set of all possible permutations; a randomly permutated sequence is denoted by z and z_i is the i-th corrupted span of z; $z_{<i}$ denotes the sub-sequence of z which ends at the $(i-1)$-th corrupted span. Within each span, GLM always follows a left-to-right order to generate each token. To control the generating process of GLM, each span starts with [START] token and ends with [END] token. That is, after an [END] being generated for the previous token, a [START] is fed into the model for the next token. GLM automatically learns a bidirectional encoder for $x_{corrupt}$ and a unidirectional decoder for the masked spans. Like many other transformer-based models, GLM also encodes positional relations by positional embeddings. The difference is that GLM uses a 2D positional encoding for both inter- and intra-span positions.

3.2 Multilingual Corpora and Language Balancing

Our multilingual pretraining corpora are originally collected from the Internet. The overall unlabeled text is around 25TB and is categorized into 101 languages with the help of language detection. The language distribution of our raw corpora is relatively unbalanced. For example, there is too much English text to train a mixed language model satisfied on low-resource languages. To illustrate the distribution of languages within our corpora, we plot the distribution in Figure 3. That unbalanced distribution implies a possible under-fitting over low-resource languages, since we would spend the majority of our training efforts on processing samples of those resourceful languages. However, we must also acknowledge that resourceful languages are more likely to contain useful common knowledge. For example, most academic papers and encyclopedias are written in English. As a result, an effective way to refine language distribution is desired for trading off between language balancing as well as common knowledge obtaining.

Here we assume a power law distribution of real-world languages. As discovered in previous statistical findings, power law distribution is suitable for various scenarios, including the distribution of word usage frequency, paper amounts of scientists, and also the wealth of individuals. We consider power law distribution as a result of free competition and the Matthew effect, and choose to take it to describe how many useful materials are created across languages.

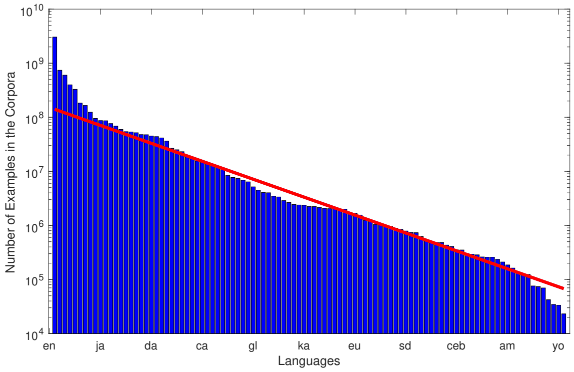


Fig. 3. Logarithmic of Language Distribution

As shown in Figure 3, we reduce the samples of high-resource languages and increase the samples of low-resource languages, to make the logarithmic language distribution fit a linear curve, which is illustrated by the red line in the figure. Basically, those who are considerably influenced by this balancing step are languages with high resources, such as English, Russian, Spanish, French, Italian, etc. It should be noted that this power-law-based balancing also aims to overcome the uncertainty of corpora which are crawled from unreliable open webspace.

According to XLM-R [8] and mT5 [10], an optimal corpora sampling technique for training a Transformer model is to set the hyper exponent of language sampling rate α to 0.3. We choose to follow their traditional manner after our power-law-balancing procedure, and refer to this as a 2-stage language sampling methodology. With the help of the methodology, $mGLM_{Large}$ is able to outperform $mT5_{Large}$ under low-resource languages while still keeping competitive under resourceful languages.

3.3 General Model Evaluation

3.3.1 Zero-shot Multilingual Benchmarks

Zero-shot is traditionally regarded as pretrained models doing unconditional generations without any finetuning. But in multilingual scenarios, a zero-shot task is usually defined as a task that finetunes models in a language and evaluates them in other languages which are not exposed during finetuning. Thus, ”zero-shot” performances of multilingual models can be regarded as their knowledge-transferring capacities across languages.

To evaluate the effectiveness of our mGLM model on zero-shot language transferring, we conduct experiments on three question answering (i.e. QA) tasks from the well-known multilingual benchmark XTREME [31], including XQuAD [32], MLQA [33], and TyDiQA-GoldP [34]. All of those are zero-shot generative benchmarks, that is, generative models are expected to generate textual answers given contexts and questions, and they are required to be finetuned on an English-only training set and evaluated in multilingual scenarios. Specifically, **XQuAD** is a cross-lingual zero-shot QA dataset consisting of 240 paragraphs and 1190 question-answer pairs translated from SQuAD [35] to 10 languages (Spanish, German, Greek, Russian, Turkish, Arabic, Vietnamese, Thai, Chinese and Hindi). **MLQA** is a highly parallel benchmark for evaluating cross-lingual QA in 7 languages (English, German, Spanish, Arabic, Chinese, Vietnamese, and Hindi). It consists of over 5,000 translated QA instances for each language to evaluate. Candidate models are also expected to be trained on SQuAD. Note that MLQA is a cross-lingual evaluation with context and answer in one language and question in another language. Thus, the final F1 and Exact Matching metrics are the averages of those measures of 49 language pairs. **TyDiQA** refers to Typologically Diverse Question Answering. It consists of 204K question-answer pairs in 11 languages (English, Arabic, Bengali, Finnish, Japanese, Indonesian, Kiswahili, Korean, Russian, Telugu, and Thai). The advantage of TyDiQA is the way they collect data: questions are written by people who construct questions without seeing contexts, and then the contexts and answers are collected directly in each language without translation.

English-only prompting strategy is utilized for finetuning and evaluating mGLM in those experiments to instruct the model to generate answers. Specifically, no matter what the input language is, we integrate the English words ”Question: ” before the input question, and ”Answer: ” before the target [sMASK] token. This setting provides consistency between finetuning and evaluating tasks, otherwise the model would lose itself in transferring task knowledge across different prompting languages. (We will deeply analyze the phenomenon in Section 3.4) The zero-shot transferring evaluations of English-finetuned models are presented in Table 1. All performance metrics of previous models are adapted from previous papers. Here EM refers to Exact Matching. It should be noted that we only present those models with a relatively similar model size (around 1B), which means larger models consisting of 10B, 100B, or even more parameters are not included in Table 1. Although they are theoretically effective because of their huge amount of parameters, they are less likely to be deployed as an efficient inferring service under very limited computing resources, compared to those smaller models. Notably, models such as mBERT and XLM-R are encoder-only models, so their effective model sizes should be doubled as they require an extra decoder for text generation. For example, for XLM-R the effective model size would be 550M * 2 = 1.1B.

Illustrated by Table 1, our $mGLM_{Large}$ exhibits competitive capacity in question answering, one of the representative scenarios of conditional text generation. It shows that $mGLM_{Large}$ beats all the previous same-scaled models in XQuAD, while also performing among the top level across MLQA and TyDiQA, especially in terms of EM. We claim the outperformance of our $mGLM_{Large}$ in EM mainly depends on the advantage of GLM framework. Specifically, models like XLM-R and mT5 use an encoder-only or encoder-decoder framework which mainly focuses on encoding the semantics of texts, thus they are more skillful at modeling the semantic distribution of their generations, but less at the coherence and integrity. In contrast, mGLM follows an autoregressive framework which mainly focuses on the conditional probability distribution of the next words or phrases, thus mGLM is more skillful at modeling the coherence and smoothness of generations. As a result, $mGLM_{Large}$ could provide more complete answers so that it can outperform encoding models on EM measures even it may be not that powerful on the F1 measures.

TABLE 1
Performances on Zero-shot Benchmarks

Model	Paramaters	XQuAD	MLQA	TyDiQA
		F1/EM	F1/EM	F1/EM
mBERT	180M	64.5/49.4	61.4/44.2	59.7/43.9
XLM-R	550M	76.6/60.8	**71.6**/53.2	65.1/45.0
MMTE	565M	64.4/46.2	60.3/41.4	43.6/29.1
$mT5_{Large}$	1.2B	77.8/61.5	71.2/51.7	**69.9**/52.2
$mGLM_{Large}$	1.0B	**83.6/71.9**	67.4/**54.3**	69.6/**55.6**

3.3.2 Cross-lingual Generation and Transferability

Cross-lingual generation can be regarded as generating sequences of an expected language on the condition of providing another language. The expected language is signaled by prompting words or phrases. Language models are expected to understand the input in multiple languages and generate the output in desired language as the prompt requires. Taking cross-lingual summarization as an example, one of possible templates can be designed as "[source text]. English Summary: [MASK]" for summarizing multilingual texts to English abstracts, and "[source text]. 中文摘要：[MASK]" for Chinese abstracts.

Basically, it is considerable hard for language models to be finetuned on a *language-A-to-language-A* generative task then can obtain the capacity of handling a *language-B-to-language-C* job, since language models only regard the tuning task as an inner-language task without any language crossing. It would always generate language-B sequences if inputs of language B are provided. As a result, we have to induce the hint of doing language-crossing during the stage of finetuning. There are four settings for multilingual models to follow when evaluating their performances of cross-lingual generation: (1). Zero-shot Transferring: Training multilingual models on a bilingual-crossing training set (e.g. the source article is in English and the summary is in Chinese), and then evaluating finetuned models on various languages and directions of cross-lingual generation beyond English-to-Chinese (such as French-to-German); (2). Zero-shot Source Transferring: Similar to zero-shot transferring, but fixing the target language as the trained one during the evaluation, such as French-to-Chinese or Russian-to-Chinese, etc.; (3). Zero-shot Target Transferring: Similar to zero-shot transferring, but fixing the source language as the trained one during the evaluation, such as English-to-French or English-to-Russian, etc.; and (4). Mixed Training: Training models on a mixed dataset consisting of all possible pairs of source and target languages, which is the same as the evaluation scenario. As we claimed in the previous section, collecting labeled data for multiple languages and directions is quite expensive, especially for low-resource languages. Therefore, we pay more attention on the first three settings, especially on the source and target language transferring respectively. The capability of handling the scenario of setting (1) actually consists of those two powers. Consequently, we start from a basic XLS task in this section, and then go on to more complex scenarios such as zero-shot transfer learning respectively for source and target languages.

We conduct evaluations of cross-lingual summarization and title generation for our mGLM model as well as $mT5_{Large}$ on the following datasets: NCLS [20] and MTG [36]. Here we each briefly introduce the two datasets, and provide our findings.

NCLS En2ZhSum NCLS dataset is constructed by [20] through a round-trip translation strategy. It contains two summarization tasks. The first one En2ZhSum is summarizing English articles to Chinese, and the second Zh2EnSum is summarizing Chinese microblogs to English. Specifically, they unite the CNN/DailyMail [37] as well as MSMO [38] and translate the summaries into Chinese to construct their En2ZhSum dataset. Here we only utilize their En2ZhSum dataset because it's more qualified. For a comparison, we construct the mT5 model by obtaining the pretrained $mT5_{Large}$ checkpoint directly from HuggingFace and implement its finetuning script. We finetune our $mGLM_{Large}$ as well as $mT5_{Large}$ on NCLS En2ZhSum dataset until they achieve their respective optimal performance over the testing set. The performances of two models are shown in Table 2. Here Rouge [39] denotes *Recall-Oriented Understudy for Gisting Evaluation* which mainly focuses on the recall of article gists. Rouge-N denotes recall on N-gram, and Rouge-L denotes recall on longest common subsequences. From Table 2, we observe that same-scaled mGLM outperforms mT5 under the circumstance of English-to-Chinese summarization. Actually, our experiments show that $mGLM_{Large}$ does not even converge yet. The performance is still improving.

TABLE 2
NCLS En2ZhSum Performances

Model	Rouge-1/Rouge-2/Rouge-L
$mT5_{Large}$	42.3/22.4/31.3
$mGLM_{Large}$	**43.2/26.7/32.1**

Test on NCLS En2ZhSum proves the effectiveness of

TABLE 3
MTG Zero-shot Cross-Title Generation Performances (Rouge-1/Rouge-2/Rouge-L)

Model			**Finetuned on English-to-Chinese**			
	en2zh	fr2zh	es2zh	de2zh	zh2zh	Avg.
$mT5_{Large}$	-	30.9/10.1/28.8	31.3/10.6/29.3	30.1/9.9/28.2	**35.0**/13.7/**32.7**	31.8/11.1/29.7
$mGLM_{Large}$	-	**32.4/12.3/30.3**	**33.0/12.6/30.8**	**31.5/11.9/29.5**	33.4/**13.8**/31.1	**32.6/12.6/30.4**

Model			**Finetuned on French-to-Chinese**			
	en2zh	fr2zh	es2zh	de2zh	zh2zh	Avg.
$mT5_{Large}$	32.9/11.8/30.9	-	32.7/11.2/30.5	31.5/10.6/29.6	**36.3/14.7/33.9**	33.4/12.1/31.2
$mGLM_{Large}$	**34.6/14.1/32.1**	-	**34.0/13.4/31.6**	**32.0/12.1/29.7**	34.3/14.2/31.9	**33.7/13.4/31.3**

mGLM on cross-lingual generation. Subsequently, We also conduct experiments on MTG zero-shot cross-lingual titling tasks, to illustrate whether this cross-lingual capability is transferable across different languages. The reason why we select a title generation task is because we expect to bring some diversity to the evaluation and prompting template beyond summarization. Differing from summaries, titles are usually much shorter than summaries and mostly appear in front of articles in the form of one sentence. Thus we follow the manner to complete prompting input, such as "中文标题：[MASK] 原文：[source text]" requiring for a Chinese title.

MTG Zero-shot Cross-lingual Titling MTG is a new benchmark suite for training and evaluating multilingual text generation. It consists of five languages as well as four multilingual text generation tasks including title generation. Data in five languages are all parallel. For title generation, MTG provides a more-than-270k training set as well as a 2k dev set represented in each of the languages. The builders of MTG originally propose their benchmark metrics for cross-lingual title generation but not include a metric for zero-shot. Here we only utilize their dataset instead of evaluating by their metrics, since we care more on the zero-shot cross-lingual generation rather than their benchmarks. Specifically, we build two tasks using MTG title generation dataset: (1) Finetuning models to generate Chinese titles for English articles and evaluating on multilingual articles; and (2) Finetuning models to generate Chinese titles for French articles and evaluating on multilingual articles. To construct a validation set, we unite articles in all the other four languages and titles in Chinese. As a result, the size of validation set for our newly-defined task equals to $2000 \times (5-1) = 8000$. Rouge metrics are then computed on the generations of models. Numbers are provided in Table 3. Here we set same configurations for both $mGLM_{Large}$ and $mT5_{Large}$: maximum length of source texts equals to 768, maximum length of target titles equals to 64, beam size equals to 5, length penalty equals to 0.7 and no-repeating N-gram size equals to 3. Both $mGLM_{Large}$ and $mT5_{Large}$ are finetuned until they achieve respective optimal performances on the validation set of our tasks.

To further explore the differences between mT5 and mGLM, we iteratively evaluate the two finetuned models on each of four languages. More results in Table 3 reveal the point: It can be discovered that, in order to achieve an optimal average performance among four languages, $mT5_{Large}$ achieves a high result on Chinese-to-Chinese evaluation, which even outperforms $mGLM_{Large}$. But at the same time, it is considerably beaten by $mGLM_{Large}$ on summarizing other languages (e.g. French, English, Spanish and German) to Chinese. That phenomenon implies that $mT5_{Large}$ may not transfer knowledge equally to different language pairs. Our mGLM has a relatively better capability of cross-lingual generation compared to mT5.

3.3.3 Domain Transferring

Domain transferring is regarded as models purely finetuned in a specific field achieving strong performance in another domain. We consider that it is important for a summarization model to have strong in-domain and out-of-domain performance. In this subsection we show that our model have strong domain transfer ability. Here we utilize Scisummnet [40] dataset, which is originally proposed by Yasunaga et al. [40] with 1,009 papers and their comprehensive summaries. Scisummnet papers are extracted from ACLanthology, and the topics of those papers include Natural Language Processing and Computational Linguistics. To perform Scisummnet for cross-lingual generation, we translate the summaries into Chinese for a cross-lingual golden data, which we refer to as Scisummnet-zh.

Both $mGLM_{Large}$ and $mT5_{Large}$ are finetuned on NCLS En2ZhSum dataset and then evaluated on Scisumment-zh to illustrate their domain transferring performances. Surprisingly, Table 4 specifies our $mGLM_{Large}$'s much higher domain transferring capability over $mT5_{Large}$. Here we have really tuned $mT5_{Large}$ through different ways such as adjusting the learning rate or batch size, etc. We consider a possible reason is that the pretraining corpora of mT5 is relatively domain-monotonous. Differences of training frameworks of mGLM and mT5 are not very crucial in this domain transferring scenarios.

TABLE 4
NCLS-tuned models Evaluated on Scisumment-zh

Model	**Rouge-1/Rouge-2/Rouge-L**
$mT5_{Large}$	22.4/ 4.4/15.9
$mGLM_{Large}$	**32.4/10.0/20.2**

3.4 Lack of Prompt Transferring

Cross-lingual capacity based on transfer learning consists of two aspects: whether models finetuned under bilingual-crossing can be transferred to process multilingual inputs; and whether they can be transferred to generate multilingual outputs through various prompts. Experiments on

NCLS and MTG only illustrate the first aspect, and here we pay more attention on the second. We claim that **current finetuning-based language models cannot perform prompt transferring**. For example, if we finetune language models to generate Chinese summaries for source articles by a template "中文摘要：", it is nearly impossible for those finetuned models to generate English summaries when we hint them by a new phrase, for example, "English Summary:". We refer to this phenomenon as **Lack of Prompt Transferring**.

TABLE 5
Transferring Summarizing Target to English

Model	Summarizing to English Avg.
$mGLM_{Large}$ (finetuned on en2zh)	14.2/3.9/12.2
$mGLM_{Large}$ (finetuned on fr2zh)	11.8/2.9/10.7
$mGLM_{Large}$ (without finetuning)	19.3/3.9/15.8

Specifically, if two prompt phrases indicate the same meaning but are in different languages, we can refer to them as *parallel prompts*. To illustrate the lack of parallel-prompt-transferring capacity of multilingual LMs, a summarizing-to-English evaluation is conducted on both two $mGLM_{Large}$ models respectively finetuned by English-to-Chinese task and French-to-Chinese task. The only difference is replacing the "中文摘要：" by "English Summary:" and also "TL;DR:". Results in Table 5 illustrate this lack of transferring. Models can only provide very limit capacity of handling prompt parallel changes after they have already been trained by another specific parallel prompt, **even worse than the model without finetuning**. Such a phenomenon makes us to reconsider the paradigm of transfer learning: There should be a difference between understanding and generation under the cross-lingual transferring scenario.

Based on researches of Deep Learning, low model layers are responsible for modeling local and low-level knowledge, while higher layers are responsible for modeling abstractive and high-level knowledge. Inspired by the researches, we propose a possible explanation for the lack of multilingual prompt transferring: We claim that inside deep multilingual models, the low layers are responsible for modeling language usages and the higher layers are responsible for modeling task knowledge. As a result, when we finetune models on a specific language pair of cross-lingual tasks, those high layers are updated much more than low layers since the skills of fundamentally understanding words and sentences are common among all tasks. Therefore, after finetuning, the low layers are still able to model low-level language-specific knowledge, such as words and grammars, while the high layers are "overfitted" for the specific task prompted by a unique word or phrase. This leads to a satisfied performance on transferring learnt understanding ability across languages, but an unsatisfied performance on transferring learnt generating ability across languages. We consider this lack of prompt transferring as the bottleneck of constructing a cross-lingual summarizer under multilingal models finetuned on limited samples. Because of that, a cross-lingual summarizer could only produce summaries in specific languages included in the labelled outputs of finetuning dataset. As a result, our proposed cross-lingual summarizer could only focus on two main scenarios: All-to-Chinese summarizing and All-to-English summarizing.

As large language models become increasingly popular in the fields of NLP, further thinkings of prompt transferring are also made by us for in-context learning instead of finetuning. In finetuning, there exhibits an issue of overfitting as discussed in the above paragraph, but for in-context learning there is no overfitting. We usually provide few examples in the context of input to illustrate the specific task LLMs are required to solve. However, we found similar issues during in-context learning when we test ChatGPT, shown in Table 6. Since examples of a summarization task consist of several soure articles and summaries which are too long to integrate into the context of ChatGPT, here we define a new task still complex enough to test LLMs, but much shorter: LLMs are required to output a double-negative sentence converted and translated from the given English. We input with a task description and several examples. To robustify the outputs, we call ChatGPT five times and collect all the responses. Table 6 illustrates that transferring the task knowledge from Chinese to Japanese is not always able to work. As a result, in-context learning by LLMs is still not able of totally overcoming parallel prompt issues.

TABLE 6
Prompting ChatGPT (3.5) via Parallel Prompts

Input1	Please double deny and then translate the sentence to the target language as it requires: For example, the input sentence "No one can give you brightness expect yourself.". You should firstly double deny it to "Only you can give yourself brightness." and translate it to "只有你自己可以带给自己光明。", if it requires a Chinese sentence by demonstrative word "Chinese: ". (Note that there maybe other languages required.) Here are some examples: I want someone who's afraid of losing me. Chinese: 我不想要一个不害怕失去我的人。 I want to be the only one. Chinese: 我不想成为任何人。 Time is really a wonderful thing. Chinese: 时间不是平庸之辈。 I will appreciate the people who helped me. Chinese: 我不会感激那些没有帮助过我的人。 No one can predict the future except the god. Chinese:
ChatGPT (3.5)	Output1: "只有上帝能预知未来" Output2: "只有神才能预测未来" (Both mean "Only god can predict the future". Both are correct.)
Input2	Same as Input1 except the last line. The last line is "Japanese: "
ChatGPT (3.5)	Output1: "神以外のも未来を予知することはできない" Output2: "神以外のも未来を予知できません" Output3: "神だけが未来を予知できる" (The first two mean "No one can predict the future except the god". The last one means "Only god can predict the future". Only the last one is correct.)

3.5 Low-resource Language

In this section, we implement experiments to evaluate the performances of our $mGLM_{Large}$ on low-resource languages. As a comparison, we also involve three same-scaled multilingual models, which we refer to as $mT5_{Large}$ (1.2B), $BLOOM_{Large}$ (1.06B), and $NLLB_{Large}$ (1.3B). Specifically, $mT5_{Large}$ and $NLLB_{Large}$ follow an encoder-decoder framework, while $BLOOM_{Large}$ follows a autoregressive decoder-only framework. We don't involve encoder-only models like mBERT, since they require an extra decoder model. Other multilingual models like mBART are also not involved because they are not capable of supporting enough low-resource languages. Here we utilize a low-resource language dataset called ISummCorp [41], which is a large-scale multilingual summarization dataset for eight Indian languages, including Hindi, Tamil, Telugu, Bengali, Gujarati, Marathi, Malayalam, and Kannada. The dataset consists of 376k articles each with a summary written in the language of itself. However, we didn't find Hindi documents on their public Github repository, thus we only use the rest seven languages to conduct our evaluation.

We finetune $mGLM_{Large}$, $mT5_{Large}$, $BLOOM_{Large}$, and $NLLB_{Large}$ respectively on ISummCorp. Experimental results are shown in Table 7. It can be illustrated from the Table that, our $mGLM_{Large}$ outperforms the other three same-scaled multilingual models in terms of Rouge metrics. Specifically, $mT5_{Large}$ and $NLLB_{Large}$ perform similarly, while $NLLB_{Large}$ performs a little bit higher than $mT5_{Large}$. Additionally, $BLOOM_{Large}$ provides an unsatisfied performance compared with the other three models. We explain this phenomenon by considering the language distribution of their pretraining corpora. For example, after the 2-stage language balancing, the seven Indian languages account for 6.12% of our mGLM's pretraining corpora. On the other hand, only 4.65% of mT5's pretraining corpora are in those seven languages. And the corresponding number of BLOOM is only 1.04%.

TABLE 7
Low-resource Languages Performance on ISummCorp

Multilingual Model	Rouge-1/Rouge-2/Rouge-L
$mT5_{Large}$	12.4/5.3/12.1
$BLOOM_{Large}$	7.8/3.6/7.6
$NLLB_{Large}$	13.6/6.1/13.3
$mGLM_{Large}$	**16.5/8.3/16.3**

4 Applicable Cross-lingual Summarizer

The zero-shot transferring ability of language models for cross-lingual generation brings much feasibility to real world applications. By a well-pretrained language model, it is easy to quickly develop a cross-lingual application through a finetuning on a bilingual-crossing task. The only gap is the lack of parallel prompt transferring. As an example, we develop an article summarizer which aims to concentrating multilingual articles to a few Chinese or English sentences. The way we implement that is to simply finetune two prompts illustrating two tasks of both summarizing to Chinese and English respectively.

4.1 Data Construction

We carefully construct the finetuning dataset. For summarizing articles to Chinese, we mix NCLS En2ZhSum dataset with part of summary-translated SCITLDR [42], a dataset consisting of 5.4K TLDRs over 3.2K papers. SCITLDR contains both author-written and expert-derived summaries with high quality; For summarizing to English, similarly, we utilize NCLS Zh2EnSum dataset as well as SCITLDR with article translated. Besides, we also insert some Xwikis [43] samples to improve the performance of English summarization. Although the construction of dataset still requires machine translation, it considerably alleviates the translating bias since we only translate across resourceful languages and leave the knowledge of low resource languages to be transferring learnt by mGLM itself.

4.2 Hyper-parameters and Inferencer Deployment

Generative models require a series of hyper-parameters to control the procedure of generation. For example, beam size controls the solution spaces in each generating step. Length penalty factor controls a regularization of output length. No repeat ngram size defines the minimal length of repeated spans prohibited. Temperature controls the diversity and randomness of a generation. Maximum sequence length controls the target length of output text. Under a proper range of cross validation through human evaluation, we finally optimize those hyper-parameters: beam size is set to 4, length penalty is set to 0.7, no repeat ngram size is set to 3, temperature is set to 0.7, and maximum sequence length is set to 200. Note that a larger beam size usually provides better results but would introduce serious computational overhead. As far as we experienced, those parameters could provide a reliable and effective solution.

To establish an efficient tool, we deploy a fast inferencer with the help of BMInf. Requirements of asynchronous service are satisfied by uvicorn. Inferencers based on BMInf are wrapped as Docker images for flexible migration. As a result, a 1B-model-based inferencer can be deployed on a GPU card using only 4GB space.

5 Conclusion

In this paper, we propose a methodology of XLS based on pretraining a large multilingual language model named mGLM. Compared with pipeline and end-to-end methods, it is globally acceptable that pretrained language models exhibit outperformance because of the huge amount of pretraining corpora. We pretrain mGLM by following the architecture of GLM on a massive mixture of 101 languages. We preprocess the corpora by a 2-stage balancing to improve the model performance under low-resource languages. After pretraining, mGLM shows effective performances across different downstream generative tasks, including multilingual QA and XLS. Moreover, mGLM outperforms other same-scaled models in summarizing low-resource languages, while still keeping competitive for resourceful languages. The number of parameters in mGLM is around 1B, which balances the trade-off between effectiveness and efficiency. Furthermore, we propose a phenomenon which we refer to as the Lack of Prompt Transferring, illustrating the lack of transferring learnt knowledge of a prompt to a parallel prompt in another language. Based on the transferability of mGLM, we implement a cross-lingual summarizer by a bilingual-crossing finetuning.

In the future, we would mainly focus on two aspects: Overcoming the lack of prompt transferring, and handling long input articles. A possible way to overcome the lack of prompt transferring would be restricting the higher layers' updates during the finetuning, such as finetuning on a mixed dataset consisting of XLS data as well as other common multilingual corpora. To handle long articles, techniques of memorized transformers could be possibly integrated into our work.

Acknowledgments

This work was supported by Technology and Innovation Major Project of the Ministry of Science and Technology of China under Grant 2020AAA0108400, NSF of China for Distinguished Young Scholars (61825602), China Postdoctoral Science Foundation (2022M711814), Tsinghua-Bosch Joint ML Center, and a research fund from Zhipu.AI.

References

[1] J. Ouyang, B. Song, and K. McKeown, "A robust abstractive system for cross-lingual summarization," 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)*, 2019, pp. 2025–2031.

[2] X. Wan, H. Li, and J. Xiao, "Cross-language document summarization based on machine translation quality prediction," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 2010, pp. 917–926.

[3] R. Xu, C. Zhu, Y. Shi, M. Zeng, and X. Huang, "Mixed-lingual pre-training for cross-lingual summarization," *arXiv preprint arXiv:2010.08892*, 2020.

[4] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe, "Training language models to follow instructions with human feedback," 2022.

[5] J. Wang, F. Meng, D. Zheng, Y. Liang, Z. Li, J. Qu, and J. Zhou, "A survey on cross-lingual summarization," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 1304–1323, 2022.

[6] X. Wan, "Using bilingual information for cross-language document summarization," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2011, pp. 1546–1555.

[7] e. jacobdevlin. (2019) bert/multilingual.md. [Online]. Available: https://github.com/google-research/bert/blob/master/multilingual.md

[8] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," *CoRR*, vol. abs/1911.02116, 2019. [Online]. Available: http://arxiv.org/abs/1911.02116

[9] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer, "Multilingual denoising pre-training for neural machine translation," *CoRR*, vol. abs/2001.08210, 2020. [Online]. Available: https://arxiv.org/abs/2001.08210

[10] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, "mT5: A massively multilingual pre-trained text-to-text transformer," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, Jun. 2021, pp. 483–498. [Online]. Available: https://aclanthology.org/2021.naacl-main.41

[11] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé *et al.*, "Bloom: A 176b-parameter open-access multilingual language model," *arXiv preprint arXiv:2211.05100*, 2022.

[12] M. R. Costa-jussà, J. Cross, O. Çelebi, M. Elbayad, K. Heafield, K. Heffernan, E. Kalbassi, J. Lam, D. Licht, J. Maillard *et al.*, "No language left behind: Scaling human-centered machine translation," *arXiv e-prints*, pp. arXiv–2207, 2022.

[13] A. Zeng, X. Liu, Z. Du, Z. Wang, H. Lai, M. Ding, Z. Yang, Y. Xu, W. Zheng, X. Xia *et al.*, "Glm-130b: An open bilingual pre-trained model," *arXiv preprint arXiv:2210.02414*, 2022.

[14] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom, "Llama 2: Open foundation and fine-tuned chat models," 2023.

[15] OpenAI, "Gpt-4 technical report," 2023.

[16] J. Manyika, "An overview of bard: an early experiment with generative ai," Technical report, Google AI, Tech. Rep., 2023.

[17] Z. Du, Y. Qian, X. Liu, M. Ding, J. Qiu, Z. Yang, and J. Tang, "GLM: General language model pretraining with autoregressive blank infilling," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 320–335. [Online]. Available: https://aclanthology.org/2022.acl-long.26

[18] A. Leuski, C.-Y. Lin, L. Zhou, U. Germann, F. J. Och, and E. Hovy, "Cross-lingual c* st* rd: English access to hindi information," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 2, no. 3, pp. 245–269, 2003.

[19] C. Orăsan and O. A. Chiorean, "Evaluation of a cross-lingual romanian-english multi-document summariser," 2008.

[20] J. Zhu, Q. Wang, Y. Wang, Y. Zhou, J. Zhang, S. Wang, and C. Zong, "Ncls: Neural cross-lingual summarization," *arXiv preprint arXiv:1909.00156*, 2019.

[21] S.-q. Shen, Y. Chen, C. Yang, Z.-y. Liu, M.-s. Sun *et al.*, "Zero-shot cross-lingual neural headline generation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 12, pp. 2319–2327, 2018.

[22] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving language understanding by generative pre-training," 2018.

[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 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)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. [Online]. Available: https://aclanthology.org/N19-1423

[24] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020. [Online]. Available: http://jmlr.org/papers/v21/20-074.html

[25] S. Wu and M. Dredze, "Are all languages created equal in multilingual bert?" in *ACL Workshop on Representation Learning for NLP (RepL4NLP)*, 2020.

[26] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized BERT pretraining approach," *CoRR*, vol. abs/1907.11692, 2019. [Online]. Available: http://arxiv.org/abs/1907.11692

[27] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv preprint arXiv:1910.13461*, 2019.

[28] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," *arXiv preprint arXiv:2107.13586*, 2021.

[29] T. Schick and H. Schütze, "Exploiting cloze-questions for few-shot text classification and natural language inference," in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Online: Association for Computational Linguistics, Apr. 2021, pp. 255–269. [Online]. Available: https://aclanthology.org/2021.eacl-main.20

[30] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020.

[31] J. Hu, S. Ruder, A. Siddhant, G. Neubig, O. Firat, and M. Johnson, "Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation," in *International Conference on Machine Learning*. PMLR, 2020, pp. 4411–4421.

[32] M. Artetxe, S. Ruder, and D. Yogatama, "On the cross-lingual transferability of monolingual representations," *arXiv preprint arXiv:1910.11856*, 2019.

[33] P. Lewis, B. Oğuz, R. Rinott, S. Riedel, and H. Schwenk, "Mlqa: Evaluating cross-lingual extractive question answering," *arXiv preprint arXiv:1910.07475*, 2019.

[34] J. H. Clark, E. Choi, M. Collins, D. Garrette, T. Kwiatkowski, V. Nikolaev, and J. Palomaki, "Tydi qa: A benchmark for information-seeking question answering in typologically diverse languages," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 454–470, 2020.

[35] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," *arXiv preprint arXiv:1606.05250*, 2016.

[36] Y. Chen, Z. Song, X. Wu, D. Wang, J. Xu, J. Chen, H. Zhou, and L. Li, "Mtg: A benchmark suite for multilingual text generation," in *Findings of the Association for Computational Linguistics: NAACL 2022*, 2022, pp. 2508–2527.

[37] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom, "Teaching machines to read and comprehend," *Advances in neural information processing systems*, vol. 28, 2015.

[38] J. Zhu, H. Li, T. Liu, Y. Zhou, J. Zhang, and C. Zong, "Msmo: Multimodal summarization with multimodal output," in *Proceedings of the 2018 conference on empirical methods in natural language processing*, 2018, pp. 4154–4164.

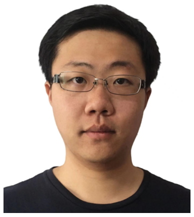
[39] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.

[40] M. Yasunaga, J. Kasai, R. Zhang, A. R. Fabbri, I. Li, D. Friedman, and D. R. Radev, "Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 7386–7393.

[41] V. L. Sireesha, "Text summarization for resource-poor languages: Datasets and models for multiple indian languages," Ph.D. dissertation, International Institute of Information Technology Hyderabad, 2023.

[42] I. Cachola, K. Lo, A. Cohan, and D. S. Weld, "Tldr: Extreme summarization of scientific documents," *arXiv preprint arXiv:2004.15011*, 2020.

[43] L. Perez-Beltrachini and M. Lapata, "Models and datasets for cross-lingual summarisation," *arXiv preprint arXiv:2202.09583*, 2022.

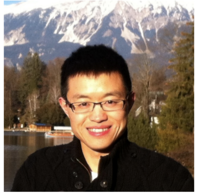


Mengyang Sun received a B.E. degree in Information Security from Shanghai Jiao Tong University, Shanghai, China, in 2016, and a M.S. degree in Computer Science from New York University, NY, USA, in 2019. He is now taking a Ph.D. program in Computer Science in the Department of Computer Science and Technology, Tsinghua University, Beijing, China, from 2019. His research is mainly on knowledge graph and large language models.

Tianjian Li received a B.A. degree in Mathematics and Computer Science from New York University, NY, USA. He is now pursuing a M.S. degree at Johns Hopkins University. He is currently doing research on multilingual language models and machine translation. His research interests lie in the field of deep learning and natural language processing, with a particular focus on the intersection of multilinguality and natural language generation.

Yifan Zhu received the BE degree in computer science from Beijing Information Science and Technology University, Beijing, China, in 2016, and the PhD degree from the School of Computer Science & Technology, Beijing Institute of Technology, Beijing, China, in 2021. He work as a postdoctoral researcher with the Department of Computer Science and Technology, Tsinghua University, China from 2021 to 2023. Currently, He is an assistant professor with Beijing University of Posts and Telecommunications. His research interests include graph mining, recommendation systems, and knowledge services.

Peng Zhang is the CTO of Zhipu.AI. He graduated from the Department of Computer Science and Technology of Tsinghua University with a Ph.D. in the 2018 Innovation Leadership Project, and his research areas vary from text data mining and semantic analysis, knowledge graph construction and application, to AIGC and large language models. As a principal researcher, he participated in many researches and platforms including AMiner (https://aminer.cn) and XLORE (http://xlore.org), etc., and published more than 10 articles in top conferences such as ICML and ISWC.



Jie Tang is a Webank Chair Professor of the Department of Computer Science at Tsinghua University. He is a Fellow of the ACM/AAAI/IEEE. His interests include artificial general intelligence, data mining, social networks, and knowledge graph. He has published more than 400 research papers in major computer science conferences and journals. He was honored with the SIGKDD Test-of-Time Award.