# LoRA

## Why need LoRA?

* Cập nhập mô hình deep learning.
* Cập nhập fine-tuning, cập nhập cả cái W 🡪 chi phí rất lớn!!.
* Các checkpoint, cần phải lưu lại, cũng expensive, lưu trữ rất lớn.

A close-up of a computer program

Description automatically generated

* Cái ý thứ 3 là mình fine tune nhiều model cùng lúc, xong mỗi lần phải load cả cái W to mỗi lần thì rất chậm + chi phí.

## Better solution LoRA

* Đóng băng pretrained-model.
* Tạo hai model nhỏ (B,A) One for each of the metrics we want to train?
* Chỗ này (4:38) chưa hiểu ý lắm, tức là không cần phải finetune hết tất cả metrics gì đó? Phải chờ khúc sau mới giải thích thêm
* A diagram of a frozen method

  Description automatically generated with medium confidence
* Tất nhiên giảm params, nhưng vẫn còn hoài nghi về việc, giảm như vậy liệu có mất thông tin? Với cả chọn r, thì r ở đây có phải là việc metrics ở trên hay không?
* Nhưng W trên, một số weights cũng không có ý nghĩa? 🡪 redundant.
* Combine. A diagram of a frozen process

  Description automatically generated
* Cộng hai W, và BxA, sau đó ra output, so sánh với target, ta tính loss, và chỉ cập nhập trên B,A này. Đó là ý tưởng.

## Benefits

* Ít params hơn!.
* Vì vậy lưu trữ nhỏ hơn.
* Faster backpropagation.
* Easily switch
* A black text on a white background

  Description automatically generated

## Why it works?

* Toán học, sử dụng kỹ thuật giảm chiều SVD.
* Giải thích trong file ipynb..

## Code LoRA from scratch

* Từ ma trận W, dxK, gọi là feature\_in x feature\_out.
* Ban đầu khởi tạo, A là theo phân phối chuẩn, B là 0.
* Thì \deltaW = BA là ma trận 0 ban đầu.
* Tức W = W+BA = W.
* À, tức là cộng cả cái model gốc, với cái BA này tính như thường, free cái trên, và finetune chạy bình thường (vì vẫn tính đạo hàm theo nhánh dưới bình thường).
* Tức tổng tham số sẽ tăng, tuy nhiên, khi trained, chỉ train nhánh dưới.