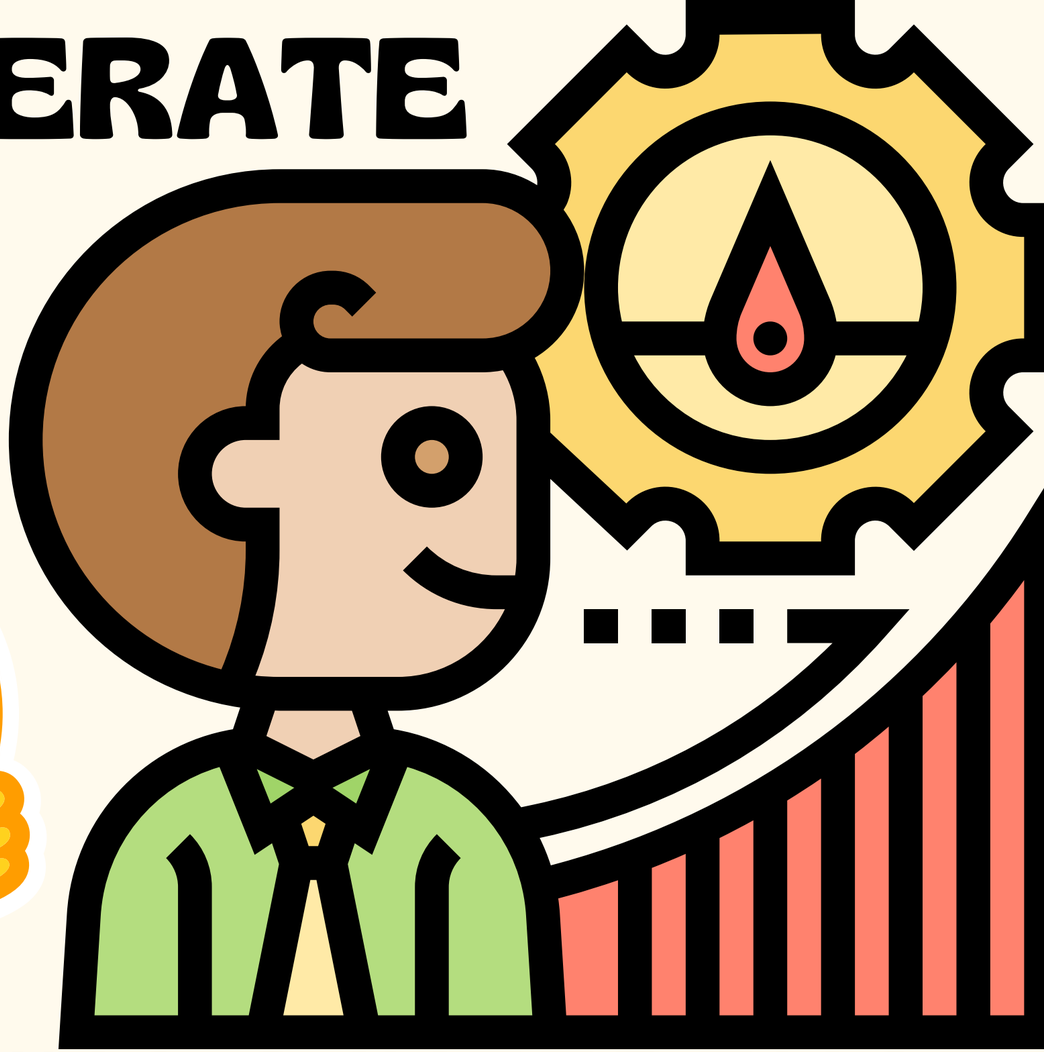
INTRODUCING ACCELERATE & PEFT TO DEMOCRATIZE LLM



TRAINING & INFERENCE OF BIG LLM WITH LESS HARDWARE RESOURCE

HTTPS://GITHUB.COM/INSIGHTBUILDER

CHALLENGE SOLVED: INFERENCE WITH LESS

- WHAT HAPPENS WHEN THE MODEL WAS LOADED IN PAST
  - CREATE THE MODEL
  - LOAD IN MEMORY ITS WEIGHTS (IN AN OBJECT USUALLY CALLED STATE_DICT)
  - LOAD THOSE WEIGHTS IN THE CREATED MODEL
  - MOVE THE MODEL ON THE DEVICE FOR INFERENCE
- WHAT HAPPENS WHEN THE MODEL IS LOADED NOW?
  I. CREATE AN EMPTY (E.G. WITHOUT WEIGHTS) MODEL (INIT EMPTY WEIGHTS)
  II. DECIDE WHERE EACH LAYER IS GOING TO GO (WHEN MULTIPLE DEVICES ARE AVAILABLE)
  III. LOAD IN MEMORY PARTS OF ITS WEIGHTS (LOAD CHECKPOINT & DISPATCH)
  IV. LOAD THOSE WEIGHTS IN THE EMPTY MODEL
  V. MOVE THE WEIGHTS ON THE DEVICE FOR INFERENCE
  VI. REPEAT FROM STEP 3 FOR THE NEXT WEIGHTS UNTIL ALL THE WEIGHTS ARE LOADED

DEVICE MAP :

- AUTO
- BALANCED
- BALANCED-LOW-0
- SEQUENTIAL

- UNDERSTANDING TRAINING LLM IN DETAILS, THE PARTS & ALL
- DIVING INTO VARIOUS MODEL CLASSES IN TRANSFORMER LIBRARY

HTTPS://YOUTU.BE/MWCSGJ9JEA0

HTTPS://GITHUB.COM/INSIGHTBUILDER

SOLUTION : ACCELERATE

- MODELS, MODEL WEIGHTS, LOCATION OF THE WEIGHTS, HOW THEY ARE ACCESSED DURING INFERENCE NEEDS TO BE UNDERSTOOD
- MODELS CAN BE LOADED IN PARTS & THEY CAN BE LOADED INTO CPU, RAM AND DISK. THE WAY IT GETS LOADED CAN BE CONTROLLED USING DEVICE MAP
- MODELS CAN BE SHARDED INTO SMALLER CHUNKS. THE SIZES OF THESE CHUNKS HAVE TO BE LESS THAN OR EQUAL TO YOUR AVAILABLE RAM
- THE MODEL LAYERS THAT NEEDS TO BE TOGETHER HAS TO BE PLACED IN SAME DEVICE

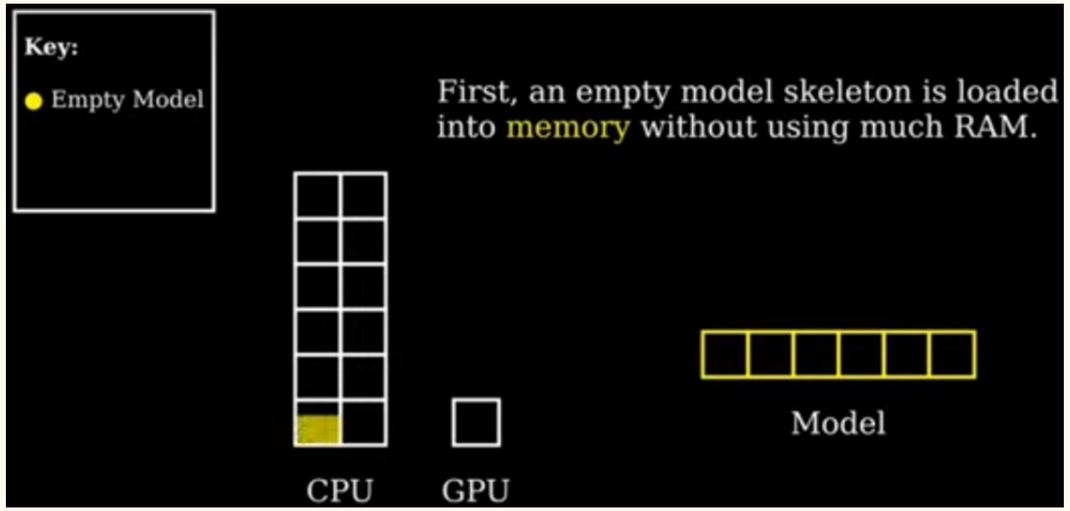
PROCESS ONLY SUPPORTS INFERENCE

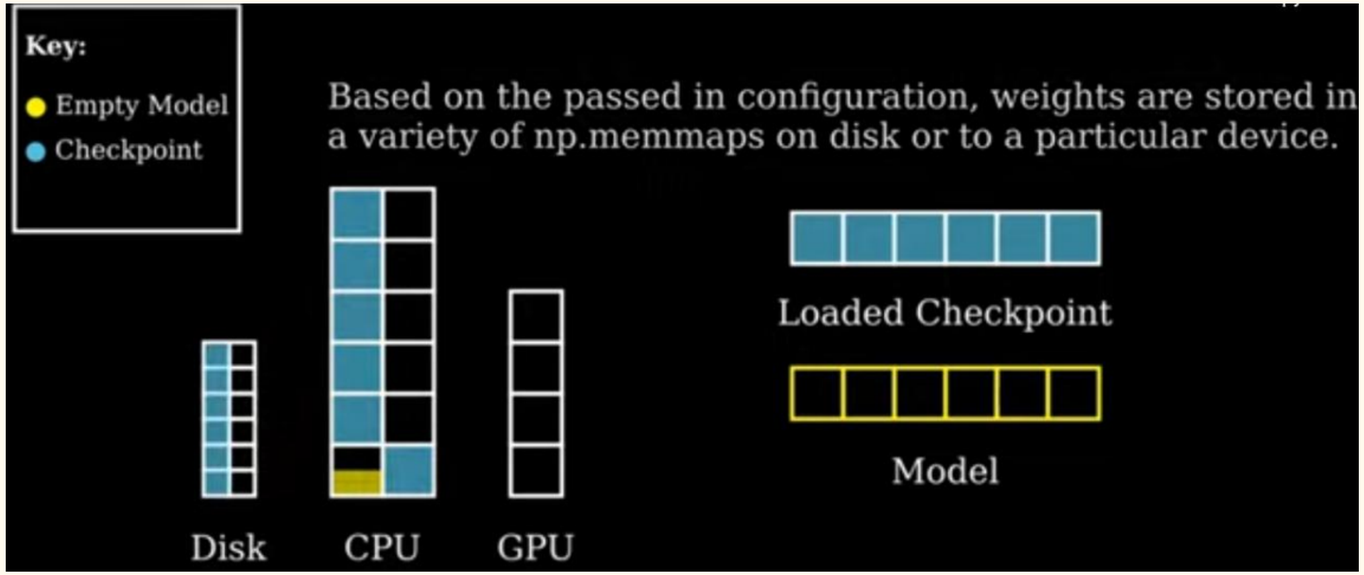
THERE ARE LIMITS

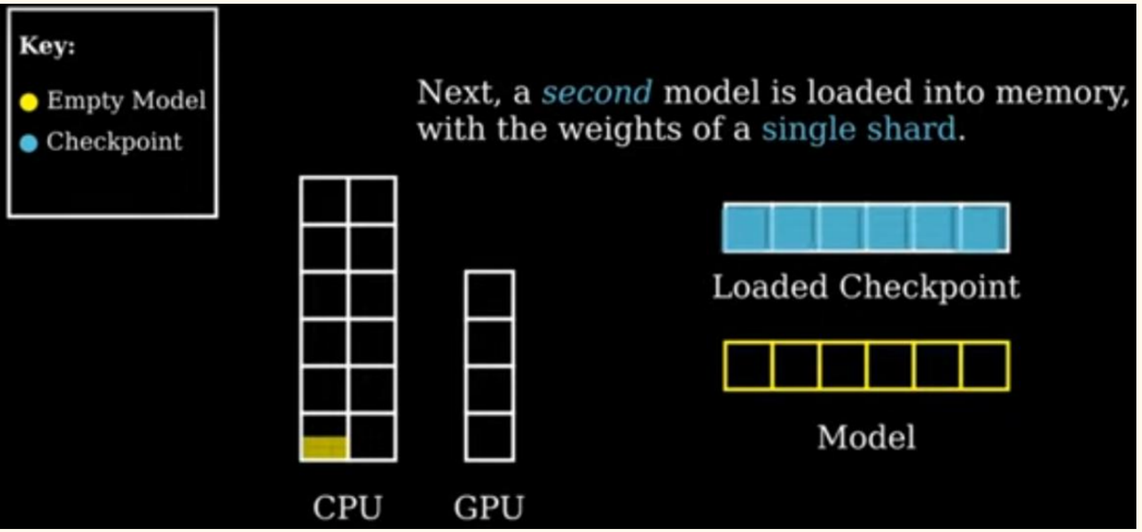
- ATLEAST 1 GPU IS REQUIRED, MEMORY MGMT WITH PYTHON CAN BE DIFFICULT
- DISK OFFLOADING IS SLOW, & MODEL PARALLELISM IS NAIVE WHEN SPLIT ON MULTIPLE GPUS

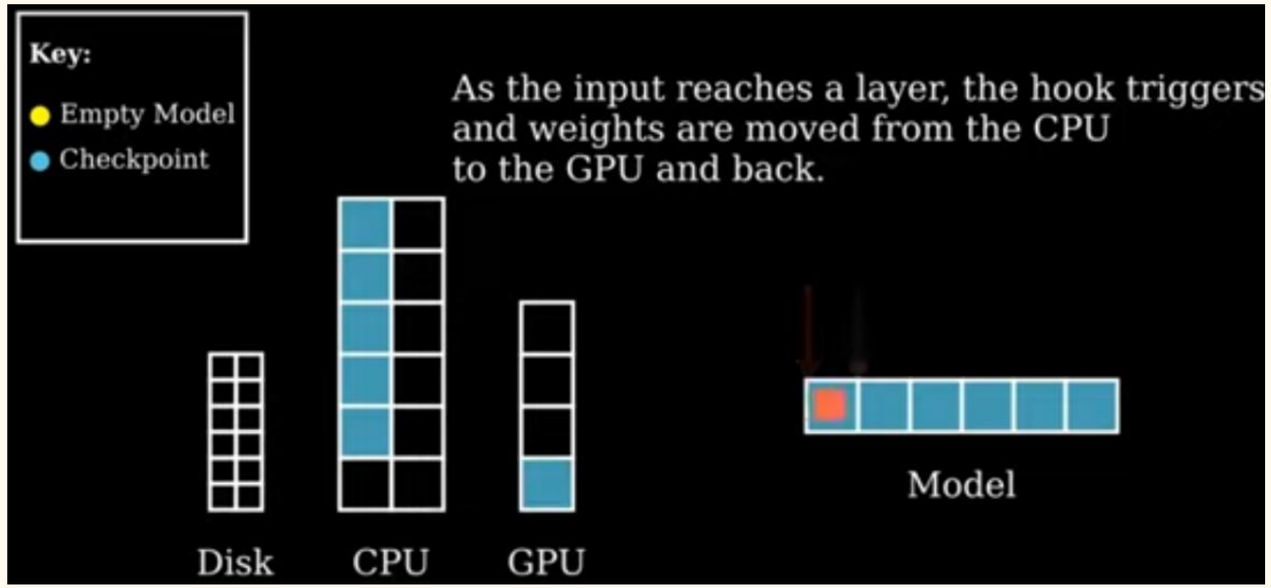
HTTPS://GITHUB.COM/INSIGHTBUILDER

STEPS IN ACCELERATE









HTTPS://GITHUB.COM/INSIGHTBUILDER

CHALLENGE SOLVED: WHAT ABOUT TRAINING

- SOTA TRANSFORMER MODEL LIKE T5, BERT HAVE PARAMETERS THAT RANGE IN BILLIONS, STORING, TRAINING & INFERENCING WITH CONSUMER GRADE HW IS CHALLENGE
- FINE TUNED MODEL ARE STILL THE SAME SIZE SINCE THE NUMBER OF PARAMETERS DON'T CHANGE
- WHEN FULL MODEL IS FINE TUNED, THE NEW CHECKPOINTS ARE EQUAL TO THE SIZE OF THE MODEL ITSELF
- DATA REQUIRED FOR TRAINING THE HUGE MODEL HAS TO BE EXTENSIVE & RICH. IF NOT THERE IS POSSIBILITY OF "CATASTROPHIC FORGETTING"
- DEPLOYING SINGLE MODEL FOR EACH TASK CAN BE PROHIBITIVELY COSTLY FROM COMPUTE AND STORAGE PERSPECTIVE

ALL THESE CAN BE MITIGATED WITH THE PARAMETER EFFICIENT FINE-TUNING. THERE IS A LIBRARY FOR THAT.

HTTPS://GITHUB.COM/INSIGHTBUILDER

SOLUTION: PEFT

- TUNES ONLY LIMITED SET OF PARAMETERS WHILE THE REST ARE FROZEN, LEADING TO
  - LESSER CHECK-POINT SIZE,
  - MITIGATING CATASTROPHIC FORGETTING
  - CONSUME LESSER COMPUTE RESOURCE
- PEFT CHECKPOINTS CAN BE ADDED ON THE EXISTING MODELS AND USED FOR NEW TASKS
- PROCESS INTEGRATES WITH ACCELERATE AND TRANSFORMERS LIBRARY & MANY OF THE SOTA MODELS ARE COMPATIBLE WITH THIS PROCESS
- PEFT CAN BE USED FOR TEXT, VIDEO & AUDIO MODALITIES WITH THE SAME PROCESS

PEFT: TWO METHODS

p

- ## LORA:

LORA'S APPROACH IS TO REPRESENT THE WEIGHT UPDATES WITH TWO SMALLER MATRICES (CALLED UPDATE MATRICES) THROUGH LOW-RANK DECOMPOSITION. THESE NEW MATRICES CAN BE TRAINED TO ADAPT TO THE NEW DATA WHILE KEEPING THE OVERALL NUMBER OF CHANGES LOW. THE ORIGINAL WEIGHT MATRIX REMAINS FROZEN AND DOESN'T RECEIVE ANY FURTHER ADJUSTMENTS. TO PRODUCE THE FINAL RESULTS, BOTH THE ORIGINAL AND THE ADAPTED WEIGHTS ARE COMBINED.

- ## PREFIX TUNING:

PREFIX TUNING IS AN ADDITIVE METHOD WHERE ONLY A SEQUENCE OF CONTINUOUS TASK-SPECIFIC VECTORS IS ATTACHED TO THE BEGINNING OF THE INPUT, OR PREFIX. ONLY THE PREFIX PARAMETERS ARE OPTIMIZED AND ADDED TO THE HIDDEN STATES IN EVERY LAYER OF THE MODEL. THE TOKENS OF THE INPUT SEQUENCE CAN STILL ATTEND TO THE PREFIX AS VIRTUAL TOKENS. AS A RESULT, PREFIX TUNING STORES 1000X FEWER PARAMETERS

- ## PROMPT TUNING

ALSO AN ADDITIVE METHOD FOR ONLY TRAINING AND UPDATING THE NEWLY ADDED PROMPT TOKENS TO A PRETRAINED MODEL. THIS WAY, YOU CAN USE ONE PRETRAINED MODEL WHOSE WEIGHTS ARE FROZEN, AND TRAIN AND UPDATE A SMALLER SET OF PROMPT PARAMETERS FOR EACH DOWNSTREAM TASK INSTEAD OF FULLY FINETUNING A SEPARATE MODEL.

REFERENCES

- HTTPS://HUGGINGFACE.CO/DOCS/ACCELERATE/USAGE_GUIDES/BIG_MODELING
- HTTPS://HUGGINGFACE.CO/DOCS/ACCELERATE/USAGE_GUIDES/TRAINING_ZOO
- HTTPS://HUGGINGFACE.CO/DOCS/ACCELERATE/INDEX
- HTTPS://HUGGINGFACE.CO/BLOG/PEFT
- HTTPS://HUGGINGFACE.CO/DOCS/PEFT/QUICKTOUR
- HTTPS://HUGGINGFACE.CO/DOCS/PEFT/INDEX
- HTTPS://HUGGINGFACE.CO/DOCS/PEFT/MAIN/EN/TASK_GUIDES/SEQ2SEQ-PREFIX-TUNING
- HTTPS://HUGGINGFACE.CO/DOCS/PEFT/MAIN/EN/TASK_GUIDES/CLM-PROMPT-TUNING

THANKS FOR WATCHING

REMEMBER TO PRACTICE WITH EXAMPLES

HTTPS://GITHUB.COM/INSIGHTBUILDER