Tesi di Laurea

Titolo della Tesi

Autore: Antonio Sessa

Relatore: Nome Primo Relatore

Relatore: Nome Secondo Relatore

2025

Contents

1 Literature Review

1.1 Vision Transformers

1.2 Vision Language Models

1.3 Parameter-Efficient Fine-Tuning through LoRA

Low-rank adaptation (LoRA) [1] is a technique first introduced to fine-tune Large Language Models (LLMs) that has also shown successful results in computer vision tasks [2] and image and video generation tasks. The core hypothesis of LoRA is that weight updates during the adaptation of a large pre-trained model to a new task have a low "intrinsic rank". This can be mathematically described as such: given a pre-trained weight matrix $W_o \in \mathbb{R}^{d \times k}$, the weight update matrix ΔW can be approximated as $\Delta W = BA$ where $B \in \mathbb{R}^{d \times r}$ and $A \in \mathbb{R}^{r \times k}$ with the rank $r << min(d, k)$. Given this hypothesis, during training, W_o can be frozen and not receive any gradient updateds, and we can reformulate the forward pass as such: $h = W_o x + BAx$

```
# dense_pt, a pre-trained nn.Linear module
dense_pt.requires_grad = False
k = dense_pt.in_features
d = dense_pt.out_features
rank = 64 # rank << min(k, d)
lora_A = nn.Parameter(torch.zeros(rank, k))
lora_B = nn.Parameter(torch.zeros(d, rank))
nn.init.normal_(self.lora_A, mean=0.0, std = (1 / rank))

def forward_lora(x, lora_A, lora_B, dense_pt):
    # original model output
    pt_model_output = dense_pt(x)

    # the matrix product of lora_B @ lora_A results in
    # a [d,r] @ [r,k] = [d,k] shaped matrix
    # that is of equal shape of the un-approximesed weight update
    lora_output = lora_B @ lora_A @ x
```

```
18
19      return F.ReLU(pt_model_output + lora_output)
```

Listing 1: LoRA pytorch code snippet

This approach has several significant benefits, making it a highly efficient and practical method for adapting large models:

- **Reduced Number of Trainable Parameters:** By freezing the large pre-trained weight matrix W_o and only optimizing the low-rank matrices A and B, LoRA drastically cuts down the number of parameters that need to be updated during training. This makes the fine-tuning process significantly less computationally intensive.
- **Lower VRAM Consumption:** The reduction in trainable parameters directly leads to a smaller memory footprint. Since gradients and optimizer states are only stored for the low-rank matrices, the overall VRAM requirement is substantially lower, enabling the fine-tuning of large models on hardware with limited memory.
- **Smaller Checkpoint Size:** Instead of saving a full copy of the fine-tuned model, only the small matrices A and B need to be stored for each task. This results in highly portable and lightweight checkpoints that are orders of magnitude smaller than the original model.
- **No Added Inference Latency:** After training, the weight update can be merged directly into the original weights by computing $W = W_o + BA$. This means the model architecture remains unchanged during inference, and there is no additional computational overhead or latency compared to the original pre-trained model.

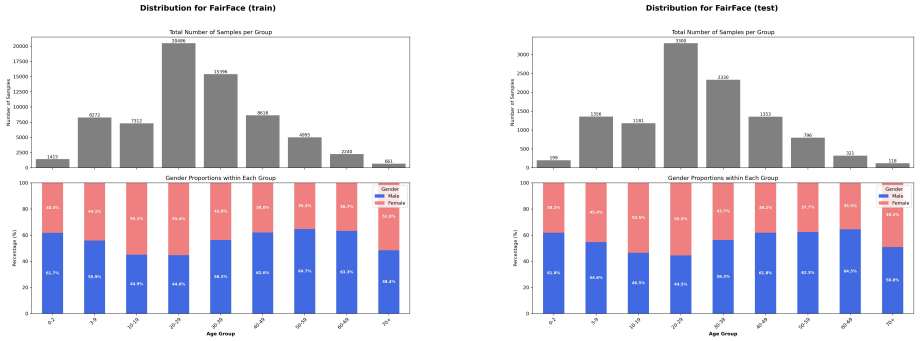
Moreover, LoRA can perform just as well as full fine-tuning in some cases [1, 2], but as task complexity increases, full fine-tune may still outperform LoRA considerably [3]. The success of this methodology has inspired many other studies on parameter-efficient adaptation trough low-rank decomposition [4]: Weight-Decomposed Low-Rank Adaptation (DoRA) [5] enchances LoRA by decomposing the W_o weight matrix in its magnitude vector $m \in \mathbb{R}^{1\times k}$ and its direction matrix $V \in \mathbb{R}^{d\times k}$, and directly trains the magnitude vector and uses LoRA to train the direction matrix; QLoRA [6] focuses on drastic reduction of VRAM requirements while maintaining performance through quantization; LoRA+ [7] proposes to set an higher learning rate to the B matrices, to more optimally fine-tune models with larger embedding dimension. Furthermore, solutions like mLoRA [8] have been developed to efficiently train numerous adapters in parallel by leveraging a single shared base model. Subsequently, for inference, systems such as S-LoRA [9] and B-LoRA [10] can serve multiple adapters concurrently, batching requests for different tasks to transform the single large model into an efficient multi-task network.

1.4 Datasets Review

In the following paragraphs we list all the datasets used for the training, validation and testing of our models. Each dataset has been moreover processed to obtain the crop of the faces[1], using the DNN available in the following repository, or using the already provided bounding boxes by the authors. For each training set listed below, also a validation set will be extracted doing an 80-20 split.

1.4.1 FairFace

The FairFace dataset [11] contains 108,501 images, with an emphasis on balanced etnicity composition. The faces are labelled for gender and age groups. It is from the FairFace dataset that we take the 9 age groups for our age classification tasks. In our experiments the FairFace dataset is used both for training and testing, using the split provided by the authors.



(a) Train Set Distribution

(b) Test Set Distribution

Figure 1: FairFace Dataset Distribution

1.4.2 UTKFace

The UTKFace dataset [12] contains 24,103 images, labelled with ages and gender. The age range spans from 0 to 116 years old. In our experiments the UTKFace dataset will be used only for testing purposes, providing a benchmark for the model's cross-dataset generalization capabilities.

[1] It is not obvious that for our tasks taking the crop of only faces is optimal, but is necessary due to the varied nature of images, which include both full-body and close-up. This preprocessing step ensures a uniform input for the models.

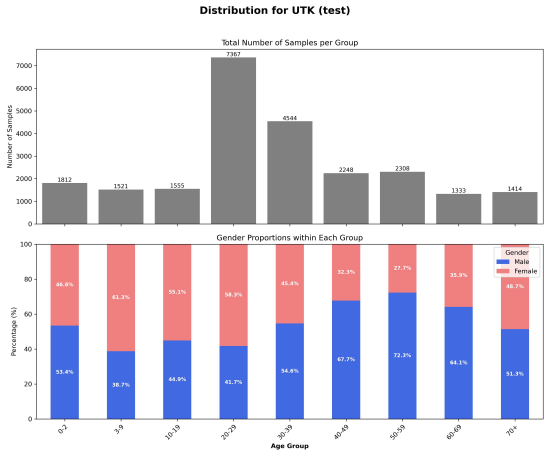


Figure 2: UTKFace Dataset Distribution (Test Set)

1.4.3 Lagenda

The Lagenda dataset [13] [14] provides 67,159 images, labelled with age and gender. The age range spans from 0 to 95 years, and is fairly balanced on the age range going from 3 to 69 years old. In our experiment, the Lagenda dataset will be used only for training purposes.

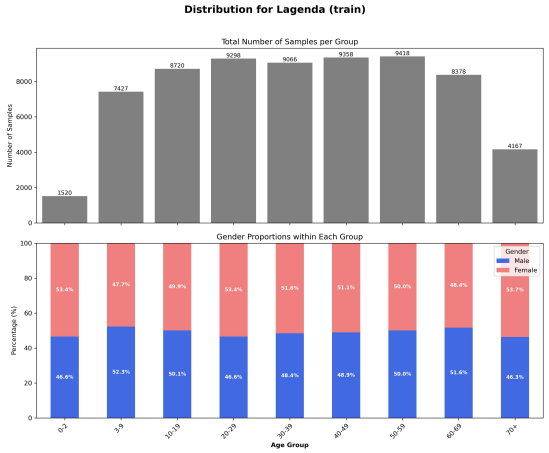
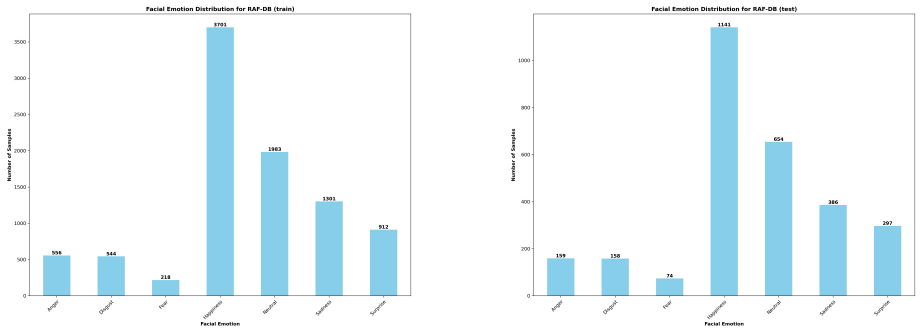


Figure 3: UTKFace Dataset Distribution (Test Set)

1.4.4 RAF-DB

The RAF-DB dataset [15] provides 14,388 images, labelled with gender and facial emotion. The facial emotion classes are the following: "Surprise", "Fear",

"Disgust", "Happy", "Sad", "Angry", and "Neutral". The RAF-DB dataset will be used both for training and testing, (come abbiamo lo split?). The samples labeled for emotion are by far the fewest and are also unbalanced; in chapter (capitolo in cui spieghiamo sampling) we explain how we tackle this problem.



(a) Train Set Distribution

(b) Test Set Distribution

Figure 4: RAF-DB Dataset Distribution

1.4.5 VggFace2

The VggFace2 [16] dataset contains 3.31 million images of 9131 subjects, labelled by gender. Moreover, each samples has been also labelled with age using (modello usato per la label di età). In our experiments the VggFace2 dataset will be used only for testing, providing a well enstablished benchmark dataset for our tasks. The choice to exclude the VggFace2 for training is to avoid to bring an heavy bias on picture of celebreties and limit the training time required. (menzoniamo il fatto che gli esperimenti fatti con vggface2 non sembrano portare a buoni risultati?)

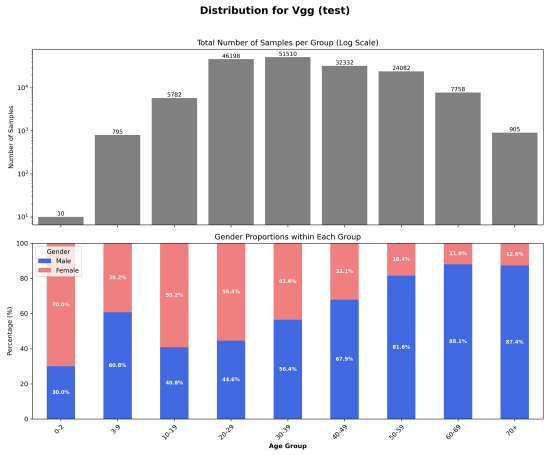
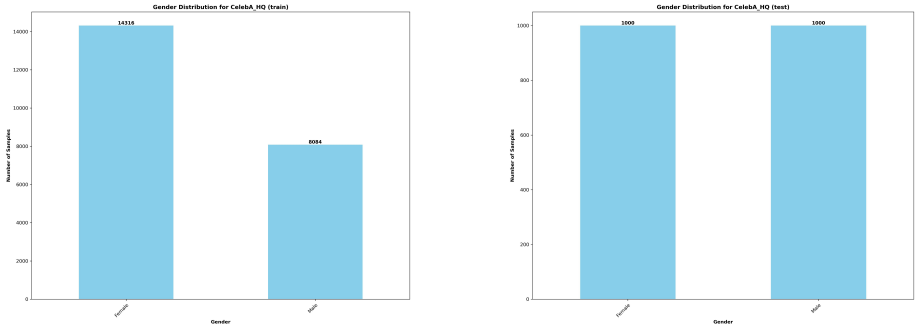


Figure 5: UTKFace Dataset Distribution (Test Set)

1.4.6 CelebA-HQ

The CelebaHQ dataset [17] [18] provides 30,000 high-quality images, labelled by gender. In our experiments the CelebaHQ dataset will be used both for training and testing.



(a) Train Set Distribution

(b) Test Set Distribution

Figure 6: FairFace Dataset Distribution

List of Figures

References

[1] Edward J. Hu et al. *LoRA: Low-Rank Adaptation of Large Language Models*. 2021. arXiv: 2106.09685 [cs.CL]. URL: https://arxiv.org/abs/2106.09685.

[2] Xuehai He et al. "Parameter-Efficient Model Adaptation for Vision Transformers". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 37.1 (June 2023), pp. 817–825. DOI: 10.1609/aaai.v37i1.25160. URL: https://ojs.aaai.org/index.php/AAAI/article/view/25160.

[3] Dan Biderman et al. *LoRA Learns Less and Forgets Less*. 2024. arXiv: 2405.09673 [cs.LG]. URL: https://arxiv.org/abs/2405.09673.

[4] Lingling Xu et al. *Parameter-Efficient Fine-Tuning Methods for Pretrained Language Models: A Critical Review and Assessment*. 2023. arXiv: 2312.12148 [cs.CL]. URL: https://arxiv.org/abs/2312.12148.

[5] Shih-Yang Liu et al. *DoRA: Weight-Decomposed Low-Rank Adaptation*. 2024. arXiv: 2402.09353 [cs.CL]. URL: https://arxiv.org/abs/2402.09353.

[6] Tim Dettmers et al. *QLoRA: Efficient Finetuning of Quantized LLMs*. 2023. arXiv: 2305.14314 [cs.LG]. URL: https://arxiv.org/abs/2305.14314.

[7] Soufiane Hayou, Nikhil Ghosh, and Bin Yu. *LoRA+: Efficient Low Rank Adaptation of Large Models*. 2024. arXiv: 2402.12354 [cs.LG]. URL: https://arxiv.org/abs/2402.12354.

[8] Zhengmao Ye et al. *mLoRA: Fine-Tuning LoRA Adapters via Highly-Efficient Pipeline Parallelism in Multiple GPUs*. 2024. arXiv: 2312.02515 [cs.LG]. URL: https://arxiv.org/abs/2312.02515.

[9] Ying Sheng et al. "S-LoRA: Serving Thousands of Concurrent LoRA Adapters". In: *arXiv preprint arXiv:2311.03285* (2023).

[10] Ali Sabet. *BLoRA: Maximize GPU util by routing inference through multiple LoRAs in same batch*. GitHub repository. 2024. URL: https://github.com/sabetAI/BLoRA.

[11] Kimmo Kärkkäinen and Jungseock Joo. "FairFace: Face Attribute Dataset for Balanced Race, Gender, and Age". In: *CoRR* abs/1908.04913 (2019). arXiv: 1908.04913. URL: http://arxiv.org/abs/1908.04913.

[12] Zhifei Zhang, Yang Song, and Hairong Qi. *Age Progression/Regression by Conditional Adversarial Autoencoder*. 2017. arXiv: 1702.08423 [cs.CV]. URL: https://arxiv.org/abs/1702.08423.

[13] Maksim Kuprashevich and Irina Tolstykh. "MiVOLO: Multi-input Transformer for Age and Gender Estimation". In: (2023). eprint: arXiv:2307.04616.

[14] Maksim Kuprashevich, Grigorii Alekseenko, and Irina Tolstykh. "Beyond Specialization: Assessing the Capabilities of MLLMs in Age and Gender Estimation". In: (2024). eprint: arXiv:2403.02302.

[15] Shan Li, Weihong Deng, and Junping Du. "Reliable Crowdsourcing and Deep Locality-Preserving Learning for Expression Recognition in the Wild". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), pp. 2584–2593. URL: https://api.semanticscholar.org/CorpusID:11413183.

[16] Qiong Cao et al. *VGGFace2: A dataset for recognising faces across pose and age*. 2018. arXiv: 1710.08092 [cs.CV]. URL: https://arxiv.org/abs/1710.08092.

[17] Weihao Xia et al. "TediGAN: Text-Guided Diverse Face Image Generation and Manipulation". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021.

[18] Weihao Xia et al. "Towards Open-World Text-Guided Face Image Generation and Manipulation". In: *arxiv preprint arxiv: 2104.08910* (2021).