Treating Models Better for Language-agnostic Understanding

Brian Yu
Kurt Keutzer, Ed.
John DeNero, Ed.

Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2023-148
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2023/EECS-2023-148.html

May 12, 2023

Treating Models Better for Language-agnostic Understanding

Brian Yu

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, in partial satisfaction of the requirements for the degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

Committee

DocuSigned by:
Kurt Keutzer
D267C13DE8B8482...

Kurt Keutzer
Research Advisor

5/9/2023

(Date)

★ ★ ★ ★ ★ ★ ★

DocuSigned by:
John DeNero
5DBBA6B64563435...

John Denero
Second Reader

5/12/2023

(Date)

Abstract

State-of-the-art foundation language models have many strengths that are under-valued. Simultaneously, multilingual NLP lacks a clear goal. In this paper, we propose language-agnostic understanding as the goal of multilingual NLP and demonstrate that leveraging foundation language model strengths directly improves on this goal. We reformulate inputs during supervised finetuning to better leverage foundation language model strengths. We obtain significant improvements on challenging translation tasks compared to a baseline mT5 setup. On a Classical Tibetan to English translation task, these reformulations improve performance up to 2.8 BLEU. On the Flores200 translation benchmark, these reformulations improve performance up to 3.1 chrF++. Our research reveals insights into how models learn from different inputs, enabling more effective training to scalably improve state-of-the-art performance. We hope our research inspires further work that leverages foundation language model strengths and further work on language-agnostic understanding. Our experiments are released here.

Acknowledgement

First, thank you to Professor Kurt Keutzer, my research and 5th years advisor. Kurt, you have many qualities that I admire. Thank you for your willingness to be frank and speak your mind. Thank you for your persistence in your pursuits, whether it be in advising during presentations or responding to directions during meeting updates. I align well with your lab management structure and the way we communicate. I appreciate all the feedback that you've given me and I appreciate your fundamental interest in our field. As I progress throughout my career, I will constantly be in touch!

Second, thank you to Hansen Lillemark, my collaborator on this project. Hansen, I hope that I have been a good mentor as a researcher, an engineer, and as a person. I'm excited to see what directions you pursue next. I strongly believe that with the right tools you can fundamentally change the field of deep learning because of your interdisciplinary background.

Third, thank you to my second reader, Professor John DeNero. John, my Berkeley undergraduate career started with you in CS61A and it's ending with you as my second reader on my thesis. I would not be the person and quality of person I am today without you. To this day, CS61A is in my top three favorite classes at Berkeley. Thank you for your dedication to education, the Berkeley community, and to the EECS community. I hope that your efforts are continually recognized and appreciated. You've raised and inspired a generation of students.

Fourth, thank you to my great friends Galen Kimball, Ashwin Rastogi, Jared Tating, Leon Chen, and Gina Wu. My undergraduate and graduate career would not be what it is without you. Thank you for understanding me and sticking with me!

Thank you to my peers in the Pallas group: Ja, Nick, Amir, Sehoon, Zhen, Sheng, Coleman, and Xiuyu. I wish you all the best in your current and future endeavors! Thank you to all my friends in HKN: Jeff, Anthony, Danny, James, Jason, Oscar, Kat, Maxwell, Shawn, and Rehan. Thank you for enriching my life and growing with me!

Finally, the most heartfelt thank you to my family, Hongbin, Jun, Nina, and Tiggy. Your unwavering support and dedication to my growth is unparalleled. Love you guys!

Contents

1 Introduction

Pretraining large language models on language understanding tasks enables them to be quickly adapted to downstream use, either through prompting or finetuning [2, 38]. **The strengths of foundation language models that underlie prompting have not been applied to finetuning**.

Simultaneously, multilingual NLP lacks a clear goal beyond simple language translation and multilingual language understanding tasks. This falls far short of our goals of **multilingual understanding, transferable understanding, and language adaptability**. Specifically, a model with full multilingual capabilities should (1) understand inputs and produce outputs in various languages, (2) use knowledge acquired in different languages, and (3) adapt quickly to novel languages. We propose the goal of multilingual NLP to be these three components, collectively termed **language-agnostic understanding** (Figure 2).

In this paper, we finetune the multilingual foundation language model mT5 [41]. mT5 is proficient at multilingual understanding but poor at transferable understanding. We measure mT5's performance on transferable understanding using two translation tasks: a Classical Tibetan to English task, and the Flores200 benchmark [37].[1] We leverage mT5's strengths as a foundation language model to reformulate inputs at training time (Figure 1). For the Classical Tibetan to English task, we see qualitative improvements in the training curves. For both tasks, we improve translation performance significantly. Finally, these input reformulations directly increase transferable understanding capabilities, reflected in the performance on these translation tasks.

In summary:

- We introduce the idea of language-agnostic understanding and pose the problem of achieving better transferable understanding with current state-of-the-art multilingual understanding models. To clarify, this is a separate theoretical contribution from our proposed techniques.
- We apply input reformulations that leverage foundation language model strengths during supervised finetuning. These techniques are simple, effective at improving translation performance, and explicitly increase transferable understanding capabilities.
- Our proposed techniques improve Classical Tibetan to English translation performance by up to 10.3% / 2.8 BLEU and Flores200 performance by up to 17.3% / 3.6 chrF++. [2]

[1] Justification for dataset and task choice can be found in section 4.1.

[2] We use different but appropriate metrics for the Classical Tibetan to English and Flores200 translation tasks. More details on why these translation tasks use different metrics can be found in sections 4.1 and 5.1.

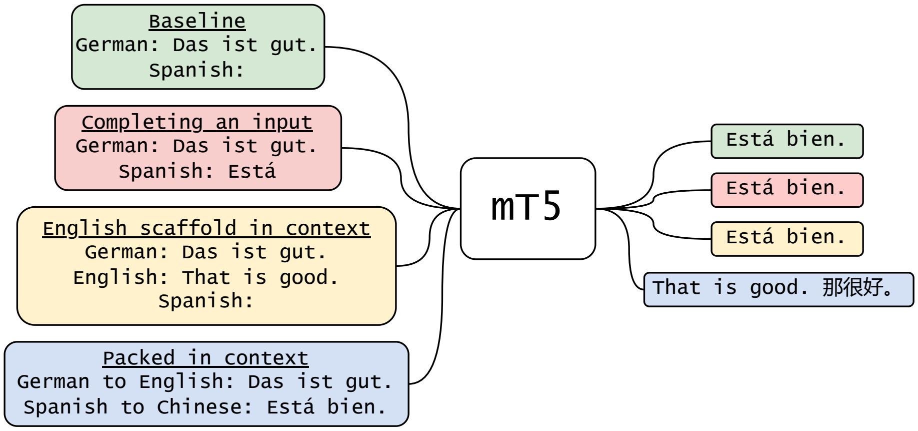


Figure 1: Task reformulations which leverage model strengths. **Green (top)**: the baseline input, a direct translation pair. **Red (second from top)**: leading the model to treat the translation task as a completion task by appending a prefix of the target output to the input. **Yellow (second from bottom)**: utilizing the model's strong English understanding capability to improve transferable understanding between two other input languages. **Blue (bottom)**: packing parallel sentences in multiple languages into a single input to improve transferable understanding.

2 Background

2.1 Foundation language models are the future of NLP

Foundation language models are powerful task-agnostic models that have become increasingly prevalent in recent years, especially beginning in 2018 with BERT [7, 20, 23, 24]. BERT was pretrained on a language understanding task, enabling strong performance on downstream classification tasks through supervised finetuning. Then, the field of NLP shifted away from classification tasks with BERT towards more general language generation models [27, 28, 30]. Since then, many other foundation language models have been developed and released [2, 3, 4, 14, 34, 38, 39, 41, 42]. Foundation language models are state-of-the-art on nearly all downstream language tasks and are easily scalable in compute, size, and data. Foundation language models are the present and future of NLP.

Unfortunately, while some of these models were trained on non-English data, **they do not focus on multilingual performance** [2, 3, 14, 29, 38, 39, 42]. Multilingual NLP enables insights into the efficiency and generality of the knowledge representations in a model.

Foundation language models are either prompted or finetuned for downstream use. Prompting is enabled by the fact that these models are strong at completing inputs and at leveraging their input contexts. These strengths come from pretraining on language understanding tasks that require the model to correctly generate the next token by leveraging their input context. **Unfortunately, foundation language model strengths at completing inputs and leveraging input contexts have not been applied to finetuning.**

2.2 Data efficient methods for translation and where they fall short

Our work can be viewed as a data efficiency technique for translation. Past works in translation have explored data augmentation [8, 33], sample re-weighting [12, 31, 35], or curriculum learning [15, 25, 37, 43, 44]. These approaches vary in effectiveness, are not generalizable, and introduce complexity into the training process. **On the other hand, our proposed technique is simple and can be directly applied to any sequence-to-sequence task.**

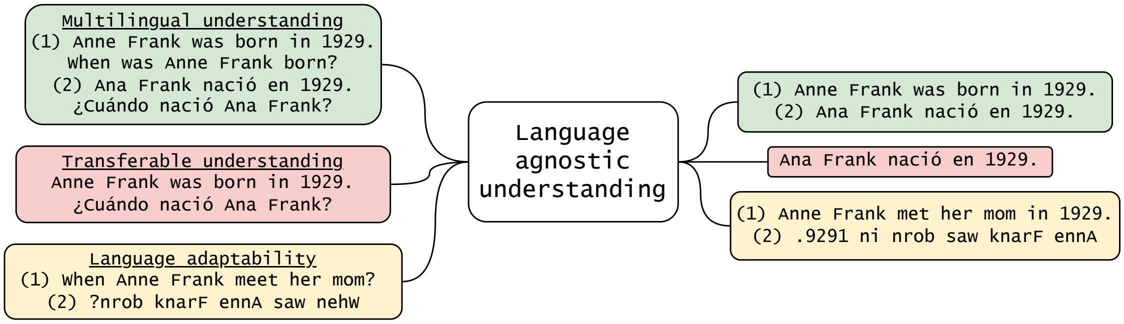

Multilingual understanding
(1) Anne Frank was born in 1929.
When was Anne Frank born?
(2) Ana Frank nació en 1929.
¿Cuándo nació Ana Frank?
Transferable understanding
Anne Frank was born in 1929.
¿Cuándo nació Ana Frank?
Language adaptability
(1) When Anne Frank meet her mom?
(2) ?nrob knarF ennA saw nehW
Language agnostic understanding
(1) Anne Frank was born in 1929.
(2) Ana Frank nació en 1929.
Ana Frank nació en 1929.
(1) Anne Frank met her mom in 1929.
(2) .9291 ni nrob saw knarF ennA

Figure 2: The constituents of language-agnostic understanding. **Top (Green)**: The model receives inputs in both English and Spanish and competently responds in both languages, separately. **Middle (Red)**: The model learns a novel fact in English, is asked a related question in Spanish, and correctly outputs the response in Spanish. **Bottom (Yellow)**: The first input is different from the typical natural language distribution, and the second input is in English but backwards. The model competently receives inputs in both cases and produces a corresponding output.

3 Language-agnostic understanding

3.1 Multilingual approaches

Following the success of BERT [7], many works applied BERT's pretraining approach to multilingual corpora. The most popular approach is to perform monolingual pretraining over the new multilingual corpus [3, 5, 10, 32, 36, 39, 41]. Other approaches shift the focus from multilingual understanding to translation [18], multilingual tokenization [17, 40], or curating translation datasets [9, 37]. **Critically, recent works in the field of multilingual NLP are decentralized, lacking a clear goal beyond simple language translation and multilingual language understanding tasks.** We hope to clarify and solidify the goal with our proposal of language-agnostic understanding.

Few approaches have deviated from this multilingual pretraining approach. In early 2019, Lample and Conneau proposed translation language modeling (TLM). TLM stacks a translation pair together for the pretraining task input [16], directly increasing transferable understanding. In late 2020, Ouyang et al extend TLM to incorporate cross-attention masking and back-translation [21]. **We believe that TLM should be revisited for better multilingual pretraining (see section 6).**

3.2 Multilingual models and where they fall short

The best multilingual language understanding model is mT5 [41] and the best translation model is NLLB [37]. Unfortunately, mT5 and NLLB lack multiple components of language-agnostic understanding. Neither model is capable of language acquisition since it takes enormous amounts of data to pretrain and finetune them, especially compared to humans. NLLB was only trained to perform translation, so it can't actually respond to inputs. **Thus, NLLB is unfit for multilingual understanding.** For example, when NLLB is asked a question in English and asked to produce an output in English, NLLB simply outputs the original input because that's what it has been trained to do. In contrast, mT5 can be finetuned for multilingual question answering. Then, when mT5 is asked a question in English, it will output an attempt at an answer in English.

Translation, as in the case of NLLB [37], is insufficient for the goal of language-agnostic understanding. While translation can be used as a middle-man for a powerful monolingual model, this simply masks the fundamental problem at stake. A model incapable of multilingual understanding will have even larger problems perceiving and acting in different modalities. In the case of multimodality, it's unclear that an analogous "translation" model can even exist. **Models trained end-to-end to have multilingual understanding will surpass two separate models of translation and monolingual language understanding.**

mT5 has been pretrained on a language understanding objective, so it's proficient at multilingual understanding. Unfortunately, mT5 lacks transferable understanding. For example, mT5 is evaluated on the XNLI task [6] where it is trained on a task in English and tested on the same task in different languages. If mT5 had perfect transferable understanding, the test set scores in different languages should match the test set scores in English, regardless of mT5's monolingual performance in each language. mT5 struggles to match English performance and the performance is correlated with the amount of pretrain data in the particular language [41]. **Thus, mT5's performance can be explained by learning about the task during finetuning and then leveraging strong monolingual capabilities, and not by transferable understanding.**

3.3 Remarks on multilingual benchmarks

Current multilingual benchmarks are designed to measure multilingual performance on a downstream task. Specifically, XNLI is a task for multilingual natural language inference [6]. The XNLI authors developed the translate-train paradigm where the original English data are translated into different languages and subsequently used for training and testing. The authors of mT5 were specifically interested in crosslingual transfer so they developed the zero-shot transfer paradigm where a model is trained only in English and tested on other languages [41].

This zero-shot transfer paradigm does measure crosslingual transfer. However, the task that the paradigm is applied to enables several confounding factors when measuring the fundamental property of interest, transferable understanding. Specifically, the zero-shot transfer paradigm applied to the natural language inference task conflates pattern matching on the input task and strong monolingual performance with actual transferable understanding. Strong performance on the zero-shot transfer paradigm by pattern matching and monolingual capabilities masks the actual underlying transferable understanding capabilities of the model.

Thus, we reduce the problem of measuring transferable understanding to reducing the effect of monolingual capabilities. In the best case, the prior given to the model on the task by monolingual capabilities in different languages is zero. An example of a task is to have the model train on made-up facts in English, and recall those facts in different languages, retaining the zero-shot transfer paradigm. This way, the model has zero monolingual priors on the task in different languages. An example can be seen in the example of transferable understanding in 2.

An alternative paradigm for measuring cross-lingual task transfer is to perform the zero-shot transfer paradigm and aggregate across different source languages. For example in XNLI, train separate models on Arabic, English, Bulgarian, etc and perform zero-shot transfer evaluation. This enables insight into how monolingual performance during training affects how much of the task is actually transferred to other languages.

4 Building intuition using the Classical Tibetan to English translation task

We evaluate mT5's transferable understanding using two challenging translation tasks based on a Classical Tibetan to English dataset and the Flores200 benchmark. We use translation tasks to measure transferable understanding because transferable understanding at least requires translation capabilities. A model with strong translation capabilities must have generalized internal representations that language can be mapped in and out of. We first perform experiments on the Classical Tibetan to English task because it's easier, then transfer our learning and findings to the harder Flores200 task.

4.1 Dataset

The Classical Tibetan to English dataset is challenging because mT5 was not pretrained on Tibetan or on classical languages. mT5's tokenizer was also not trained on Classical Tibetan input, so the inputs on this dataset are much less dense than mT5 typically prefers them to be (see 5.5). This choice of dataset enables salient differences to be shown when using better input reformulations, while still being tractable for the model to learn as we perform ablations on specific input reformulations. An alternative is to use datasets similar to the WMT German to English dataset [1], but mT5 already has strong performance on this task because it was pretrained on large amounts of monolingual German and English data. Another alternative is to directly use the Flores200 dataset, but there are many other confounding factors that prevent isolation of effects of our input reformulations. This dataset contains no personally identifiable information or offensive content and is available by request here. The Classical Tibetan to English dataset consists of 450k train, 5k validation, and 5k test translation pairs.

mT5's tokenizer was not trained on Tibetan. One approach is to train a new tokenizer on the new corpus, but this would require mT5 to be re-pretrained. As a result, we use mT5's current tokenizer and use the byte-level fallback capabilities of the underlying SentencePiece tokenizer to encode unknown tokens [41]. mT5's tokenizer yields inputs of mean 72 / median 51 on the Classical Tibetan to English task. Given the dataset sentence characteristics, we use a max sequence length of 256 for the Tibetan to English task.

We perform evaluation on the Tibetan to English task using the BLEU metric [22], suitable for use because the outputs are in English. Additionally, we perform qualitative evaluation of training curves. An alternative is to use chrF++ metric, but this character n-gram based score inflates English output scores. Furthermore, translation into English is typically measured in BLEU.

4.2 Input reformulations

As a foundation model, mT5 is strong at completing a given input. To leverage this strength, we append a uniformly randomly length prefix of the target to the input. **This leads the model to treat the task as a completion task rather than a direct input-output task.** Intuitively, we improve the model's single step probabilities of outputting the correct next token. An example can be found in 3. We apply this input reformulation to the Classical Tibetan to English task.

Table 1: Summary of results on the Classical Tibetan to English translation task. Values shown are test set BLEU scores.

Model	Baseline	Reformulated	Diff
mT5 600M	23.5	24.6	**+1.1**
mT5 1B	27.2	28.3	**+1.1**
mT5 3B	27.3	30.1	**+2.8**

4.3 Experiment setup

We use the stochastic gradient descent optimization algorithm AdamW [19] with the gradient exponential moving average (EMA) parameter $\beta_1 = 0.9$, the hessian approximation EMA parameter $\beta_2 = 0.999$, and weight decay value of 0. Since we use AdamW which requires a minimum of several thousand updates to converge, we use a total of 10,000 updates, plenty of time for all of our training runs to fully converge. We use no warmup and a constant learning rate. On the Classical Tibetan to English task, we ablate over learning rates in {1e-3, 2e-3, 3e-3} for 600M and 1B parameter models (the default finetuning learning rate for mT5 [41]) and {3e-4, 5e-4, 1e-3} for 3B parameter models, where we found lower learning rates to be empirically better.

We use a per step batch size of 512 examples / 35,000 tokens, very small compared to the finetuning setups of mT5 (per step batch size of $2^{17} \approx 131,000$) [41]. This covers about 11 epochs on the Tibetan to English translation task.

We explore several different experimental setups to find the optimal strategy of incorporating our input reformulation. The best setup was presenting the reformulated dataset for the first 2000 steps with the remaining 8000 steps remaining unchanged. All ablations are performed on mT5 600M.

We perform evaluation on the models and save checkpoints every 200 steps, for a total of 50 evaluations, and we use the highest scoring checkpoint for all results.

Models were trained on GPU nodes of either 8 NVIDIA A5000 24GB GPUs or 8 NVIDIA A6000 48GB GPUs. The typical train time varied from 3 hours for the smallest models to 36 hours for the largest. We leverage the Deepspeed library `https://www.deepspeed.ai/` for multi-GPU training and for training in the half precision bf16.

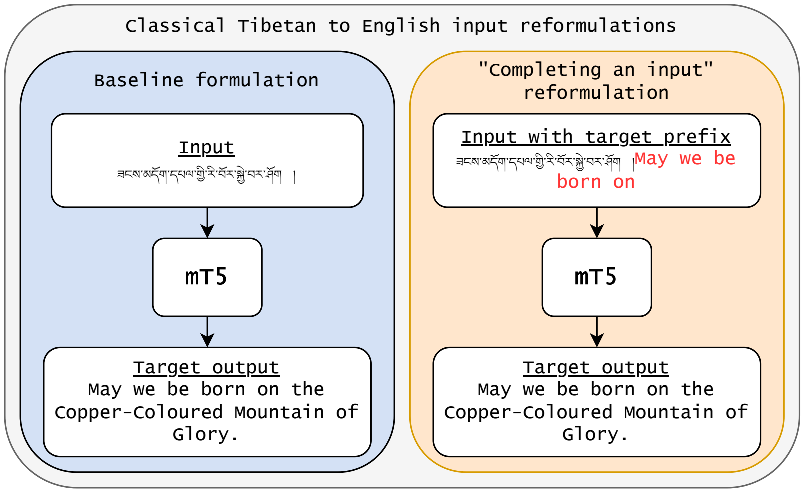

Classical Tibetan to English input reformulations
Baseline formulation
Input
ཟངས་མདོག་དཔལ་གྱི་རི་བོར་སྐྱེ་བར་ཤོག །
mT5
Target output
May we be born on the Copper-Coloured Mountain of Glory.
"Completing an input" reformulation
Input with target prefix
ཟངས་མདོག་དཔལ་གྱི་རི་བོར་སྐྱེ་བར་ཤོག །May we be born on
mT5
Target output
May we be born on the Copper-Coloured Mountain of Glory.

Figure 3: Examples of input reformulations applied to the Classical Tibetan to English translation task. The changes to the original input are highlighted in **red**.

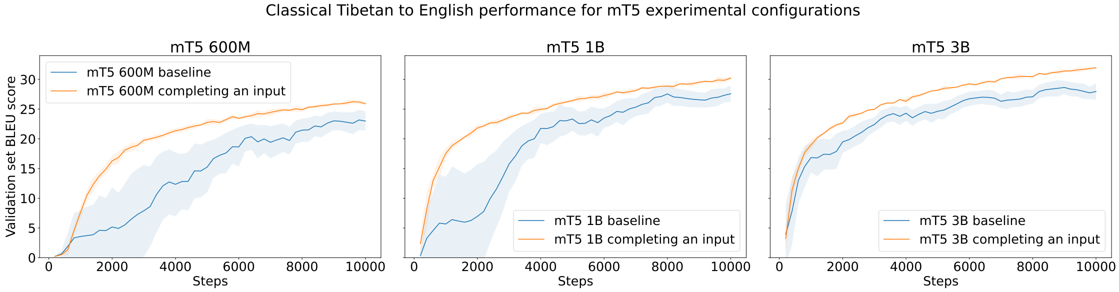


Figure 4: Classical Tibetan to English translation task reformulation experiment results. These results compare the **mT5 baseline (blue)** and the **mT5 "completing an input" input reformulation (orange)** experimental configurations. Each line represents performance on the range of learning rates specified in section 4.3, where the solid line is the mean and the shaded area around each line is the standard deviation. Left: 600M. Center: 1B. Right: 3B.

4.4 Results and analysis

For the Classical Tibetan to English task, we seek to answer two questions: (1) can input reformulations improve evaluation set performance and (2) how do input reformulations affect the behavior of model training?

A summary of the ablations performed can be found in 2. From the results, we learn: easier tasks stabilize model training, especially in the beginning of training (1, 2, 3); simple setups have less variance than complex setups (4, 5, 6); input reformulations should have enough noise for the model to successfully denoise (3, 6); input reformulations are target-substring independent (7, 8, 9); input reformulations cannot deviate too far from the actual task and input reformulations should be only in natural language (10, 11, 12, 13). We choose setup (3) to represent our input reformulations because it was the most stable of the input reformulations while performing significantly better than the baseline.

Clearly, changing the input reformulation from the baseline to "completing an input" can improve evaluation set performance (see Figure 4 and Table 1). Furthermore, the input reformulation consistently outperforms the baseline. The baseline training curves have high variance over learning rates and are unstable throughout training. **Adding our simple reformulation significantly reduces the variance of the training curves, smooths out training, and improves performance. Furthermore, the input reformulation experiment training curves could be extrapolated to improve performance even further, while the baseline has already begun converging.**

5 Putting it all together on the challenging Flores200 translation task

5.1 Dataset

The Flores200 dataset consists of around 3,000 parallel sentences in 204 different languages [11, 13, 37] and is available via the Creative Commons Attribution Share Alike 4.0 license. This dataset is challenging because of the sheer number of languages, and because mT5 was not pretrained on over half of the languages present in the dataset. The Flores200 dataset is purported for evaluation with a separate non-parallel train set NLLB, but the parallel nature of the Flores200 dataset enables better training techniques leveraging mT5's strengths. To formulate a translation task, we take translation pairs from the Flores200 dev set as our training set and translation pairs from the

Table 2: Classical Tibetan to English task ablations. All ablations setups use mT5 600M. The reformulation process is to select a substring of the target English output and append it to the Tibetan input. The length of the substring is selected in two different ways: uniformly randomly and linearly scaling. In the linearly scaling condition, earlier steps use proportionally more of the target output. The value shown in the "BLEU" score column is the maximum test set BLEU score over the learning rates.

No.	Setup	Substring	Stable?	BLEU
(1)	50% baseline, 50% reformulated	Prefix	Yes	23.9
(2)	100% reformulated	Prefix	Yes	21.1
(3)	**First 2000 steps reformulated**	**Prefix**	**Yes**	**24.6**
(4)	Staged linearly scaling	Prefix	No	24.7
(5)	Linearly scaling	Prefix	No	17.4
(6)	(3) with linearly scaling	Prefix	No	24.9
(7)	(3)	Prefix and suffix	Yes	24.5
(8)	(3) with 4000 steps	Prefix and suffix	Yes	24.0
(9)	(3) with 1200 steps	Prefix and suffix	Yes	24.8
(10)	(3) with masked target reformulation with p=0.1	Prefix	No	24.9
(11)	(3) + last 2000 steps mask the input with p=0.1	Prefix	No	23.6
(12)	(3) + last 5000 steps mask the input with p=0.25	Prefix	Yes	23.0
(13)	(3) + last 5000 steps span-mask input with p=0.25	Prefix	Yes	23.4

devtest set as our validation and test sets. Our reformulated Flores200 dataset for training consists of 20M train, 5k validation, and 10k test translation pairs.

Following the tokenization setup for the Classical Tibetan to English task, mT5's tokenizer yields inputs of mean 52 / median 46 on the Flores200 task. Given the dataset sentence characteristics, we use a max sequence length of 256 for both the Flores200 task. This is lower than the maximum sequence length of 512 used to develop NLLB [37], but it makes no mechanical difference since none of the inputs are truncated.

We perform evaluation on the Flores200 task using the chrF++ metric [26], in line with evaluation of the NLLB team [37]. BLEU poorly represents relative scores across the 204 typologically diverse languages in the Flores200 dataset, so chrF++ is more suitable.

5.2 Input reformulations

We apply our learnings from the Classical Tibetan to English translation task to the Flores200 task. Any input reformulations on the Flores200 task should only be in natural language and be just "noisy" enough for the model to denoise on. The "noise" that we leverage is presenting the same content but in different languages. To better leverage a model's input context, we either "scaffold" an input context with the parallel English translation or "pack" multiple parallel translation pairs into a single input context 5. For both reformulations, the model is able to directly attend between pairs of parallel inputs, directly increasing transferable understanding. For the English scaffold input, the model is able to heavily exploit its knowledge of English to perform better. For the packed input, the model is able to increase transferable understanding between languages to perform better. We apply both of these reformulations to the Flores200 translation task.

5.3 Experiment setup

We apply the same setup as the Classical Tibetan to English translation task setup unless noted otherwise. On the Flores200 task, we ablate over the learning rates {1e-4, 2e-4, 3e-4}, where we again found lower learning rates to be empirically better. We use a per step batch size of 2048 examples / 105,000 tokens for the Flores200 task, very small compared to the training setup of NLLB (per step batch size of 1 million tokens) [37]. This covers about 0.5 epochs on the Flores200 train set. We use a larger data budget and a larger per step batch size for Flores200 compared to the Classical Tibetan to English task because Flores200 is a harder task.

For the Flores200 task, we use a ratio of 20% baseline and 80% reformulated dataset. In all of the scenarios, we hold the data and compute budgets constant to ensure the validity and veracity of

Table 3: Summary of results on the Flores200 translation task. Values shown are test set chrF++ scores.

Model	Baseline	Reformulated	Diff
mT5 600M	18.4	21.5	**+3.1**
mT5 1B	20.8	24.4	**+3.6**
mT5 3B	23.7	25.7	**+2.0**

our results. Specifically, the Flores200 reformulations use up to twice the number of examples per input so we reduce the per-step batch size by a factor of two.

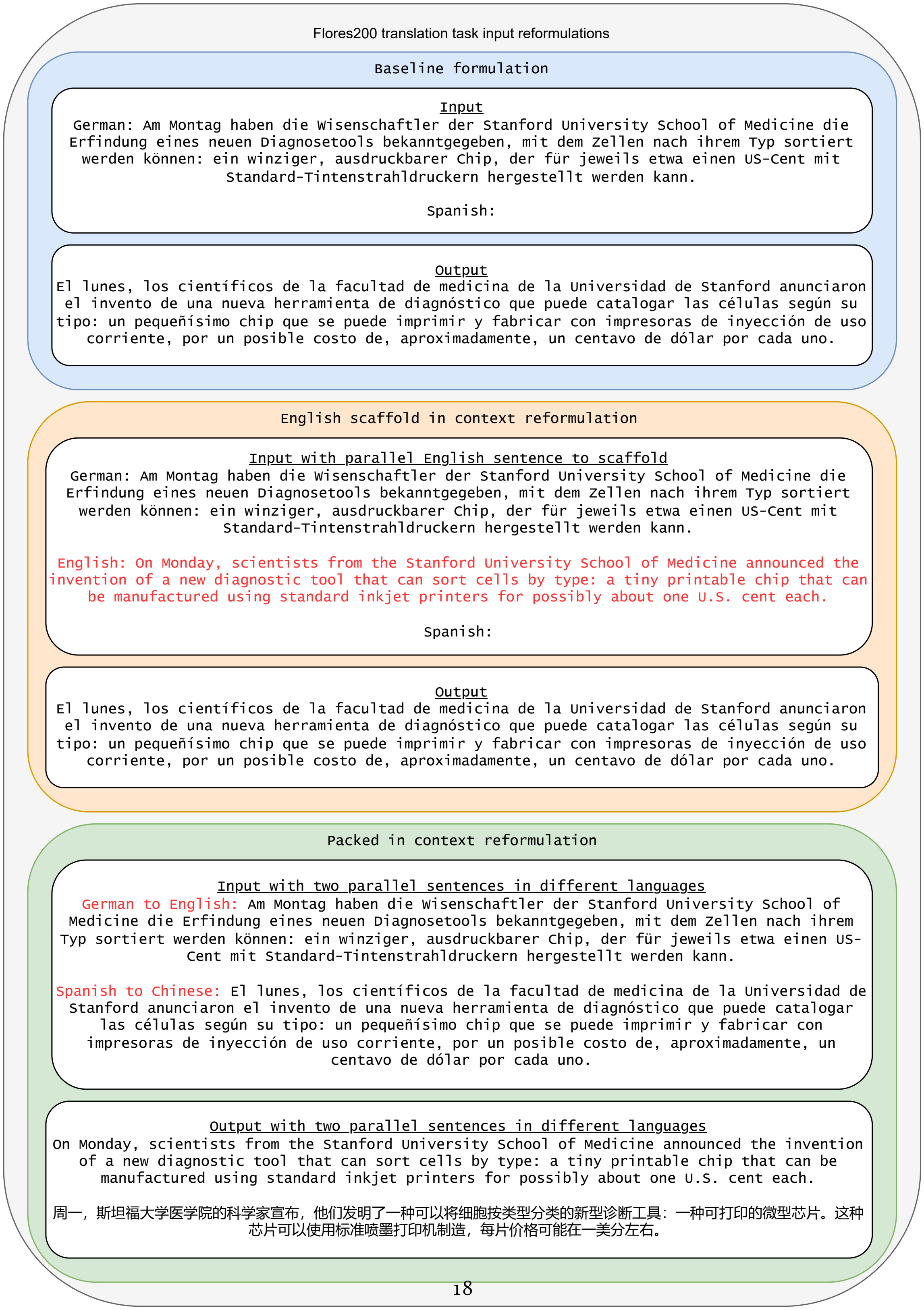
5.4 Results and analysis

For the Flores200 task, we seek to answer two further questions: (1) how general can input reformulations be, and (2) how much can input reformulations enable the model to leverage other strengths?

We observe similar effects compared to the input reformulations on the Classical Tibetan to English task (see Figure 6 and Table 3). For the "packed in context" reformulation, the model learns faster and better. For the "English scaffold in context" reformulation, the model learns slightly slower initially but learns much more over the course of training. Critically, the English scaffold in context reformulation enables the model to not only leverage its input context better, but also exploit its strong knowledge of English. These input reformulations are different and more general than those in the Classical Tibetan to English translation task, but have similar positive effects.

Input reformulations that leverage model strengths are effective at improving model performance. Beyond just improving absolute scores on the translation task metrics, the input reformulations can also help to better condition the model for finetuning on challenging downstream tasks. Packing multiple translation examples in a single input context or scaffolding an input context with a parallel English sentence enables the model to directly attend to the same input sentence in different languages. This direct attention from the input context helps the model align the same content from different languages and directly increases transferable understanding. Packing multiple parallel examples into a single input context enables the model to better denoise on the difference between languages.

Interestingly, the English scaffold condition performs the best but has the highest variance over the learning rates. The need for lower learning rates typically indicates poor conditioning, so the



Flores200 translation task input reformulations

Baseline formulation

Input

German: Am Montag haben die Wisenschaftler der Stanford University School of Medicine die Erfindung eines neuen Diagnosetools bekanntgegeben, mit dem Zellen nach ihrem Typ sortiert werden können: ein winziger, ausdruckbarer Chip, der für jeweils etwa einen US-Cent mit Standard-Tintenstrahldruckern hergestellt werden kann.

Spanish:

Output

El lunes, los científicos de la facultad de medicina de la Universidad de Stanford anunciaron el invento de una nueva herramienta de diagnóstico que puede catalogar las células según su tipo: un pequeñísimo chip que se puede imprimir y fabricar con impresoras de inyección de uso corriente, por un posible costo de, aproximadamente, un centavo de dólar por cada uno.

English scaffold in context reformulation

Input with parallel English sentence to scaffold

German: Am Montag haben die Wisenschaftler der Stanford University School of Medicine die Erfindung eines neuen Diagnosetools bekanntgegeben, mit dem Zellen nach ihrem Typ sortiert werden können: ein winziger, ausdruckbarer Chip, der für jeweils etwa einen US-Cent mit Standard-Tintenstrahldruckern hergestellt werden kann.

English: On Monday, scientists from the Stanford University School of Medicine announced the invention of a new diagnostic tool that can sort cells by type: a tiny printable chip that can be manufactured using standard inkjet printers for possibly about one U.S. cent each.

Spanish:

Output

El lunes, los científicos de la facultad de medicina de la Universidad de Stanford anunciaron el invento de una nueva herramienta de diagnóstico que puede catalogar las células según su tipo: un pequeñísimo chip que se puede imprimir y fabricar con impresoras de inyección de uso corriente, por un posible costo de, aproximadamente, un centavo de dólar por cada uno.

Packed in context reformulation

Input with two parallel sentences in different languages

German to English: Am Montag haben die Wisenschaftler der Stanford University School of Medicine die Erfindung eines neuen Diagnosetools bekanntgegeben, mit dem Zellen nach ihrem Typ sortiert werden können: ein winziger, ausdruckbarer Chip, der für jeweils etwa einen US-Cent mit Standard-Tintenstrahldruckern hergestellt werden kann.

Spanish to Chinese: El lunes, los científicos de la facultad de medicina de la Universidad de Stanford anunciaron el invento de una nueva herramienta de diagnóstico que puede catalogar las células según su tipo: un pequeñísimo chip que se puede imprimir y fabricar con impresoras de inyección de uso corriente, por un posible costo de, aproximadamente, un centavo de dólar por cada uno.

Output with two parallel sentences in different languages

On Monday, scientists from the Stanford University School of Medicine announced the invention of a new diagnostic tool that can sort cells by type: a tiny printable chip that can be manufactured using standard inkjet printers for possibly about one U.S. cent each.

周一，斯坦福大学医学院的科学家宣布，他们发明了一种可以将细胞按类型分类的新型诊断工具：一种可打印的微型芯片。这种芯片可以使用标准喷墨打印机制造，每片价格可能在一美分左右。

Figure 5: Examples of input reformulations applied to the Flores200 translation task. The changes to the original input are highlighted in **red**.

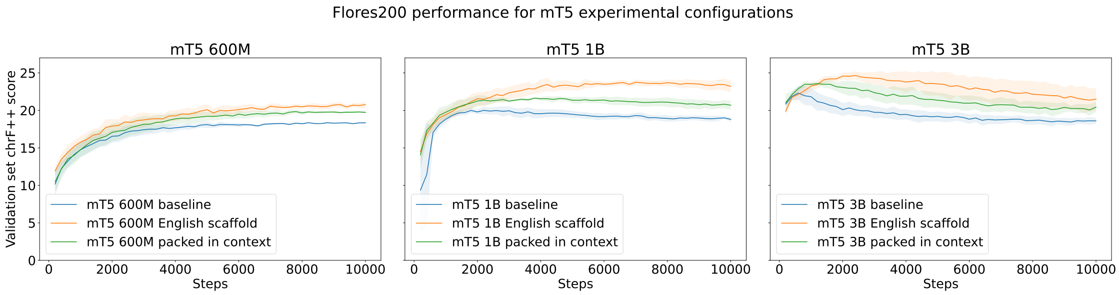


Figure 6: Flores200 translation task reformulation experiment results. These results compare the **mT5 baseline (blue)**, **mT5 "packed in context" (green)**, and the **mT5 "English scaffold" input reformulation (orange)** experimental configurations. Each line represents performance on the range of learning rates specified in section 5.3, where the solid line is the mean and the shaded area around each line is the standard deviation. Left: 600M. Center: 1B. Right: 3B.

input task is more ill-conditioned than the baseline. One possible explanation is that mT5 begins to actually learn the languages in Flores200 that were not present in its training set.

5.5 Comparison of mT5 and NLLB performance

Since we use translation tasks as a proxy for transferable understanding in mT5, we compare the performance of mT5 with the performance of NLLB, the best translation model today [37]. Our goal is not to achieve better translation results than NLLB; rather, we aim to show improvements on mT5's performance at transferable understanding. Furthermore, the data and compute budget of our finetuning setup is much smaller than that of NLLB.

We first compare the input lengths using mT5's tokenizer and NLLB's tokenizer which has been trained on all languages in Flores200. On the Tibetan inputs of the Tibetan to English dataset, mT5's encoded input lengths are mean 72 / median 51 tokens and NLLB's encoded input lengths are mean 26 / median 19 tokens. This large gap highlights that mT5's tokenizer was not trained on the Tibetan language. In addition, the Classical Tibetan dataset contains many novel entity names that require many additional tokens under the byte-level fallback scheme. On the Flores200 dataset, mT5's encoded input lengths are mean 52 / median 46 tokens and NLLB's encoded input lengths are mean 41 / median 39. The gap between the token lengths for the two tokenizers is much closer because mT5's tokenizer was trained on about half of the languages in the Flores200 dataset and many of the languages present in Flores200 are represented using a Latin or Arabic script.

Differences in tokenization can have large effects on downstream performance. Specifically, mT5's tokenizer yields word or character pieces that are much smaller than that of NLLB's tokenizer. This presents a significant challenge to models today that have no explicit notion of summarization over the sequence length dimension. Specifically, the current attention mechanism cannot group multiple tokens together to attend to them as a unit or have that unit attend to something else. For example, the word "packed" may be broken up into several characters. When translating the word "packed", we certainly want a more holistic view of the word, perhaps grouped as "pack" and "-ed". This reduces the modeling capacity and performance of mT5 on these translation tasks.

Second, NLLB has actually been trained on translation while mT5 has only been trained on the monolingual pretraining task on a multilingual corpus. NLLB also has been trained on Tibetan to English translation. As a result, NLLB has much stronger translation and Tibetan priors than mT5.

Third, the NLLB translation model was trained on the full NLLB translation dataset while our setup trains on the Flores200 set. Our Flores200 training set reformulations actually only come from 1000 sentences. This reduces the information that mT5 can learn during our finetuning. Furthermore, while a specific translation pair is seen rarely throughout training, the input sentences themselves are commonplace. Specifically, the model runs through each input sentence *once* per training step.

All of these factors combine to put our mT5 training setup at a severe disadvantage compared to NLLB. To reiterate, we do not intend to achieve better or even comparable results to NLLB.

Because NLLB is a translation-only model, our input reformulations cannot be applied to it. For example using the "English scaffold in context" input reformulation, the inputs to the model are two parallel sentences in two different languages. NLLB would receive these inputs and output a repeat of the same sentence in the first language specified. In contrast, mT5 can be finetuned to dynamically receive inputs and produce outputs in different languages.

For NLLB on the Tibetan to English task, we ablate over learning rates in {3e-4, 5e-4, 1e-3}. For the NLLB Tibetan to English baseline, we use a linear warmup of 1000 steps, 10% of the total number of updates, with constant learning rate afterwards. The NLLB column is the task performance of a corresponding size NLLB model. We evaluate Flores200 scores using the xx-yy condition [37].

Clearly, NLLB outperforms mT5 on both the Classical Tibetan to English and Flores200 tasks (4, 7). Only the mT5 3B reformulated inputs condition reaches the smallest NLLB 600M model performance. Despite NLLB's strong translation and Tibetan prior, NLLB still struggles on the Classical Tibetan to English translation task. This is most likely due to the the fact that NLLB

Table 4: Comparison of results to NLLB

Task	Metric	Model	NLLB	Baseline	Reformulated
Classical Tibetan to English	BLEU	mT5 600M	*29.3*	23.5	24.6
		mT5 1B	*32.3*	27.2	28.3
		mT5 3B	*34.4*	27.3	30.1
Flores200	chrF++	mT5 600M	*39.5*	18.4	21.5
		mT5 1B	*41.5*	20.8	24.4
		mT5 3B	*42.7*	23.7	25.7

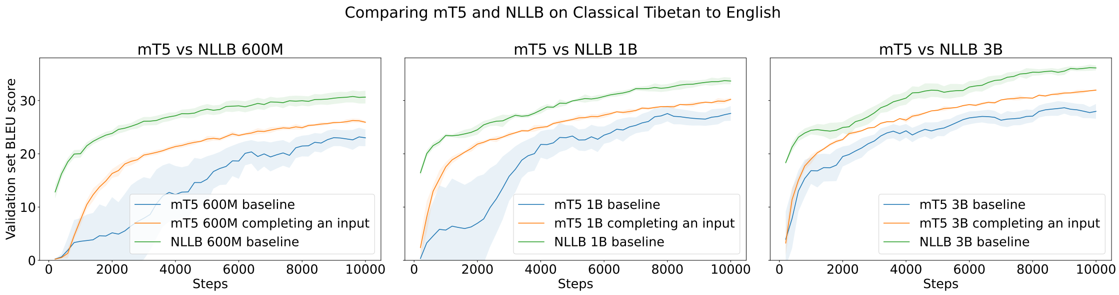


Figure 7: Results comparing the NLLB baseline, mT5 baseline, and mT5 completing an input experimental configurations. Each line represents performance on the range of learning rates specified above and in section 4.3, where the solid line is the mean and the shaded area around each line is the standard deviation. Left: 600M. Center: 1B. Right: 3B.

was only trained on modern languages and the language distribution of Classical Tibetan is very distinct from that of modern Tibetan.

6 Conclusion

We identify two shortcomings of current research in the NLP space. First, foundation language models have many strengths that are undervalued. Specifically, the strengths of foundation language models that underlie prompting have not been applied to finetuning. Second, multilingual NLP lacks a clear goal. We introduce the concept of language-agnostic understanding and its three constituents: multilingual understanding, transferable understanding, and language adaptability. We perform supervised finetuning on mT5, a state-of-the-art multilingual foundation language model that has good multilingual understanding capabilities but poor transferable understanding capabilities. We leverage the strengths of mT5 as a foundation language model to reformulate inputs at training time. These input reformulations are simple and effective, only requiring changes to data processing. Critically, these input reformulations directly increase the transferable understanding capabilities of mT5.

Furthermore, we show that transferable understanding can be improved under a very small data and compute budget around 20M examples over 10k finetuning steps. As alluded to earlier, we believe that this same input reformulation paradigm should be additionally applied to multilingual pretraining, for example by revisiting translation language modeling (TLM) [16]. The translation language modeling task can be directly added into the baseline multilingual pretraining data mix. The input reformulation can be exactly the packed in context reformulation applied to the Flores200 dataset, where the number of parallel translation examples per input can be scaled further from 2 up to 8 in a single input context.

Our proposed technique has only been applied to two challenging translation tasks where the input and output are both information rich and sequential in nature. Mechanically, there is no reason why this technique cannot be applied to other tasks such as sequence classification. Intuitively, doing so would enable the model to attend to multiple inputs in its input context in order to better denoise the inputs. This allows the model to learn more effectively. The specific techniques explored here are applicable to other tasks. Fundamentally, the idea is to leverage strengths of foundation models better during finetuning. This allows for more creative input reformulations to better exploit the specific task at hand.

We hope that the proposed direction of language-agnostic understanding for multilingual NLP is comprehensive and intuitive, and that further works expand on it. We hope that the simplicity and efficacy of our technique inspires further research and techniques that better leverage foundation models strengths.

Bibliography

1. L. Barrault, M. Biesialska, O. Bojar, M. R. Costa-jussà, C. Federmann, Y. Graham, R. Grundkiewicz, B. Haddow, M. Huck, E. Joanis, T. Kocmi, P. Koehn, C.-k. Lo, N. Ljubešić, C. Monz, M. Morishita, M. Nagata, T. Nakazawa, S. Pal, M. Post, and M. Zampieri. "Findings of the 2020 Conference on Machine Translation (WMT20)". In: *Proceedings of the Fifth Conference on Machine Translation.* Association for Computational Linguistics, Online, 2020, pp. 1–55. URL: `https://aclanthology.org/2020.wmt-1.1`.

2. 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. arXiv: `2005.14165 [cs.CL]`.

3. A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel. *PaLM: Scaling Language Modeling with Pathways.* 2022. arXiv: `2204.02311 [cs.CL]`.

4. H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei. *Scaling Instruction-Finetuned Language Models.* 2022. arXiv: `2210.11416 [cs.LG]`.

5. 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.* 2020. arXiv: `1911.02116 [cs.CL]`.

6. A. Conneau, G. Lample, R. Rinott, A. Williams, S. R. Bowman, H. Schwenk, and V. Stoyanov. *XNLI: Evaluating Cross-lingual Sentence Representations.* 2018. arXiv: `1809.05053 [cs.CL]`.

7. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2019. arXiv: 1810.04805 [cs.CL].

8. M. Fadaee, A. Bisazza, and C. Monz. "Data Augmentation for Low-Resource Neural Machine Translation". In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 567–573. DOI: 10.18653/v1/P17-2090. URL: https://aclanthology.org/P17-2090.

9. A. Fan, S. Bhosale, H. Schwenk, Z. Ma, A. El-Kishky, S. Goyal, M. Baines, O. Celebi, G. Wenzek, V. Chaudhary, N. Goyal, T. Birch, V. Liptchinsky, S. Edunov, E. Grave, M. Auli, and A. Joulin. *Beyond English-Centric Multilingual Machine Translation*. 2020. arXiv: 2010.11125 [cs.CL].

10. N. Goyal, J. Du, M. Ott, G. Anantharaman, and A. Conneau. *Larger-Scale Transformers for Multilingual Masked Language Modeling*. 2021. arXiv: 2105.00572 [cs.CL].

11. N. Goyal, C. Gao, V. Chaudhary, P.-J. Chen, G. Wenzek, D. Ju, S. Krishnan, M. Ranzato, F. Guzmán, and A. Fan. "The FLORES-101 Evaluation Benchmark for Low-Resource and Multilingual Machine Translation". In: 2021.

12. J. Gu, Y. Wang, Y. Chen, K. Cho, and V. O. K. Li. *Meta-Learning for Low-Resource Neural Machine Translation*. 2018. arXiv: 1808.08437 [cs.CL].

13. F. Guzmán, P.-J. Chen, M. Ott, J. Pino, G. Lample, P. Koehn, V. Chaudhary, and M. Ranzato. "Two New Evaluation Datasets for Low-Resource Machine Translation: Nepali-English and Sinhala-English". In: 2019.

14. J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de Las Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millican, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, J. W. Rae, O. Vinyals, and L. Sifre. *Training Compute-Optimal Large Language Models*. 2022. arXiv: 2203.15556 [cs.CL].

15. T. Kocmi and O. Bojar. "Curriculum Learning and Minibatch Bucketing in Neural Machine Translation". In: *RANLP 2017 - Recent Advances in Natural Language Processing Meet Deep Learning*. Incoma Ltd. Shoumen, Bulgaria, 2017. DOI: 10.26615/978-954-452-049-6_050.

16. G. Lample and A. Conneau. *Cross-lingual Language Model Pretraining*. 2019. arXiv: 1901.07291 [cs.CL].

17. D. Liang, H. Gonen, Y. Mao, R. Hou, N. Goyal, M. Ghazvininejad, L. Zettlemoyer, and M. Khabsa. *XLM-V: Overcoming the Vocabulary Bottleneck in Multilingual Masked Language Models*. 2023. arXiv: 2301.10472 [cs.CL].

18. 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*. 2020. arXiv: 2001.08210 [cs.CL].

19. I. Loshchilov and F. Hutter. *Decoupled Weight Decay Regularization*. 2019. arXiv: 1711.05101 [cs.LG].

20. B. McCann, J. Bradbury, C. Xiong, and R. Socher. *Learned in Translation: Contextualized Word Vectors.* 2018. arXiv: 1708.00107 [cs.CL].

21. X. Ouyang, S. Wang, C. Pang, Y. Sun, H. Tian, H. Wu, and H. Wang. *ERNIE-M: Enhanced Multilingual Representation by Aligning Cross-lingual Semantics with Monolingual Corpora.* 2021. arXiv: 2012.15674 [cs.CL].

22. K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. "Bleu: a Method for Automatic Evaluation of Machine Translation". In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics.* Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, 2002, pp. 311–318. DOI: 10.3115/1073083.1073135. URL: https://aclanthology.org/P02-1040.

23. J. Pennington, R. Socher, and C. Manning. "GloVe: Global Vectors for Word Representation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).* Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1532–1543. DOI: 10.3115/v1/D14-1162. URL: https://aclanthology.org/D14-1162.

24. M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. *Deep contextualized word representations.* 2018. arXiv: 1802.05365 [cs.CL].

25. E. A. Platanios, O. Stretcu, G. Neubig, B. Poczos, and T. M. Mitchell. *Competence-based Curriculum Learning for Neural Machine Translation.* 2019. arXiv: 1903.09848 [cs.CL].

26. M. Popović. "chrF: character n-gram F-score for automatic MT evaluation". In: *Proceedings of the Tenth Workshop on Statistical Machine Translation.* Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 392–395. DOI: 10.18653/v1/W15-3049. URL: https://aclanthology.org/W15-3049.

27. A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. *Improving language understanding by generative pre-training.* 2018.

28. A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. *Language models are unsupervised multitask learners.* 2019.

29. J. W. Rae, S. Borgeaud, T. Cai, K. Millican, J. Hoffmann, F. Song, J. Aslanides, S. Henderson, R. Ring, S. Young, E. Rutherford, T. Hennigan, J. Menick, A. Cassirer, R. Powell, G. van den Driessche, L. A. Hendricks, M. Rauh, P.-S. Huang, A. Glaese, J. Welbl, S. Dathathri, S. Huang, J. Uesato, J. Mellor, I. Higgins, A. Creswell, N. McAleese, A. Wu, E. Elsen, S. Jayakumar, E. Buchatskaya, D. Budden, E. Sutherland, K. Simonyan, M. Paganini, L. Sifre, L. Martens, X. L. Li, A. Kuncoro, A. Nematzadeh, E. Gribovskaya, D. Donato, A. Lazaridou, A. Mensch, J.-B. Lespiau, M. Tsimpoukelli, N. Grigorev, D. Fritz, T. Sottiaux, M. Pajarskas, T. Pohlen, Z. Gong, D. Toyama, C. de Masson d'Autume, Y. Li, T. Terzi, V. Mikulik, I. Babuschkin, A. Clark, D. de Las Casas, A. Guy, C. Jones, J. Bradbury, M. Johnson, B. Hechtman, L. Weidinger, I. Gabriel, W. Isaac, E. Lockhart, S. Osindero, L. Rimell, C. Dyer, O. Vinyals, K. Ayoub, J. Stanway, L. Bennett, D. Hassabis, K. Kavukcuoglu, and G. Irving. *Scaling Language Models: Methods, Analysis & Insights from Training Gopher.* 2022. arXiv: 2112.11446 [cs.CL].

30. 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.* 2020. arXiv: 1910.10683 [cs.LG].

31. M. Ren, W. Zeng, B. Yang, and R. Urtasun. *Learning to Reweight Examples for Robust Deep Learning.* 2019. arXiv: 1803.09050 [cs.LG].

32. R. Ri, I. Yamada, and Y. Tsuruoka. *mLUKE: The Power of Entity Representations in Multilingual Pretrained Language Models.* 2022. arXiv: 2110.08151 [cs.CL].

33. R. Sennrich, B. Haddow, and A. Birch. "Improving Neural Machine Translation Models with Monolingual Data". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* Association for Computational Linguistics, Berlin, Germany, 2016, pp. 86–96. DOI: 10.18653/v1/P16-1009. URL: https://aclanthology.org/P16-1009.

34. M. Shoeybi, M. Patwary, R. Puri, P. LeGresley, J. Casper, and B. Catanzaro. *Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism.* 2020. arXiv: 1909.08053 [cs.CL].

35. J. Shu, Q. Xie, L. Yi, Q. Zhao, S. Zhou, Z. Xu, and D. Meng. *Meta-Weight-Net: Learning an Explicit Mapping For Sample Weighting.* 2019. arXiv: 1902.07379 [cs.LG].

36. S. Soltan, S. Ananthakrishnan, J. FitzGerald, R. Gupta, W. Hamza, H. Khan, C. Peris, S. Rawls, A. Rosenbaum, A. Rumshisky, C. S. Prakash, M. Sridhar, F. Triefenbach, A. Verma, G. Tur, and P. Natarajan. *AlexaTM 20B: Few-Shot Learning Using a Large-Scale Multilingual Seq2Seq Model.* 2022. arXiv: 2208.01448 [cs.CL].

37. N. Team, M. R. Costa-jussà, J. Cross, O. Çelebi, M. Elbayad, K. Heafield, K. Heffernan, E. Kalbassi, J. Lam, D. Licht, J. Maillard, A. Sun, S. Wang, G. Wenzek, A. Youngblood, B. Akula, L. Barrault, G. M. Gonzalez, P. Hansanti, J. Hoffman, S. Jarrett, K. R. Sadagopan, D. Rowe, S. Spruit, C. Tran, P. Andrews, N. F. Ayan, S. Bhosale, S. Edunov, A. Fan, C. Gao, V. Goswami, F. Guzmán, P. Koehn, A. Mourachko, C. Ropers, S. Saleem, H. Schwenk, and J. Wang. *No Language Left Behind: Scaling Human-Centered Machine Translation.* 2022. arXiv: 2207.04672 [cs.CL].

38. H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample. *LLaMA: Open and Efficient Foundation Language Models.* 2023. arXiv: 2302.13971 [cs.CL].

39. B. Workshop et al. *BLOOM: A 176B-Parameter Open-Access Multilingual Language Model.* 2023. arXiv: 2211.05100 [cs.CL].

40. L. Xue, A. Barua, N. Constant, R. Al-Rfou, S. Narang, M. Kale, A. Roberts, and C. Raffel. *ByT5: Towards a token-free future with pre-trained byte-to-byte models.* 2022. arXiv: 2105.13626 [cs.CL].

41. 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.* 2021. arXiv: 2010.11934 [cs.CL].

42. S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin, T. Mihaylov, M. Ott, S. Shleifer, K. Shuster, D. Simig, P. S. Koura, A. Sridhar, T. Wang, and L. Zettlemoyer. *OPT: Open Pre-trained Transformer Language Models*. 2022. arXiv: 2205.01068 [cs.CL].

43. X. Zhang, G. Kumar, H. Khayrallah, K. Murray, J. Gwinnup, M. J. Martindale, P. McNamee, K. Duh, and M. Carpuat. *An Empirical Exploration of Curriculum Learning for Neural Machine Translation.* 2018. arXiv: 1811.00739 [cs.CL].

44. X. Zhang, P. Shapiro, G. Kumar, P. McNamee, M. Carpuat, and K. Duh. *Curriculum Learning for Domain Adaptation in Neural Machine Translation.* 2019. arXiv: 1905.05816 [cs.CL].