日本語特化の視覚と言語を組み合わせた事前学習モデルの開発
Developing Vision-Language Pre-Trained Models for Japanese

王直 [1]　細野健人 [1]　石塚湖太 [1]
奥田悠太 [1]　川上孝介 [1]
[1] 株式会社博報堂テクノロジーズ
{zhi.wang,kento.hosono,kota.ishizuka,yuta.okuda,kosuke.kawakami}
@hakuhodo-technologies.co.jp

Abstract

視覚と言語を組み合わせた事前学習（Vision-Language Pre-training; VLP）は、多くのタスクに対して Fine-Tuning なしでも一定の性能を発揮する。特に、VLP モデルの一つである CLIP は、ゼロショットで教師あり ResNet-50 と同等の画像分類性能を持つとされるが、その多くは英語向けであり、日本語特化 CLIP での性能は 10–25%劣る。我々は、画像エンコーダと訓練データを 7–10 倍大にし、さらに言語エンコーダの拡大を行うことで、日本語特化 CLIP の画像検索性能 R@5 を 14%改善させた。これは、OpenAI が公開する英語版 CLIP の精度を 2%上回るものである。追加実験でハイパーパラメータの影響を調査し、大きなバッチサイズが性能向上に重要であることを明らかにした。

1 Introduction

Task-agnostic foundation models [1], owing to their adaptability across a diverse spectrum of downstream tasks without compromising quality, have attracted significant public attention subsequent to the unveiling of ChatGPT empowered by the underlying Generative Pre-trained Transformer (GPT) model [2], concurrently presenting numerous opportunities and challenges. The pre-training methodology adopted by the foundation models has expanded into the domain of multimodal learning, catalyzed by the release of the Contrastive Language-Image Pre-training (CLIP) model [3] developed by OpenAI.

The CLIP model integrates an image encoder and a text encoder to align multimodal inputs within a shared embedding space. The image encoder and text encoder undergo joint training to maximize the cosine similarities between the image and text embeddings of matching pairs within the batch, while simultaneously minimizing those for mismatching pairs. The paper [3] demonstrates CLIP model's noteworthy *zero-shot* classification capability on ImageNet-1k [4] comparable to that of the original ResNet-50 [5], suggesting its potential generalizability to downstream tasks such as image captioning, image retrieval, and visual question answering (VQA). Nevertheless, the aforementioned CLIP model is devoid of multilingual support, rendering it unsuitable for non-English environments.

Several endeavors have been undertaken to address this challenge. Chinese CLIP [6], for instance, adopted a two-stage training strategy. Italian CLIP [7] was developed contemporaneously with the original CLIP release. The "sonoisa" model [8] implemented transfer learning to align the embeddings of Japanese text with their English counterparts. The company rinna has introduced a series of Japanese models [9]. The recent release of Japanese Stable CLIP [10] has attained an unprecedented top-1 accuracy of 62.06% on Japanese ImageNet-1k [9, 11].

Notwithstanding these efforts, the accuracy remains more than 10 points lower than that of the English CLIP model of a comparable size [12] (75.3%). It remains unexplored whether this performance gap is intrinsic to language differences or is a consequence of inferior training methods and/or a smaller dataset.

Compounding the issue is that the evaluation metric for Japanese CLIP is presently limited to ImageNet-1k classification accuracy. As articulated in the OpenAI CLIP paper [3], validating CLIP's transferability on image and text retrieval—the tasks for which it is pre-trained—is necessary. It is noteworthy that the CLIP models demonstrate weaker performance on these tasks [3, 13], particularly on the MS-COCO dataset [14]. We opt to address the more challenging task to genuinely test the model's versatility.

Therefore, we define our evaluation metric as zero-shot image recall and text recall on MS-COCO, while reassessing the baseline CLIP models using the same metric.

For the broad community of Japanese AI developers and users with an interest in multimodal representation, we strive to enhance Japanese CLIP to the level of English models.

2 Accuracy gap

To clarify the underlying reason for the accuracy gap between English models and Japanese models, we began by evaluating the text-to-image recall and image-to-text recall on the 5K validation set of MS-COCO [14] for English models and STAIR Captions [15] (Japanese MS-COCO)[1)] for Japanese models.

The obtained results are summarized in Table 1. Notably, the accuracy gap ranges between 10 to 25 points, which can be decomposed into the following:

- The gap of ∼10 percentage points between English and Japanese, as indicated by the comparison between JA ViT-B/16 and EN ViT-B/16, potentially due to differences intrinsic to the language and/or inconsistent size of the dataset (12M for JA vs 400M for EN).
- ∼3 percentage points attributable to model size, demonstrated by comparison between EN ViT-B/16 and EN ViT-L/14.
- 1 to 2 percentage points associated with prompt engineering, inferred from differences between our results without prompt engineering and the previously reported ones [3] for the same model, which is consistent with the discussion in the paper [3].

In contrast to prompt engineering boosting the performance of English CLIP models, Japanese CLIP models suffer from such a technique, as evidenced by our results of 1 to 4 points weaker R@1 performance after prepending and/or appending prompts to the original text. As the Japanese captions consist of a mix of nominals and complete sentences, no single prompt might consistently function well in terms of grammatical structure. This could be the source of the aforementioned detrimental effect of prompt engineering. See details in Appendix A.

1) While MS-COCO [14] and STAIR Captions [15] share a common set of images, the captions in the latter dataset were manually labeled, irrespective of the English text, leading to a lack of 1-to-1 correspondence. However, we use the term "Japanese MS-COCO" throughout the paper to underscore the correspondence on images.

3 Larger image encoder and dataset

3.1 Training strategy

The rinna CLIP model [9] was chosen as the starting point and the baseline since it was the one with the highest zero-shot top-k ($k = 1, 5$) accuracy on the ImageNet validation set amongst Japanese CLIP models accessible to the public at the time of our study.

Note that the model was trained on Conceptual 12M (CC12M) [16] with captions translated into Japanese, the size of which, ∼12 million, is merely 3% of the dataset size employed in the development of the OpenAI CLIP models [3]. In light of the preference for a larger albeit noisy dataset over a smaller one in VLP [3, 17], CC12M is deemed unfavorable compared with the Japanese subset of LAION-5B [18], consisting of approximately 120 million Japanese captions paired with corresponding images. For this reason, the latter dataset was employed in this work.

It is additionally observed that, in terms of the number of parameters, the image encoder, ViT-B/16, of the rinna baseline model [9] (86M) is less than a 50th of that in the currently best-performing publicly available CLIP model, EVA-02-CLIP-E/14+ [13] (4.4B). However, the size of the image encoder is shown to be positively correlated with the model's zero-shot performance on image classification, video classification, and retrieval [3, 12, 13]. Balancing between training/inference time and the resulting performance, we selected the "Huge" variant of Vision Transformer with 14×14 input patch size (ViT-H/14) [19], with 632M parameters, as the architecture for the image encoder. This size is approximately the geometric mean of the base size and the size of the best-performing one.

Considering training efficiency, we initialized our model by combining the well-performing ViT-H/14 image encoder pre-trained for vision-language tasks by OpenCLIP [12] and the text encoder from the rinna Japanese CLIP model [9]. The rationale is that images should be perceived similarly regardless of the native language of the viewer, whereas text is intrinsically language-dependent.

For enhanced data- and compute-efficiency, we applied "Locked-image Tuning" (LiT) [20], the strategy of optimizing the model with a frozen pre-trained image encoder, since it has yielded superior models compared with from-scratch CLIP [3] or A Large-scale ImaGe and Noisy-text

	Text Retrieval			Image Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10
EN ViT-B/16	51.7	76.8	84.3	32.7	57.7	68.2
EN ViT-L/14	57.5	80.3	87.6	36.1	60.8	70.8
EN ViT-L/14*	**58.4**	**81.5**	**88.1**	**37.8**	**62.4**	**72.2**
JA ViT-B/16	36.9	64.3	74.3	24.8	48.8	60.0
JA Base (*ours*)	39.2	66.3	76.6	28.9	53.3	63.9
JA Deeper (*ours*)	**48.7**	74.0	82.4	36.5	61.5	71.8
JA Wider (*ours*)	47.9	**74.2**	**83.2**	**37.3**	**62.8**	**72.7**

Table 1: **Zero-shot retrieval on MS-COCO [14, 15].** The Japanese (JA) ViT-B/16 CLIP model is developed by rinna [9], while the English (EN) models are OpenAI CLIP models [3], in which the ViT-L/14 model is the best OpenAI model pre-trained at a 336-pixel resolution. All external models are re-evaluated without prompt engineering, using an internal script employed consistently throughout the study, with the exception that the row labeled asterisk corresponds to previously published data [3]. Unless stated otherwise, all model comparisons in this study are conducted without prompt engineering to ensure equal conditions. Bold indicates best in category performance for each metric. R@k ($k = 1, 5, 10$) is short for recall@k (%).

embedding (ALIGN) [17] models and has proven its capability to train a non-English CLIP model [6].

3.2 Results and discussion

Combining the ideas of replacing the dataset with a 10-times larger one [18], employing a 7.3-times larger image encoder [19], combining pre-trained encoders from various sources [9, 12], and the efficient "LiT" training strategy [20], we trained our CLIP model for Japanese, named "Base", with details delineated in the Appendix B.1. As indicated in Table 1, our Base model beats the baseline rinna model [9] by 2 to 5 points in zero-shot retrieval on Japanese MS-COCO [15].

For ablation study on hyperparameters, multiple models were trained in parallel, with various settings. We summarize our discoveries as follows.

Batch size. We have conducted two sets, A and B, of experiments, with the batch size in set B being 8 times that in set A. Image retrieval R@5 on Japanese MS-COCO [15] was evaluated for each saved checkpoint, and the last values of R@5 for each experiment were extracted for the following analyses. Since the sample standard deviations are consistently less than 0.2 point, the experiments should be considered converged. For comparison, the expected R@5

ID	1	2	3	4	5	6	7	8
Set	A	A	A	B	B	B	B	B
LR (10^{-4})	2.5	5	10	4	10	20	40	80
R@5	52.2	52.3	52.1	53.0	52.9	52.9	52.9	52.8

Table 2: **Estimated image retrieval R@5 on Japanese MS-COCO [15] for each setting.** The BS for set B (131072) is 8 times that for set A (16384). The width of 95% confidence intervals is consistently 0.5 point.

are listed in Table 2, with details described in Appendix C. Notably, two sets of confidence intervals are disjoint, manifesting a significant improvement from A to B.

Careful scrutiny of Table 2 signifies that such improvement is not influenced by the learning rate (LR) and, therefore, should be attributable to the increased the batch size. Upon comparing the 3rd and 5th settings, we discern a 0.8-point increase despite the fixed LR. In reference to the linear scaling rule for LR [21], one might argue that LR should scale proportionally with the batch size (BS). Comparing 1st vs 6th, 2nd vs 7th, and 3rd vs 8th experiments, we recognize consistent enhancements under a controlled LR-to-BS ratio.

Having seen such results, we hypothesize that the significance of BS may arise from the construction of the loss function calculated from the cosine similarity matrix, containing $O(N^2)$ mismatching pairs, i.e., off-diagonal elements, with N representing the batch size. Given that the information encoded into the loss function grows faster than the BS, a larger BS is expected to be beneficial.

Learning rate. The overlapping confidence intervals within each set suggest that the LR is not a determinant. However, as revealed in the set B, an increase in LR tends to reduce R@5, albeit insignificantly, implying that an overlarge LR could potentially lead to accuracy degradation. Conversely, an excessively small LR might adversely impact training efficiency, as demonstrated in another experiment with an LR of 10^{-6} where the R@5 shows a slow increase across 3 epochs without any sign of convergence.

4 Larger text encoder

Despite its improvement over the Japanese baseline model [9], our Base model still lags behind ViT-L/14, the best OpenAI model trained for English [3], by 7.5 points of text-to-image recall@5 (see Table 1).

As both the image encoder and dataset are enlarged, the

text encoder (111M parameters) may become a bottleneck, limiting the full potential of the model. Building on previous studies of CLIP that involved scaling the text encoder along with the image encoder [3, 12, 13], we explored the possibility of increasing the size of the text encoder to enhance our Japanese CLIP model, Base.

4.1 Model initialization

Considering the risks associated with training the text encoder from scratch, such as the possibility of falling into local minima or even divergence and low training efficiency, we proposed two potential solutions, both of which involve initializing the larger text encoder using our original model, Base.

Modified ZerO. ZerO [22], a fully deterministic initialization technique using only zeros and ones, is reported to maintain the expressivity of the model architecture while ensuring training reproducibility. Noticing its effectiveness in handling ultra-deep networks due to dynamical isometry [23, 24], we employed it with minor modifications to initialize a doubly deeper text encoder (Appendix D).

Model Fusion. As an alternative to expanding the Transformer-based text encoder [25] along the layer dimension, we constructed a wider text encoder by doubling the number of attention heads, followed by initialization that we refer to as "Model Fusion" of two Base models, which involves concatenating all vectors and stacking all weight matrices into block diagonal matrices, with the exceptions explained in Appendix E.

4.2 Results and discussion

We have trained the Deeper model with modified ZerO and the Wider model with Model Fusion. Details are delineated in the Appendix B.2. As shown in Table 1, our models with larger text encoder greatly improve from the rinna baseline [9] as well as the previous model, Base, and attain best performances in Japanese models. Notably, the Wider model outperforms the best OpenAI CLIP model [3] by a margin of 2.0 point of text-to-image recall@5.

Initially, there was apprehension that the Wider model might suffer from the degeneracy issue, i.e., not exhibiting expressivity surpassing that of Base, given that it was initialized with identical copies of two Base models. However, as the model evolved over iterations, the two identical components appeared to diverge, potentially influenced by accumulated errors (such as floating-point errors and numerical errors due to parallelization [26]) and/or randomness introduced by Dropout layers [27]. As a result, the Wider model turned out to outperform the Deeper model across all metrics but text retrieval R@1 (Table 1).[2)]

5 Conclusion and future work

In summary, our contributions comprise:

- The development of a high-quality Japanese CLIP model, Wider, outperforming the best OpenAI CLIP model [3] in image retrieval R@5 by 2.0 points.
- The ZerO initialization adapted for the Transformer architecture, and the Model Fusion technique aiming at the inheritance of previously achieved accuracy by initializing a larger model with two smaller ones.
- Investigation into hyperparameters for training CLIP, highlighting the significance of a large batch size and the learning rate within an appropriate range.
- Insights into the accuracy gap observed between English CLIP models and Japanese CLIP models, primarily attributed to different languages and slightly influenced by model size and prompt engineering.
- Additional evaluation on MS-COCO [15] for existing Japanese CLIP models.
- Applicability of the scaling laws (image encoder, text encoder, dataset) for Japanese CLIP.

Our future work on this project entails a comparison of our models with Japanese Stable CLIP [10], a release that was not available at the time of our study.[3)]

Another avenue for future work involves the application of our models in downstream tasks beyond retrieval, concurrently with a comparative examination against other available Japanese CLIP models [8, 9, 10].

A third idea would be proportional scaling of all dimensions for either/both encoder(s), as is recommended for Vision Transformer [19].

Lastly, alternative initialization techniques, for example, Linear Growth Operator (LiGO) [28], could be investigated and compared with our approaches.

2) However, this does not necessarily indicate that expanding width is superior to increasing depth or that Model Fusion is better than modified ZerO, given the inconsistent parameter counts, with 196M for the Deeper text encoder and 278M for the Wider text encoder.

3) It is noteworthy that the latter model was partially trained on Japanese MS-COCO [15], impeding the assessment of its zero-shot performance on the dataset. Consequently, the evaluation metric must be carefully designed.

Acknowledgments

Special thanks to all current and past members in the same team, Hiroshi Kamiya, Xuan Zhang, Tamotsu Kurioka, team members incentivizing the development of Model Fusion, Tomoki Hoshino, Kai Kurogi, Nobuhito Nishihara, Marcus Jackson, Joe Foran, and the representative director Dai Shibayama. We are grateful to NVIDIA and Amazon Web Services for their infrastructures this project used. We would also like to thank the developers of software packages used in this project including but not limited to PyTorch [29], NumPy [30], Matplotlib [31], pandas [32], the Transformers library [33], Accelerate [34], Black [35], Jupyter [36], Wolfram Engine [37]. Last but not least, we appreciate the Association for Natural Language Processing organizing NLP2024.

References

[1] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, et al. On the opportunities and risks of foundation models. **arXiv preprint arXiv:2108.07258**, 2021.

[2] Tom Brown, Benjamin Mann, Nick Ryder, et al. Language models are few-shot learners. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria Florina Balcan, and Hsuan-Tien Lin, editors, **Advances in Neural Information Processing Systems**, Vol. 33, pp. 1877–1901. Curran Associates, Inc., 2020.

[3] Alec Radford, Jong Wook Kim, Chris Hallacy, et al. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, **Proceedings of the 38th International Conference on Machine Learning**, Vol. 139 of **Proceedings of Machine Learning Research**, pp. 8748–8763. PMLR, 18–24 Jul 2021.

[4] Jia Deng, Wei Dong, Richard Socher, et al. ImageNet: A large-scale hierarchical image database. In **2009 IEEE Conference on Computer Vision and Pattern Recognition**, pp. 248–255, 2009.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. **arXiv preprint arXiv:1512.03385**, 2015.

[6] An Yang, Junshu Pan, Junyang Lin, et al. Chinese CLIP: Contrastive vision-language pretraining in Chinese. **arXiv preprint arXiv:2211.01335**, 2022.

[7] Federico Bianchi, Giuseppe Attanasio, Raphael Pisoni, et al. Contrastive language-image pre-training for the Italian language. **arXiv preprint arXiv:2108.08688**, 2021.

[8] sonoisa. https://huggingface.co/sonoisa/clip-vit-b-32-japanese-v1.

[9] 沢田 慶シーン 誠. 日本語における言語画像事前学習モデルの構築と公開. In **The 25th Meeting on Image Recognition and Understanding**, 2022.

[10] Makoto Shing and Takuya Akiba. Japanese Stable CLIP ViT-L/16. https://huggingface.co/stabilityai/japanese-stable-clip-vit-l-16.

[11] Gregor Geigle, Radu Timofte, and Goran Glavaš. Babel-ImageNet: Massively multilingual evaluation of vision-and-language representations. **arXiv preprint arXiv:2306.08658**, 2023.

[12] Mehdi Cherti, Romain Beaumont, Ross Wightman, et al. Reproducible scaling laws for contrastive language-image learning. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**, pp. 2818–2829, June 2023.

[13] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. EVA-CLIP: Improved training techniques for CLIP at scale. **arXiv preprint arXiv:2303.15389**, 2023.

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, et al. Microsoft COCO: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, **Computer Vision – ECCV 2014**, pp. 740–755, Cham, 2014. Springer International Publishing.

[15] STAIR Captions: Constructing a large-scale Japanese image caption dataset. In **Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)**, pp. 417–421, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[16] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**, pp. 3558–3568, June 2021.

[17] Chao Jia, Yinfei Yang, Ye Xia, et al. Scaling up visual and vision-language representation learning with noisy text supervision. In Marina Meila and Tong Zhang, editors, **Proceedings of the 38th International Conference on Machine Learning**, Vol. 139 of **Proceedings of Machine Learning Research**, pp. 4904–4916. PMLR, 18–24 Jul 2021.

[18] Christoph Schuhmann, Romain Beaumont, Richard Vencu, et al. LAION-5b: An open large-scale dataset for training next generation image-text models. In **Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track**, 2022.

[19] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, et al. An image is worth 16x16 words: Transformers for image recognition at scale. **arXiv preprint arXiv:2010.11929**, 2021.

[20] Xiaohua Zhai, Xiao Wang, Basil Mustafa, et al. LiT: Zero-shot transfer with locked-image text tuning. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**, pp. 18123–18133, June 2022.

[21] Priya Goyal, Piotr Dollár, Ross Girshick, et al. Accurate, large minibatch SGD: Training ImageNet in 1 hour. **arXiv preprint arXiv:1706.02677**, 2018.

[22] Jiawei Zhao, Florian Schäfer, and Anima Anandkumar. ZerO initialization: Initializing neural networks with only zeros and ones. **arXiv preprint arXiv:2110.12661**, 2022.

[23] Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. **arXiv preprint arXiv:1312.6120**, 2014.

[24] Minmin Chen, Jeffrey Pennington, and Samuel Schoenholz. Dynamical isometry and a mean field theory of RNNs: Gating enables signal propagation in recurrent neural networks. In Jennifer Dy and Andreas Krause, editors, **Proceedings of the 35th International Conference on Machine Learning**, Vol. 80 of **Proceedings of Machine Learning Research**, pp. 873–882. PMLR, 10–15 Jul 2018.

[25] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2019.

[26] Fabienne Jézéquel, Jean-Luc Lamotte, and Issam Saïd. Estimation of numerical reproducibility on CPU and GPU. In **2015 Federated Conference on Computer Science and Information Systems (FedCSIS)**, pp. 675–680, 2015.

[27] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. **arXiv preprint arXiv:1207.0580**, 2012.

[28] Peihao Wang, Rameswar Panda, Lucas Torroba Hennigen, et al. Learning to grow pretrained models for efficient transformer training. **arXiv preprint arXiv:2303.00980**, 2023.

[29] Adam Paszke, Sam Gross, Francisco Massa, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Hanna Wallach, Hugo Larochelle, Alina Beygelzimer, et al., editors, **Advances in Neural Information Processing Systems**, Vol. 32, pp. 8024–8035. Curran Associates, Inc., 2019.

[30] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, et al. Array programming with NumPy. **Nature**, Vol. 585, No. 7825, pp. 357–362, September 2020.

[31] John D. Hunter. Matplotlib: A 2d graphics environment. **Computing in Science & Engineering**, Vol. 9, No. 3, pp. 90–95, 2007.

[32] The pandas development team. pandas-dev/pandas: Pandas, February 2020.

[33] Thomas Wolf, Lysandre Debut, Victor Sanh, et al. Transformers: State-of-the-art natural language processing. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations**, pp. 38–45, Online, October 2020. Association for Computational Linguistics.

[34] Sylvain Gugger, Lysandre Debut, Thomas Wolf, et al. Accelerate: Training and inference at scale made simple, efficient and adaptable. https://github.com/huggingface/accelerate, 2022.

[35] Lukasz Langa and contributors to Black. Black: The uncompromising Python code formatter.

[36] Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, et al. Jupyter notebooks – a publishing format for reproducible computational workflows. In F. Loizides and B. Schmidt, editors, **Positioning and Power in Academic Publishing: Players, Agents and Agendas**, pp. 87 – 90. IOS Press, 2016.

[37] Wolfram Research, Inc. Wolfram engine, Version 13.3. Champaign, IL, 2023.

[38] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In **2015 IEEE International Conference on Computer Vision (ICCV)**, pp. 1026–1034, Los Alamitos, CA, USA, December 2015. IEEE Computer Society.

[39] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. **arXiv preprint arXiv:1608.03983**, 2017.

[40] Harald Cramér. **Mathematical Methods of Statistics**. Princeton University Press, Princeton, NJ, US, 1946.

[41] Gokarna Aryal and Saralees Nadarajah. Information matrix for beta distributions. **Serdica Mathematical Journal**, Vol. 30, No. 4, pp. 513–526, 2004.

[42] Sean Rangel. Confidence interval for the mean of a beta distribution, December 2021.

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, et al., editors, **Advances in Neural Information Processing Systems**, Vol. 30. Curran Associates, Inc., 2017.

[44] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. **arXiv preprint arXiv:1607.06450**, 2016.

A Prompt engineering

Prepending "a photo of" boosted OpenAI CLIP's R@1 by 1–2 points [3]. However, we tested multiple prompts on rinna CLIP [9] where braces were replaced with captions, resulting in the following sequence with increasing retrieval performance: 「{}」の画像 < 「{}」の写真 < {} の写真 < {} の画像 < 画像の中で {} < 写真の中で {} ≤ {}. Note that the captions in Japanese MS-COCO [15] are a blend of full sentences such as "テーブルの上に花瓶に入った花が置いてある", and nominals such as "草むらに座る亜麻色のクマの近影".

B Details of training

B.1 Base model

Due to the discrepancy in embedding dimensions (i.e., the output dimensions of the final image/text projection layer) between the rinna Japanese CLIP model [9] (512) and the OpenCLIP ViT-H/14 model [12] (1024), the final text projection layer was reshaped to match the final image projection layer and randomized with a Kaiming uniform initializer [38].

Given the re-initialization of the text projection layer from scratch, the training commenced with a BS of 8192 and a linearly increasing LR from 0 to 10^{-6} for 2000 iterations, subsequently transitioning to constant-LR training, amounting to 3 epochs. The BS was subsequently elevated to 131072, accompanied by a linearly increasing LR reaching its maximum value of 4×10^{-4} within 500 iterations. Such training persisted for 1 epoch, succeeded by an additional training of ∼2 epochs, equipped with a cosine annealing LR scheduler [39] decaying to 1% of its max value. Other configurations remained consistent with the procedures previously published [3, 9].

B.2 Deeper and Wider model

Deeper. The LR decayed from 10^{-4} to 10^{-6} using a cosine annealing LR scheduler [39] for 1 epoch, with frozen pre-trained component (i.e., the shallow half of the text encoder), followed by repetitive warm restarts, spanning ∼8 epochs, with the relaxation of all constraints on the text encoder.

Wider. The initialized text encoder was freely trained for ∼7 epochs, through a sequence of 3 warm restarts of LR, cosine-annealed [39] from 10^{-4} to 10^{-6}.

C Confidence interval

Consider retrievals using the i-th checkpoint within an experiment as $n_i \in \mathbb{Z}^+$ independent and identically distributed (i.i.d.) Bernoulli trials, out of which $s_i \in \{0, \dots, n_i\}$ are successful, in which case the likelihood of s_i has a binomial distribution $\mathrm{Bin}(n_i, p)$ given the recall p as a parameter assumed to follow the conjugate prior, i.e., beta distribution $\mathrm{Beta}(a, b)$, with hyperparameters $\alpha, \beta > 0$ and support $p \in [0, 1]$. With the assumption that all observations of s_i are independent, of which the set S consists, it can be derived that the posterior probability $\mathbb{P}(p \mid S; a, b)$ is $\mathrm{Beta}(a + \sum_i s_i, b + \sum_i (n_i - s_i))$. We can calculate the confidence interval (CI) with a significance level of $\alpha \in (0, 1)$ as $(p_{\min}(a, b; \alpha), p_{\max}(a, b; \alpha))$ such that

$$\mathbb{P}[p \le p_{\min}(a, b; \alpha) \mid S; a, b] = \frac{\alpha}{2},$$
$$\mathbb{P}[p \ge p_{\max}(a, b; \alpha) \mid S; a, b] = \frac{\alpha}{2}.$$

Note that $\sum_i s_i \gg 1$, $\sum_i (n_i - s_i) \gg 1$. When a and b are small, CI can be estimated as $(p_{\min}(1, 1; \alpha), p_{\max}(1, 1; \alpha))$, the latter of which is equivalently the case of uniform prior.

Alternatively, it can be assumed that R@5 values for an experiment are i.i.d. samples from beta distribution $\mathrm{Beta}(\phi\mu, \phi(1-\mu))$, with $\mu \in (0, 1)$ being the mean and precision denoted as $\phi \in (0, +\infty)$. Since each R@5 is calculated from $n_i = 25000$ samples, with 5 captions corresponding each of the 5000 images, the posterior probability derived above implies that the precision parameter ϕ should be no less than n_i. Therefore, ϕ is fixed to the value for the most conservative estimation of CI. Thanks to the asymptotic normality of maximum likelihood estimators (MLE) [40], we can calculate the CI from Fisher information matrix [41] following established procedures [42]. Note that the MLE for the mean μ coincides with the average R@5, $(\sum_i s_i) / (\sum_i n_i)$, and that two approaches of CI calculation yield identical results—the mean μ, summarized in Table 2, is the midpoint of the 95% CI whose width is 5 points, consistently.

D Modified ZerO initialization

All layers of the Base text encoder are inherited by the larger text model, and the newly added deeper Transformer layers [43] are initialized with zeros, with exceptions listed as follows.

- ZerO is applied to both the query weight matrix and the dimension-increasing weight matrix of the feed-forward network for each added layer.
- The weights of newly added LayerNorm [44] are set to 1.
- The output weight matrices of multi-head attention in deeper layers are either randomized with a Kaiming uniform initializer [38] or initialized with ZerO. Both yield models with similar performance, and therefore, we report results generated from the former initialization.

E Model Fusion

Denote two text models with a shared tokenizer as A and B. Let the common dimensionality of the inner-layer of the feed-forward networks within the layers be d_{ff}, and the Transforer hidden size be $d_{\mathrm{model},A}$ and $d_{\mathrm{model},B}$, respectively. Model Fusion generates a combined model by concatenating all vectors and stacking all weight matrices into block diagonal matrices, with the exceptions listed as follows.

- The feed-forward networks expressed by

$$\mathbf{z}_i \in \mathbb{R}^{d_{\mathrm{model},i}} \mapsto a\left(\mathbf{z}_i \mathbf{W}_{1,i}^\mathsf{T} + \mathbf{b}_{1,i}\right) \mathbf{W}_{2,i}^\mathsf{T} + \mathbf{b}_{2,i},$$

  where i is either A or B, a denotes activation function, $\mathbf{z}$ is the input row vector, $\mathbf{W}_{1,i} \in \mathbb{R}^{d_{ff} \times d_{\mathrm{model},i}}$, $\mathbf{W}_{2,i} \in \mathbb{R}^{d_{\mathrm{model},i} \times d_{ff}}$ are weight matrices, and $\mathbf{b}_{1,i} \in \mathbb{R}^{d_{ff}}$, $\mathbf{b}_{2,i} \in \mathbb{R}^{d_{\mathrm{model},i}}$ are row vectors for the bias term, are combined into

$$\mathbf{z} \in \mathbb{R}^{\sum_i d_{\mathrm{model},i}} \mapsto a\left(\mathbf{z}\mathbf{W}_1^\mathsf{T} + \mathbf{b}_1\right) \mathbf{W}_2^\mathsf{T} + \mathbf{b}_2,$$

  where

$$\mathbf{W}_1 = \begin{bmatrix} \frac{1}{2}\mathbf{W}_{1,A} & \frac{1}{2}\mathbf{W}_{1,B} \end{bmatrix},$$
$$\mathbf{b}_1 = \frac{1}{2}\left(\mathbf{b}_{1,A} + \mathbf{b}_{1,B}\right),$$
$$\mathbf{W}_2 = \begin{bmatrix} \mathbf{W}_{2,A}^\mathsf{T} & \mathbf{W}_{2,B}^\mathsf{T} \end{bmatrix}^\mathsf{T},$$

  and $\mathbf{z}$ and $\mathbf{b}_2$ are obtained from concatenation.
- The final projection matrix of the text encoder is treated similar to $\mathbf{W}_1$ to generate averaged outputs from both models.