FLAVA: A Foundational Language And Vision Alignment Model
(Supplementary Material)

Amanpreet Singh* Ronghang Hu* Vedanuj Goswami*
Guillaume Couairon Wojciech Galuba Marcus Rohrbach Douwe Kiela
Facebook AI Research (FAIR)

A. Hyperparameters and details of FLAVA

We summarize the hyperparameters in our FLAVA model in Table A.1. We also list the sampling probabilities of the datasets for joint pretraining in Table A.2, including PMD (multimodal paired image and text), ImageNet-1k (unimodal unpaired images), and CCNews & BookCorpus (unimodal unpaired text).

We find that a large batch size, a large weight decay, and a long warmup are helpful to stabilize training and prevent divergence under a large learning rate. Based on this finding, we performed a hyperparameter search based by monitoring the learning curve as well as monitoring the zero-shot image classification accuracy based on the image-text contrastive loss on using the text templates from CLIP [7] to obtain the hyperparameters above.

B. Training and evaluation details

B.1. Pretraining details

Language encoder pretraining. We follow RoBERTa$_{base}$ pretraining hyperparameters to train our pre-norm ViT-based text encoder [5]. Specifically, we pretrain our text encoder using masked language modeling (MLM) [3] on CC-News and BookCorpus for 125K iterations with a batch size of 2048 and a learning rate of 5e-4. We pick the best checkpoint based on the MLM loss without any further hyperparameter sweeps over RoBERTa's default configuration.

Vision encoder pretraining. We pretrain the image encoder in FLAVA on the ImageNet-1k dataset following either BEiT [1] or DINO [2]. When pretraining a ViT-B/16 image encoder with BEiT, we adopt the hyperparameters and training details in [1] with a masked image modeling loss by predicting the dVAE visual tokens of the masked image patches. We also follow the training protocols in [2] to pretrain a DINO ViT-B/16 model as our image encoder. As discussed in Sec. C, we empirically find that the DINO-pretrained image encoder gives better final performance.

Hyperparameter	**Value**
Image Encoder	
hidden size	768
number of heads	12
intermediate size	3072
number of layers	12
dropout prob.	0
patch size	16×16
input image size (pretraining)	224×224
input image size (VQAv2 fine-tuning)	480×480
input image size (all other evaluation)	224×224
Text Encoder	
hidden size	768
number of heads	12
intermediate size	3072
number of layers	12
dropout prob.	0
Multimodal Encoder	
hidden size	768
number of heads	12
intermediate size	3072
number of layers	6
dropout prob.	0
Others	
text vocabulary size	30522
image dVAE codebook size	8192
global contrastive loss projection dim	512
Training	
batch size	8192
learning rate	1e-3
learning schedule	`warmup_cosine`
warmup updates	10000
AdamW weight decay	0.1
AdamW β_1	0.9
AdamW β_2	0.999

Table A.1. A summary of various hyperparameters in FLAVA.

*Equal contribution.

Dataset	Sampling probability
PMD	0.70
ImageNet-1k	0.15
CCNews & BookCorpus	0.15

Table A.2. Sampling probabilities of PMD (multimodal paired image and text), ImageNet-1k (unimodal unpaired images), and CCNews & BookCorpus (unimodal unpaired text) for joint FLAVA pretraining on the three modalities.

Full FLAVA pretraining. We pretrain jointly on the unimodal and multimodal datasets, following the sampling probabilities of these datasets as provided in Table A.2. Specifically, for each update, we pick a dataset based on its sampling probability and obtain a complete batch from it. In all our ablations, we use a training schedule such that the PMD dataset is sampled for a total of 150K iterations. We monitor the zero-shot accuracy on ImageNet classification [9] every 8K updates and select the best checkpoint based on the ImageNet-1k zero-shot accuracy. We follow [7] to calculate the zero-shot accuracy.

B.2. Vision, language and multimodal evaluation

We evaluate the pretrained FLAVA model across a broad range of vision, natural language, and multimodal tasks. We discuss our evaluation details of these tasks below.

Linear probing on vision tasks. We perform linear probe evaluations on the datasets by closely following the setup described in [7]. We extract image features from the final layer of the image encoder (before the multi-modal encoder) and train a logistic regression classifier (`L-BFGS` implementation from [6]) on the extracted image features. We follow the hyperparameters similar to [7] : 1000 iterations, logistic regression λ parameter sweep from 1e-6 to 1e6.

Fine-tuning on NLP tasks. For NLP tasks, we finetune the language encoder end to end for all the GLUE tasks. We add a classification head on top of the language encoder for all the tasks, except for the STS-B task, where we use a regression head. The hyperparameters we use for finetuning follow the setup of RoBERTa [5].[1]

Fine-tuning on multimodal VQA, SNLI-VE, and Hateful Memes. We adopt the following settings when finetuning on VQA, SNLI-VE, and Hateful Memes, adding a 2-layer classifier head with a hidden dimension of 1536 on top of $\mathbf{h}_{\mathrm{CLS},M}$ from the multimodal encoder (corresponding to `[CLS_M]`). For VQAv2, we use 1e-4 learning rate, 44000 updates, and an input image size of 480×480. For SNLI-VE and Hateful Memes, we use 1e-5 learning rate, a total iteration number of 24000, and an input image size of 224×224 (we use the OCR tokens extracted from the images as the textual input for Hateful Memes). On all these three tasks, we use the AdamW optimizer with a batch size of 256, 1e-2 weight decay, and 2000 warm-up iterations followed by a cosine decay schedule.

We use the same approach above to also evaluate the CLIP model on VQAv2, SNLI-VE, and Hateful Memes datasets. Since CLIP does not have a multimodal encoder, we concatenate the image feature vector from its image encoder and the text feature vector from its text encoder, apply a 2-layer classifier head (with the same hidden dimension of 1536) over the concatenated feature, and finetune the model following the same hyperparameters as for FLAVA.

Zero-shot multimodal text and image retrieval. We also evaluate the FLAVA model on the multimodal zero-shot retrieval tasks over the Flickr30K and COCO datasets, where the model needs to select a text caption based on a query image or select an image based on a query caption. We use the cosine similarities between the image and text feature computed in the global contrastive loss in FLAVA as the matching scores between the image and text modalities. Then, the text caption (or image) with the highest matching score to the query is retrieved. Similarly, we also evaluate the zero-shot text and image retrieval performance of the CLIP model using the cosine similarities between its image and text features.

C. Additional ablations and analyses

Unimodal-pretrained vision encoders. We experiment with initializing our model from different pretrained vision encoders (while keeping the language encoder the same). We study two different self-supervised ViT-B/16 models trained on ImageNet-1k: i) BEiT and ii) DINO. Under three FLAVA pretraining settings, $\mathrm{FLAVA_C}$, $\mathrm{FLAVA_{MM}}$ and FLAVA (full pretraining), initializing from any of the two pretrained vision encoders (along with pretrained language encoders) leads to significant improvement in all tasks. In Table C.1, comparing columns 5 vs 6, 8 vs 9, and 11 vs 12 between BEiT and DINO initialization, DINO gives better performance on vision and multimodal tasks. On NLP tasks, the results are mixed and comparable, as the language encoder is initialized from the same pretrained weights.

Global vs. local contrastive losses. In our FLAVA model, we apply a global contrastive loss, where the image and text features are gathered across GPUs and the loss is back-propagated through the gathering operation to all GPUs. This is in contrast with the implementation in [4], where the loss is only back-propagated to local features from the same GPU. It can be seen from Table C.1 (columns 3 vs 4) that the global contrastive loss (column 4) leads to a noticeable gain in the average vision and NLP performance compared to its local contrastive counterpart and also provides a slight boost in multimodal performance.

Observations on SST and VQA. Some of our vision tasks

[1]We follow hyperparameters used in FairSeq RoBERTa repo for finetuning on GLUE tasks without any further sweeping.

	MIM	MLM	FLAVA$_C$				FLAVA$_{MM}$			FLAVA			CLIP	CLIP
			local contrastive		BEiT init.	DINO init.		BEiT init.	DINO init.		BEiT init.	DINO init.		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Datasets	PMD	PMD	PMD				PMD			PMD+IN-1k+CCNews+BC			PMD	400M [7]
MNLI	–	73.22	70.65	70.99	74.12	74.23	76.82	78.59	78.74	78.06	**80.96**	80.32	32.84	33.52
COLA	–	39.55	9.76	17.58	15.30	14.92	38.97	39.41	45.04	44.22	44.52	**50.65**	11.02	25.37
MRPC	–	73.24	73.20	76.31	74.28	73.50	79.14	79.30	80.66	78.90	**85.96**	84.16	68.74	69.91
QQP	–	86.68	85.08	85.94	87.29	87.02	88.48	88.52	88.82	88.60	**89.27**	88.74	59.16	65.33
SST-2	–	87.96	85.78	86.47	88.30	89.22	89.33	91.51	90.02	90.14	**91.74**	90.94	83.49	88.19
QNLI	–	82.32	70.25	71.85	80.67	80.93	84.77	86.05	86.23	86.40	**88.52**	87.31	49.46	50.54
RTE	–	50.54	49.10	51.99	52.71	49.82	51.99	**57.76**	50.90	54.87	51.62	**57.76**	53.07	55.23
STS-B	–	78.89	60.08	57.28	76.93	76.17	84.29	**86.70**	85.86	83.21	86.64	85.67	13.70	15.98
NLP Avg.	–	71.55	62.99	64.80	68.70	68.22	74.22	75.98	75.78	75.55	77.40	**78.19**	46.44	50.51
ImageNet	41.79	–	70.64	74.09	74.07	75.87	74.34	74.37	**76.23**	73.49	74.59	75.54	72.95	80.20
Food101	53.30	–	85.02	87.77	88.04	**88.94**	87.53	87.82	88.88	87.39	88.02	88.51	85.49	91.56
CIFAR10	76.20	–	91.74	**93.44**	91.65	92.49	92.37	91.17	92.29	92.63	91.91	92.87	91.25	94.93
CIFAR100	55.57	–	73.54	**78.37**	74.58	76.32	78.01	74.76	76.97	76.49	75.29	77.68	74.40	81.10
Cars	14.71	–	60.86	**72.12**	69.92	71.83	72.07	69.44	71.84	66.81	69.44	70.87	62.84	85.92
Aircraft	13.83	–	42.96	**49.74**	46.11	49.17	48.90	44.73	48.63	44.73	45.81	47.31	40.02	51.40
DTD	55.53	–	73.51	76.86	76.97	**77.77**	76.91	75.80	77.18	75.80	76.54	77.29	73.40	78.46
Pets	34.48	–	80.10	84.98	84.63	86.26	84.93	84.55	**86.75**	82.77	84.60	84.82	79.61	91.66
Caltech101	67.36	–	92.98	94.91	94.95	**95.94**	95.32	95.46	95.45	94.95	94.89	95.74	93.76	95.51
Flowers	67.23	–	94.42	96.36	96.08	**96.86**	96.39	96.03	96.49	95.58	96.34	96.37	94.94	97.12
MNIST	96.40	–	97.75	98.39	98.28	98.49	98.58	97.94	98.38	**98.70**	98.38	98.42	97.38	99.01
STL10	80.12	–	97.52	98.06	98.71	98.75	98.31	98.50	**98.94**	98.32	98.55	98.89	97.29	99.09
EuroSAT	95.48	–	95.76	97.00	97.04	97.24	96.98	97.36	96.72	97.04	**97.40**	97.26	95.70	95.38
GTSRB	63.14	–	73.81	78.92	74.76	79.27	77.93	76.13	79.01	77.71	76.96	**79.46**	76.34	88.61
KITTI	86.03	–	87.77	87.83	89.04	88.03	88.84	**89.77**	89.71	88.70	88.57	89.04	84.89	86.56
PCAM	85.10	–	**86.04**	85.02	85.09	85.25	85.51	85.29	85.27	85.72	84.84	85.31	83.99	83.72
UCF101	46.34	–	77.82	82.69	80.60	82.90	82.90	81.52	**83.40**	81.42	81.60	83.32	77.85	85.17
CLEVR	61.51	–	73.86	79.35	80.24	79.84	**81.66**	80.96	79.81	80.62	80.88	79.66	73.64	75.89
FER 2013	50.98	–	57.40	59.96	60.91	60.30	60.87	60.34	**61.12**	58.99	60.43	**61.12**	57.04	68.36
SUN397	52.45	–	79.43	81.27	81.96	**82.75**	81.41	81.99	82.16	81.05	81.76	82.17	79.96	82.05
SST	57.77	–	58.65	56.67	58.05	58.98	**59.25**	56.29	57.17	56.40	56.12	57.11	56.84	74.68
Country211	8.87	–	22.98	27.27	26.87	27.84	26.75	26.64	27.69	27.01	27.28	**28.92**	25.12	30.10
Vision Avg.	57.46	–	76.12	79.14	78.57	**79.59**	79.35	78.49	79.55	78.29	78.65	79.44	76.12	82.57
VQAv2	–	–	65.82	67.13	66.98	68.34	71.69	73.14	**73.75**	71.29	72.23	72.49	59.81	54.83
SNLI-VE	–	–	74.03	73.27	74.37	73.59	78.36	**79.05**	79.01	78.14	78.49	78.89	73.53	74.27
Hateful Memes	–	–	59.31	55.58	63.20	59.65	70.72	69.61	**79.69**	77.45	74.10	76.09	56.59	63.93
Flickr30K TR R@1	–	–	68.80	68.30	64.90	70.80	69.30	**71.00**	69.80	64.50	69.50	67.70	60.90	82.20
Flickr30K TR R@5	–	–	91.80	93.50	92.20	92.90	92.90	91.80	92.00	90.30	93.00	**94.00**	88.90	96.60
Flickr30K IR R@1	–	–	59.24	60.56	63.14	65.06	63.16	64.60	64.84	60.04	63.78	**65.22**	56.48	62.08
Flickr30K IR R@5	–	–	84.58	86.68	87.94	89.32	87.70	87.98	88.94	86.46	87.94	**89.38**	83.60	85.68
COCO TR R@1	–	–	**48.28**	43.08	44.00	45.06	43.48	42.44	44.62	39.88	42.24	42.74	37.12	52.48
COCO TR R@5	–	–	76.96	75.82	75.90	77.04	76.76	75.66	**77.34**	72.84	75.38	76.76	69.48	76.68
COCO IR R@1	–	–	37.34	37.59	38.28	**39.20**	38.46	37.54	38.99	34.95	37.89	38.38	33.29	33.07
COCO IR R@5	–	–	64.41	67.28	67.29	**68.20**	67.68	66.71	67.70	64.63	66.96	67.47	62.47	58.37
Multimodal Avg.	–	–	66.42	66.25	67.11	68.11	69.11	69.05	**70.61**	67.32	69.23	69.92	62.02	67.29
Macro Avg.	28.73	35.77	69.55	71.97	73.64	73.91	76.79	77.24	77.67	76.92	78.03	**78.82**	61.28	66.54

Table C.1. **Comparing our full FLAVA pretraining with other settings** (similar to Table 4 in the main paper) with additional ablations (see Sec. C for details). The overall best result is underlined while **bold** signifies the best on public data (PMD and unimodal).

involve classifying an image using the text written on the image pixels, and require the model to perform OCR to read text from images. For example, in the SST task in Table C.1 (which is also evaluated as an image classification task in [7]), the model is asked to classify the sentiment of a natural language sentence by printing the sentence words onto an image and providing the image pixels to the model. It can be seen from Table C.1 that our FLAVA model does not perform well on this SST task, which we believe is mostly because our PMD dataset does not contain enough scene text information for the model to acquire text reading ability from images. We note that the CLIP model pretrained on PMD (column 13) has a similar lower performance on SST than the variant pretrained on 400M image-text pairs (column 14), and we anticipate that FLAVA will also be able to perform scene text reading when pretrained on a larger dataset with enough scene text information.

Our FLAVA model reaches a final accuracy of 72.49 on the VQAv2 dataset. While this accuracy is below the state-of-the-art on VQAv2, we note that this is a reasonable performance given the amount of data used in FLAVA pretraining. Recent models such as SimVLM [12] often use a much larger dataset (*e.g*. 1.8B image-text pairs [12]), and we believe more pretraining data will also benefit FLAVA.

D. Architectural differences between FLAVA and CLIP encoders

method	Vision Avg.	NLP Avg.	Multi-modal Avg.	Macro Avg.
1 CLIP (PMD)	76.12	46.44	62.02	61.52
2 arch optimizations	76.12	62.99	66.42	68.51
Δ	+0.00	+16.55	+4.40	+6.99

Table D.1. Comparing our FLAVA image and text encoders to the original CLIP when trained under same settings on PMD.

FLAVA and CLIP [7] use transformers [11] as the image and text encoders in their comparable variations (column 3, FLAVA$_C$-local contrastive and column 13, CLIP-ViT-B/16 in Table C.1). Compared to CLIP which uses a text vocabulary of size 49152, in FLAVA we use BERT's text vocabulary with a size of 30522. CLIP uses lower-cased byte pair encoding similar to [8,10] whereas we use BERT's tokenizer from [13] to tokenize the text. Furthermore, we use a hidden size of 768 instead of 512 and use the ViT architecture (based on the implementation in Hugging Face [13]) instead of the GPT-style transformer architecture in CLIP for both text and image encoders [14]. Table D.1 shows the comparison of macro averages for the three domains between the original CLIP architecture and our optimized FLAVA architecture trained on PMD under the same settings with local contrastive loss (corresponding to columns 13 and 3 in Table C.1, respectively). A comparison between rows 1 and 2 in Table D.1 shows that our architecture optimizations help achieve a better macro average overall.

References

[1] Hangbo Bao, Li Dong, and Furu Wei. BEiT: BERT pre-training of image transformers. *CoRR*, abs/2106.08254, 2021. 1

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021. 1

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, 2019. 1

[4] Gabriel Ilharco, Mitchell Wortsman, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. 2

[5] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 1, 2

[6] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. 2

[7] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021. 1, 2, 3, 4

[8] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *OpenAI Blog. https://openai.com/blog/language-unsupervised/*, 2018. 4

[9] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 2015. 2

[10] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*, 2015. 4

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of NeurIPS*, 2017. 4

[12] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. *CoRR*, abs/2108.10904, 2021. 3, 4

[13] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface's transformers: State-of-the-art natural language processing, 2020. 4

[14] Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *Proceedings of ICML*, pages 10524–10533. PMLR, 2020. 4