Breaking Language Barriers: Cross-Lingual Continual Pre-Training at Scale

Anonymous ACL submission

Abstract

In recent years, Large Language Models (LLMs) have made significant strides towards Artificial General Intelligence. However, training these models from scratch requires substantial computational resources and vast amounts of text data. In this paper, we explore an alternative approach to constructing an LLM for a new language by *continually pre-training* (CPT) from existing pre-trained LLMs, instead of using randomly initialized parameters. Based on parallel experiments on 40 model sizes ranging from 40M to 5B parameters, we find that 1) CPT converges faster and saves significant resources in *a scalable manner*; 2) CPT adheres to an extended scaling law derived from Hoffmann et al. (2022) with a joint data-parameter scaling term; 3) The compute-optimal data-parameter allocation for CPT markedly differs based on our estimated scaling factors; 4) The effectiveness of transfer at scale is influenced by training duration and linguistic properties, while robust to *data replaying*, a method that effectively mitigates catastrophic forgetting in CPT. We hope our findings provide deeper insights into the transferability of LLMs at scale for the research community.

1 Introduction

In recent years, Large Language Models (LLMs) pre-trained on web-scale corpora have achieved significant success in various language tasks (Radford et al., 2019; Brown et al., 2020; Achiam et al., 2023). As the scale of pre-training increases, LLMs have exhibited remarkable abilities, particularly in transferring knowledge across different domains (Wei et al., 2022; Tan et al., 2018).

Training an LLM from scratch is prohibitively expensive. To address this, some practitioners leverage *transfer learning* to adapt LLMs to new domains or tasks. This usually involves fine-tuning the models on a small dataset within the target domain. Previous works have showcased multiple benefits of transfer learning in fine-tuning when the transfer gap is small, including faster convergence and better final performance (Zhang et al., 2024; Hernandez et al., 2021). However, it remains unclear if these benefits hold when fine-tuning on massive data or across large distribution shifts (e.g., different languages). This becomes a crucial consideration if one aims to efficiently build an LLM using transfer learning, especially when there is a sufficient amount of data available from different distributions.

To fill this gap, we investigate training LLMs with transfer learning on large pre-training corpora. To be specific, we create LLMs for a new language by using pre-trained LLMs as initialization instead of starting from scratch. We refer to this approach as *continual pre-training* (CPT). The motivation for our work stems from the inherent ability of meta-knowledge to transfer across various languages (Pan and Yang, 2009; Zhuang et al., 2020; Tang et al., 2020; Eronen et al., 2023). By leveraging this transferability, LLMs can use existing linguistic knowledge to enable more efficient training.

In this paper, we conduct pre-training with parameter sizes ranging from 40M to 5B, spanning 40 different sizes, to systematically study the effect of CPT at different conditions and scales. Specifically, we use English as the *source language* for the source model and Chinese as the *target language* for CPT. We compare two different training strategies:

1. **Training from Scratch**: The pre-training of Chinese LLM begins with completely randomly initialized parameters and is trained using Chinese language corpora.

2. **Continual Pre-Training (CPT)**: The parameters of a Chinese LLM are initialized with those from an equivalent English LLM and then trained using Chinese language corpora.

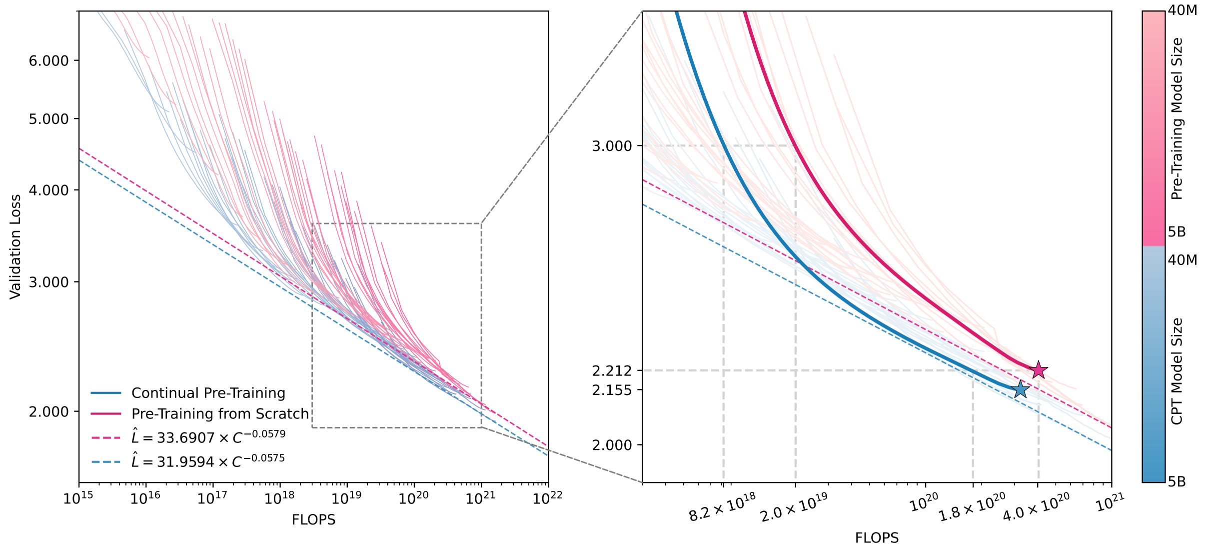

Figure 1: Loss curves of *pre-training* and *continual pre-training* (CPT) across different model sizes. All models are pre-trained on Chinese text while CPT models are initialized from pre-trained English checkpoints. Dashed lines predict optimal loss at each computation level, as estimated in Section 4.2. *(Left)* Overlapped loss-compute power-law visualization, with each line representing one model. *(Right)* CPT LLM (2B parameters) reaches the same loss with approximately 50% fewer FLOPs.

Figure 1 summarizes our main training results. We find that, CPT models of different sizes exhibit a power-law relationship between loss and compute similar to models trained from scratch, but achieve lower loss at each computational level. For models of a given parameter size, CPT consistently outperforms training from scratch, particularly during the initial stages. Throughout the whole training process, CPT saves 25% to 50% of tokens when achieving the same loss.

Our main focus lies in the comparative analysis between the two strategies, including their scaling behaviors, the robustness of scaling, and their corresponding impact factors. For this purpose, we fit a new extended scaling law for CPT, derived from Hoffmann et al. (2022). Our findings are outlined as follows:

- CPT demonstrates persistent training advantages even at the pre-training scale. For example, after training on 70B tokens, the 5.5B model with CPT reaches the same loss as a model trained from scratch with 110B tokens.
- Our extended scaling law more accurately captures the scaling behavior in CPT, revealing a positive multiplicative joint scaling effect between data and parameter size.
- Based on the extended scaling law, we determine the compute-optimal data-parameter allocation for CPT, which favor larger parameter sizes over larger datasets compared to training from scratch.
- The transfer scaling effect in CPT is stronger with fewer training tokens or when the target language is more similar to the source language, but robust to *data replaying*.
- CPT is susceptible to catastrophic forgetting; however, replaying 10% to 30% of the source language data effectively mitigates this issue.

2 Setup

2.1 Training Framework

To compare the transfer effects in CPT versus pre-training from scratch, we train two sets of models with the same parameter sizes. Additionally, another set of model checkpoints is trained in the source language to serve as the initialization for the continually pre-trained models. The training configurations for the three sets of models are shown in Table 1.

To simplify the experiments, we use identical training strategies for all three pre-training sets. All models are pre-trained with a context length of 2048 and undergo training on tokens equivalent to 20 times the model size (e.g., a 5B model is trained on 100B tokens). Although this is far from the extensive pre-training seen in recent practices (Tou-

Table 1: Training configurations for pre-training. All three sets of models are trained with identical parameter sizes, which cover 40 sizes spanning from 50M to 5.5B. Note that the batch size is based on token counts.

Model Set	Initialization	Training Language	Parameter Size & Batch Size (Same for Each Set)
Source Checkpoints	Random	English	50M-1B(23 models) ,1M
Pre-trained from Scratch	Random	Chinese	1B-2.5B(12 models), 2M
Continually Pre-trained	Source Checkpoints	Chinese	2.7B-5.5B(5 models), 4M

vron et al., 2023), as outlined in Hoffmann et al. (2022), the 20x trained token count is sufficient to demonstrate the loss-data scaling relationship. Our learning rate (LR) schedule features a cosine LR decay from a maximum LR of 2×10^{-4} and an LR warm-up, which increases the LR to the maximum in the first 5% of the training session. We use different batch sizes for different parameter sizes, as shown in Table 1.

2.2 Model and Data

Model Architecture We adopt the same decoder-only Transformer architecture as LLaMA2 (Touvron et al., 2023) for all pre-training. We choose LLaMA2 because it is widely studied and proven to scale well across different parameter sizes. Following Muennighoff et al. (2023), we derive architectural parameters for models of each parameter size, which are listed in Appendix C.

Data Sources Our English training data is primarily sampled from the RedPajama dataset (Computer, 2023), while the Chinese training data was acquired from the public web, undergoing filtering and deduplication processes. To study language robustness of the CPT strategy, we also conduct experiments on other languages, including French and Russian. We take their corresponding subsets from mC4 (Raffel et al., 2019) as pre-training data. An total of 10^6 tokens are held out from each respective training set as validation sets, remaining consistent across different models.

2.3 Evaluation Tasks

Throughout experiments, we primarily use *cross-entropy loss* on held-out validation sets as an indicator of model performance. To further validate the generalizability of CPT, we also evaluate LLMs using widely adopted language modeling benchmarks. To assess models in different languages, we choose multilingual versions of existing benchmarks, including XNLI (Conneau et al., 2018), Multilingual Winograde (Sakaguchi et al., 2019), Multilingual Hellaswag (Dac Lai et al., 2023), XStorycloze (Lin et al., 2021), XCopa (Ponti et al., 2020), and PiQA (Bisk et al., 2019). Note that for French and Russian, we exclude XCopa (Ponti et al., 2020) and PiQA (Bisk et al., 2019) as they do not contain splits for these two languages. All evaluations are performed under zero-shot settings. We report normalized accuracy as the metric for each task.

3 Methodology

3.1 Scaling Law for Pre-Training from Scratch

We follow the Chinchilla Scaling Law (Hoffmann et al., 2022) to express cross-entropy loss (L) as a function of parameters (N) and training tokens (D):

$$L(N, D) = E + \frac{A}{N^\alpha} + \frac{B}{D^\beta} \tag{1}$$

where $\{E, A, B, \alpha, \beta\}$ are learned variables. The Chinchilla law further determines the optimal allocation of compute (C) to N and D as:

$$\begin{aligned} N_{\text{opt}}(C) &= G\left(\frac{C}{6}\right)^a \\ D_{\text{opt}}(C) &= G^{-1}\left(\frac{C}{6}\right)^b \end{aligned} \tag{2}$$

where $G = \left(\frac{\alpha A}{\beta B}\right)^{\frac{1}{\alpha+\beta}}$, with $a = \frac{\beta}{\alpha+\beta}$, $b = \frac{\alpha}{\alpha+\beta}$. The ratio of a to b represents the optimal data-to-parameter size allocation.

Additionally, as shown in Kaplan et al. (2020), the optimal loss, independent of parameters and data, also scales with compute C following a power-law relationship:

$$L_{opt}(C) = E' + \frac{A'}{C^\gamma} \tag{3}$$

3.2 Scaling Law for Continual Pre-Training

The Chinchilla law assumes that LLM pre-training is initialized with no prior knowledge, which does not apply to continual pre-training (CPT). To extend the Chinchilla law for CPT, we incorporate insights from Hernandez et al. (2021), introducing an *effectively transferred data* term. According to Hernandez et al. (2021), effective data transfer is modeled as $k(D_F)^{\alpha}(N)^{\beta}$, capturing the idea that larger models store more transferable knowledge. Thus, we extend the D term to include a multiplicative joint effect of both D and N, resulting in our CPT loss function:

$$L(N,D) = E + \frac{A}{N^{\alpha}} + \frac{B'}{D^{\beta'}N^{\gamma}} \tag{4}$$

Accordingly, we update Equation 2 for the extended scaling law:

$$G = \left(\frac{\alpha A}{(\beta'-\gamma)B'}\right)^{\frac{1}{\alpha+\beta'-\gamma}}, \quad a = \frac{\beta'}{\alpha+\beta'-\gamma}, b = \frac{\alpha-\gamma}{\alpha+\beta'-\gamma} \tag{5}$$

Note that we do not update A, E, and α during optimization for CPT. Preliminary experiments show minimal impact of CPT on the N term, so we keep these variables from Equation 1 to reduce variance. Empirical experiments demonstrate that the extended scaling law achieves a lower fitting error than the Chinchilla law for CPT. Additionally, the introduced data-parameter joint term captures meaningful features in scaling behavior, as shown in Section 4.3. We provide fitting error comparison for both scaling laws in Appendix B, where we show that extended scaling law performs better for CPT. We also give more theoretical analysis and interpretation of the extended scaling law in Appendix C.

3.3 Parametric Fit

To fit the learnable variables in Equation 4, we minimize the Huber loss (Huber, 1992) between predicted and observed log loss, with δ set to 10^{-3}. For pre-training from scratch, we minimize Equation 1:

$$\min_{a,b,e,\alpha,\beta} \sum_{\text{Run } I} \text{Huber}_{\delta}\left(\text{LSE}(a-\alpha\log N_i, b-\beta\log D_i, e) - \log L_i\right) \tag{6}$$

where LSE is the *log-sum-exp* operator. We set $A = \exp(a)$, $B = \exp(B)$, $B' = \exp(b')$, and $E = \exp(e)$. For continual pre-training, using the fixed values of a, α, and e from the previous optimization step, we subsequently optimize B', β', and γ in Equation 4:

$$\min_{b',\beta',\gamma} \sum_{\text{Run } I} \text{Huber}_{\delta}\left(\text{LSE}(a-\alpha\log N_i, b'-\beta'\log D_i - \gamma\log N_i, e) - \log L_i\right) \tag{7}$$

We use the Optuna library for hyperparameter search and the L-BFGS algorithm (Nocedal, 1980) for optimal local search, yielding the best hyperparameters. The final parameter values are presented in Table 2a, and the optimized allocation coefficients are shown in Table 2b.

4 Results

4.1 CPT Reaches Lower Loss Throughout Training

Figure 1 reports the validation loss over training for all trained models. It can be seen that pre-training language models from existing checkpoints generally yield lower loss given certain compute constraints. This effect exists across both various model sizes and training stages of the same model. At the start of training, CPT converges significantly faster, advancing pre-training from scratch by orders of magnitudes. The absolute difference of loss becomes smaller as training continues, but a substantial gap in loss persists. Note that Figure 1 is presented on a logarithmic scale. This gap may require several orders of magnitude more iterations before it disappears.

4.2 CPT Preserves Loss-Compute Scaling Relationship

As indicated by Equation 3, optimal validation loss scales with compute following a power-law relationship. We conducted parametric fits for CPT and pre-training from scratch on Equation 3, using the lowest loss at each compute level. The fit results are depicted as dotted lines in Figure 1. For pre-training from scratch, the relationship is represented by $L = 33.69907 \times C^{-0.0579}$. In comparison, the loss for CPT is lower, described by $L = 31.9594 \times C^{-0.0575}$.

The results of the parametric fit indicate that the advantage of lower loss is consistent across each unit of compute expended. This is supported by the significantly reduced coefficient term (from

Table 2: Comparison of parameter estimation and optimization coefficients for Equation 4 and Equation 5. For Continual Pre-Training, parameters E, A, and α are fixed based on values from Training from Scratch.

(a) Estimations for Equation 4.

Model	E	A	B	α	β	γ
Training from Scratch	1.55	420.0	719.5	0.40	0.30	-
Continual Pre-training	1.55	420.0	433.3	0.40	0.20	0.08

(b) Approximated optimization coefficients for Equation 2.

Model	Coeff. a where $N_{\text{opt}} \propto C^a$	Coeff. b where $D_{\text{opt}} \propto C^b$
Training from Scratch	0.429	0.571
Continual Pre-training	0.385	0.615

33.69907 to 31.9594) and the nearly unchanged exponent (from -0.0579 to -0.0575). The nearly unchanged exponent suggests that CPT does not alter the underlying dynamics of the loss-compute relationship, but rather provides an advantageous initial condition.

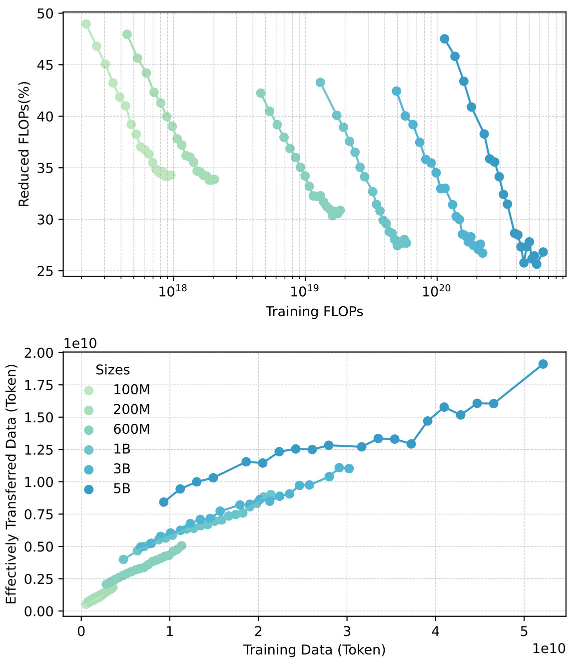

Figure 2: Reduced computational resources (top) and data consumption (bottom) with CPT. Only a subset of models of typical sizes is displayed for simplicity. *(Top)* Percentage reduction in FLOPs C relative to pre-training from scratch PT, as estimated by $(C_{PT} - C_{CPT})/C_{PT}$ at the same loss level for both strategies. *(Bottom)* Effectively Transferred Data, calculated by subtracting the tokens D used by CPT from those used in pre-training from scratch at the same loss level, i.e. $D_{PT} - D_{CPT}$.

4.3 Extended Scaling Law Measures Effectively Transferred Data in CPT

We conducted a further analysis to study the impact of individual factors, specifically data and model size, on loss. Table 2a compares the estimated parameters for CPT with those for training from scratch. As discussed in Section 3.2, only the parameters in the term $\frac{B'}{D^{\beta'} N^{\gamma}}$ are updated for CPT. For CPT, the parameters are $B = 433$, $\gamma = 0.08$, and $\beta = 0.20$. The lower β and positive γ suggest that in CPT, the cross-lingual transfer effect positively correlates with parameter size.

In Figure 2, we measure the transferred training FLOPs and data during CPT to visualize the scaling transfer effect of parameter size, which corroborates our theoretical results. We find that the percentage of reduced training FLOPs steadily decreases during the individual training process, resulting in 25% to 50% FLOPs saved during CPT. On the other hand, effectively transferred data linearly increases with training tokens, with larger models reducing more training FLOPs and data during CPT, indicating a stronger transfer effect. A plausible explanation could be that a larger optimization space contains more linguistic-agnostic knowledge that can transfer more easily.

4.4 CPT Models Generalize to Downstream Tasks

Besides validation losses, we also evaluate cross-lingual CPT on several multi-lingual benchmarks. Using 1.4B parameters, we continually trained models in French (Fr.), Russian (Ru.), and Chinese (Zh) from the same English checkpoint and compared them to models trained from scratch and the original English checkpoints. The results, shown in Figure 3, reveal that CPT improves performance across all languages.

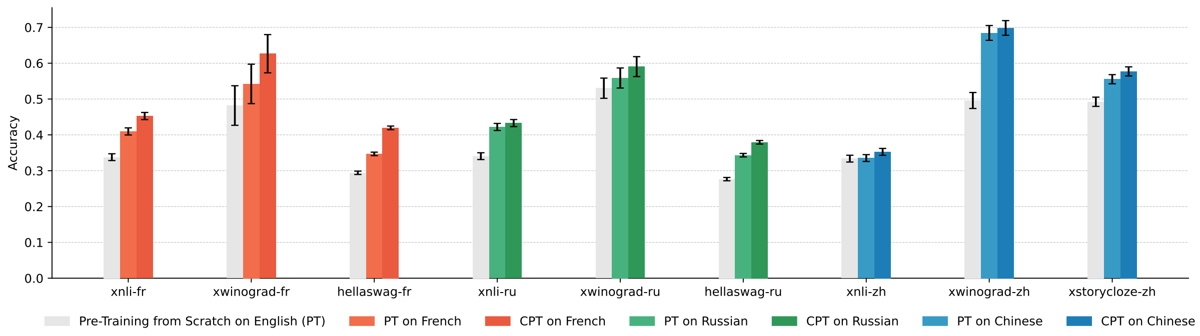

Figure 3: Zero-shot evaluation for pre-trained and continually pre-trained (CPT) models of different languages. CPT models of various languages are initialized from the same checkpoint (light gray).

We assessed the models on their respective language splits of multi-lingual benchmarks to ensure fair comparison. The results indicate that all three languages show improved performance compared to the original pre-trained model, demonstrating that CPT enhances benchmark performance across different languages and scenarios.

We find that French models benefit the most from CPT. This is likely due to the high similarity between French and English, which share many common words and grammatical structures, facilitating more effective cross-lingual transfer compared to Russian and Chinese.

Key Takeaways

- Continual pre-training converges to lower loss faster throughout training, saving 25% to 50% of training FLOPs.
- The transfer effect is most pronounced in the early stages and positively correlated with parameter size.
- The effect generalizes well to downstream evaluations, with languages more similar to English experiencing greater benefits.

5 Discussion

5.1 What is the Compute-Optimal Allocation between Parameter Size and Data?

When total computational resources are limited, there exists a trade-off between model parameter size and the amount of training data during pre-training.

According to the framework established in Section 3, we can determine the optimal allocation between model parameters N_{opt} and training data

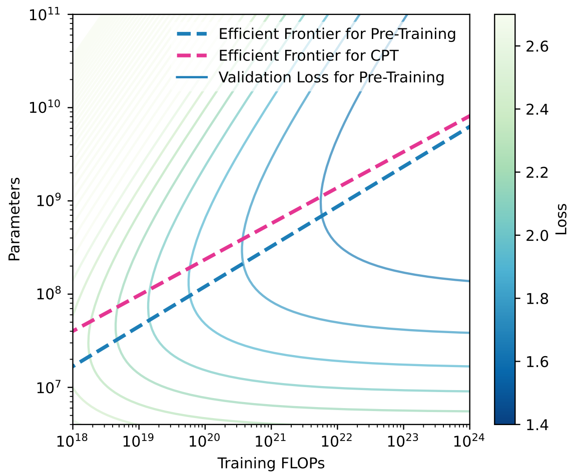

Figure 4: Predicted compute-optimal efficient frontiers on IsoLoss contour for both strategies.

D_{opt} by minimizing the predicted loss L with respect to data D and parameter size N. More specifically, by optimizing Equation 2, we estimate the optimal training data and model parameters for pre-training from scratch to be:

$$\begin{aligned} N_{opt}(C) &= 0.324C^{0.429} \\ D_{opt}(C) &= 0.514C^{0.571} \end{aligned} \tag{8}$$

In comparison, for continual pre-training, the optimal allocations are:

$$\begin{aligned} \hat{N}_{opt}(C) &= 4.79C^{0.385} \\ \hat{D}_{opt}(C) &= 0.035C^{0.615} \end{aligned} \tag{9}$$

A visualization of the efficient frontier of model parameter N with respect to compute over the IsoLoss contour is shown in Figure 4. We find that the optimal parameters for continual pre-training differ from those for pre-training from scratch, favoring less compute for the same model sizes.

This aligns with the nature of cross-lingual transfer learning, where the model in continual pre-training is "pre-matured" due to prior knowledge acquired in the source language. This suggests that, in continual pre-training, using a larger language model is preferred over pre-training on a larger dataset.

It is worth noting that under our settings, larger models not only imply higher model capacity but also involve training on more data in the source language. This may explain why the compute-optimal allocation favors larger base models to some extent. However, this preference may not hold when a larger initialization model checkpoint is under-trained.

5.2 Does Replaying from Source Language Prevent Catastrophic Forgetting?

By continually pre-training a model from the source language, its performance on the target language can be greatly improved. However, with straightforward pre-training strategies, the model's performance on the source language degrades significantly. For example, in a 1.4 billion parameter model, the validation loss on English increases from 2.40 to 3.68 during pre-training. This issue is even more severe in smaller models.

To prevent catastrophic forgetting of the original distributions during continual pre-training, we investigate methods that replay data from the source language during pre-training. We use the term *replaying* to refer to the practice of mixing data from the source language during continual pre-training on the target language.

For models with 1.4B parameters, we continually train several models with mixed training corpora by replaying data at various ratios. We visualize the training curves of these English-replaying models in Figure 5. Note that in Figure 5, the compute is specific to each language rather than the total compute during training.

Figure 5 demonstrates that replaying data from the source language significantly alters the scaling behavior in an intricate manner. As shown on the right side of Figure 5, different ratios of replaying only affect the early stage of training. Models reach the same validation loss when the same amount of compute is used, regardless of the varying ratios of original data, ranging from 1% to 80%.

The left side of Figure 5 compares the relationship between compute and validation loss on the original distribution throughout continual pre-training, which can be viewed as the *"scaling law of forgetting"*. Interestingly, the scaling behavior depicts a power-law relationship similar to that during pre-training from scratch. Validation losses of models at different English replaying ratios increase at the early stage of training and then decline, eventually returning to a lower value than at the start. This suggests that a large amount of original knowledge is preserved throughout continual training, even with a very low English replaying ratio (1% - 5%). Above discoveries suggest that higher levels of replaying original data are beneficial, as replaying does not hinder the scaling properties on the target language while preserving the model's performance on the original distribution.

Key Takeaways

- Under computational constraints, a larger parameter size is preferred over pretraining on a larger dataset in CPT.
- Continual pre-training without replaying data from source language causes severe catastrophic forgetting, especially in smaller models.
- 5% - 30% of source language replaying effectively prevents forgetting while not hindering efficiency of continual pre-training.

6 Related Work

Scaling Law for Large Language Models Scaling laws help us understand how model performance changes with the size of the model and the amount of data. Kaplan et al. (2020) first introduced a detailed scaling law for large language models, demonstrating a clear relationship between model size, training data, and performance. Hoffmann et al. (2022) further explored this by emphasizing the trade-off between model size and data quantity, suggesting a compute-optimal allocation of data and parameters. Recent studies have examined scaling laws under specific conditions. Hernandez et al. (2022) and Muennighoff et al. (2023) focused on the diminishing returns from repeated tokens and excessive parameters. Tay et al. (2022) and Frantar et al. (2023) investigated how different model architectures impact scaling. Scaling laws are also relevant in the context of

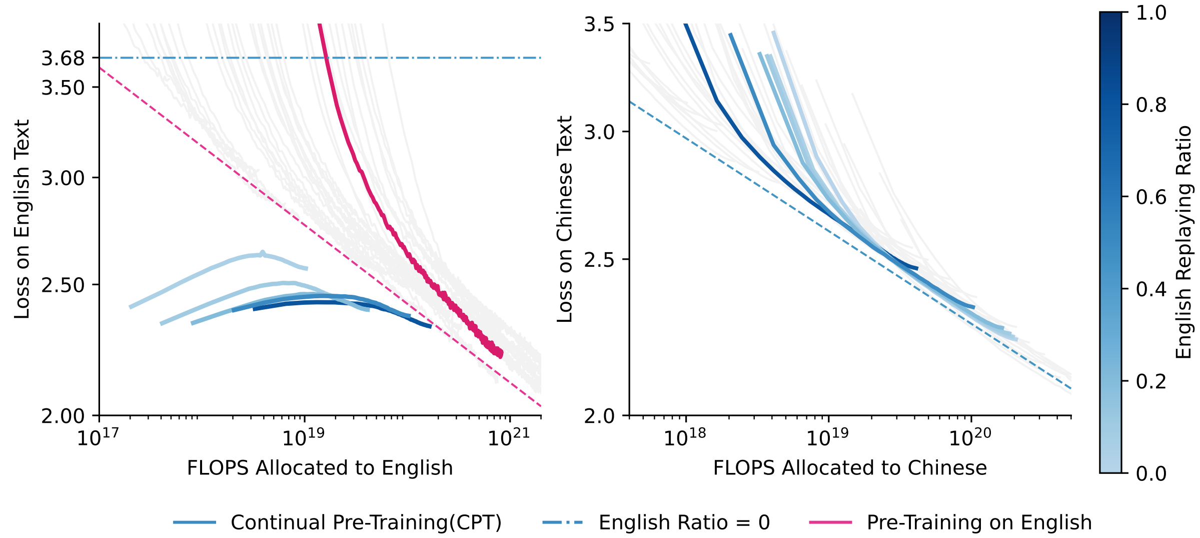

Figure 5: Scaling of CPT with different English replaying ratios. Each blue line represents a 1.4B model continually pre-trained with various replaying ratios and evaluated on two validation sets: English (*left*) and Chinese (*right*). Models with English replaying ratios of 1%, 5%, 10%, 20%, 50%, and 80% are shown from light to dark blue, respectively. FLOPs allocated to each language are calculated by multiplying the corresponding language ratios by the total FLOPs.

newer pre-training methods, such as parameter-efficient fine-tuning (PEFT) (Kalajdzievski, 2024) and Mixture-of-Experts (MoE) (Krajewski et al., 2024).

Cross-Lingual Transfer Learning Transfer learning aims to enhance performance on new tasks by adapting pre-trained models with out-of-domain data. This process is more efficient when the source and target domains are closely related (Pan and Yang, 2009; Zhuang et al., 2020). Cross-lingual pre-training leverages language-independent knowledge embedded in pre-trained LLMs to improve performance in the target language (Wu et al., 2019; Yosinski et al., 2014). Transfer learning is often studied within the context of limited-scale post-training, but it has been shown to be effective at a large pre-training scale with proper techniques (Gupta et al., 2023). A significant challenge in transfer learning is *catastrophic forgetting* (Winata et al., 2023), where the model's ability in the original training domain degrades during transfer learning. Various strategies have been proposed to mitigate catastrophic forgetting, including modified learning rate schedules (Ibrahim et al., 2024; Gupta et al., 2023; Winata et al., 2023), data replay (Ostapenko et al., 2022), and regularization (Farajtabar et al., 2020). Our work combines data replay and modified learning rate schedules to combat catastrophic forgetting.

Our research is closely related to Hernandez et al. (2021), which focused on meta-knowledge transfer between English and code under self-supervised fine-tuning settings. In contrast, we expand continual pre-training to larger-scale and cross-lingual settings, addressing the gap in effective transfer at scale for continual pre-training with significant distribution shifts.

7 Conclusion

In this paper, we explored continual pre-training (CPT), analyzing its principles, influencing factors, and best practices. Through training multiple LLMs with varying sizes, language distributions, and conditions, we derived an extended scaling law for CPT. Our results quantitatively demonstrate that CPT achieves lower loss more quickly, saving 25% to 50% of training resources. However, CPT is particularly sensitive to factors such as language type, training duration, and catastrophic forgetting. Based on these insights, we provide best practices for CPT, including optimal data-to-parameter allocation and replay ratios. These findings motivate future practitioners to apply CPT, offering deeper insights into factors like dataset distribution and training budgets.

Limitations

Language Contamination In this study, we utilized publicly accessible datasets for pre-training. Although the Chinese dataset and mC4 dataset attempt to clean and create language-specific training splits, they cannot entirely prevent the contamination of English at a more granular level. This is particularly challenging due to the inherent nature of many languages, such as French, which often incorporate English words. To estimate the computational effort for different languages, we counted the number of samples processed in each language training split. This approach may be imprecise if the dataset contains a large amount of text in other languages. This issue highlights the need for future research to conduct a more in-depth analysis of the impact of language contamination in multilingual pre-training.

Hyper-Parameter Sensitivity In the training of models across various scales, we selected hyper-parameters based on experience and trial and error. Our preliminary results showed that deviating from optimal hyper-parameters can significantly harm model optimization and disrupt the scaling laws. To maintain consistency, we selected a constant learning rate, optimizer, learning rate scheduler, and batch size that matched the scale of the model for different experiments. This approach is in line with the conclusions of previous studies. Future research should explore the finding of optimal hyper-parameters from the perspective of language-specific scaling laws, which could lead to more effective pre-training configurations.

Scaling Constraints Due to computational limitations, we were unable to cover a wide range of experiments, particularly in cases where the training data was extensive or the model size was very large. This limitation may reduce the generalizability of our findings to scenarios involving larger-scale models or datasets. In this study, we focused exclusively on the LLaMA2 architecture, which is recognized as a practical and effective transformer architecture for measuring scaling properties in pre-training. However, it is important to note that different architectures may have distinct scaling behaviors. This variability is a critical area for future investigation, as understanding these differences could provide deeper insights into optimizing and scaling various model architectures.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2019. Piqa: Reasoning about physical commonsense in natural language. *ArXiv*, abs/1911.11641.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Together Computer. 2023. Redpajama: an open dataset for training large language models.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Viet Dac Lai, Chien Van Nguyen, Nghia Trung Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan A Rossi, and Thien Huu Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. *arXiv e-prints*, pages arXiv–2307.

Juuso Eronen, Michal Ptaszynski, and Fumito Masui. 2023. Zero-shot cross-lingual transfer language selection using linguistic similarity. *Information Processing & Management*, 60(3):103250.

Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. 2020. Orthogonal gradient descent for continual learning. In *International Conference on Artificial Intelligence and Statistics*, pages 3762–3773. PMLR.

Elias Frantar, Carlos Riquelme, Neil Houlsby, Dan Alistarh, and Utku Evci. 2023. Scaling laws for sparsely-connected foundation models. *arXiv preprint arXiv:2309.08520*.

Kshitij Gupta, Benjamin Thérien, Adam Ibrahim, Mats L Richter, Quentin Anthony, Eugene Belilovsky, Irina Rish, and Timothée Lesort. 2023. Continual pre-training of large language models: How to (re) warm your model? *arXiv preprint arXiv:2308.04014*.

Danny Hernandez, Tom Brown, Tom Conerly, Nova DasSarma, Dawn Drain, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Tom Henighan, Tristan Hume, et al. 2022. Scaling laws and interpretability of learning from repeated data. *arXiv preprint arXiv:2205.10487*.

Danny Hernandez, Jared Kaplan, Tom Henighan, and Sam McCandlish. 2021. Scaling laws for transfer. *arXiv preprint arXiv:2102.01293*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Peter J Huber. 1992. Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution*, pages 492–518. Springer.

Adam Ibrahim, Benjamin Thérien, Kshitij Gupta, Mats L Richter, Quentin Anthony, Timothée Lesort, Eugene Belilovsky, and Irina Rish. 2024. Simple and scalable strategies to continually pre-train large language models. *arXiv preprint arXiv:2403.08763*.

Damjan Kalajdzievski. 2024. Scaling laws for forgetting when fine-tuning large language models. *arXiv preprint arXiv:2401.05605*.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Jakub Krajewski, Jan Ludziejewski, Kamil Adamczewski, Maciej Pióro, Michał Krutul, Szymon Antoniak, Kamil Ciebiera, Krystian Król, Tomasz Odrzygóźdź, Piotr Sankowski, et al. 2024. Scaling laws for fine-grained mixture of experts. *arXiv preprint arXiv:2402.07871*.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona T. Diab, Ves Stoyanov, and Xian Li. 2021. Few-shot learning with multilingual generative language models. In *Conference on Empirical Methods in Natural Language Processing*.

Niklas Muennighoff, Alexander M Rush, Boaz Barak, Teven Le Scao, Aleksandra Piktus, Nouamane Tazi, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. 2023. Scaling data-constrained language models. *arXiv preprint arXiv:2305.16264*.

Jorge Nocedal. 1980. Updating quasi-newton matrices with limited storage. *Mathematics of computation*, 35(151):773–782.

Oleksiy Ostapenko, Timothee Lesort, Pau Rodríguez, Md Rifat Arefin, Arthur Douillard, Irina Rish, and Laurent Charlin. 2022. Continual learning with foundation models: An empirical study of latent replay. In *Conference on lifelong learning agents*, pages 60–91. PMLR.

Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

E. Ponti, Goran Glavavs, Olga Majewska, Qianchu Liu, Ivan Vulic, and Anna Korhonen. 2020. Xcopa: A multilingual dataset for causal commonsense reasoning. In *Conference on Empirical Methods in Natural Language Processing*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv e-prints*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Winogrande. *Communications of the ACM*, 64:99 – 106.

Jonathan W Siegel and Jinchao Xu. 2020. Approximation rates for neural networks with general activation functions. *Neural Networks*, 128:313–321.

Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. 2018. A survey on deep transfer learning. In *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27*, pages 270–279. Springer.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. *arXiv preprint arXiv:2008.00401*.

Yi Tay, Mostafa Dehghani, Samira Abnar, Hyung Won Chung, William Fedus, Jinfeng Rao, Sharan Narang, Vinh Q Tran, Dani Yogatama, and Donald Metzler. 2022. Scaling laws vs model architectures: How does inductive bias influence scaling? *arXiv preprint arXiv:2207.10551*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Genta Indra Winata, Lingjue Xie, Karthik Radhakrishnan, Shijie Wu, Xisen Jin, Pengxiang Cheng, Mayank Kulkarni, and Daniel Preotiuc-Pietro. 2023.

Overcoming catastrophic forgetting in massively multilingual continual learning. *arXiv preprint arXiv:2305.16252*.

Shijie Wu, Alexis Conneau, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Emerging cross-lingual structure in pretrained language models. *arXiv preprint arXiv:1911.01464*.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? *Advances in neural information processing systems*, 27.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. 2024. When scaling meets llm finetuning: The effect of data, model and finetuning method. *arXiv preprint arXiv:2402.17193*.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. 2020. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76.

A Downstream Performance of English-Replaying Models at Various Ratios

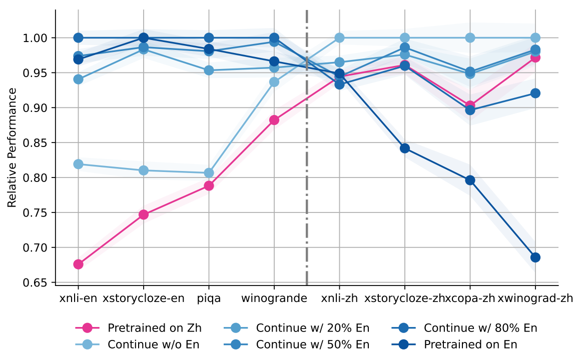

Figure 6: Model performance on English and Chinese benchmarks at different English data replaying ratios with 1.4B parameters. Relative Performance refers to accuracy relative to the highest accuracy achieved across different training settings with 1.4B parameters.

To further analyze the impacts of mixing original data in continual pre-training, we evaluate model performance on English and Chinese benchmarks at different English data mix ratios in Figure 6. The results show that pre-training solely on one language leads to sub-optimal performance on the other language. However, incorporating even a small amount of English data can effectively maintain performance on both original distributions. In practice, around 30% of original data is sufficient to keep the validation loss lower than at the start of continual pre-training.

Models pre-trained only on English excel on English benchmarks but perform poorly on Chinese benchmarks, and vice versa. Adding English data to models initially pre-trained on Chinese improves their English performance without significantly harming their Chinese performance. This improvement is observed across different proportions of English data (20%, 50%, and 80%).An optimal ratio is around 30% English data, balancing low validation loss and high relative performance across both languages. Beyond 50% English data, there are diminishing returns, with marginal gains in English performance and a slight decline in Chinese performance.

B Fitting Error for Extended Scaling Law

Table 3: Comparison of fitting errors L for the Chinchilla Law (Hoffmann et al., 2022) and our extended scaling law on empirical data. The fitt error in huber loss is denoted as $L_{equation}$. Our extended scaling law performs better for CPT, comparable to Chinchilla in pre-training.

Fit Data	Pre-Training	CPT
$L_{Chinchilla}$	0.0090	0.0108
L_{Ours}	0.0094	0.0093
γ in Eq. 4	-0.005	0.080

We applied the Chinchilla Law (Hoffmann et al., 2022) and our extended scaling law to empirical data from both pre-training from scratch and continual pre-training (CPT) on Chinese text. The fitting process minimized the average loss across all trained models for both strategies using the same procedures described in Section 3.3. The results, shown in Table 3, indicate that for pre-training from scratch, the extended scaling law performs similarly to the Chinchilla Law, with the factor γ close to zero. In contrast, for continual pre-training, the joint data-parameter term in the extended scaling law significantly reduces the fitting error, with $\gamma = 0.080$.

C Theoretical Analysis and Interpretation of Extended Scaling Law

First, we review the formulated scaling law proposed by Hoffmann et al. (2022), where they derived and fit a formula for the loss. They decompose the loss $L(N, D)$ into three terms in the abstract functional space:

$$\begin{aligned} L(N, D) &\triangleq L(\bar{f}_{N,D}) \\ &= L(f^*) + \Big(L(\hat{f}_N) - L(f^*)\Big) \\ &\quad + \Big(L(\bar{f}_{N,D}) - L(\hat{f}_N)\Big) \end{aligned} \tag{10}$$

Here, N represents the parameters, D represents the training tokens, f^* represents the optimal Bayesian classifier, $\hat{f}_N$ denotes the optimal transformer model under the constraint of parameters N, $\bar{f}_{N,D}$ represents the outcome obtained through gradient descent under the constraints of parameters N and training tokens D in the experiments.

This functional space decomposition includes three parts:the Bayes risk $L(f^*)$, which is the smallest possible loss for predicting the next token based on the full distribution P, also known as the "entropy of natural text", a term $\Big(L(\hat{f}_N) - L(f^*)\Big)$ related to how well the function approximates based on the hypothesis space size, and a stochastic approximation term $\Big(L(\bar{f}_{N,D}) - L(\hat{f}_N)\Big)$.

Functional space decomposition Our goal is to modify the Equation 1 to fit the scenario of continual pre-training. Consider Continual Pre-training as initialization from a specific model weight state, recalling the functional space decomposition – Equation 10. It serves as a loss decomposition under token and model size constraints, discuss in the abstract functional space. This decomposition method has no relation to the training process (including initialization, naturally), but is a theoretical analysis and summary, so we think that the structure of the entire decomposition is unaffected.

Keeping the structure of Equation 10, let's continue to analyze the impact on the each three term. When considering continual pre-training as a form of random initialization, recall the meaning of the first two terms: the entropy of natural text and the restrictions on the scale of the parameter space, they are both independent of the specific training process and only depend on the model's architecture, as well as the scale of N and D. Therefore, different initialization will only affect The process we implement gradient descent, which is the last term: $L(\bar{f}_{N,D}) - L(\hat{f}_N)$.

Overall, in this scenario, we inherit Equation 10 and then fine-tuned Equation 1.

Inheriting learned variables Pay attention to the detailed settings of our training scenario. the dataset used for training and the details of the entire training process are consistent. We will discuss the expected forms and explain the reasons for inheriting learned variables

(1) For the first term, $L(f^*)$, due to the consistency of the dataset, the entropy of training data naturally maintain consistency between continual pre-training and training from scratch. Numerically, this is equivalent to the same constant E.

(2) For the second term, $L(\hat{f}_N) - L(f^*)$, depends entirely on the number of parameters N that defines the size of the functional approximation space. Siegel and Xu (2020)(Siegel and Xu, 2020) analyzed this term and found it is related to the power of N. We inherit this perspective and believe that its estimated form is $\frac{A}{N^\alpha}$. From the principle of decomposition, this second term does not involve the training phase and only represents the abstract restriction of model's parameter scale. When comparing to training from scratch, the models size N and architecture are completely consistent, so we inherits the values of A and α.

D Model Structural Parameters

Table 4: Structural Parameters for Models of Different Sizes.

Parameter Size(M)	Hidden Layer Size	Intermediate Layer	Attention Head Count	Number of Layers
49	512	3072	8	8
66	576	3584	9	9
86	640	3584	10	10
105	640	3584	10	13
125	640	3584	10	16
137	768	4608	12	12
166	768	4608	12	15
194	768	4608	12	18
208	896	5120	14	14
234	896	5120	14	16
259	896	5120	14	18
301	1024	5632	16	16
334	1024	5632	16	18
368	1024	5632	16	20
512	1280	7168	10	18
591	1280	7168	10	21
616	1408	7680	11	18
670	1280	7168	10	24
711	1408	7680	11	21
766	1536	8704	12	19
806	1408	7680	11	24
879	1536	8704	12	22
992	1536	8704	12	25
1085	1792	9728	14	20
1239	1792	9728	14	23
1393	1792	9728	14	26
1542	2048	11264	16	22
1736	2176	11776	17	22
1743	2048	11264	16	25
1944	2048	11264	16	28
1963	2176	11776	17	25
2112	2304	12800	18	24
2191	2176	11776	17	28
2452	2304	12800	18	28
2791	2304	12800	18	32
2808	2560	13824	20	26
3227	2560	13824	20	30
3647	2560	13824	20	34
4016	2688	14848	22	34
4248	2688	14848	21	36
4657	2816	15360	22	36
5534	3072	16896	24	36