Fine-tuning (deep learning)

From Wikipedia, the free encyclopedia

In deep learning, **fine-tuning** is an approach to transfer learning in which the parameters of a pre-trained neural network model are trained on new data.[1] Fine-tuning can be done on the entire neural network, or on only a subset of its layers, in which case the layers that are not being fine-tuned are "frozen" (i.e., not changed during backpropagation).[2] A model may also be augmented with "adapters" that consist of far fewer parameters than the original model, and fine-tuned in a parameter-efficient way by tuning the weights of the adapters and leaving the rest of the model's weights frozen.[3]

For some architectures, such as convolutional neural networks, it is common to keep the earlier layers (those closest to the input layer) frozen, as they capture lower-level features, while later layers often discern high-level features that can be more related to the task that the model is trained on.[2][4]

Models that are pre-trained on large, general corpora are usually fine-tuned by reusing their parameters as a starting point and adding a task-specific layer trained from scratch.[5] Fine-tuning the full model is also common and often yields better results, but is more computationally expensive.[6]

Fine-tuning is typically accomplished via supervised learning, but there are also techniques to fine-tune a model using weak supervision.[7] Fine-tuning can be combined with a reinforcement learning from human feedback-based objective to produce language models such as ChatGPT (a fine-tuned version of GPT models) and Sparrow.[8][9]

Robustness [edit]

Fine-tuning can degrade a model's robustness to distribution shifts.[10][11] One mitigation is to linearly interpolate a fine-tuned model's weights with the weights of the original model, which can greatly increase out-of-distribution performance while largely retaining the in-distribution performance of the fine-tuned model.[12]

Variants [edit]

Low-rank adaptation [edit]

Low-rank adaptation (LoRA) is an adapter-based technique for efficiently fine-tuning models. The basic idea is to design a low-rank matrix that is then added to the original matrix.[13] An adapter, in this context, is a collection of low-rank matrices which, when added to a base model, produces a fine-tuned model. It allows for performance that approaches full-model fine-tuning with lower space requirements. A language model with billions of parameters may be LoRA fine-tuned with only several millions of parameters.

LoRA-based fine-tuning has become popular in the Stable Diffusion community.[14] Support for LoRA was integrated into the diffusers library from Hugging Face.[15] Support for LoRA and similar techniques is also available for a wide range of other models through Hugging Face's *parameter-efficient fine-tuning (PEFT)* package.[16]

Representation fine-tuning [edit]

This section **relies largely or entirely upon a single source**. Relevant discussion may be found on the talk page. Please help improve this article by introducing citations to additional sources at this section.
Find sources: "Representation fine-tuning (ReFT)" – news · newspapers · books · scholar · JSTOR *(May 2024) (Learn how and when to remove this message)*

Representation fine-tuning (ReFT) is a technique developed by researchers at Stanford University aimed at fine-tuning large language models (LLMs) by modifying less than 1% of their representations. Unlike parameter-efficient fine-tuning (PEFT) methods, which mainly focus on updating weights, ReFT targets representations, suggesting that modifying representations might be a more effective strategy than updating weights.[17]

ReFT methods operate on a frozen base model and learn task-specific interventions on hidden representations and train interventions that manipulate a small fraction of model representations to steer model behaviors towards solving downstream tasks at inference time. One specific method within the ReFT family is *low-rank linear subspace ReFT (LoReFT)*, which intervenes on hidden representations in the linear subspace spanned by a low-rank projection matrix.[17] LoReFT can be seen as the representation-based equivalent of low-rank adaptation (LoRA).

Applications [edit]

Natural language processing [edit]

Fine-tuning is common in natural language processing (NLP), especially in the domain of language modeling. Large language models like OpenAI's series of GPT foundation models can be fine-tuned on data for specific downstream NLP tasks (tasks that use a pre-trained model) to improve performance over the unmodified pre-trained model.[6]

Commercial models [edit]

Commercially-offered large language models can sometimes be fine-tuned if the provider offers a fine-tuning API. As of June 19, 2023, language model fine-tuning APIs are offered by OpenAI and Microsoft Azure's

Azure OpenAI Service for a subset of their models, as well as by Google Cloud Platform for some of their PaLM models, and by others.[18][19][20]

See also [edit]

- Catastrophic forgetting
- Continual learning
- Domain adaptation
- Foundation model
- Hyperparameter optimization
- Overfitting

References [edit]

1. ^ Quinn, Joanne (2020). *Dive into deep learning: tools for engagement*. Thousand Oaks, California. p. 551. ISBN 978-1-5443-6137-6. Archived from the original on January 10, 2023. Retrieved January 10, 2023.
2. ^ *a* *b* "CS231n Convolutional Neural Networks for Visual Recognition". *cs231n.github.io*. Retrieved 9 March 2023.
3. ^ Liu, Haokun; Tam, Derek; Muqeeth, Mohammed; Mohta, Jay; Huang, Tenghao; Bansal, Mohit; Raffel, Colin A (2022). Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; Oh, A. (eds.). *Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning* (PDF). Advances in Neural Information Processing Systems. Vol. 35. Curran Associates, Inc. pp. 1950–1965.
4. ^ Zeiler, Matthew D; Fergus, Rob (2013). "Visualizing and Understanding Convolutional Networks". *ECCV*. arXiv:1311.2901.
5. ^ Dodge, Jesse; Ilharco, Gabriel; Schwartz, Roy; Farhadi, Ali; Hajishirzi, Hannaneh; Smith, Noah (2020). "Fine-Tuning Pretrained Language Models: Weight Initializations, Data Orders, and Early Stopping". arXiv:2002.06305. {{cite journal}}: Cite journal requires |journal= (help)
6. ^ *a* *b* Dingliwal, Saket; Shenoy, Ashish; Bodapati, Sravan; Gandhe, Ankur; Gadde, Ravi Teja; Kirchhoff, Katrin (2021). "Prompt Tuning GPT-2 language model for parameter-efficient domain adaptation of ASR systems". *InterSpeech*. arXiv:2112.08718.
7. ^ Yu, Yue; Zuo, Simiao; Jiang, Haoming; Ren, Wendi; Zhao, Tuo; Zhang, Chao (2020). "Fine-Tuning Pre-trained Language Model with Weak Supervision: A Contrastive-Regularized Self-Training Approach". *Association for Computational Linguistics*. arXiv:2010.07835.
8. ^ "Introducing ChatGPT". *openai.com*. Retrieved 9 March 2023.
9. ^ Glaese, Amelia; McAleese, Nat; Trębacz, Maja; Aslanides, John; Firoiu, Vlad; Ewalds, Timo; Rauh, Maribeth; Weidinger, Laura; Chadwick, Martin; Thacker, Phoebe; Campbell-Gillingham, Lucy; Uesato, Jonathan; Huang, Po-Sen; Comanescu, Ramona; Yang, Fan; See, Abigail; Dathathri, Sumanth; Greig, Rory; Chen, Charlie; Fritz, Doug; Elias, Jaume Sanchez; Green, Richard; Mokrá, Soňa; Fernando, Nicholas; Wu, Boxi; Foley, Rachel; Young, Susannah; Gabriel, Iason; Isaac, William; Mellor, John; Hassabis, Demis; Kavukcuoglu, Koray; Hendricks, Lisa Anne; Irving, Geoffrey (2022). "Improving alignment of dialogue agents via targeted human judgements". *DeepMind*. arXiv:2209.14375.
10. ^ Radford, Alec; Kim, Jong Wook; Hallacy, Chris; Ramesh, Aditya; Goh, Gabriel; Agarwal, Sandhini; Sastry, Girish; Askell, Amanda; Mishkin, Pamela; Clark, Jack; Krueger, Gretchen; Sutskever, Ilya (2021). "Learning Transferable Visual Models From Natural Language Supervision". arXiv:2103.00020 [cs.CV].

11. ^ Kumar, Ananya; Raghunathan, Aditi; Jones, Robbie; Ma, Tengyu; Liang, Percy (2022). "Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution". *ICLR*. arXiv:2202.10054.
12. ^ Wortsman, Mitchell; Ilharco, Gabriel; Kim, Jong Wook; Li, Mike; Kornblith, Simon; Roelofs, Rebecca; Gontijo-Lopes, Raphael; Hajishirzi, Hannaneh; Farhadi, Ali; Namkoong, Hongseok; Schmidt, Ludwig (2022). "Robust fine-tuning of zero-shot models". arXiv:2109.01903 [cs.CV].
13. ^ Hu, Edward J.; Shen, Yelong; Wallis, Phillip; Allen-Zhu, Zeyuan; Li, Yuanzhi; Wang, Shean; Wang, Lu; Chen, Weizhu (2022-01-28). "LoRA: Low-Rank Adaptation of Large Language Models". *ICLR*. arXiv:2106.09685.
14. ^ Ryu, Simo (February 13, 2023). "Using Low-rank adaptation to quickly fine-tune diffusion models". *GitHub*. Retrieved June 19, 2023.
15. ^ Cuenca, Pedro; Paul, Sayak (January 26, 2023). "Using LoRA for Efficient Stable Diffusion Fine-Tuning". *Hugging Face*. Retrieved June 19, 2023.
16. ^ "Parameter-Efficient Fine-Tuning using 🤗 PEFT". *huggingface.co*. Retrieved 2023-06-20.
17. ^ *a* *b* Wu, Zhengxuan; Arora, Aryaman; Wang, Zheng; Geiger, Atticus; Jurafsky, Dan; Manning, Christopher D.; Potts, Christopher (2024-04-07), *ReFT: Representation Finetuning for Language Models*, arXiv:2404.03592
18. ^ "Fine-tuning". OpenAI. Retrieved 2023-06-19.
19. ^ "Learn how to customize a model for your application". Microsoft. Retrieved 2023-06-19.
20. ^ "Tune text foundation models". Retrieved 2023-06-19.

V · T · E

Generative AI [hide]

Concepts		Autoencoder · Deep learning · **Fine-tuning** · Foundation model · Generative adversarial network · Generative pre-trained transformer · Large language model · Model Context Protocol · Neural network · Prompt engineering · Reinforcement learning from human feedback · Retrieval-augmented generation · Self-supervised learning · Stochastic parrot · Synthetic data · Top-p sampling · Transformer · Variational autoencoder · Vibe coding · Vision transformer · Waluigi effect · Word embedding
Models	**Text**	Character.ai · ChatGLM · Claude · DBRX · DeepSeek · ERNIE · Gemini · GPT (1 · 2 · 3 · J · ChatGPT · 4 · 4o · o1 · o3 · 4.5 · 4.1 · o4-mini) · Granite · Grok · Hunyuan · Kimi · Llama · Microsoft Copilot · MiniMax-M1 · Mistral Large · PanGu-Σ · Qwen · YandexGPT · Yi-Lightning
	Coding	Claude Code · Cursor · Devstral · GitHub Copilot · Granite Code · Kimi-Dev · Replit · Xcode
	Image	Aurora · Firefly · Flux · GPT Image 1 · Ideogram · Imagen · Leonardo · Midjourney · Recraft · Seedream · Stable Diffusion
	Video	Dream Machine · Runway Gen · Hailuo AI · Kling · Midjourney Video · Seedance · Sora · Veo · Wan
	Speech	15.ai · ElevenLabs · Speech-02 · WaveNet
	Music	Endel · Lyria · Riffusion · Suno AI · Udio
Agents		AutoGLM · AutoGPT · Devin AI · Manus · OpenAI Codex · Operator
Companies		01.AI · Aleph Alpha · Alibaba · Anthropic · Baichuan · Baidu · ByteDance · Canva · Cognition AI · Cohere · Contextual AI · DeepL · DeepSeek · ElevenLabs · Google DeepMind · HeyGen · Hugging Face · Krikey AI · Kuaishou · Meta AI · MiniMax · Mistral AI · Moonshot AI · OpenAI · Runway · Safe Superintelligence · Scale AI · Stability AI · Synthesia · Tencent · Thinking Machines Lab · xAI · Yandex · Zhipu AI

Category · **Commons**

Categories: Machine learning | Deep learning

This page was last edited on 30 May 2025, at 15:08 (UTC).

Text is available under the Creative Commons Attribution-ShareAlike 4.0 License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.