Compressing Large-Scale Transformer-Based Models: A Case Study on BERT

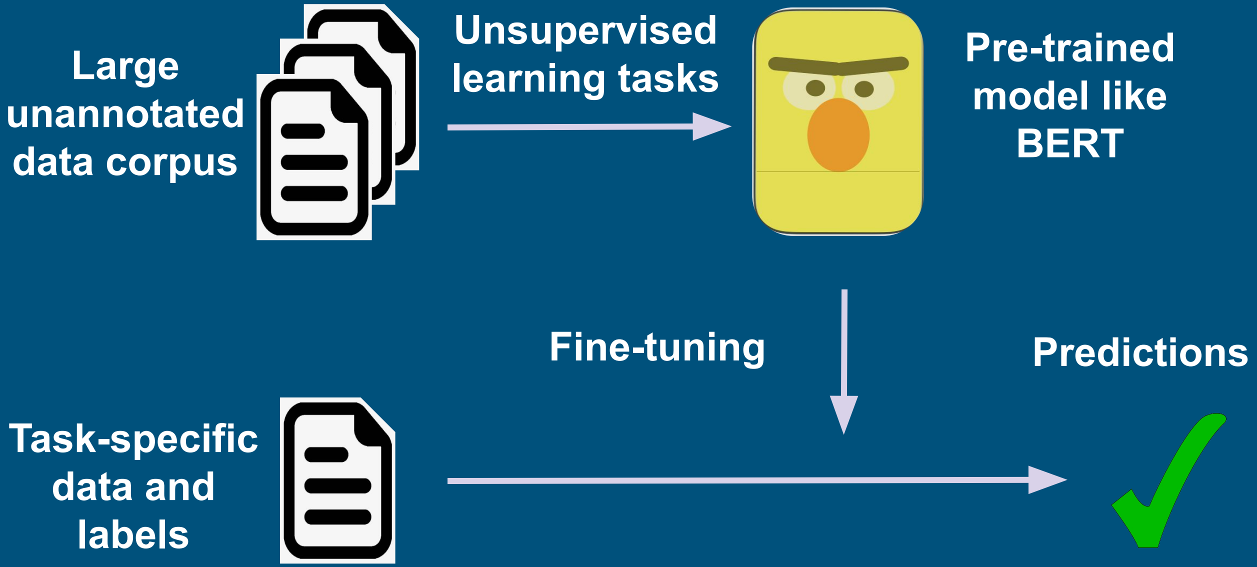
Prakhar Ganesh[1*], Yao Chen[1*], Xin Lou[1], Mohammad Ali Khan[1],
Yin Yang[2], Hassan Sajjad[3], Preslav Nakov[3], Deming Chen[1,4], Marianne Winslett[1,4]

[1] Advanced Digital Sciences Center, Illinois at Singapore
[2] College of Science and Engineering, Hamad Bin Khalifa University, Qatar
[3] Qatar Computing Research Institute, Hamad Bin Khalifa University, Qatar
[4] University of Illinois at Urbana-Champaign, USA

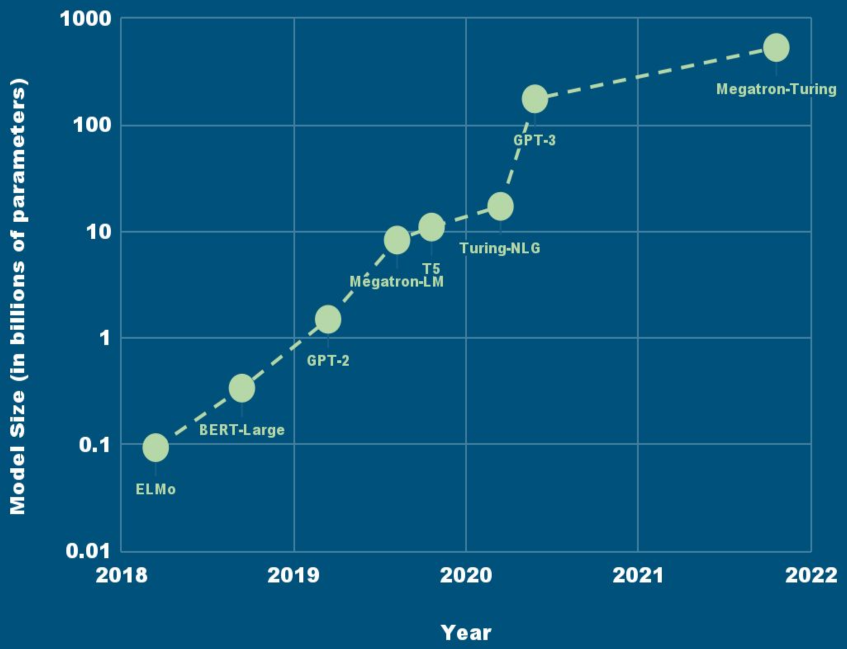
Large-Scale
Pre-Trained Models

Large-Scale Pre-Trained Models

Large-Scale Pre-Trained Models



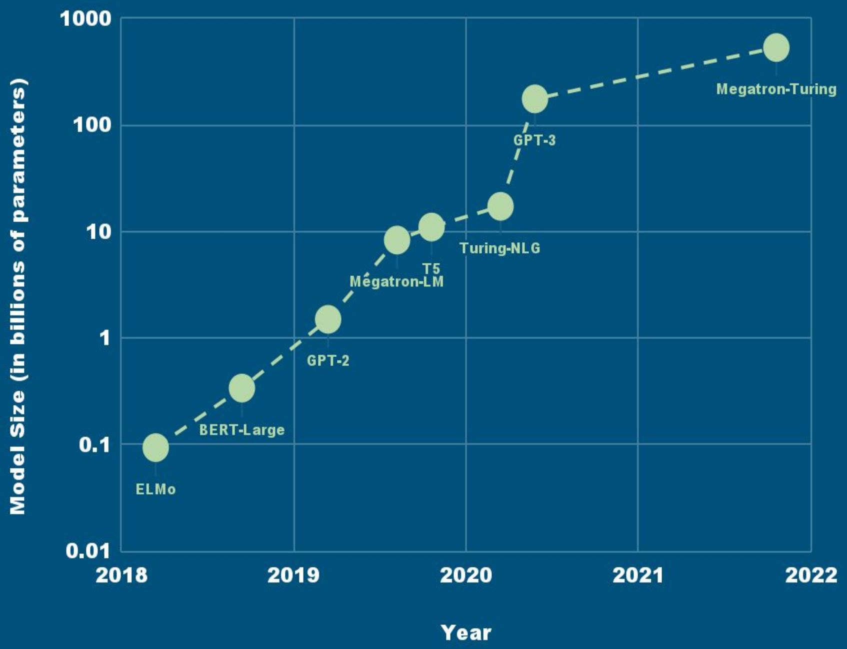
Why Do We Need Compression?



- **Rapidly increasing size of pre-trained language models**

Data Source : https://analyticsindiamag.com/we-might-see-a-100t-language-model-in-2022/

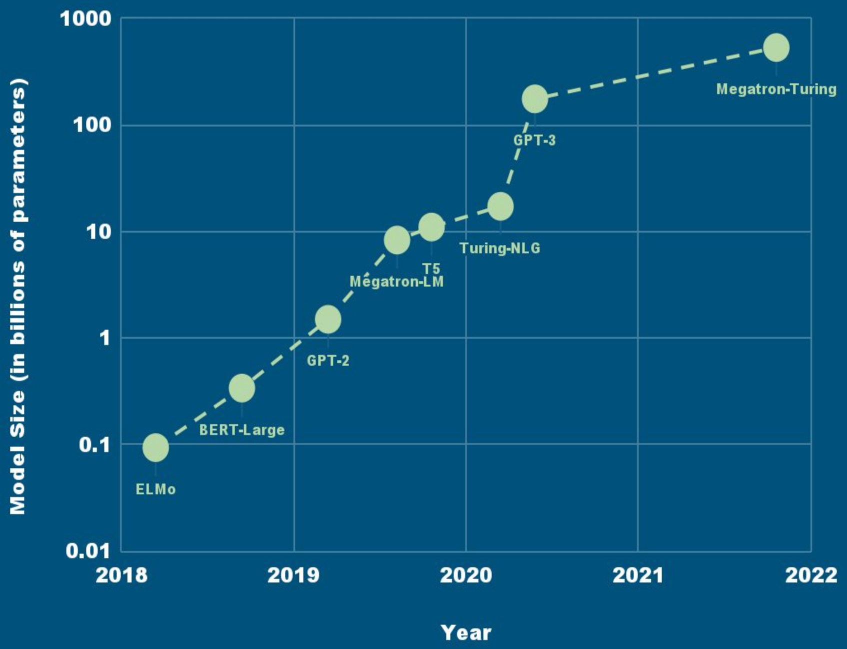
Why Do We Need Compression?



- **Rapidly increasing size of pre-trained language models**
- **Even a highly advanced data center GPU like Nvidia H100 (80 GB Memory) can only handle a few billion parameters during inference!!**

Data Source : https://analyticsindiamag.com/we-might-see-a-100t-language-model-in-2022/

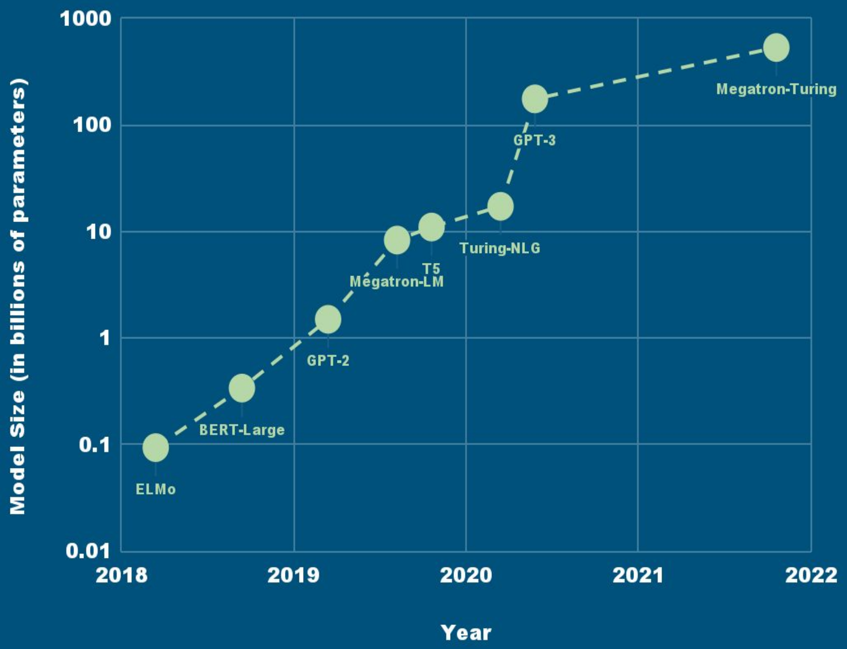
Why Do We Need Compression?



- Rapidly increasing size of pre-trained language models
- Even a highly advanced data center GPU like Nvidia H100 (80 GB Memory) can only handle a few billion parameters during inference!!
- Deploying these models requires access to high-performance clusters

Data Source : https://analyticsindiamag.com/we-might-see-a-100t-language-model-in-2022/

Why Do We Need Compression?

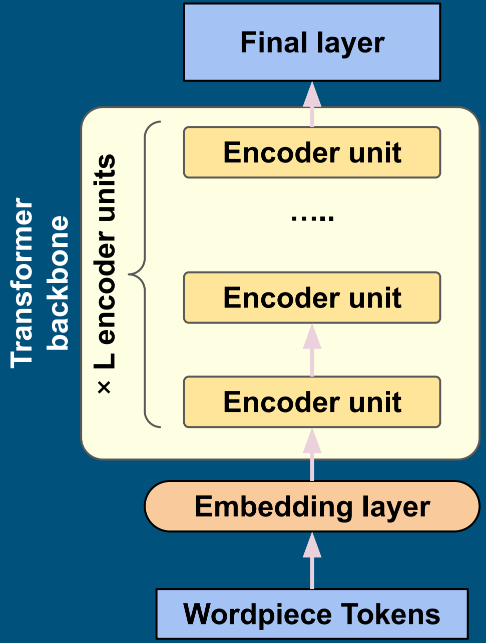


- Rapidly increasing size of pre-trained language models
- Even a highly advanced data center GPU like Nvidia H100 (80 GB Memory) can only handle a few billion parameters during inference!!
- Deploying these models requires access to high-performance clusters
- Solution: Model Compression!

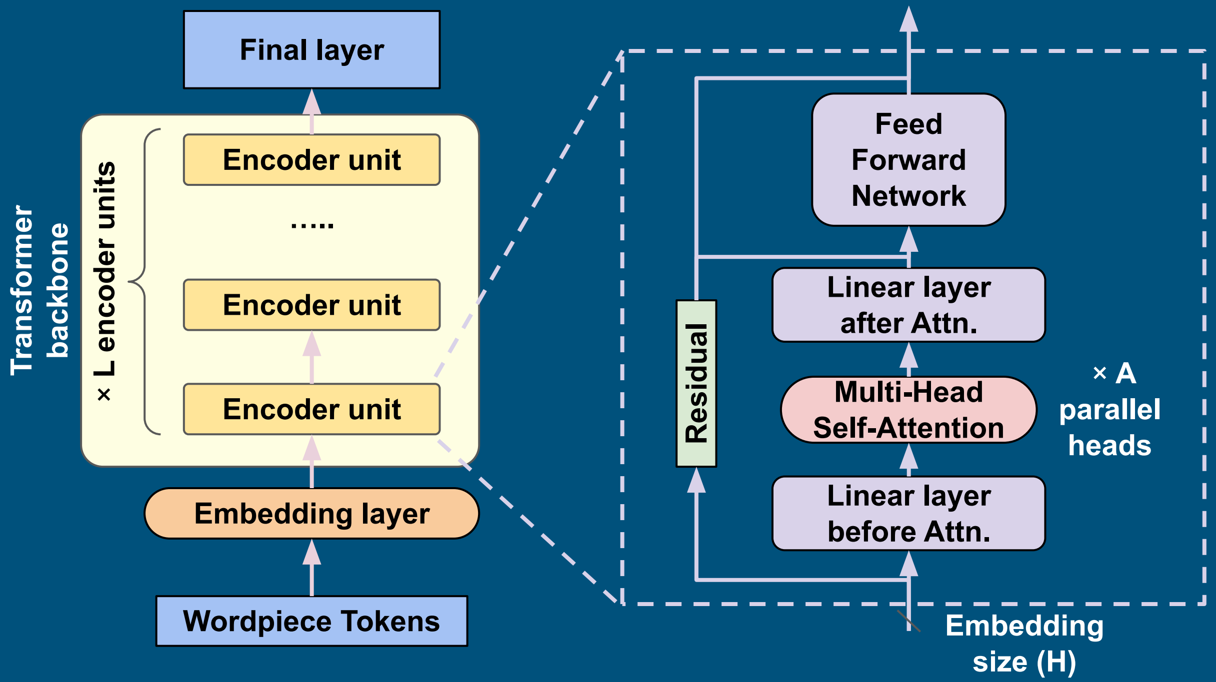
Data Source : https://analyticsindiamag.com/we-might-see-a-100t-language-model-in-2022/

BERT Breakdown Analysis

BERT Model

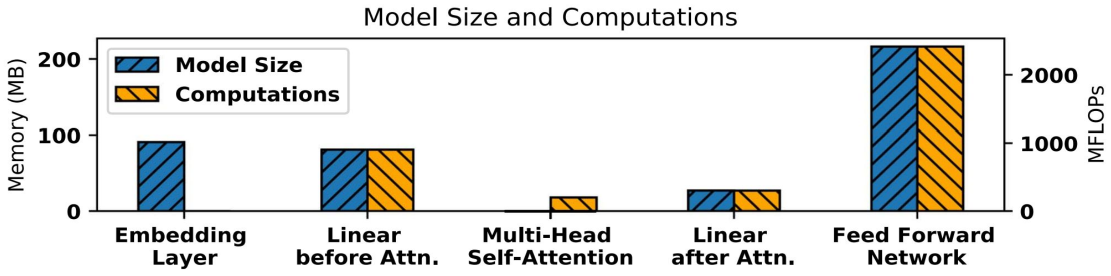

Final layer
Encoder unit
.....
Encoder unit
Encoder unit
× L encoder units
Transformer backbone
Embedding layer
Wordpiece Tokens

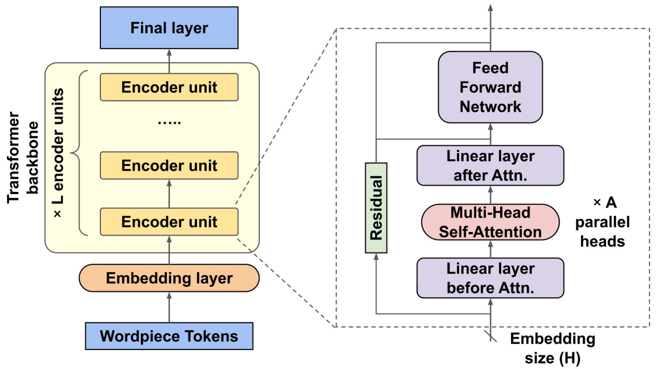
BERT Model



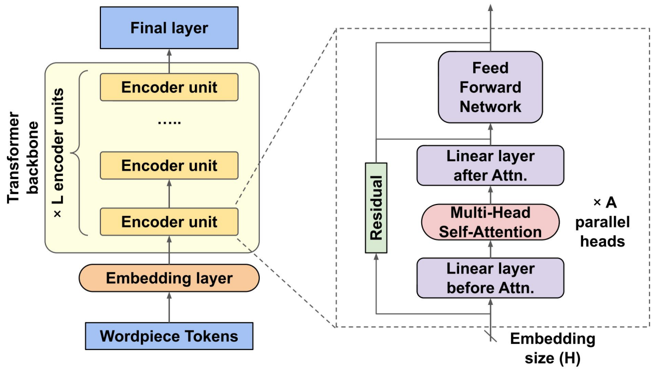
Final layer
Transformer backbone
× L encoder units
Encoder unit
.....
Encoder unit
Encoder unit
Embedding layer
Wordpiece Tokens
Feed Forward Network
Residual
Linear layer after Attn.
Multi-Head Self-Attention
× A parallel heads
Linear layer before Attn.
Embedding size (H)

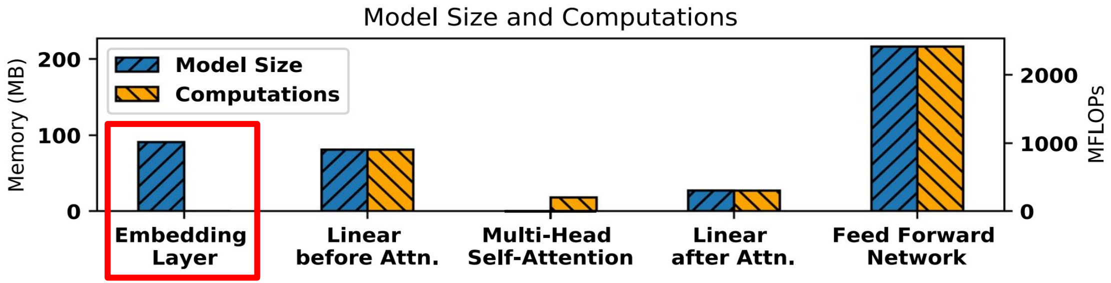
BERT Breakdown: Computation & Memory



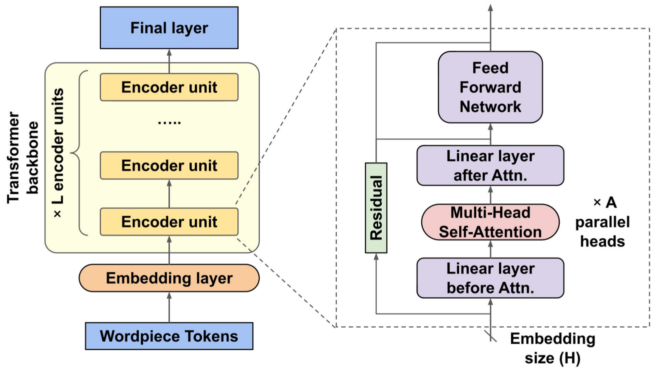


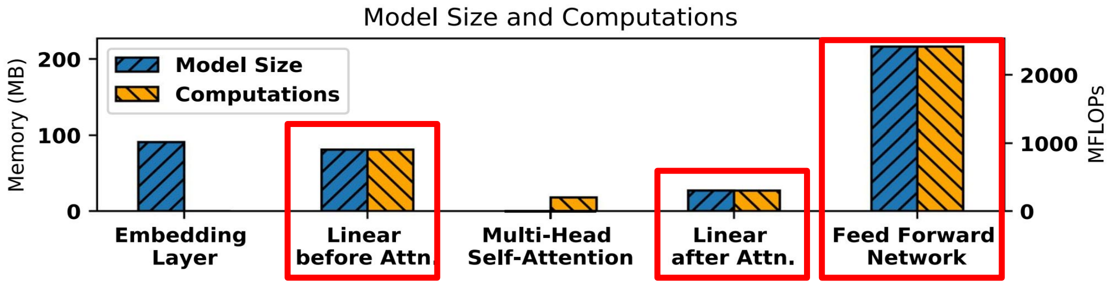
BERT Breakdown: Computation & Memory



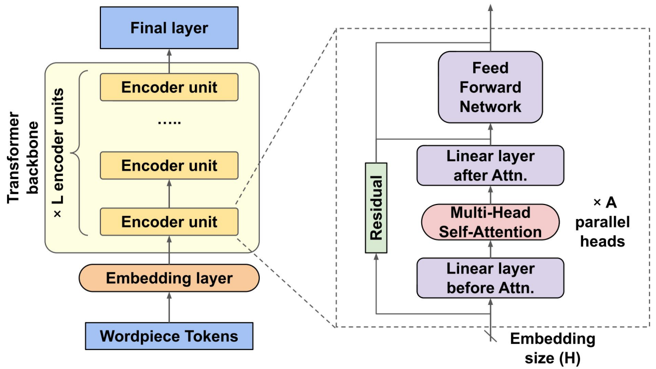


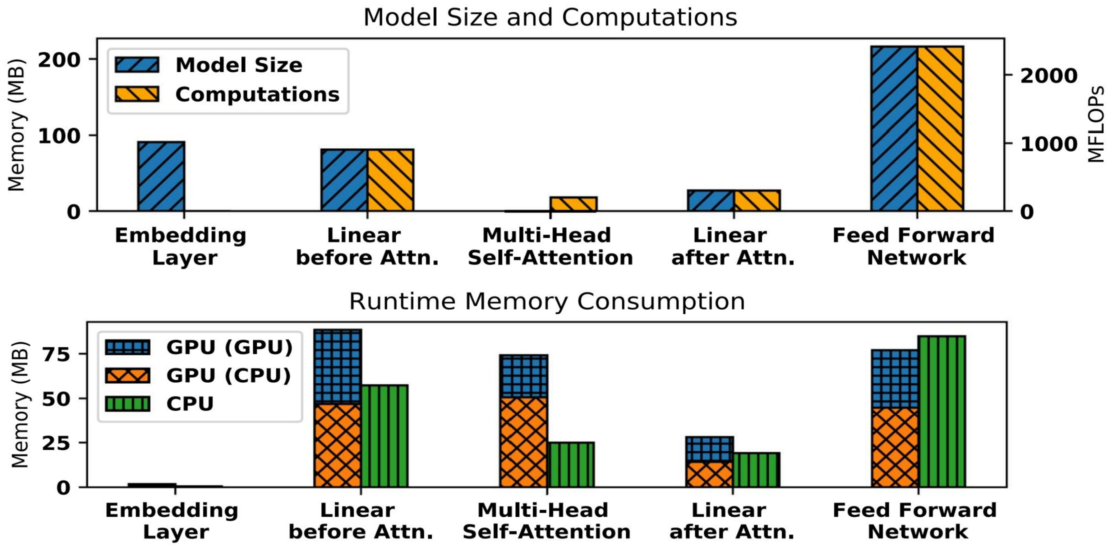
BERT Breakdown: Computation & Memory



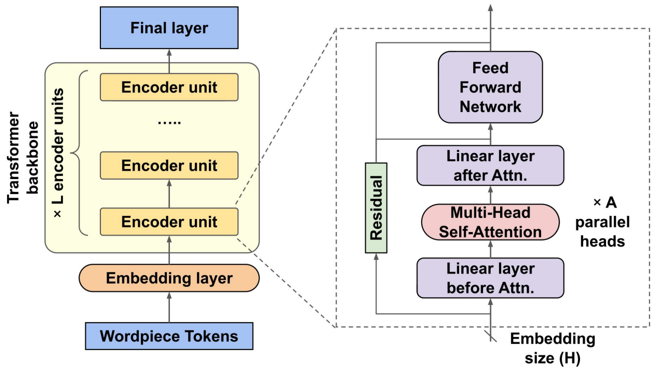


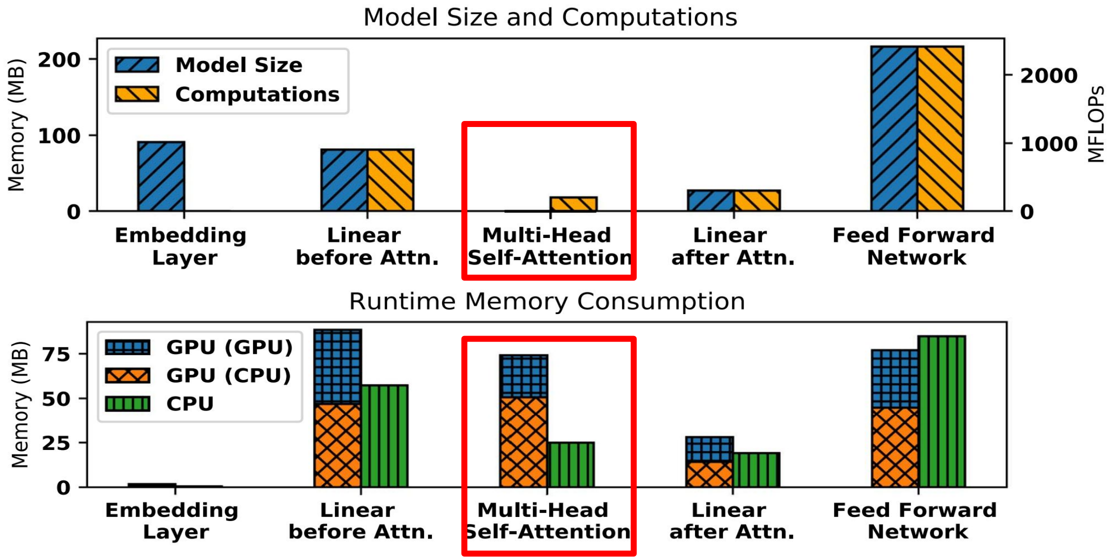
BERT Breakdown: Runtime Memory



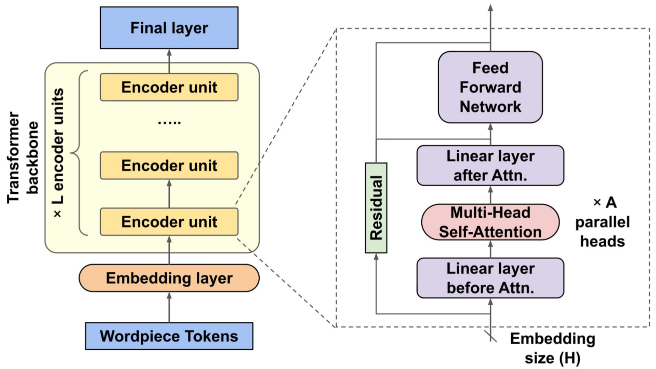


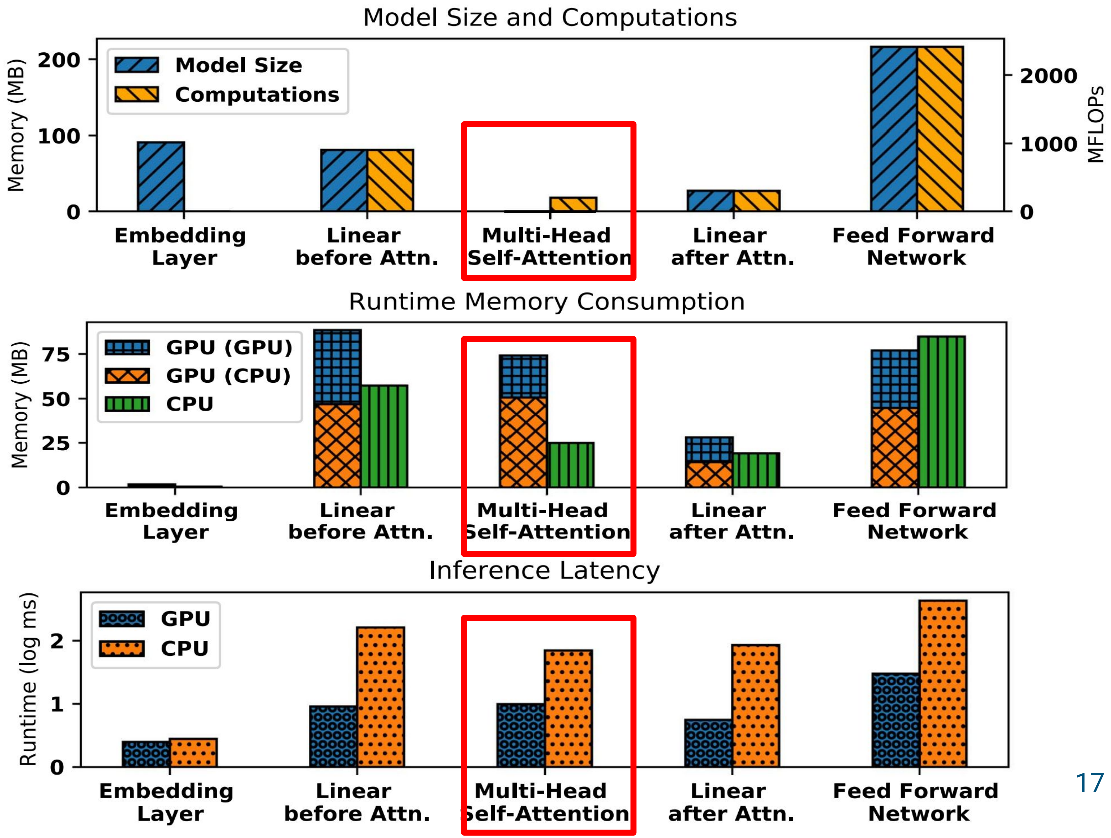
BERT Breakdown: Runtime Memory





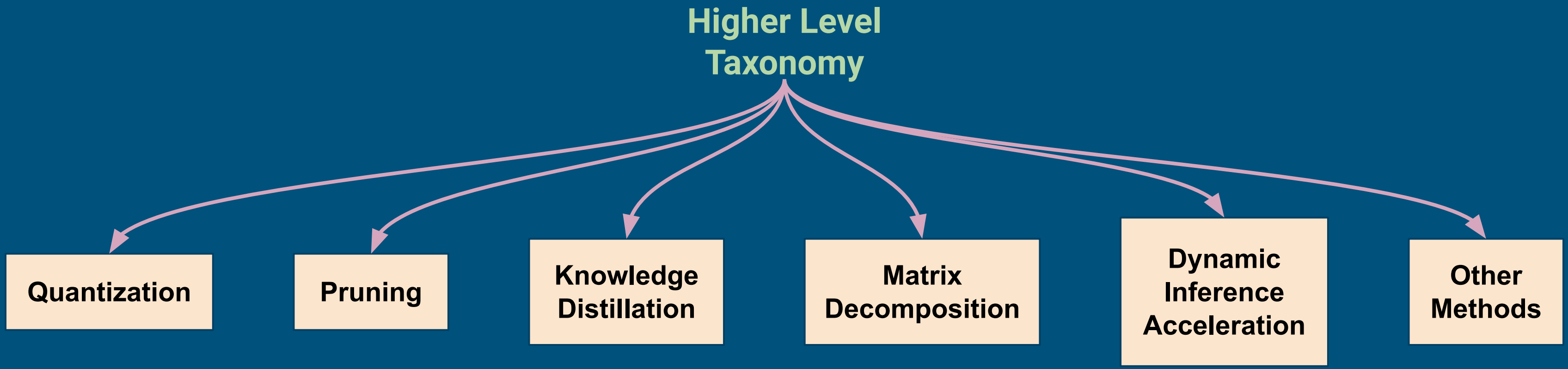
BERT Breakdown: Inference Latency



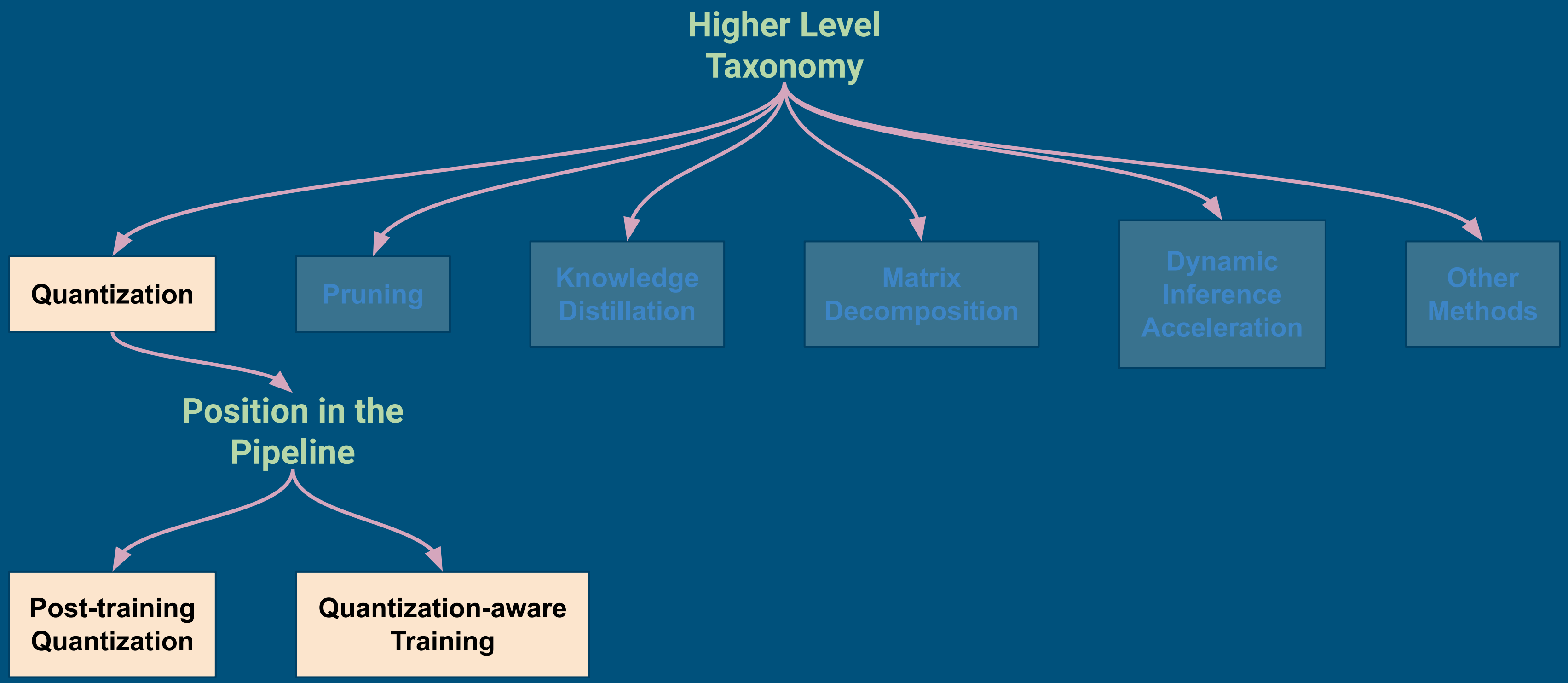


Model Compression

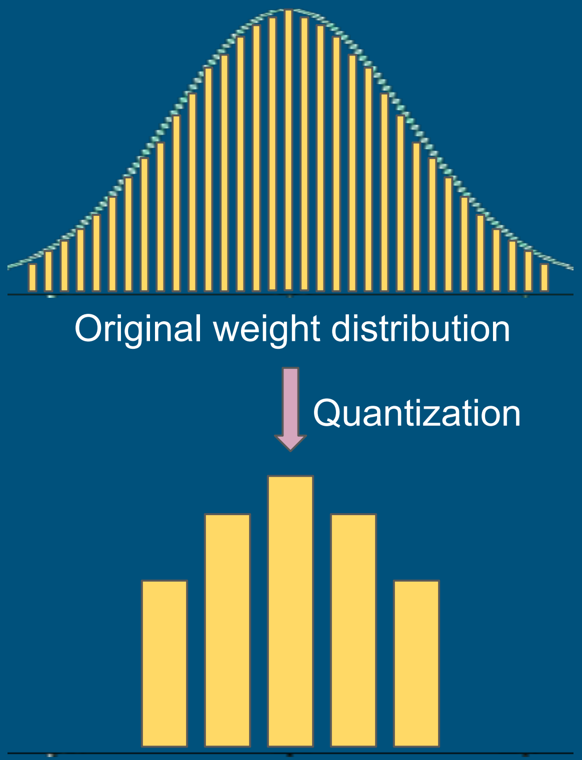
Compression Methods for BERT



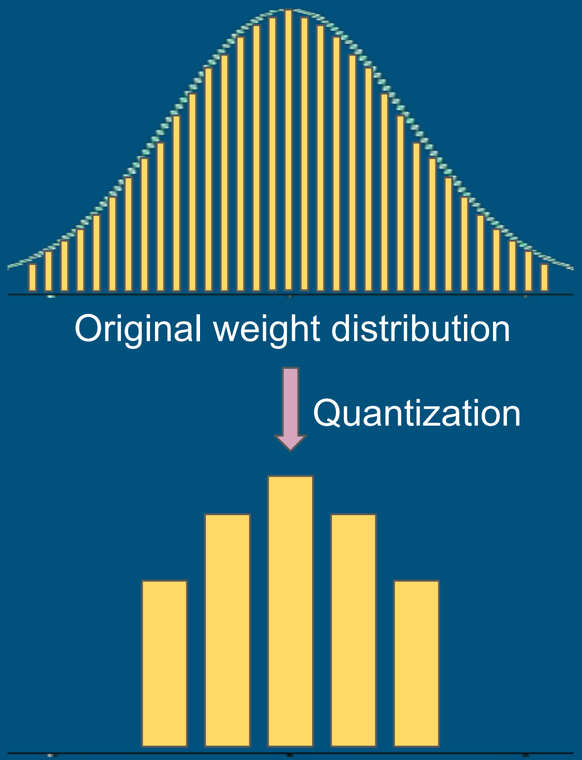
Quantization



Quantization



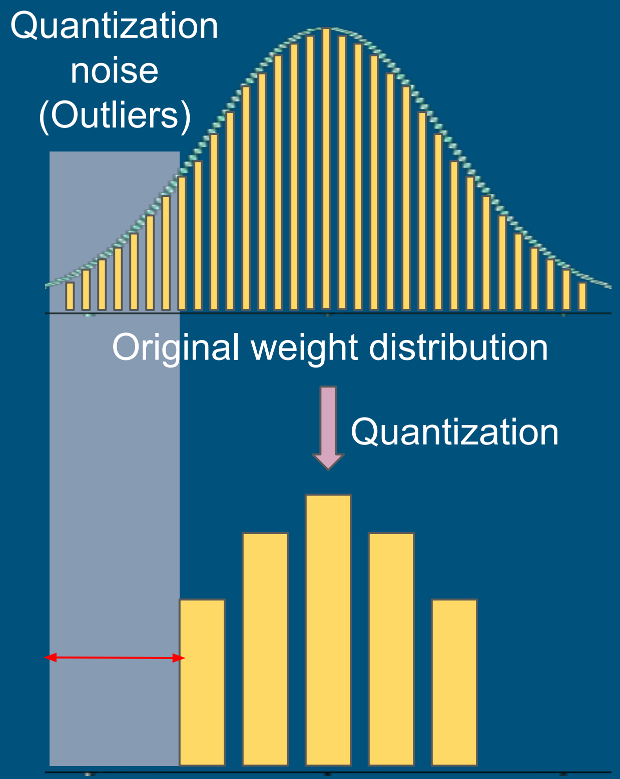
Quantization



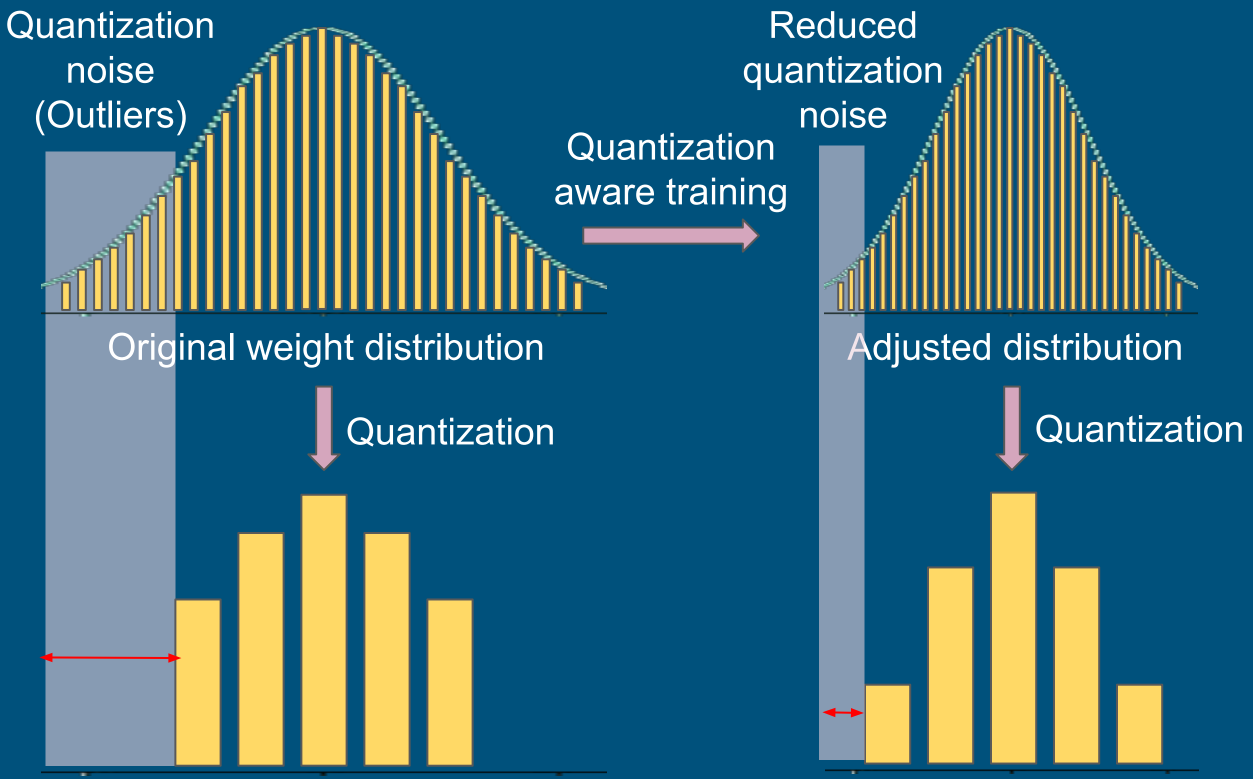
FP32/FP64

FP16, INT8, INT4, etc.

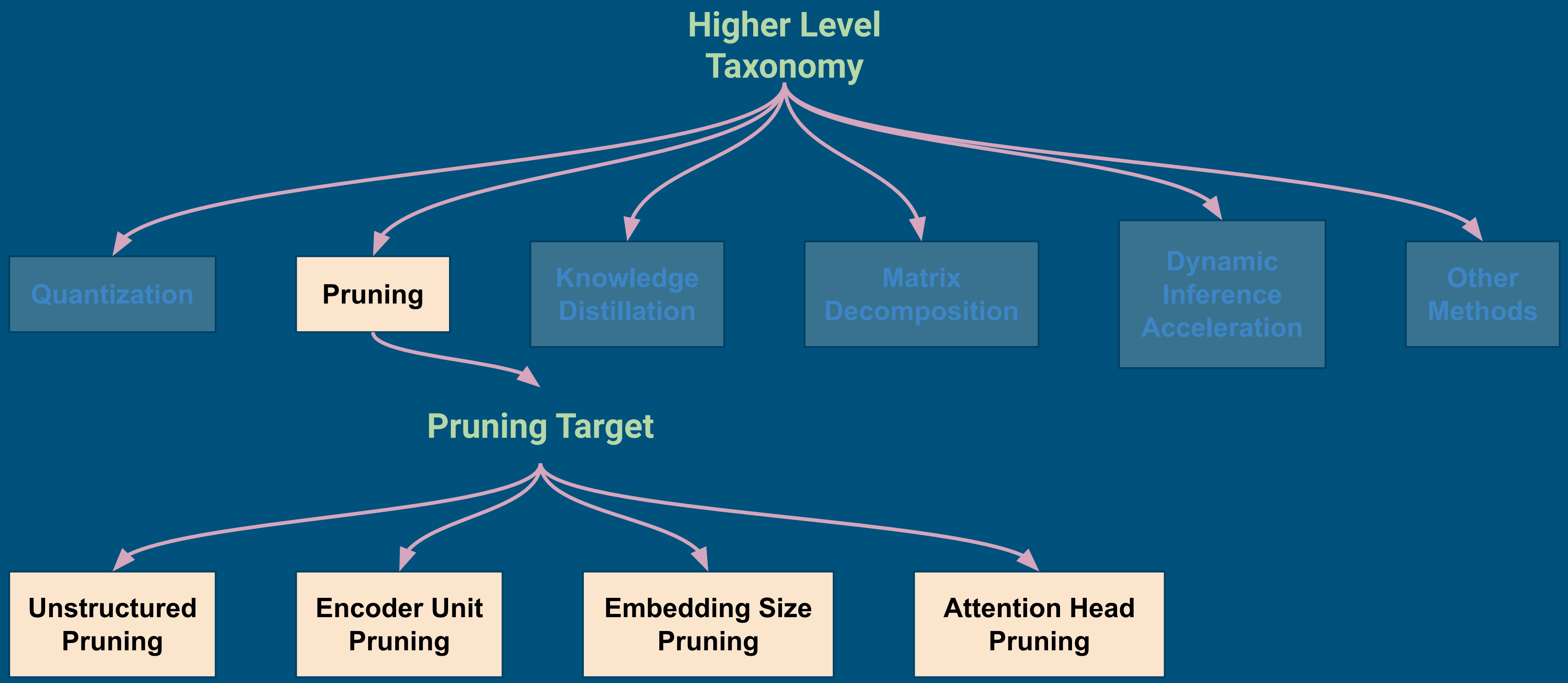
Quantization



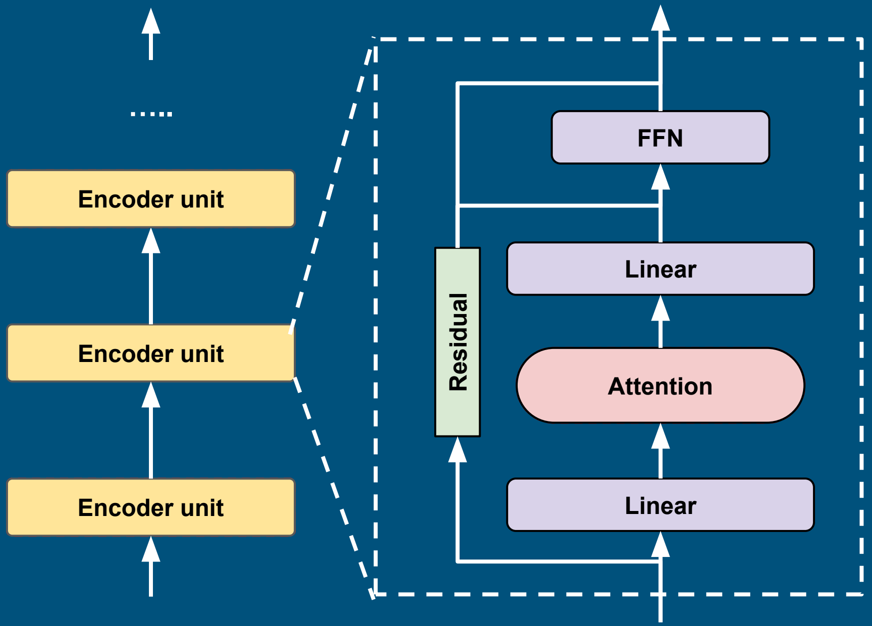
Quantization



Pruning

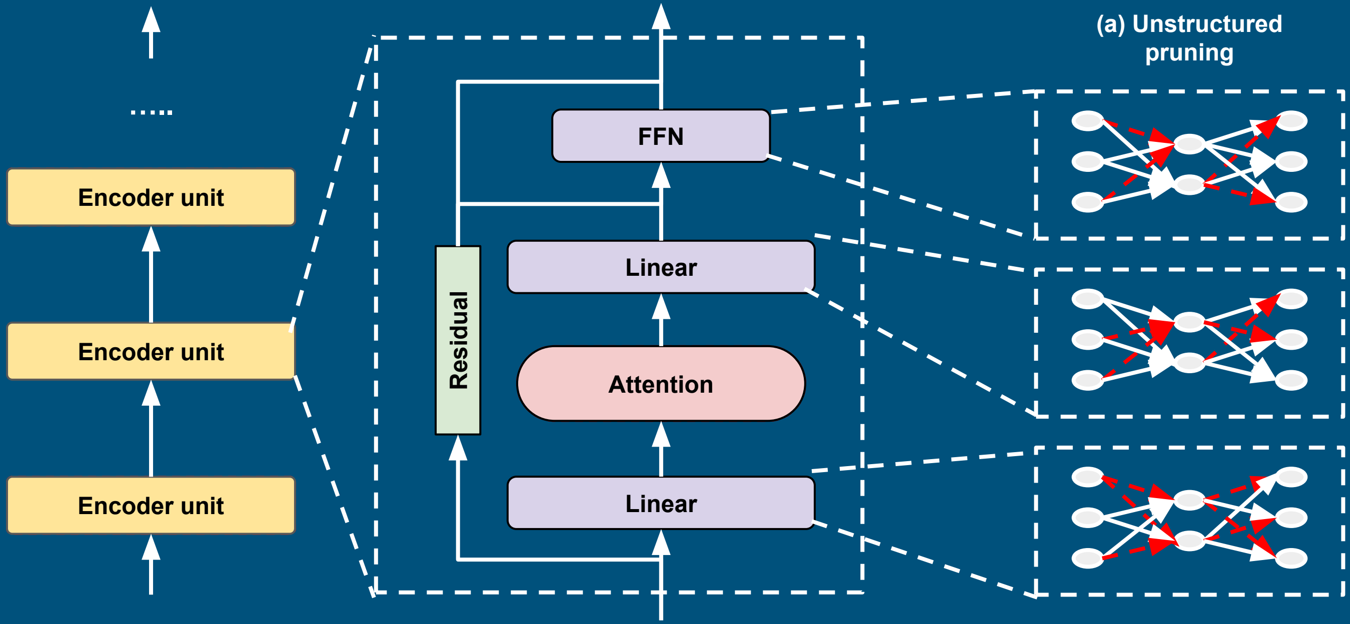

Higher Level Taxonomy
Quantization
Pruning
Knowledge Distillation
Matrix Decomposition
Dynamic Inference Acceleration
Other Methods
Pruning Target
Unstructured Pruning
Encoder Unit Pruning
Embedding Size Pruning
Attention Head Pruning

Pruning

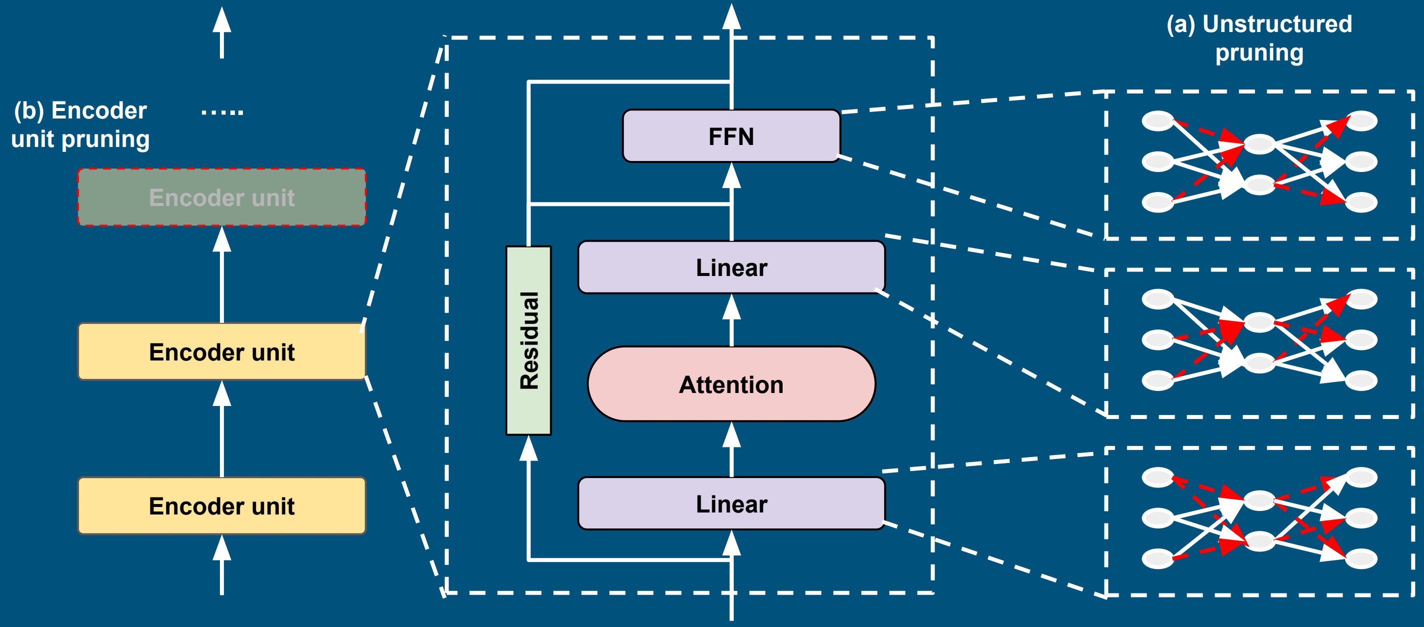


.....
Encoder unit
Encoder unit
Encoder unit
FFN
Linear
Residual
Attention
Linear

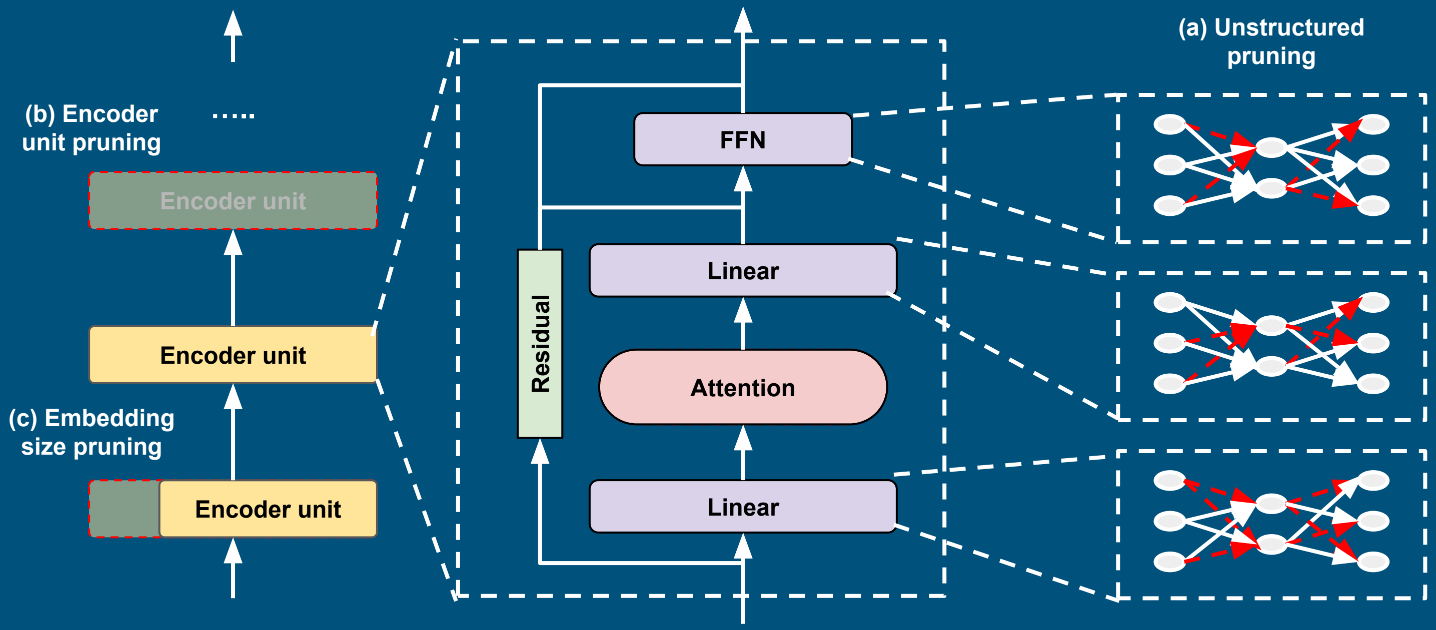
Pruning: Unstructured Pruning



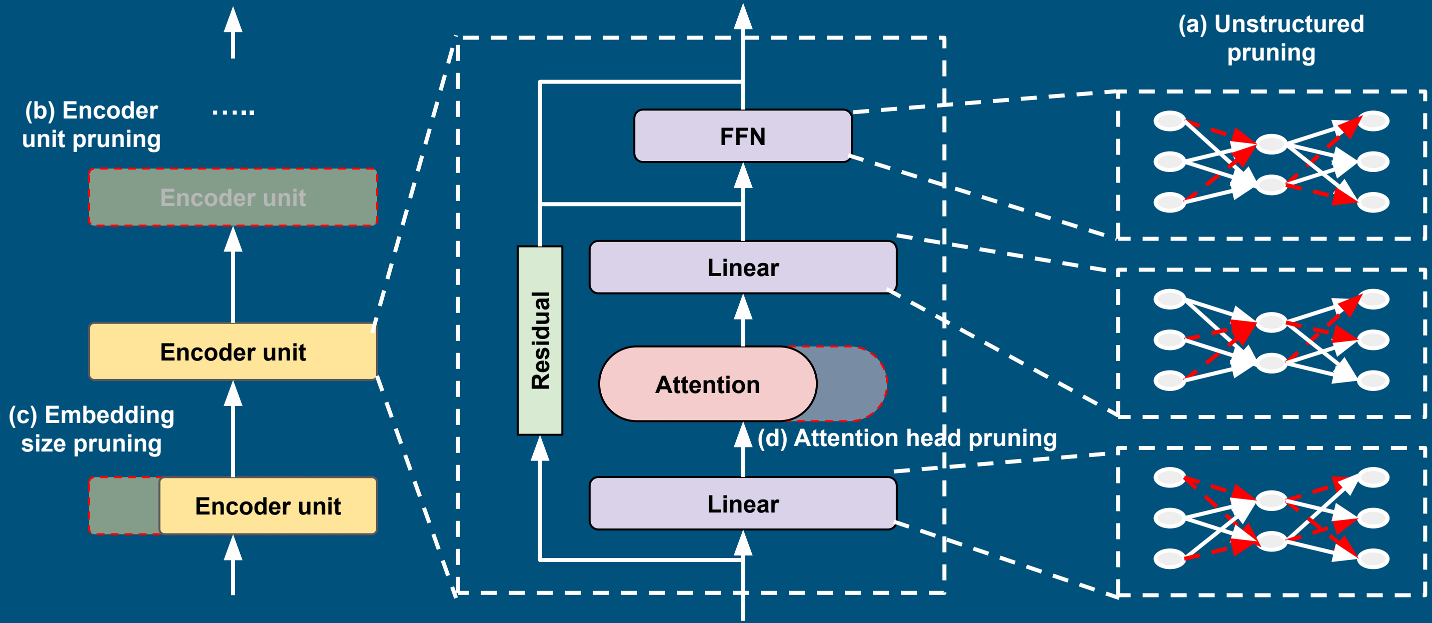
Pruning: Encoder Unit Pruning



Pruning: Embedding Size Pruning



Pruning: Attention Head Pruning

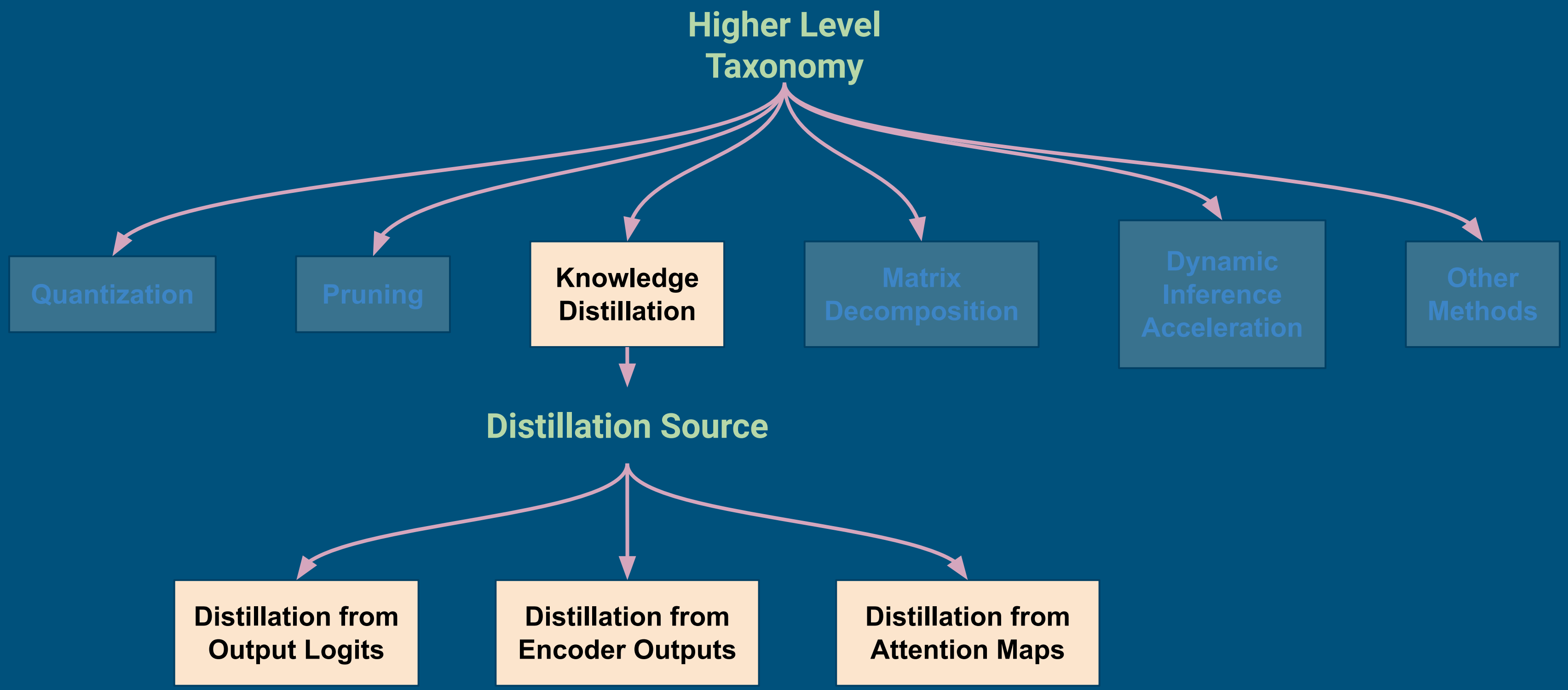


Recent Work in BERT Pruning

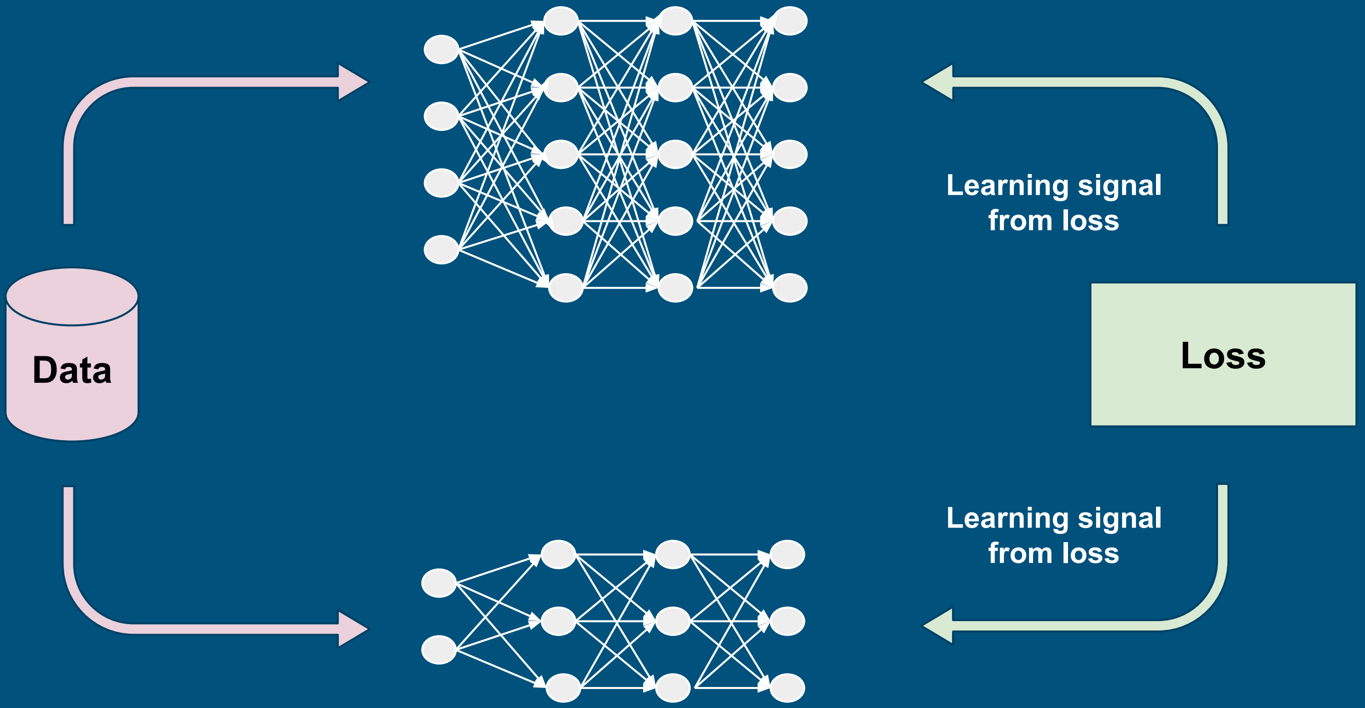
- Several papers have been published on pruning for BERT compression since the publication of our survey!!

- Guo, Demi, et al. "Parameter-Efficient Transfer Learning with Diff Pruning." ACL-IJCNLP 2021.
- Xu, Dongkuan, et al. "Rethinking Network Pruning–under the Pre-train and Fine-tune Paradigm." NAACL 2021.
- Rotman, Guy, et al. "Model compression for domain adaptation through causal effect estimation." TACL 2021.
- Kovaleva, Olga, et al. "BERT Busters: Outlier Dimensions that Disrupt Transformers." ACL-IJCNLP 2021.
- Fan, Chun, et al. "Layer-wise Model Pruning based on Mutual Information." EMNLP 2021.
- Peer, David, et al. "Greedy-layer Pruning: Speeding up Transformer Models for Natural Language Processing." Pattern Recognition Letters 2022.

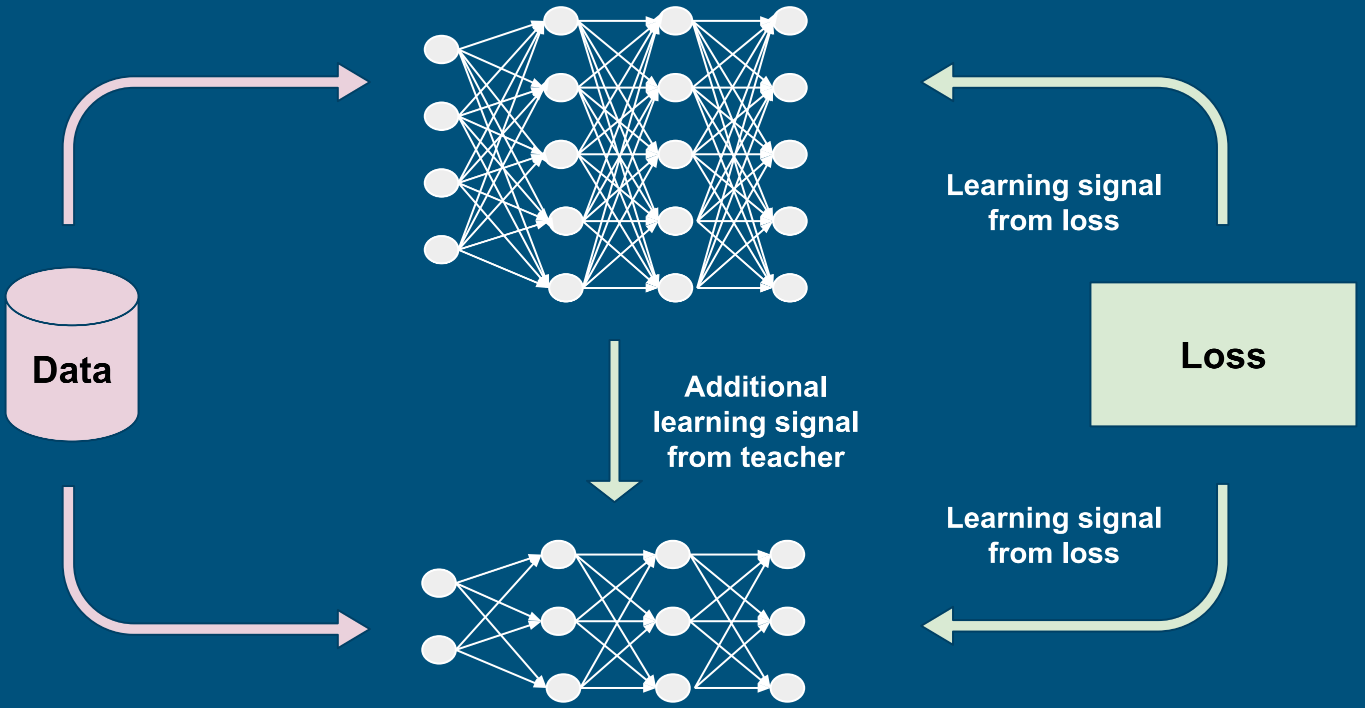
Knowledge Distillation



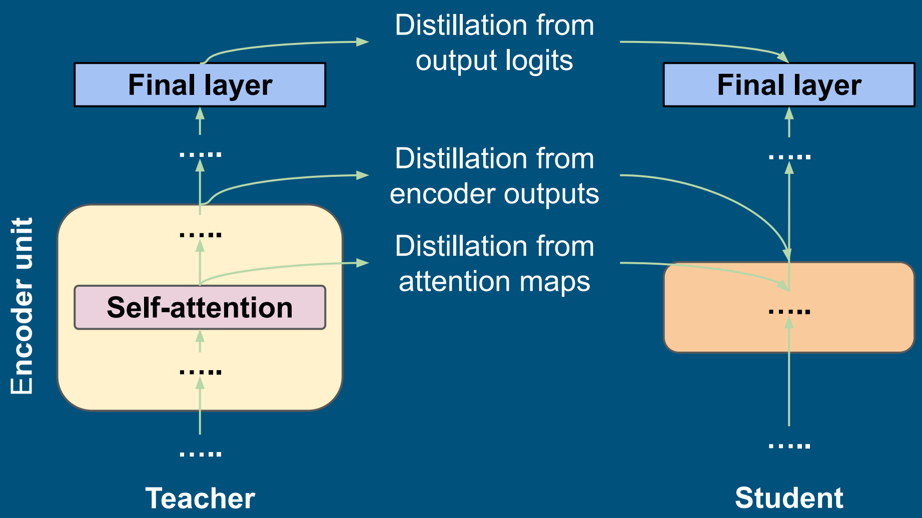
Knowledge Distillation



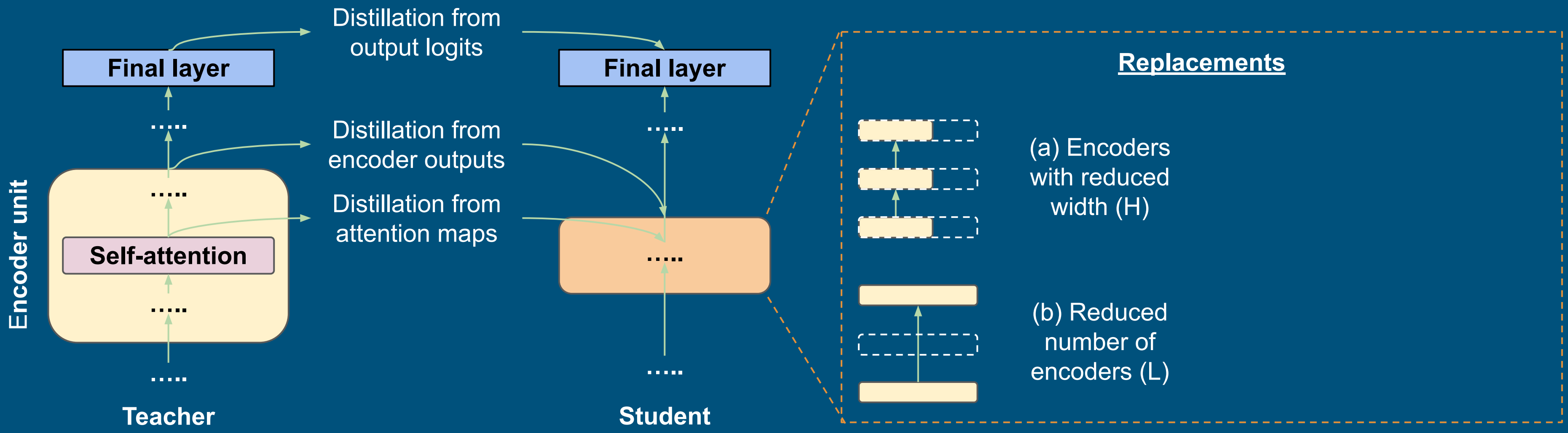
Knowledge Distillation



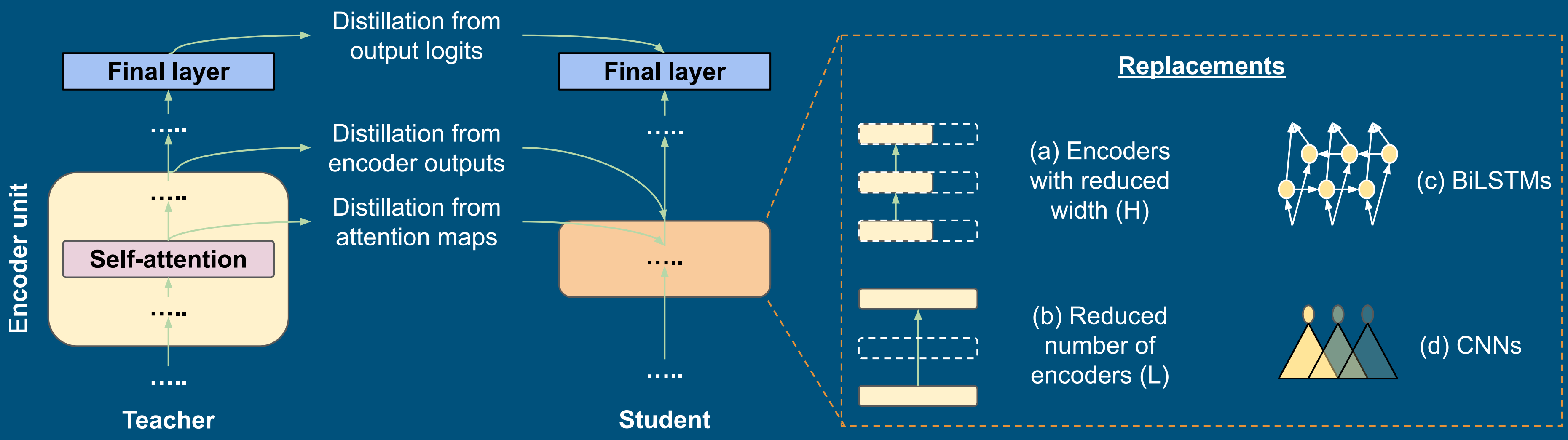
Knowledge Distillation



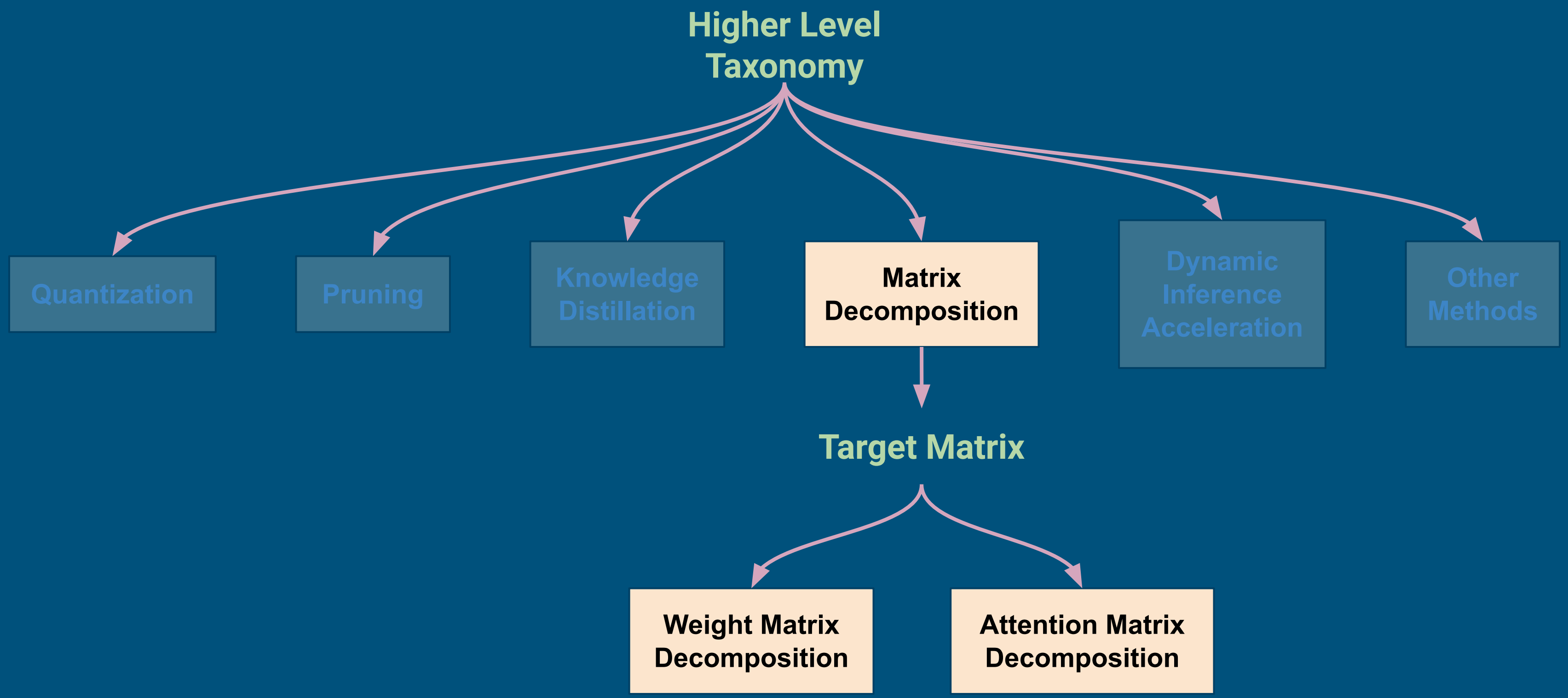
Knowledge Distillation



Knowledge Distillation

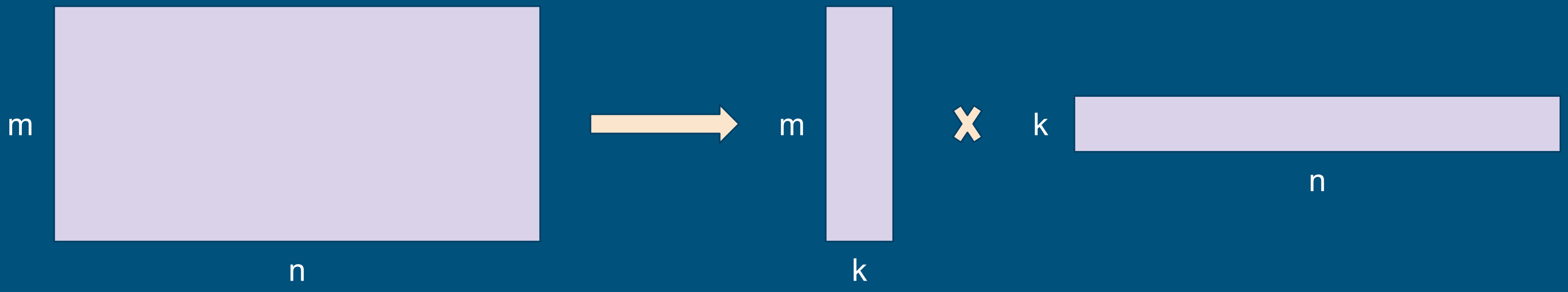


Matrix Decomposition



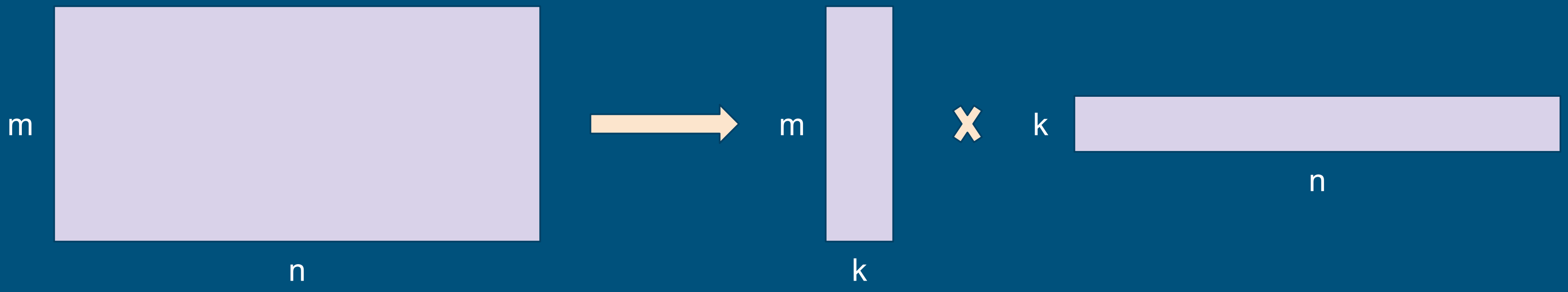
Matrix Decomposition

Weight matrix Decomposition



Matrix Decomposition

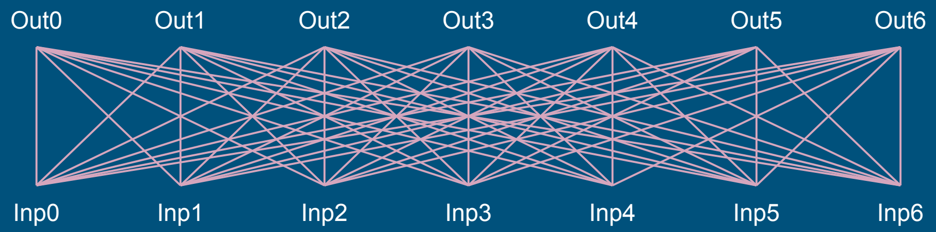
Weight matrix Decomposition



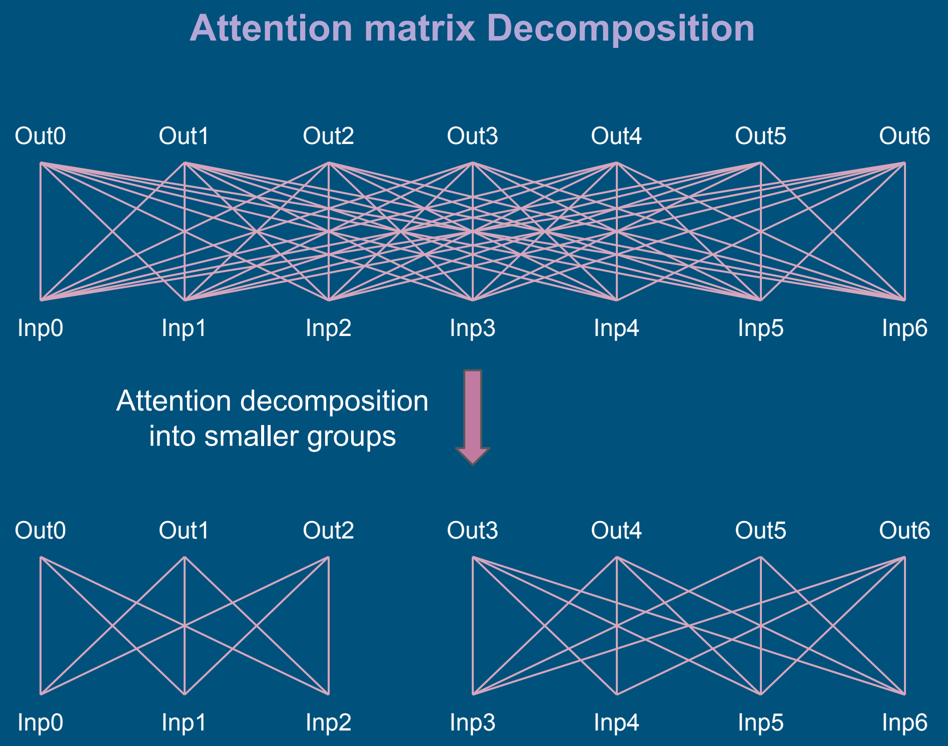
Significantly reduces number of parameters and computations for m,n >> k

Matrix Decomposition

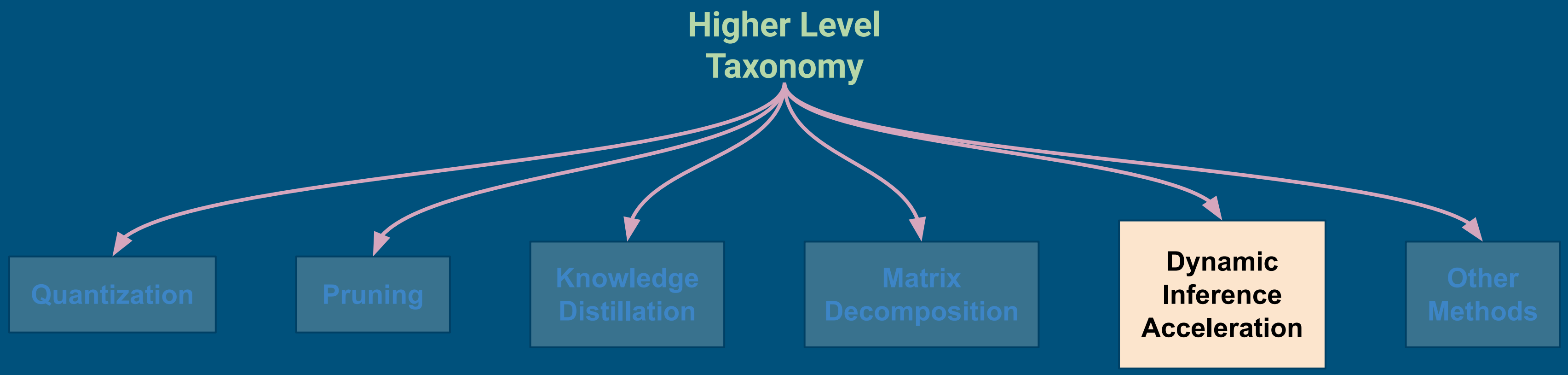
Attention matrix Decomposition



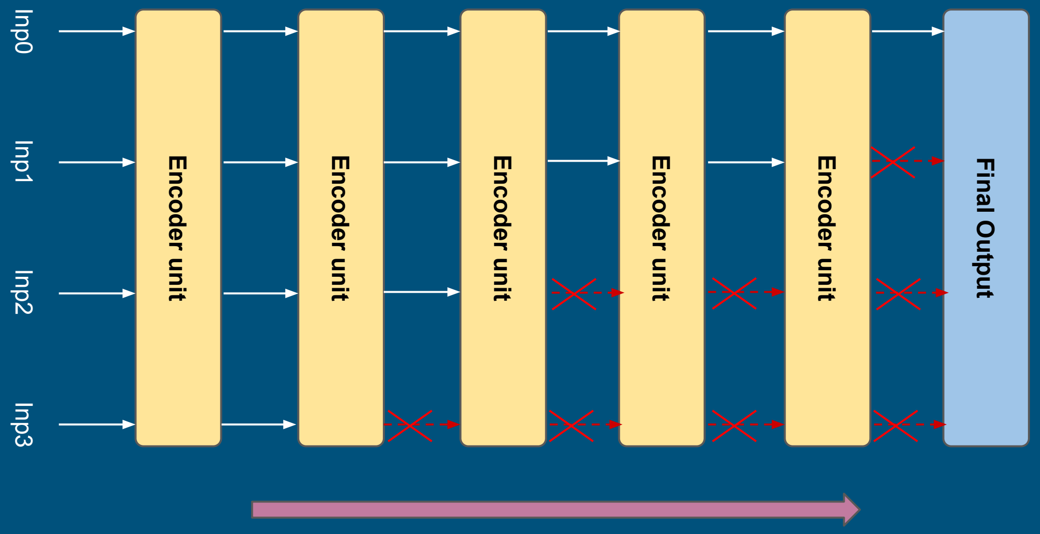
Matrix Decomposition



Dynamic Inference Acceleration

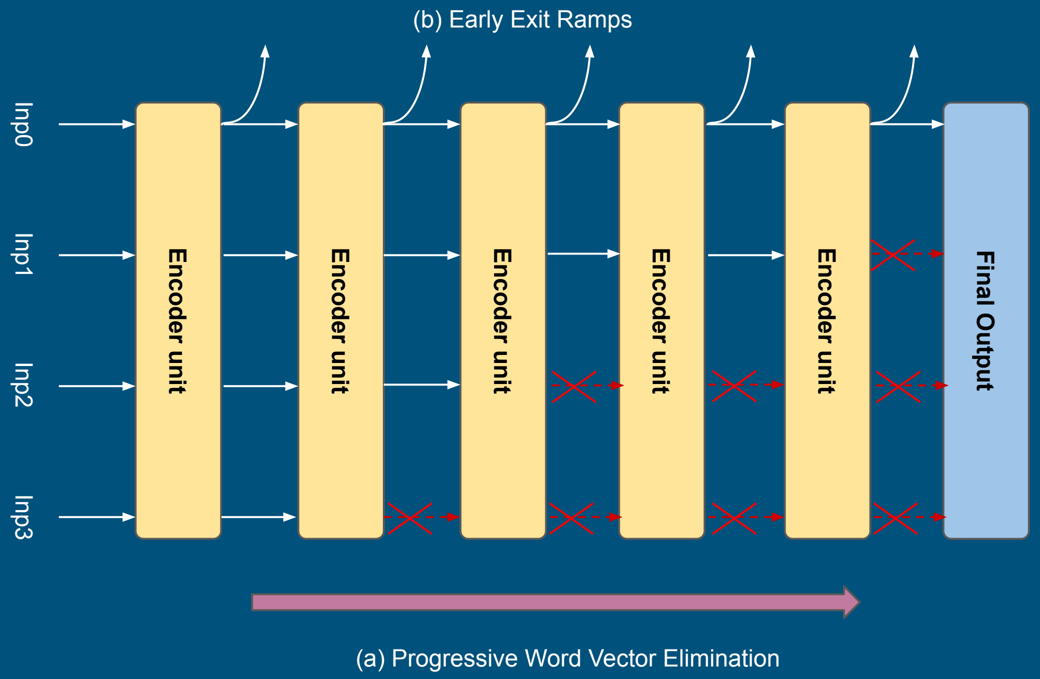


Dynamic Inference Acceleration

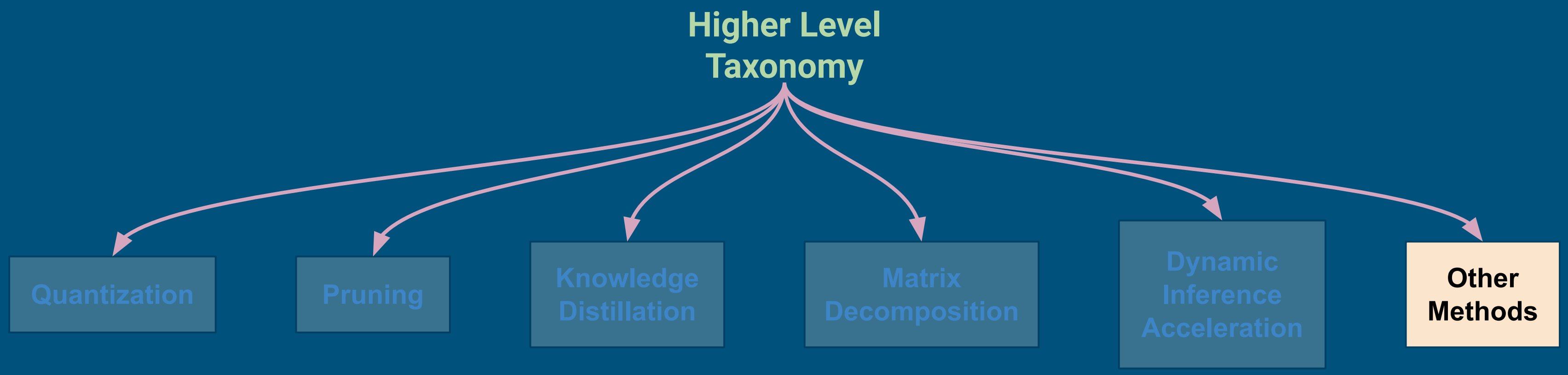


(a) Progressive Word Vector Elimination

Dynamic Inference Acceleration

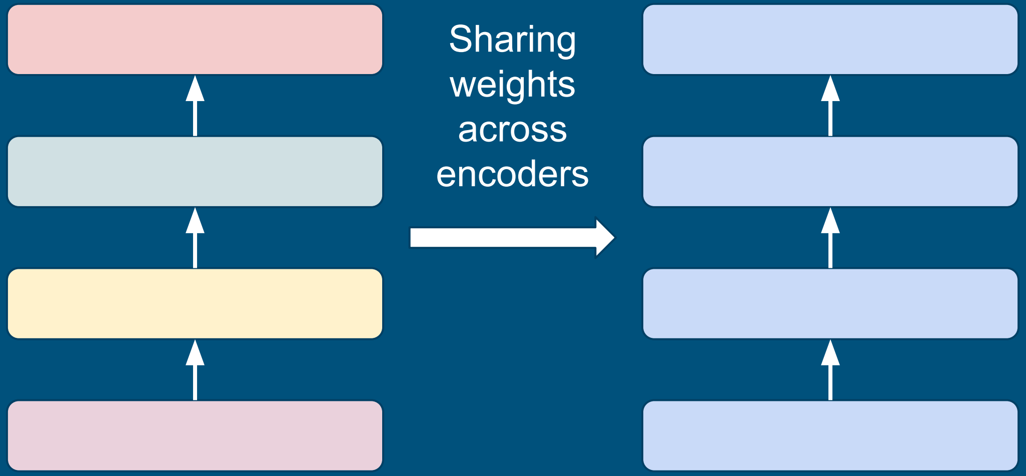


Other Methods



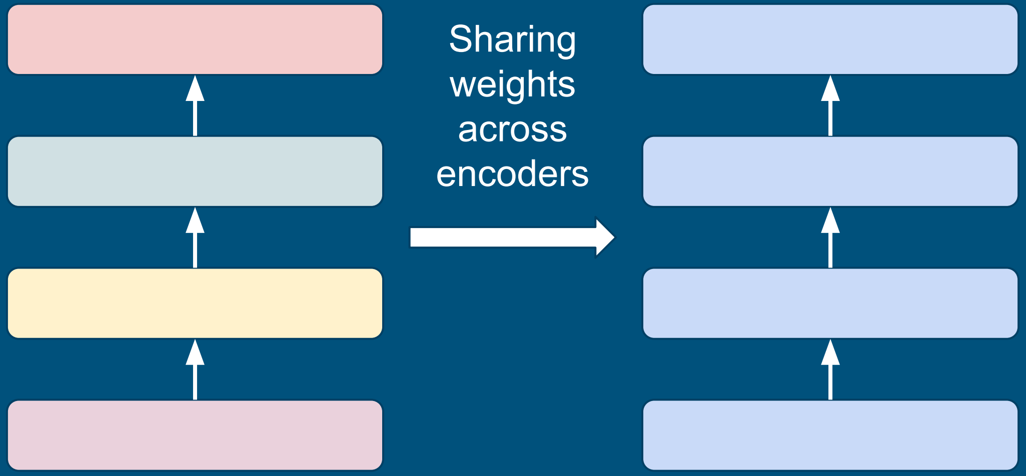
Other Methods

- Parameter Sharing: Sharing parameters across various encoders



Other Methods

- Parameter Sharing: Sharing parameters across various encoders

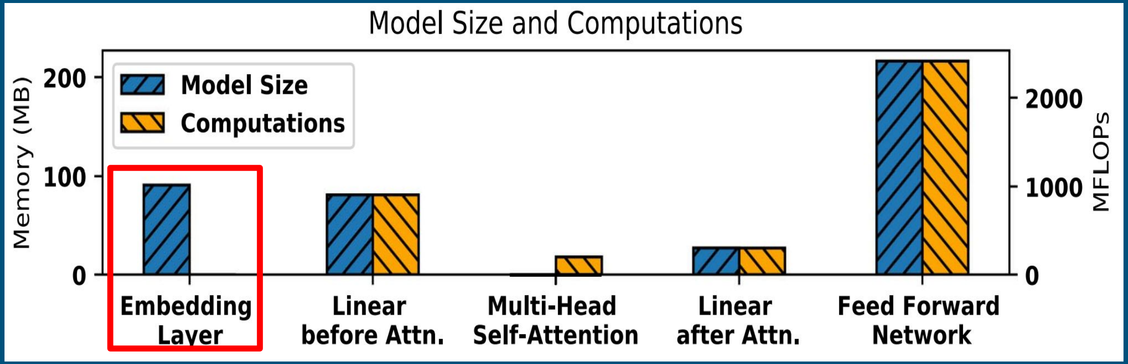


More nuanced methods
of parameter sharing
have also been explored

Reid, Machel et al.. "Subformer: Exploring Weight Sharing for Parameter Efficiency in Generative Transformers." EMNLP 2021.

Other Methods

- Parameter Sharing: Sharing parameters across various encoders
- Embedding Matrix Compression: Compressing the embedding matrix (e.g., by reducing the vocabulary size)



21% of total size

Other Methods

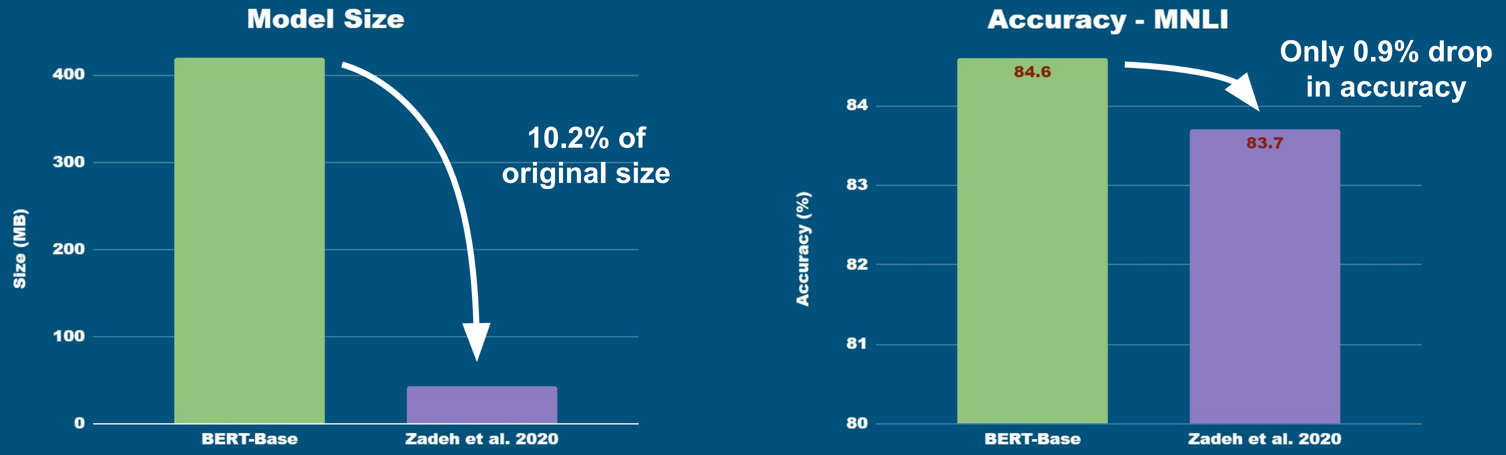
- Parameter Sharing: Sharing parameters across various encoders
- Embedding Matrix Compression: Compressing the embedding matrix (e.g., by reducing the vocabulary size)
- Weight Squeezing: Distilling ‘weight’ signal instead of output signals

Effectiveness of the Compression Methods

Quantization

Quantization is the best single compression method

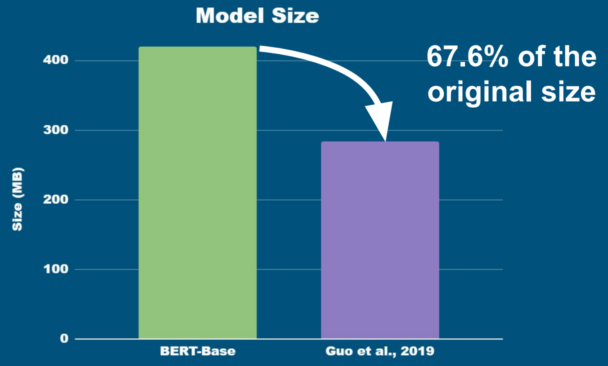
- can reduce model tenfold, with only 0.9% drop in accuracy
- yet, requires specialised hardware for inference speedup!

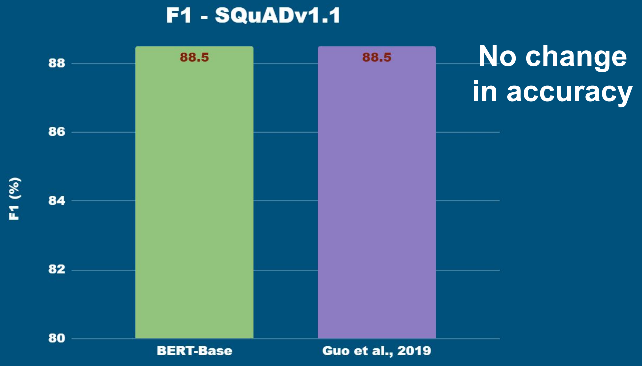


Zadeh, Ali Hadi, et al. "GOBO: Quantizing attention-based nlp models for low latency and energy efficient inference." IEEE/ACM MICRO 2020.

Pruning

- A great method to reduce completely redundant weights
- Can reduce model size up to 67% of original size with no drop in accuracy
- Unstructured pruning has not been used to reduce the size of the embedding matrix (which takes 21% of the total model size)

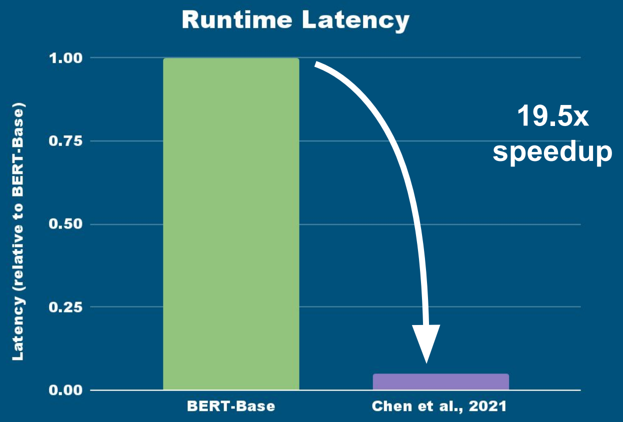


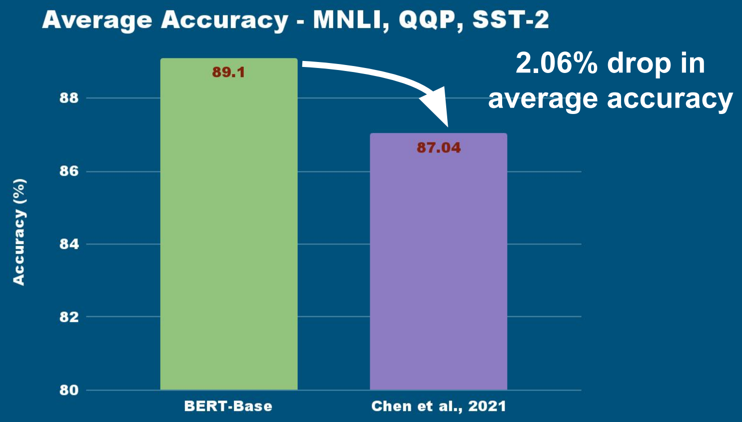


Guo, Fu-Ming, et al. "Reweighted proximal pruning for large-scale language representation." arXiv preprint arXiv:1909.12486 (2019).

BiLSTM and CNN Student Models

- Knowledge distillation allows to train BiLSTM- and CNN-based students
- Existing work can achieve up to 19.5x speedup with only a 2.06% accuracy drop
- CNNs can provide special caching benefits due to local processing

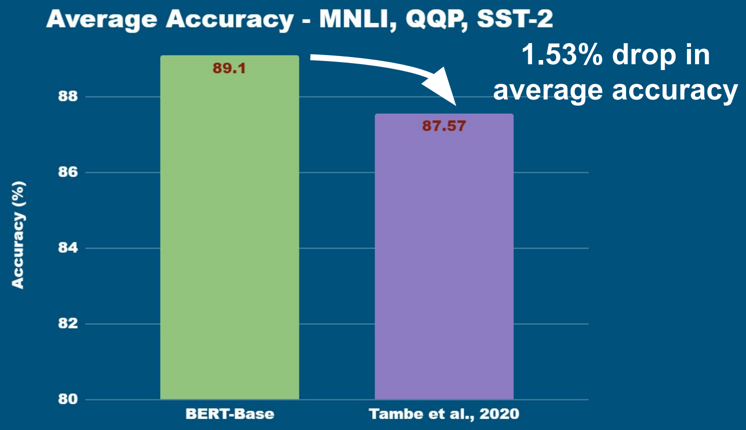




Chen, Daoyuan, et al. "AdaBERT: task-adaptive BERT compression with differentiable neural architecture search." IJCAI 2021.

Combining Compression Methods

- Combining multiple compression methods can filter out more redundancies
- Existing work in combining compression methods can reduce model size to only 1.3% of its original size with just 1.53% drop in accuracy!



Tambe, Thierry, et al. "Edgebert: Sentence-level energy optimizations for latency-aware multi-task nlp inference." IEEE/ACM MICRO 2021.

Combining Compression Methods

- Combining multiple compression methods can filter out more redundancies
- Existing work in combining compression methods can reduce model size to only 1.3% of its original size with just 1.53% drop in accuracy!
- More work in combining various compression methods was done since our publication!

Liu, Yuanxin, et al. "ROSITA: Refined BERT cOmpreSsion with InTegrAted techniques." AAAI 2021.

Practical Advice

Practical Advice

- Choose an appropriate baseline

Practical Advice

- Choose an appropriate baseline
- Use specialised hardware and accelerators

Practical Advice

- Choose an appropriate baseline
- Use specialised hardware and accelerators
- Investigate the target setup
  - Choose appropriate quantization and pruning settings
  - Choose an appropriate student model

Practical Advice

- Choose an appropriate baseline
- Use specialised hardware and accelerators
- Investigate the target setup
  - Choose appropriate quantization and pruning settings
  - Choose an appropriate student model
- Compound different compression methods
  - Combine multiple forms of compatible compression methods
  - Use knowledge distillation as a guide for other forms of compression

Thank You