PennState

BERT Pruning: *Structural vs. Sparse*

Dongkuan (DK) Xu, Ph.D. Candidate
Pennsylvania State University, Advisor: Xiang Zhang
Web: www.personal.psu.edu/dux19/
04-28-2021

Agenda

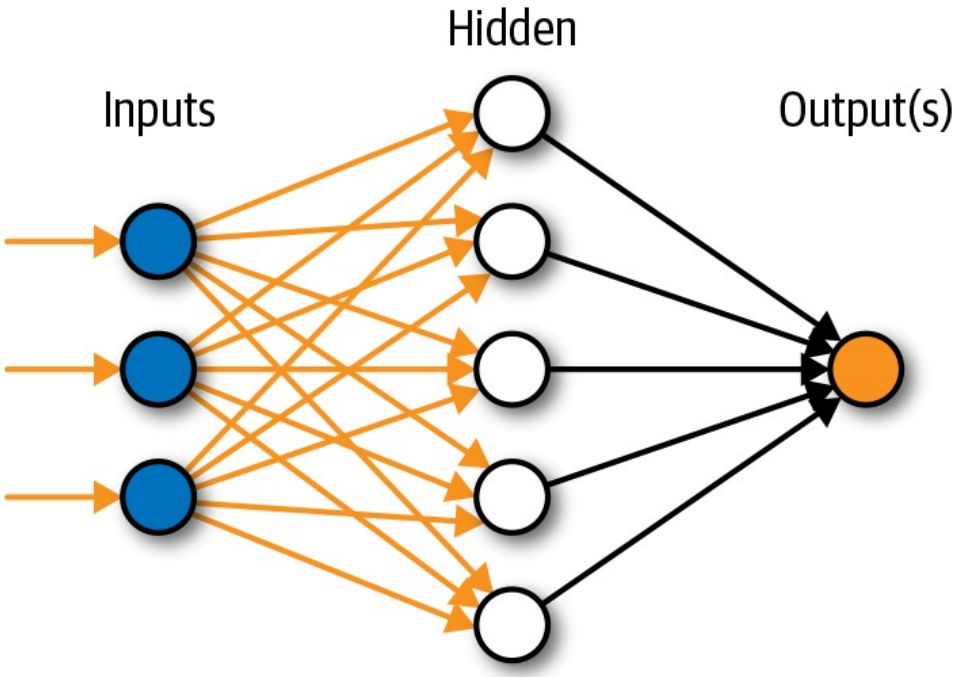
- Motivation
- Neural Network Pruning
- BERT Architecture
- Structural Pruning vs. Sparse Pruning

Agenda

- **Motivation**
- Neural Network Pruning
- BERT Architecture
- Structural Pruning vs. Sparse Pruning

Deep Neural Networks

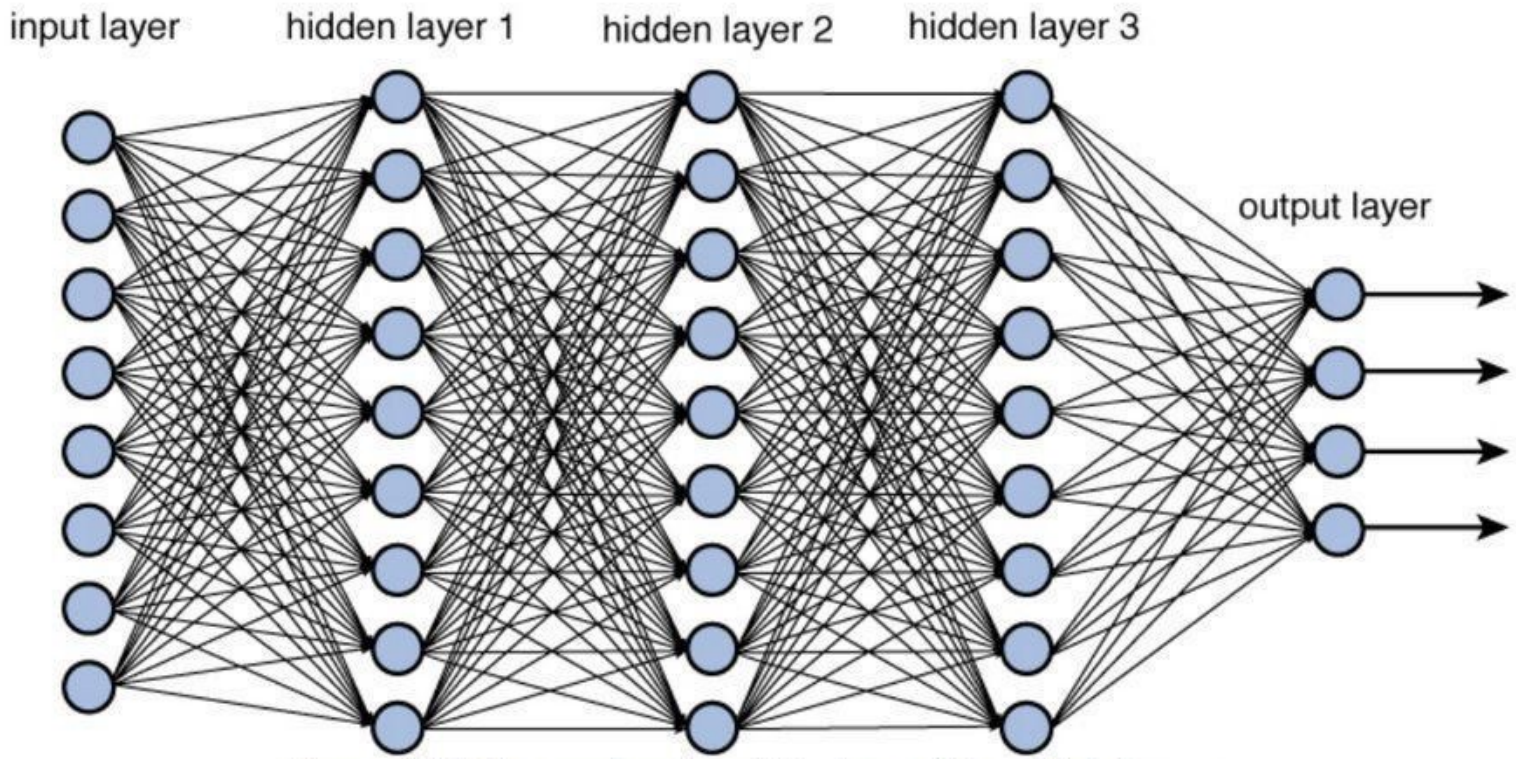
- What you think usually looks like [1]



[1] https://www.innoarchitech.com/blog/artificial-intelligence-deep-learning-neural-networks-explained

Deep Neural Networks

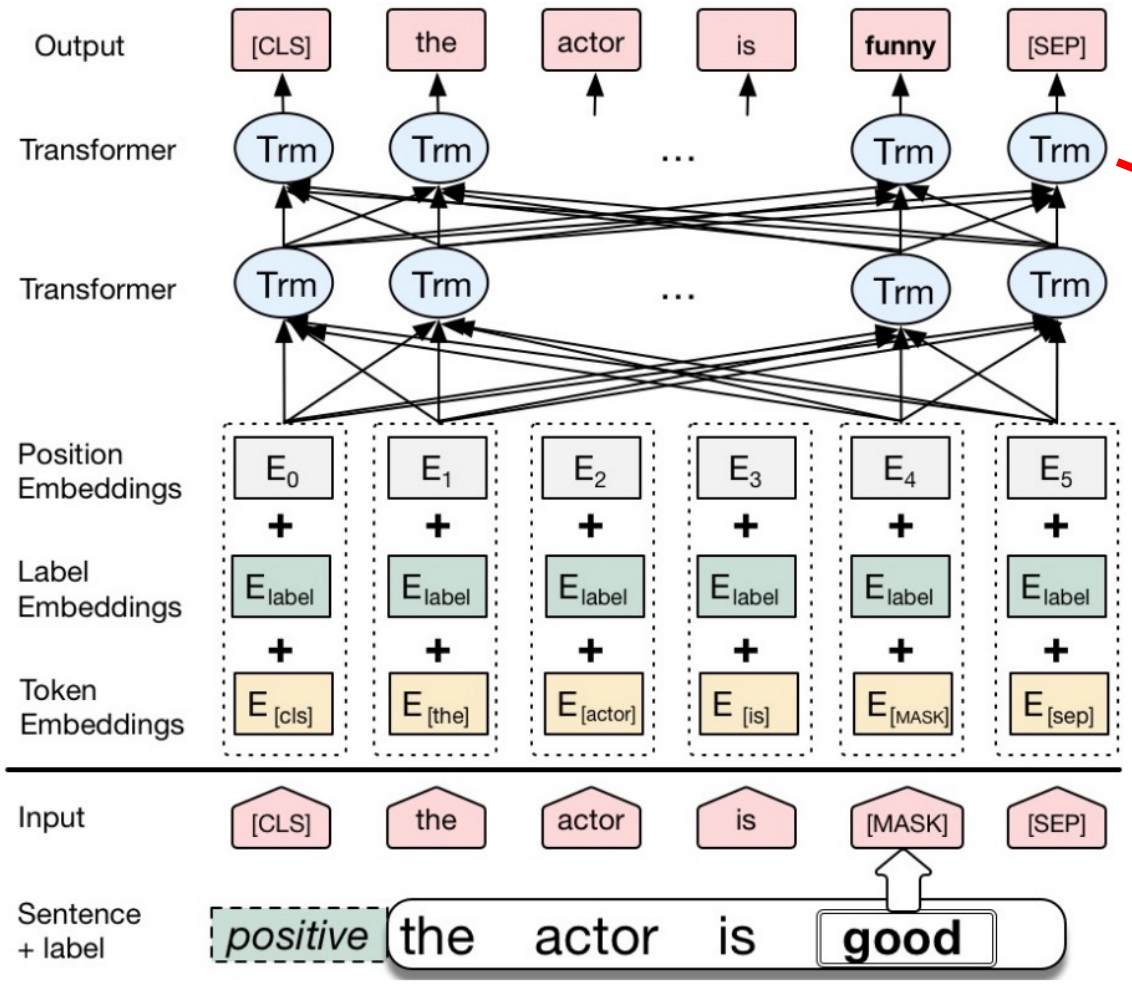
- But it could be multiple layers [1]

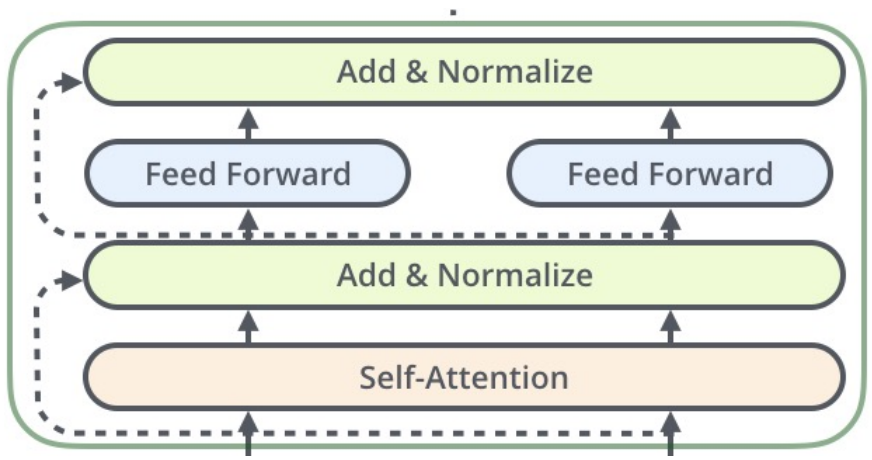


[1] https://towardsdatascience.com/training-deep-neural-networks-9fdb1964b964

Deep Neural Networks

- Even more complex [1]



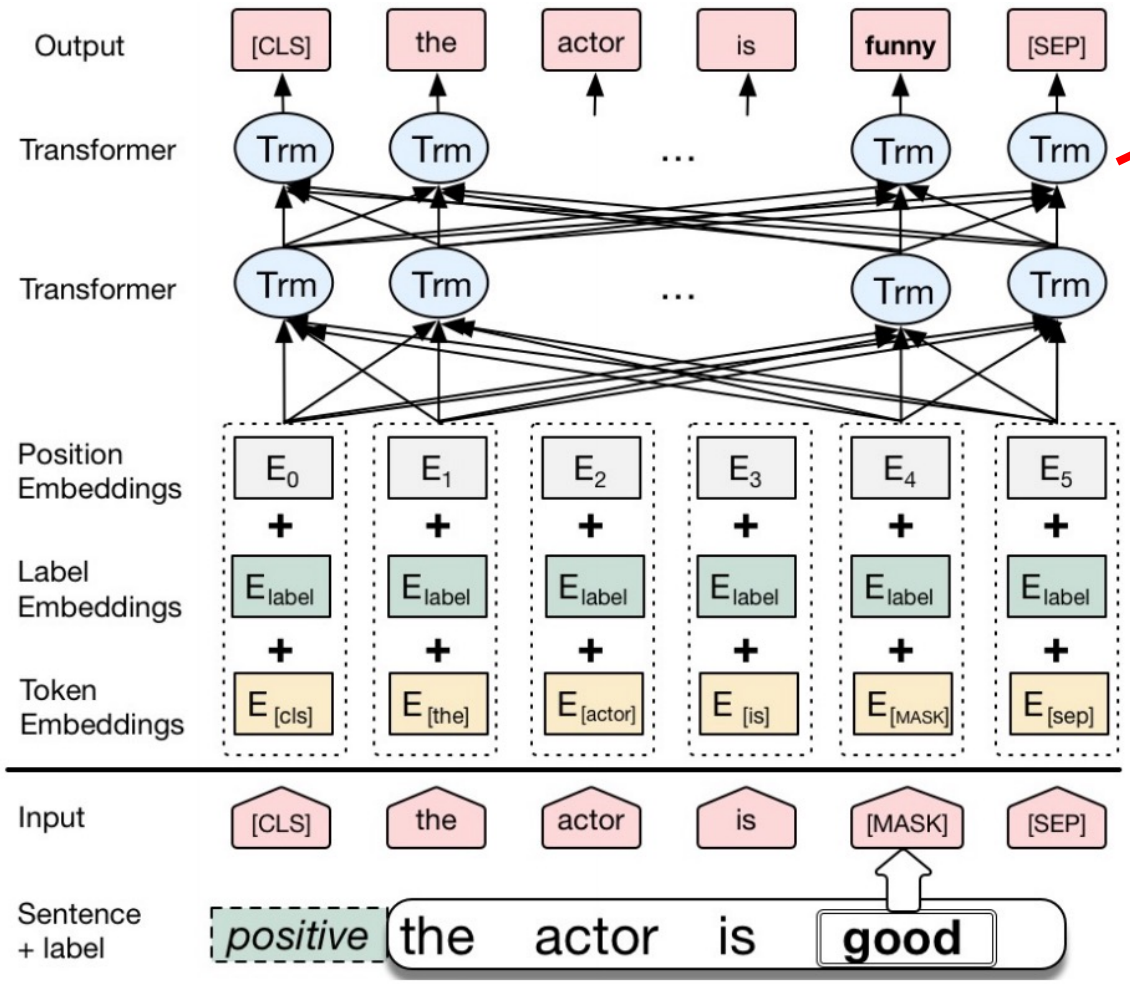


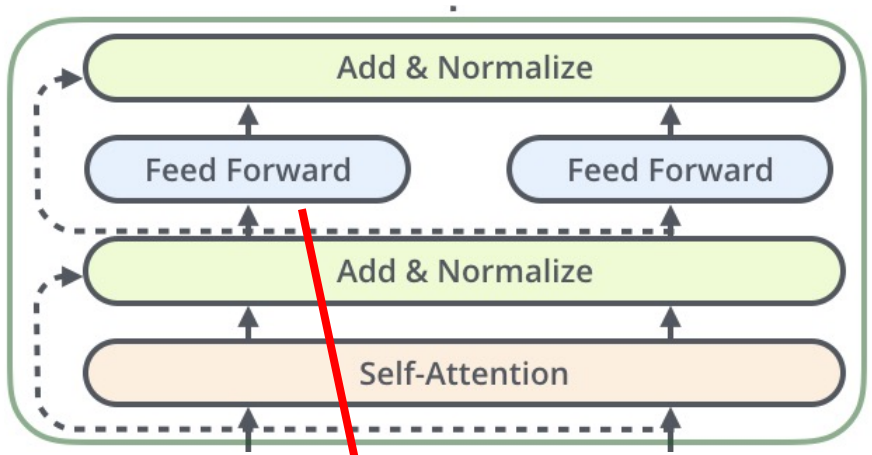
Architecture of Transformer Encoder

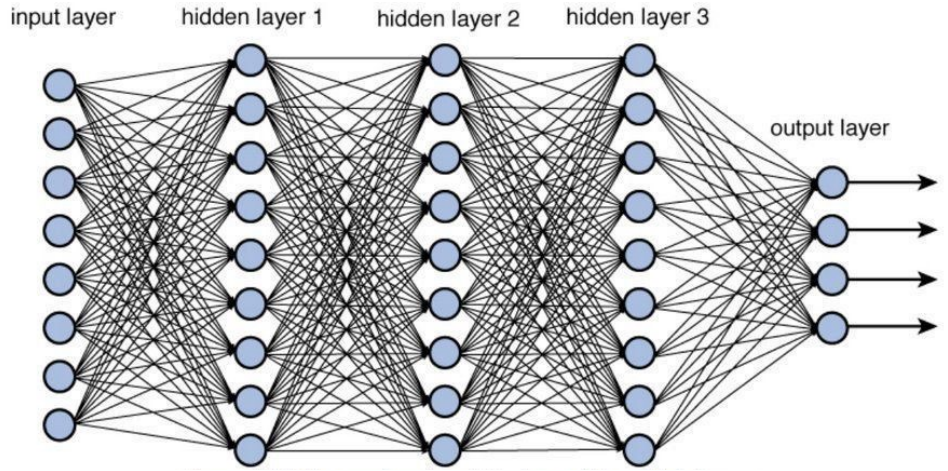
[1] Wu, X., Lv, S., Zang, L., Han, J., & Hu, S. (2019, June). Conditional bert contextual augmentation. In ICCS.

Deep Neural Networks

- Even more complex [1]







[1] Wu, X., Lv, S., Zang, L., Han, J., & Hu, S. (2019, June). Conditional bert contextual augmentation. In ICCS.

Neural Networks Are Too Large

- Evolution of deep learning models over time

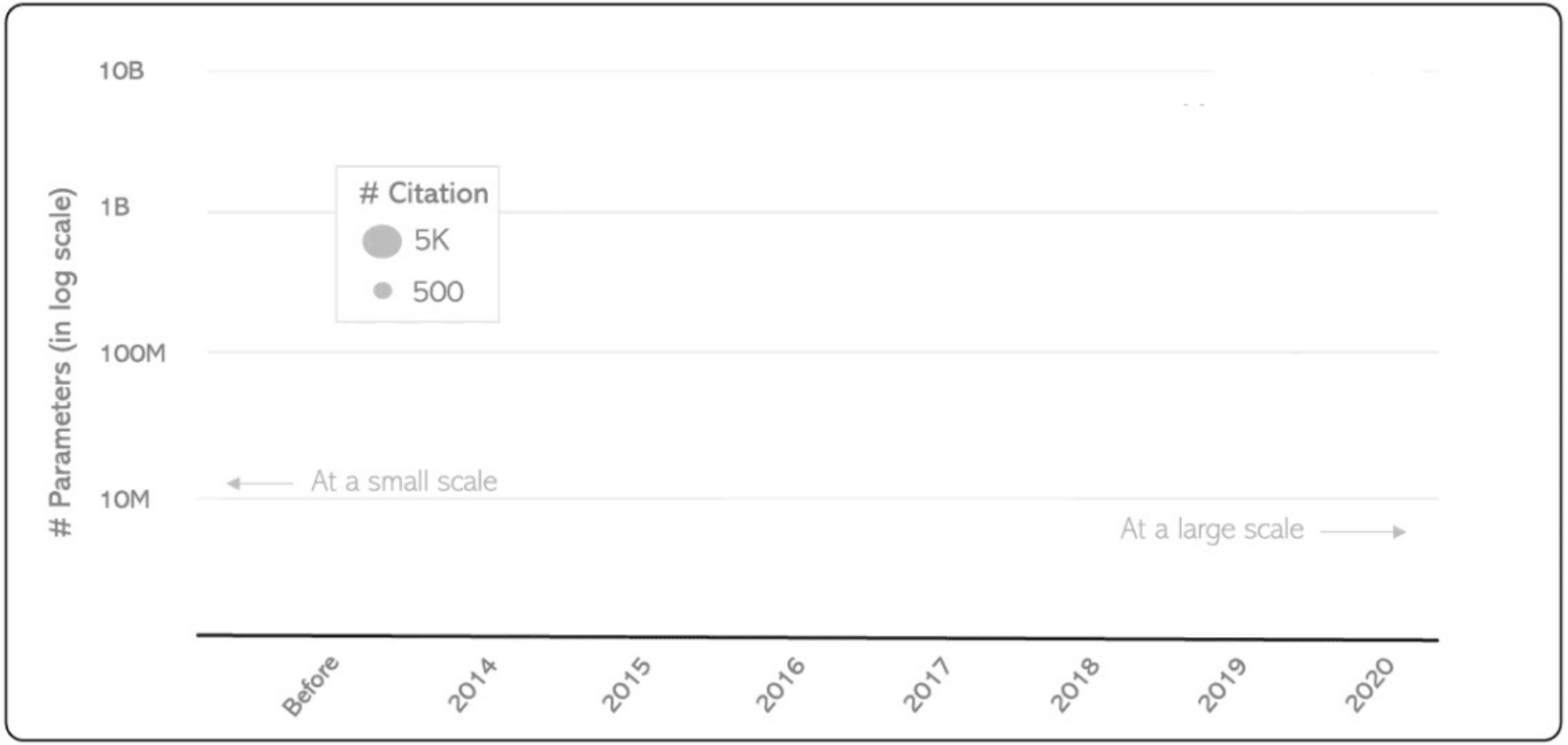


Figure: A brief evolution of deep learning models over time, measured by **model size (number of parameters)** and scientific impact (**number of citations to date**) [1]

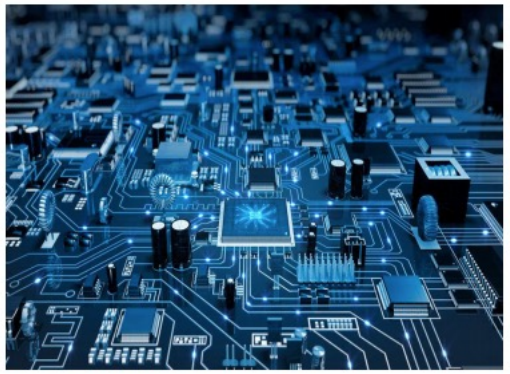
[1] https://www.microsoft.com/en-us/research/blog/a-deep-generative-model-trifecta-three-advances-that-work-towards-harnessing-large-scale-power/

Resources Are Limited

- Large neural networks require huge memory, computations, power
- Resource constrained environments[1]

Memory & Computations

Power Consumption



Embedded Systems e.g., Mobile Devices

Real-Time Tasks e.g., Autonomous Car

[1] https://www.robots.ox.ac.uk/~namhoon/doc/slides-snip-msr.pdf

Compression Is Desirable

- The goal
  - Reduce the size of network without compromising accuracy
- Pruning is a popular compression approach

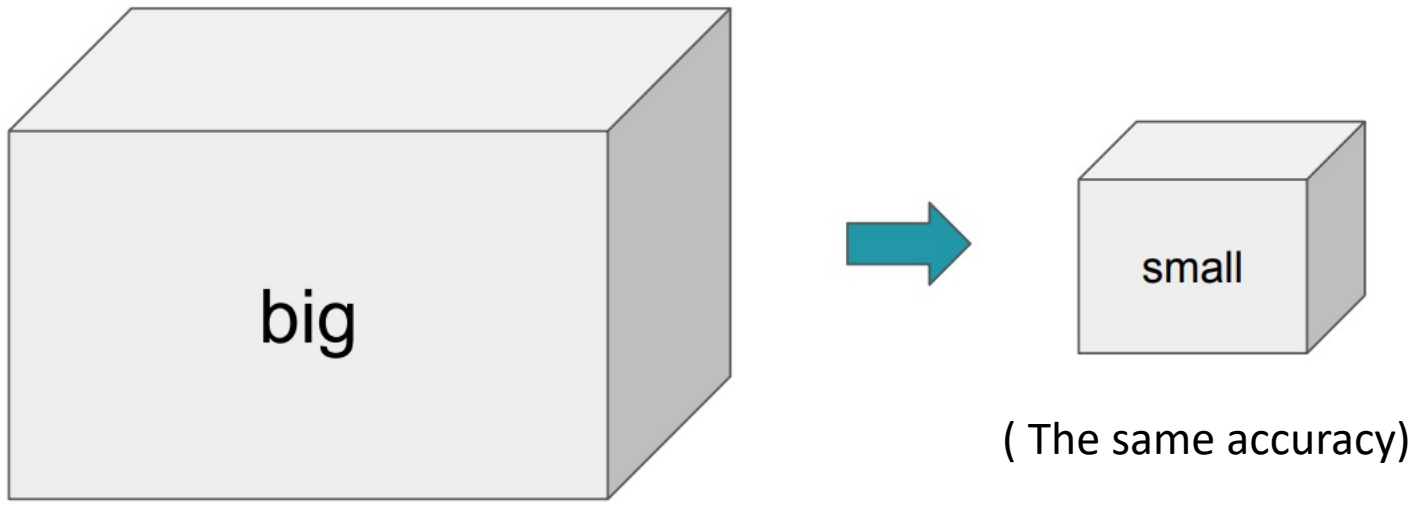


Illustration of model compression [1]

[1] https://www.robots.ox.ac.uk/~namhoon/doc/slides-snip-msr.pdf

Agenda

- Motivation
- **Neural Network Pruning**
- BERT Architecture
- Structural Pruning vs. Sparse Pruning

Neural Network Pruning

- Systematically removing parameters/connections from a network

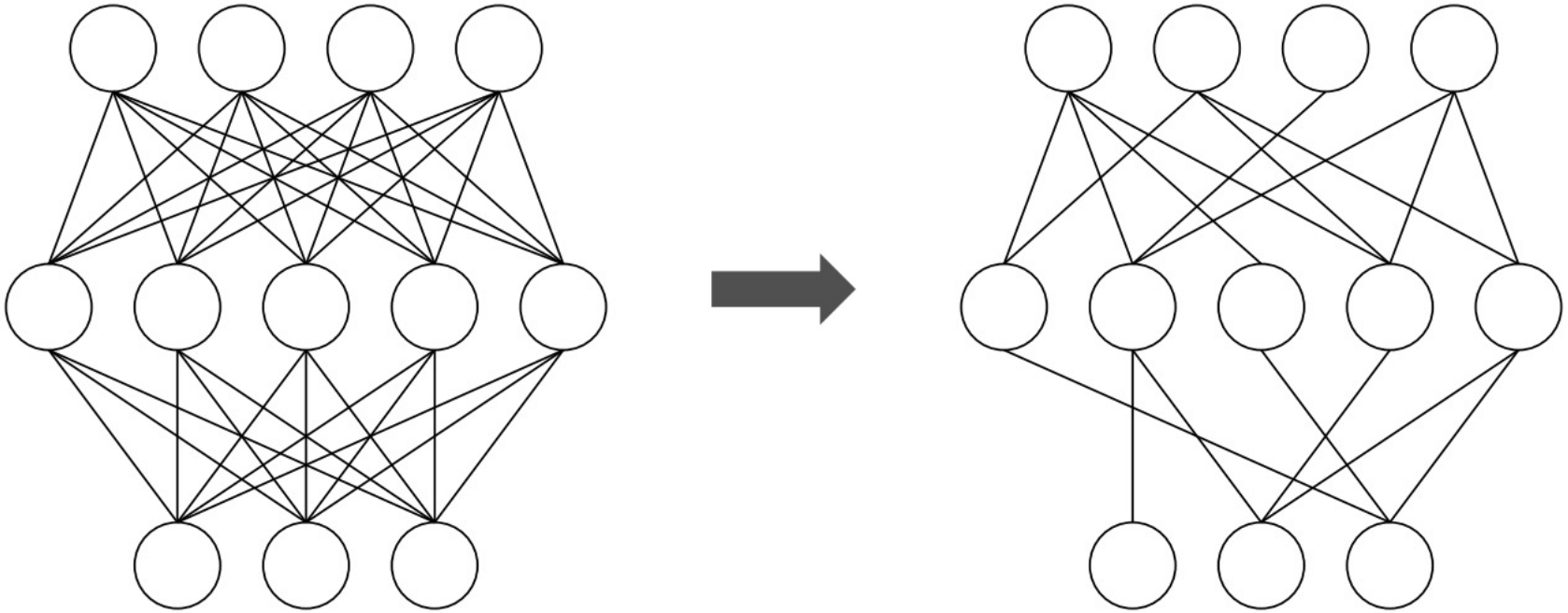


Illustration of neural network pruning [1]

[1] https://mlsys.org/media/Slides/mlsys/2020/balla(02-14-30)-02-14-30-1413-what_is_the.pdf

Typical Pruning Pipeline

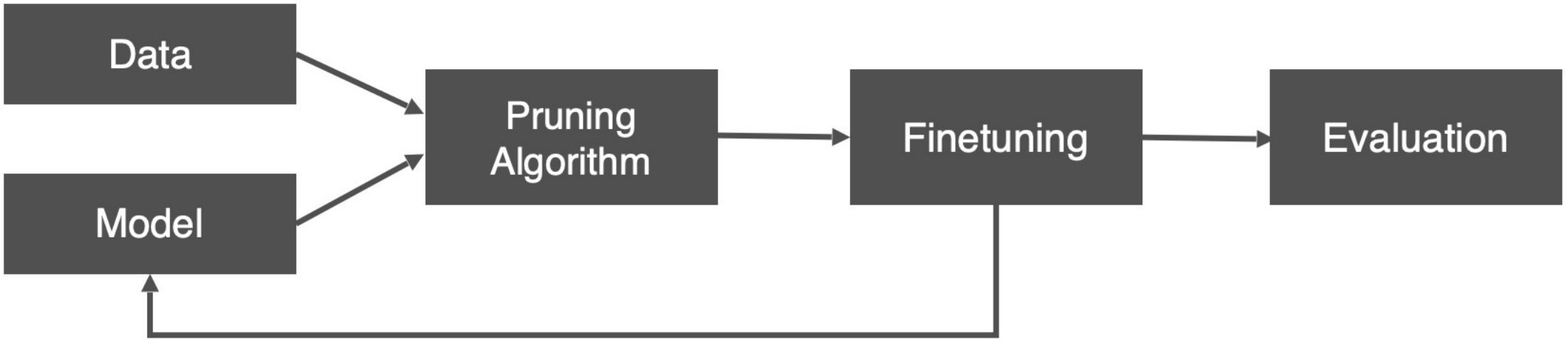


Illustration of pruning pipeline [1]

[1] https://mlsys.org/media/Slides/mlsys/2020/balla(02-14-30)-02-14-30-1413-what_is_the.pdf

Typical Pruning Pipeline: Cyclic Compression

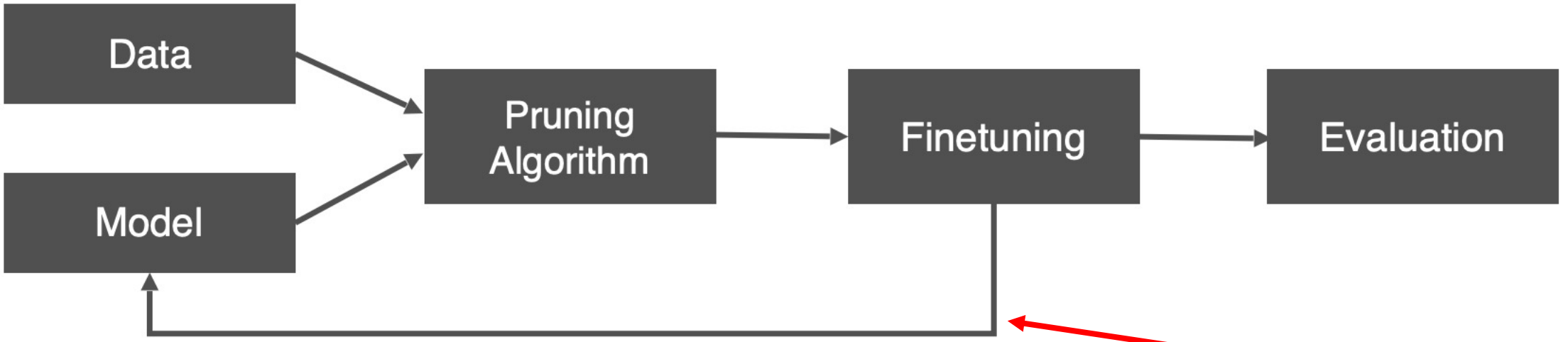


Illustration of pruning pipeline [1]

Usually, we do not compress to the compression ratio we want at once

[1] https://mlsys.org/media/Slides/mlsys/2020/balla(02-14-30)-02-14-30-1413-what_is_the.pdf

Typical Pruning Pipeline

- Many design choices
  - Scoring importance of parameters
  - Structure of induced sparsity
  - Schedule of pruning, training / finetuning

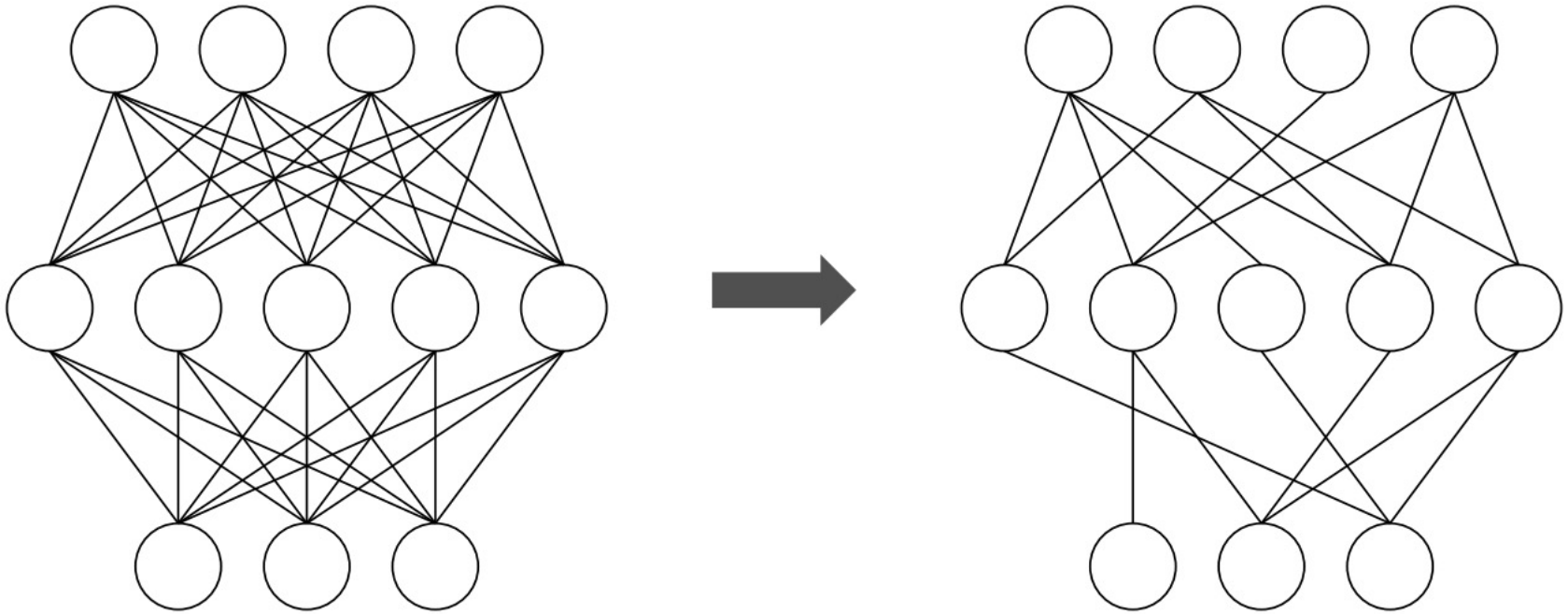
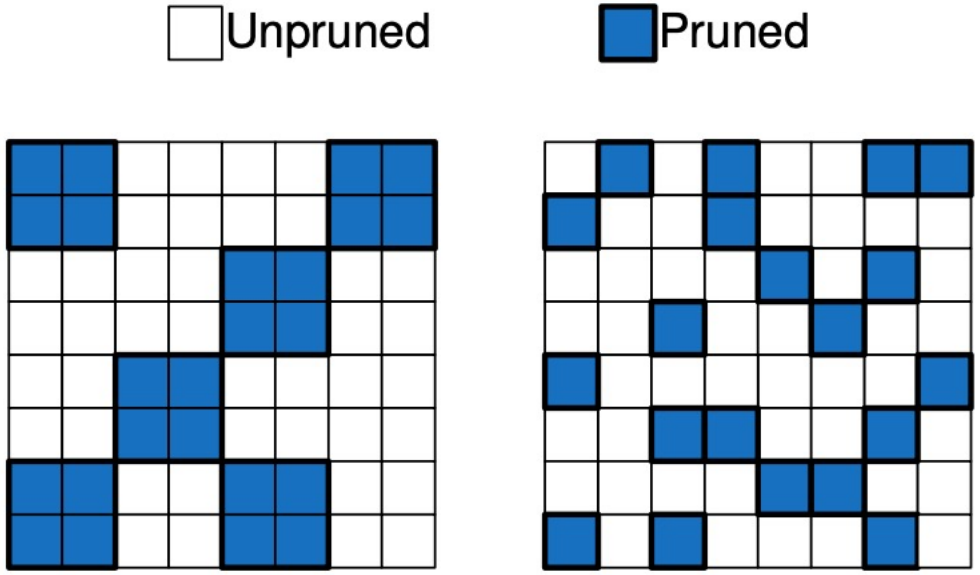


Illustration of neural network pruning

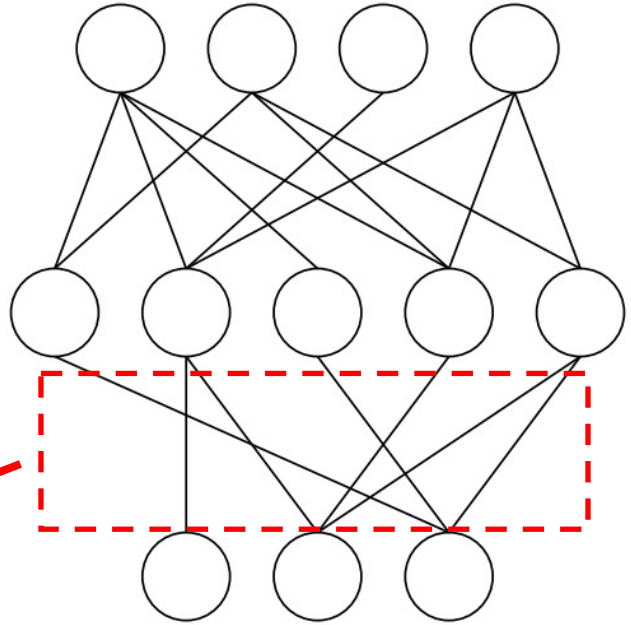
Typical Pruning Pipeline

- Many design choices
  - Scoring importance of parameters
  - Structure of induced sparsity
  - Schedule of pruning, training / finetuning



Different structures of sparsity [1]

Connections are represented by a transformation matrix



Pruned neural network

[1] Kwon, S. J., Lee, D., Kim, B., Kapoor, P., Park, B., & Wei, G. Y. (2020). Structured compression by weight encryption for unstructured pruning and quantization. In CVPR.

Typical Pruning Pipeline

- Many design choices
  - Scoring importance of parameters
  - Structure of induced sparsity
  - Schedule of pruning, training / finetuning

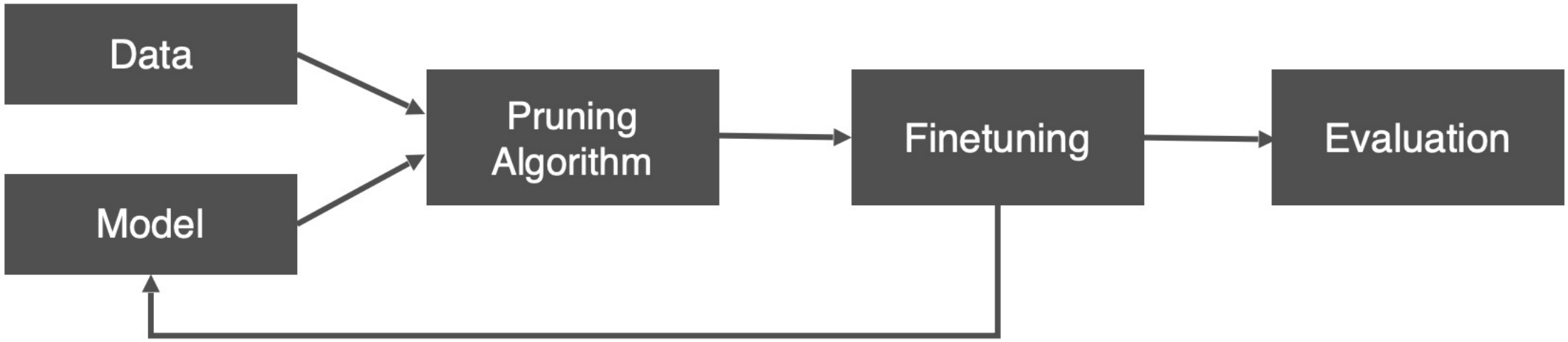


Illustration of pruning pipeline [1]

[1] https://mlsys.org/media/Slides/mlsys/2020/balla(02-14-30)-02-14-30-1413-what_is_the.pdf

Evaluating Neural Network Pruning [1]

- Goal
  - Increase efficiency of network as much as possible with **minimal drop** in quality
- Metrics
  - Quality = Accuracy
  - Efficiency = FLOPs (floating point operations per second), compression, latency
- Trade-off
  - Between **accuracy** and **compression**

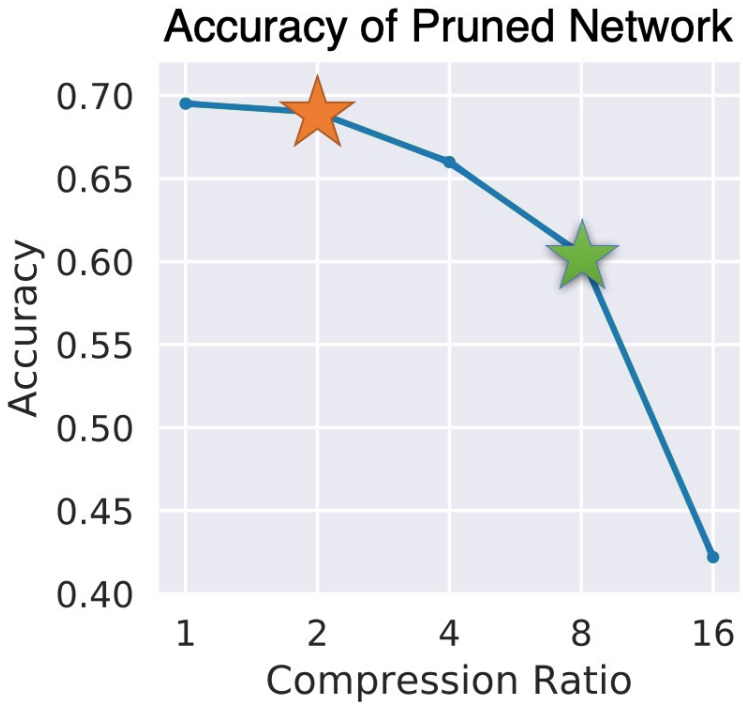
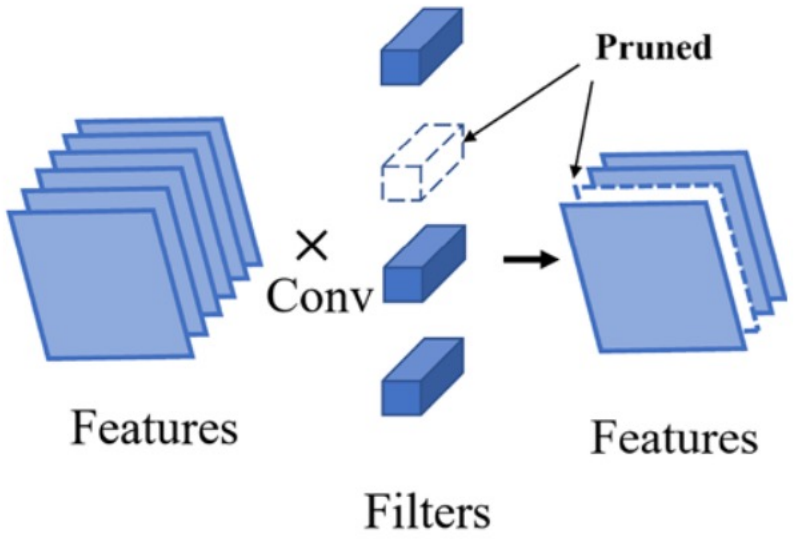


Illustration of trade-off [1]

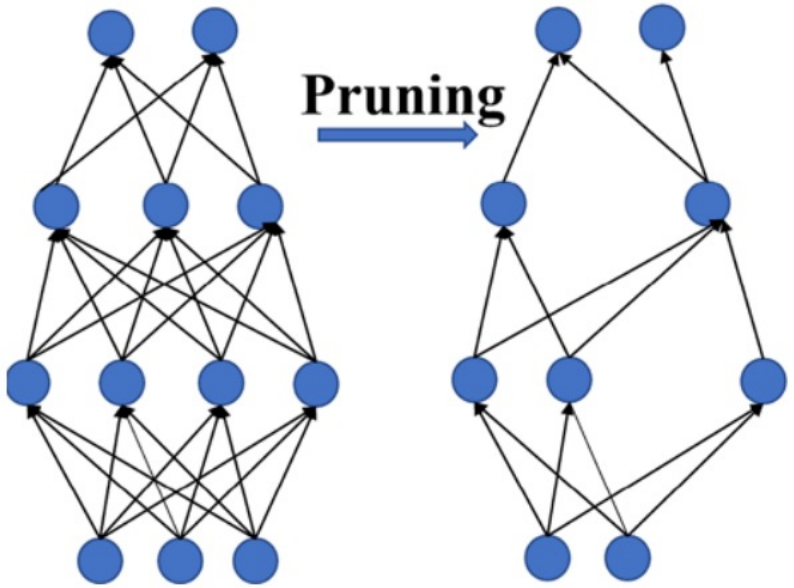
[1] https://mlsys.org/media/Slides/mlsys/2020/balla(02-14-30)-02-14-30-1413-what_is_the.pdf

Structural Pruning vs. Sparse Pruning [1]

- Structural pruning: **a channel, a layer**
- Sparse pruning: **a neuron**



Structural pruning for CNN

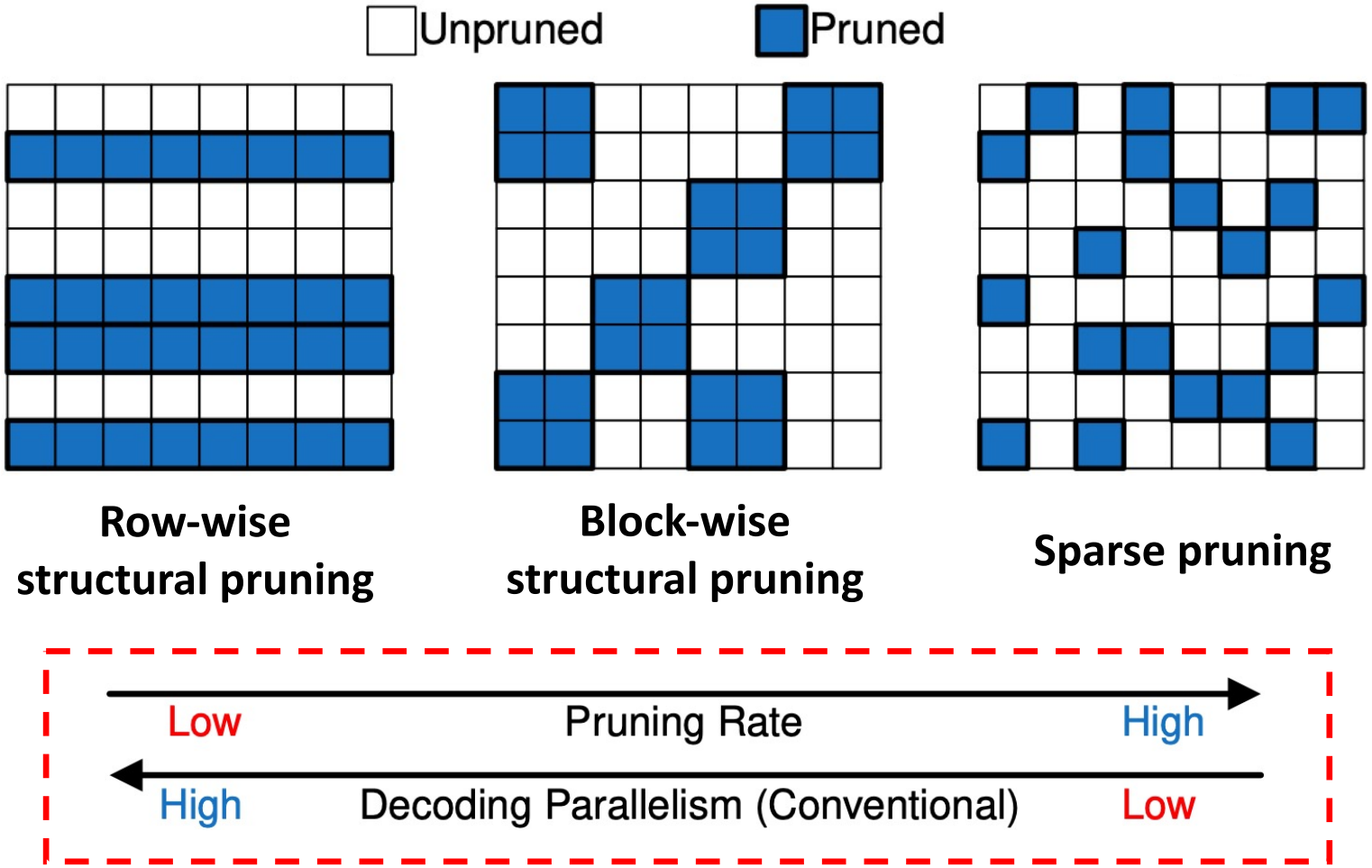


Sparse pruning for fully connected networks

[1] Chen, L., Chen, Y., Xi, J., & Le, X. (2021). Knowledge from the original network: restore a better pruned network with knowledge distillation. Complex & Intelligent Systems, 1-10.

Structural Pruning vs. Sparse Pruning [1]

- Structural pruning: a group of neurons
- Sparse pruning: a neuron



[1] Kwon, S. J., Lee, D., Kim, B., Kapoor, P., Park, B., & Wei, G. Y. (2020). Structured compression by weight encryption for unstructured pruning and quantization. In CVPR.

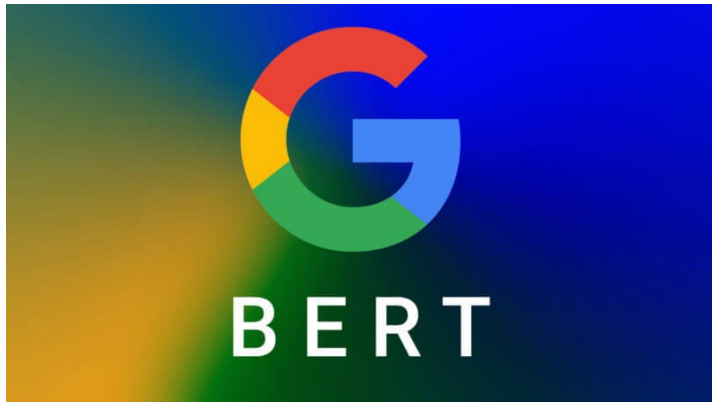
Agenda

- Motivation
- Neural Network Pruning
- **BERT Architecture**
- Structural Pruning vs. Sparse Pruning

Background

- BERT (Bidirectional Encoder Representations from Transformers)
  - Published by Google AI Language [1]
  - Achieved state-of-the-art results in various NLP/CV tasks

- Key Innovation
  - **Bidirectional training** for language modelling
  - **Previous efforts** looked at a text sequence from **left-to-right or right-to-left**

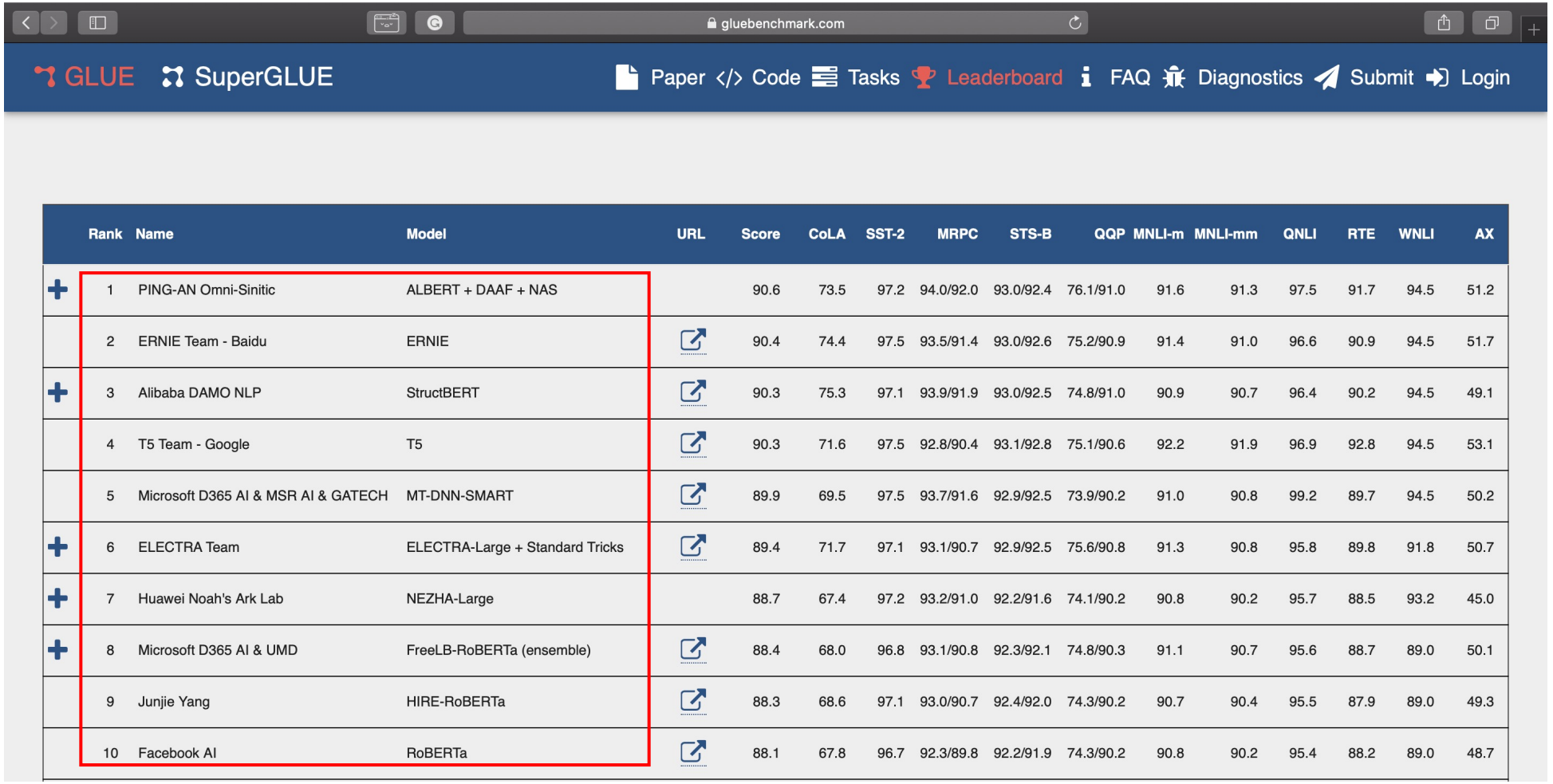


[1] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*

Background

- Performance on **GLUE** (General Language Understanding Evaluation) Benchmark
  - The most popular collection for training, evaluating and analyzing NLP systems [1]
  - Constructed by NYU, UW and DeepMind

Support BERT
-> Support All

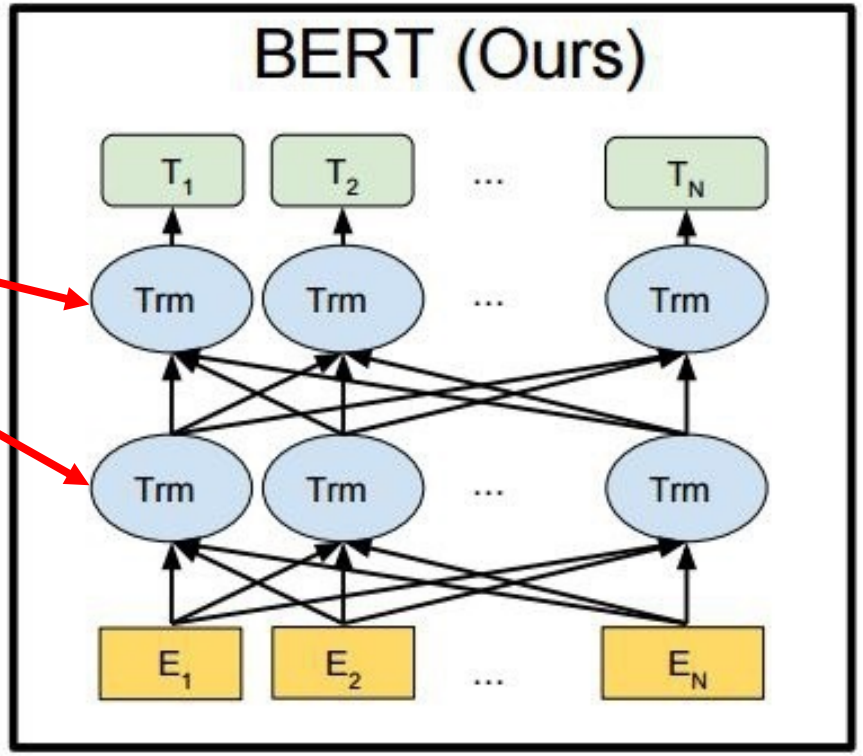


Rank	Name	Model	URL	Score	CoLA	SST-2	MRPC	STS-B	QQP	MNLI-m	MNLI-mm	QNLI	RTE	WNLI	AX
1	PING-AN Omni-Sinitic	ALBERT + DAAF + NAS		90.6	73.5	97.2	94.0/92.0	93.0/92.4	76.1/91.0	91.6	91.3	97.5	91.7	94.5	51.2
2	ERNIE Team - Baidu	ERNIE		90.4	74.4	97.5	93.5/91.4	93.0/92.6	75.2/90.9	91.4	91.0	96.6	90.9	94.5	51.7
3	Alibaba DAMO NLP	StructBERT		90.3	75.3	97.1	93.9/91.9	93.0/92.5	74.8/91.0	90.9	90.7	96.4	90.2	94.5	49.1
4	T5 Team - Google	T5		90.3	71.6	97.5	92.8/90.4	93.1/92.8	75.1/90.6	92.2	91.9	96.9	92.8	94.5	53.1
5	Microsoft D365 AI & MSR AI & GATECH	MT-DNN-SMART		89.9	69.5	97.5	93.7/91.6	92.9/92.5	73.9/90.2	91.0	90.8	99.2	89.7	94.5	50.2
6	ELECTRA Team	ELECTRA-Large + Standard Tricks		89.4	71.7	97.1	93.1/90.7	92.9/92.5	75.6/90.8	91.3	90.8	95.8	89.8	91.8	50.7
7	Huawei Noah's Ark Lab	NEZHA-Large		88.7	67.4	97.2	93.2/91.0	92.2/91.6	74.1/90.2	90.8	90.2	95.7	88.5	93.2	45.0
8	Microsoft D365 AI & UMD	FreeLB-RoBERTa (ensemble)		88.4	68.0	96.8	93.1/90.8	92.3/92.1	74.8/90.3	91.1	90.7	95.6	88.7	89.0	50.1
9	Junjie Yang	HIRE-RoBERTa		88.3	68.6	97.1	93.0/90.7	92.4/92.0	74.3/90.2	90.7	90.4	95.5	87.9	89.0	49.3
10	Facebook AI	RoBERTa		88.1	67.8	96.7	92.3/89.8	92.2/91.9	74.3/90.2	90.8	90.2	95.4	88.2	89.0	48.7

[1] *https://gluebenchmark.com*

Overall

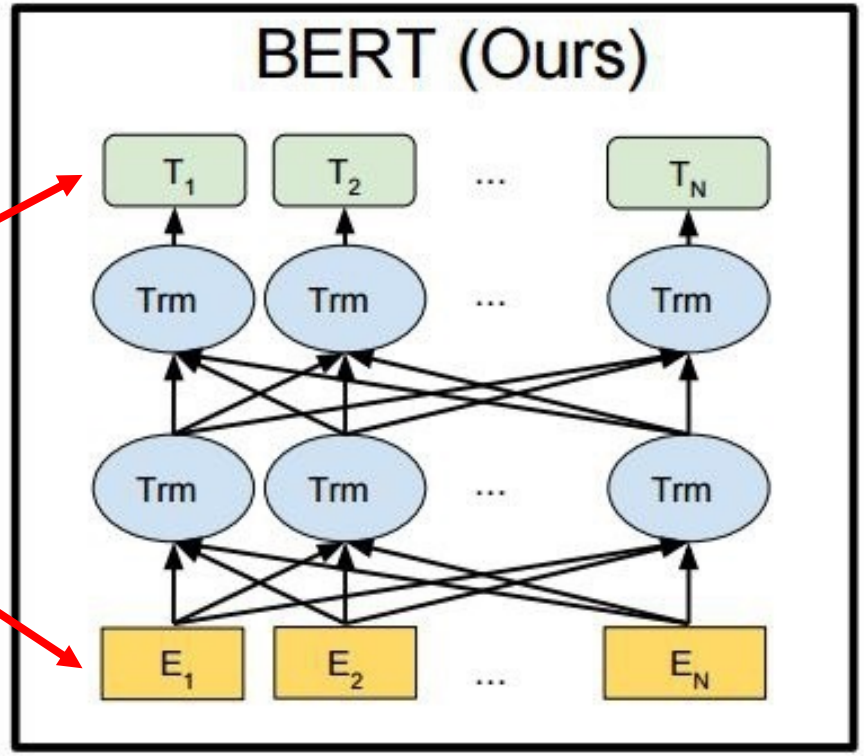
- General architecture
  - Multiple Transformer encoders
  - Input: Embeddings of words
  - Output: Hidden representations of words
- Downstream task
  - e.g., sentence classification
- How BERT works
  - Pre-training
    - The model is trained on unlabeled data over different pre-training tasks.
  - Fine-tuning
    - The model is first initialized with the pre-trained parameters, and all the parameters are fine-tuned using labeled data from the downstream tasks.



Architecture of BERT

Overall

- General architecture
  - Multiple Transformer encoders
  - Input: Embeddings of words
  - Output: Hidden representations of words



Architecture of BERT

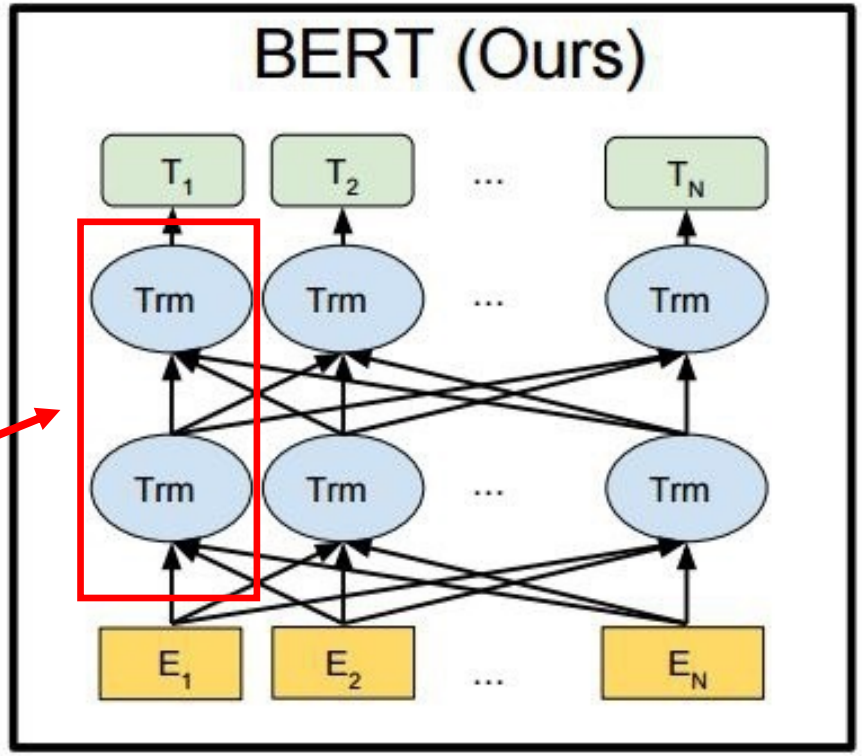
- Downstream task
  - e.g., sentence classification

- How BERT works
  - Pre-training
    - The model is trained on unlabeled data over different pre-training tasks.
  - Fine-tuning
    - The model is first initialized with the pre-trained parameters, and all the parameters are fine-tuned using labeled data from the downstream tasks.

Overall

- General architecture
  - Multiple Transformer encoders
  - Input: Embeddings of words
  - Output: Hidden representations of words
- Downstream task
  - e.g., sentence classification

1. Only one series of encoders
2. Shared by all word embeddings



Architecture of BERT

- How BERT works
  - Pre-training
    - The model is trained on unlabeled data over different pre-training tasks.
  - Fine-tuning
    - The model is first initialized with the pre-trained parameters, and all the parameters are fine-tuned using labeled data from the downstream tasks.

Overall

- General architecture
  - Multiple Transformer encoders
  - Input: Embeddings of words
  - Output: Hidden representations of words

- Downstream task
  - e.g., sentence classification

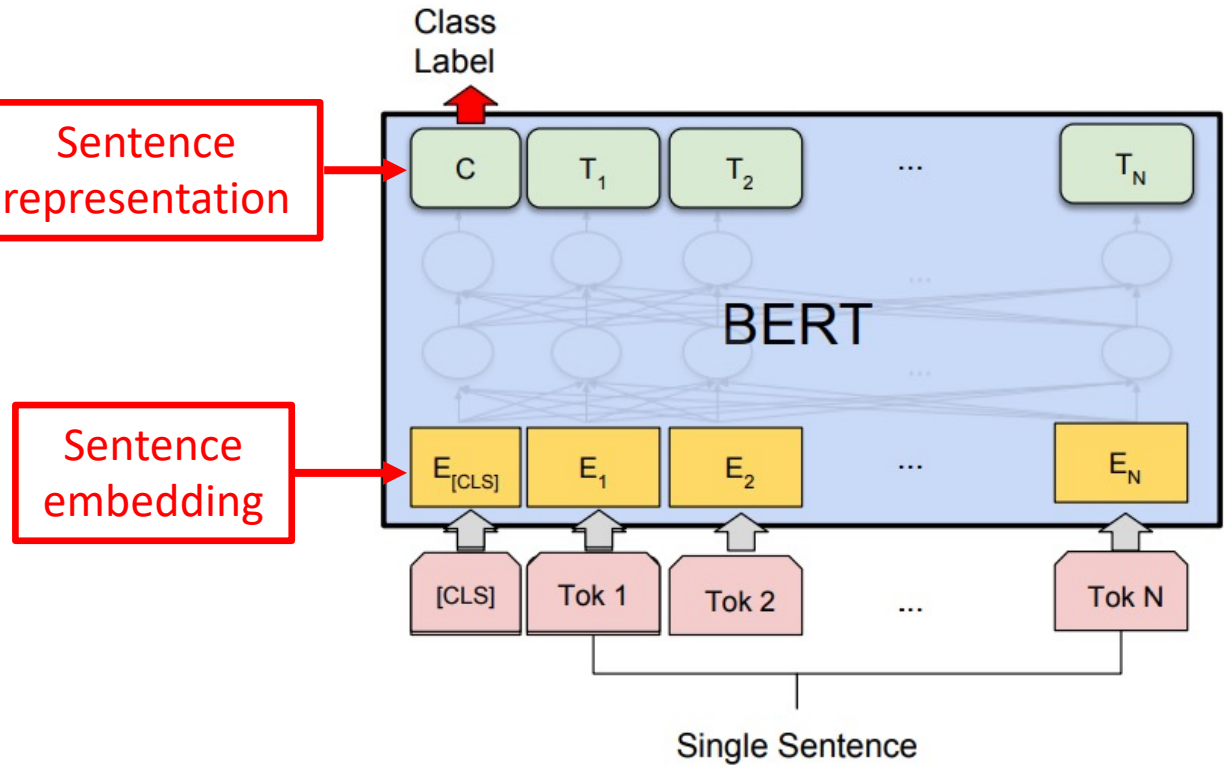


Illustration of BERT on classification

- How BERT works
  - Pre-training
    - The model is trained on unlabeled data over different pre-training tasks.
  - Fine-tuning
    - The model is first initialized with the pre-trained parameters, and all the parameters are fine-tuned using labeled data from the downstream tasks.

Overall

- General architecture
  - Multiple Transformer encoders
  - Input: Embeddings of words
  - Output: Hidden representations of words

- Downstream task
  - e.g., sentence classification

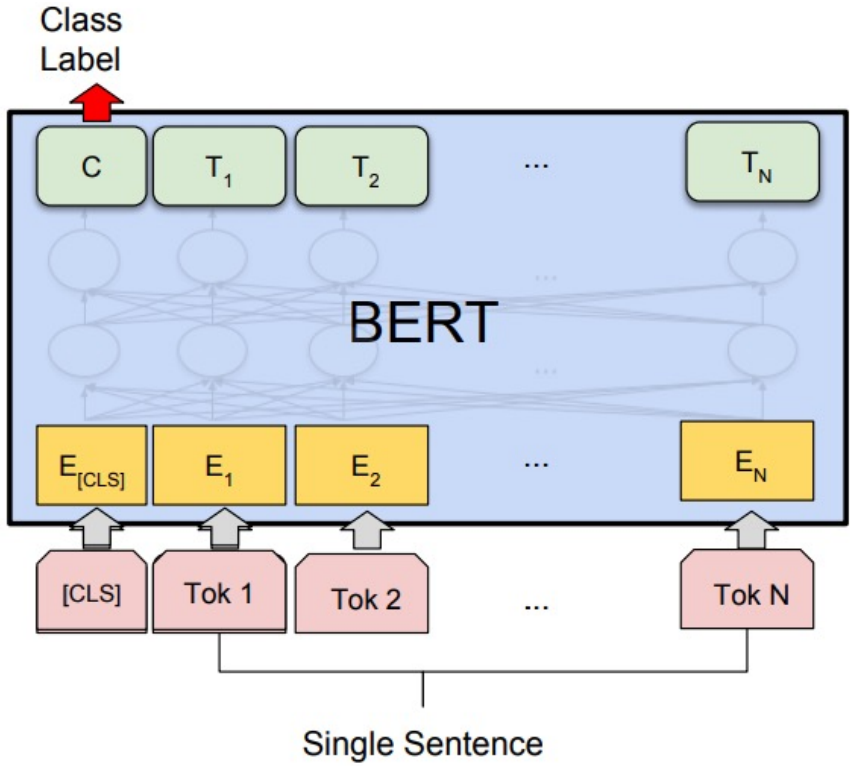
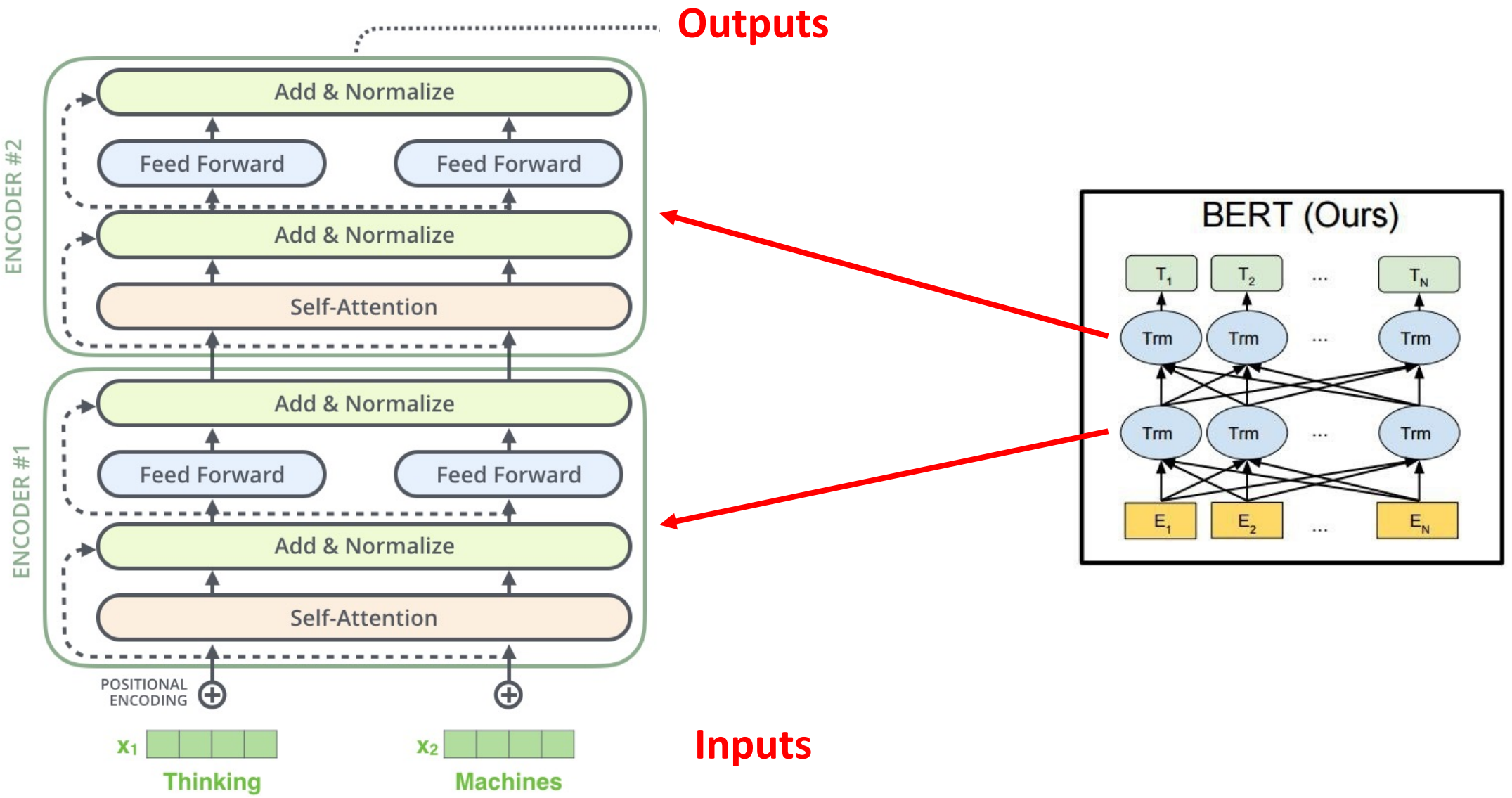


Illustration of BERT on classification

- How BERT works
  - Pre-training
    - The model is trained on unlabeled data over different pre-training tasks.
  - Fine-tuning
    - The model is first initialized with the pre-trained parameters, and all the parameters are fine-tuned using labeled data from the downstream tasks.

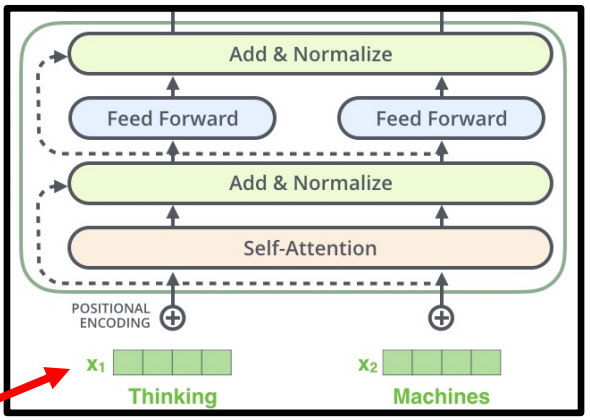
Transformer Encoder



Architecture of Encoder (Two Encoders Here)

Embedding Layer

- Embedding is the element-wise sum of three embeddings
- Goal
  - To give the model a sense of the order of the words



Architecture of Transformer Encoder

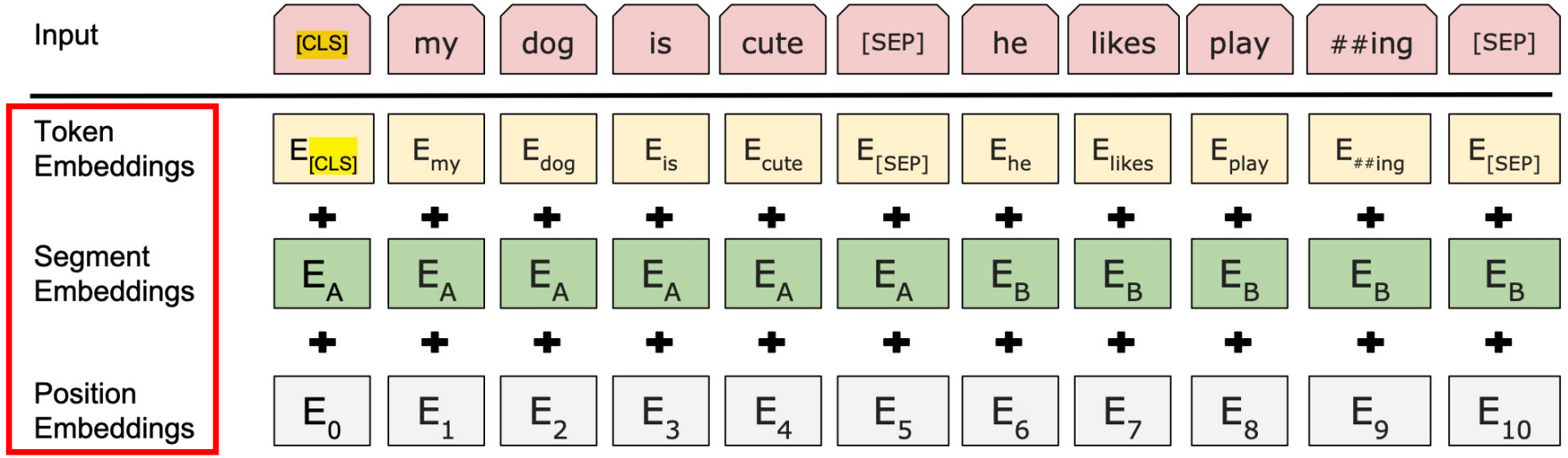
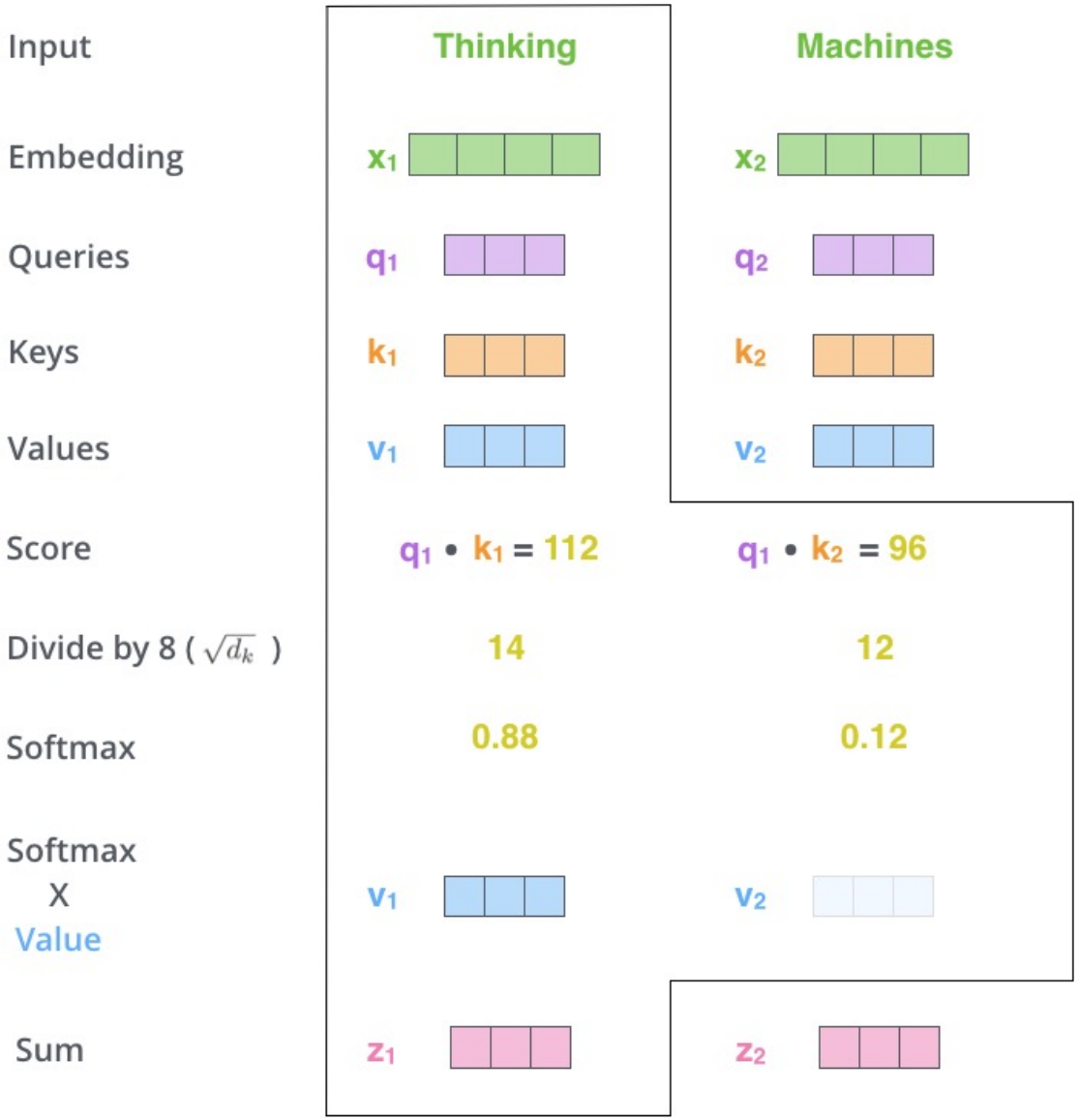
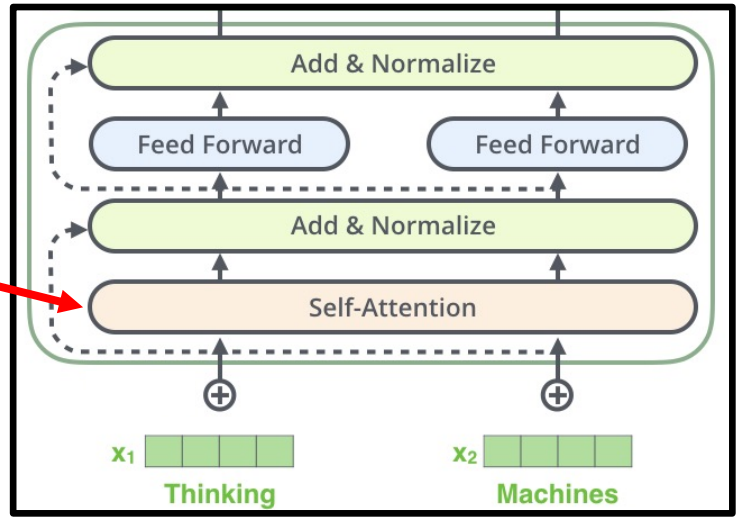


Illustration of BERT Input Representation

Self-Attention

- Take two words as an example

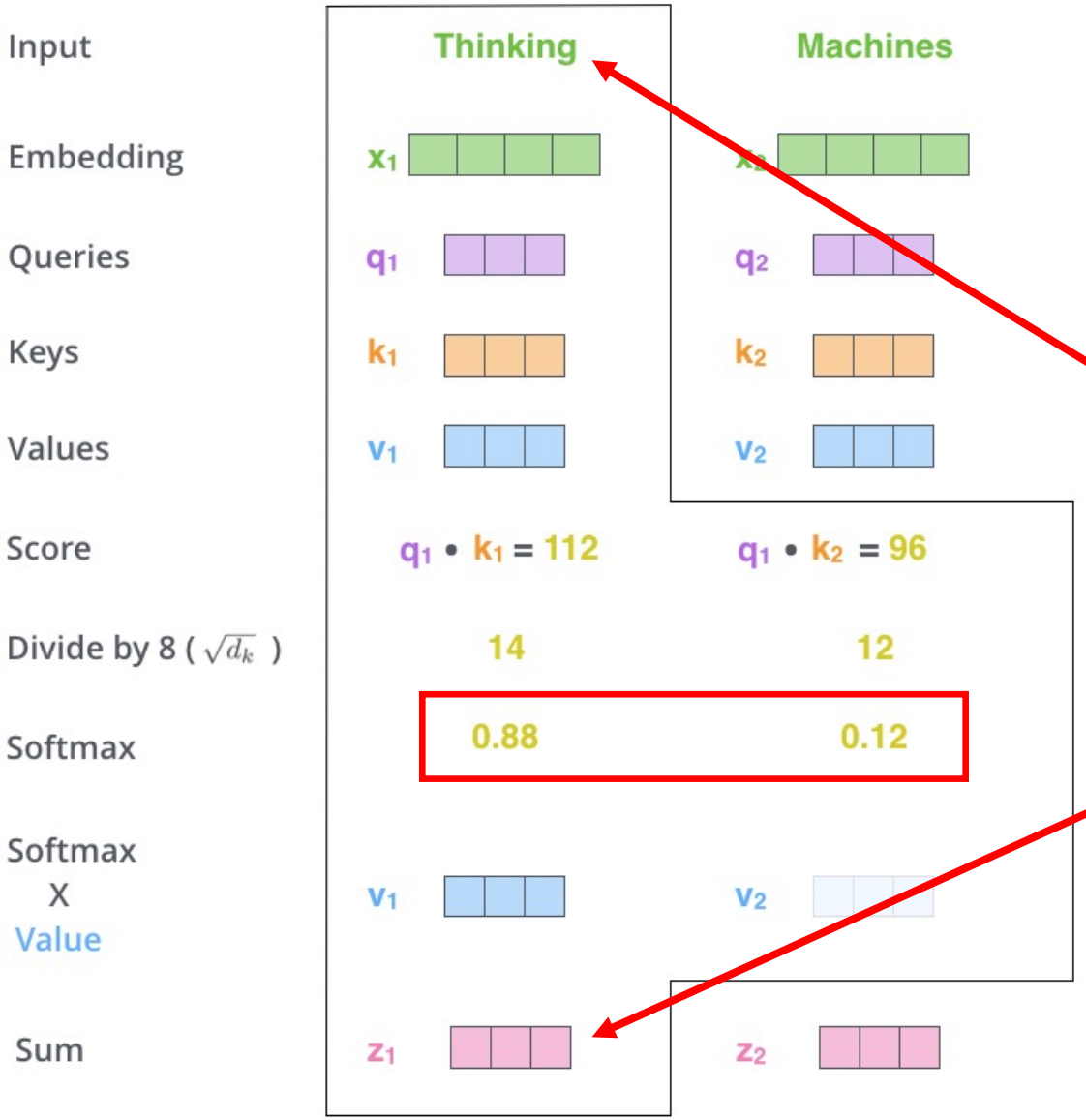


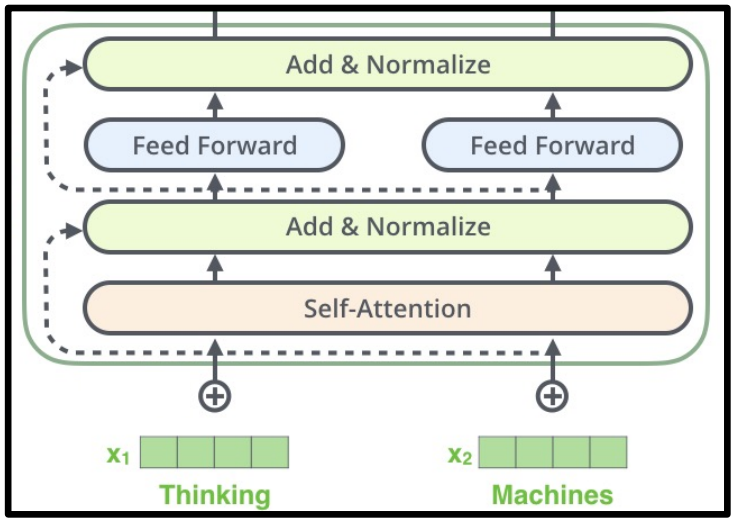


Architecture of Transformer Encoder

Self-Attention

- Take two words as an example



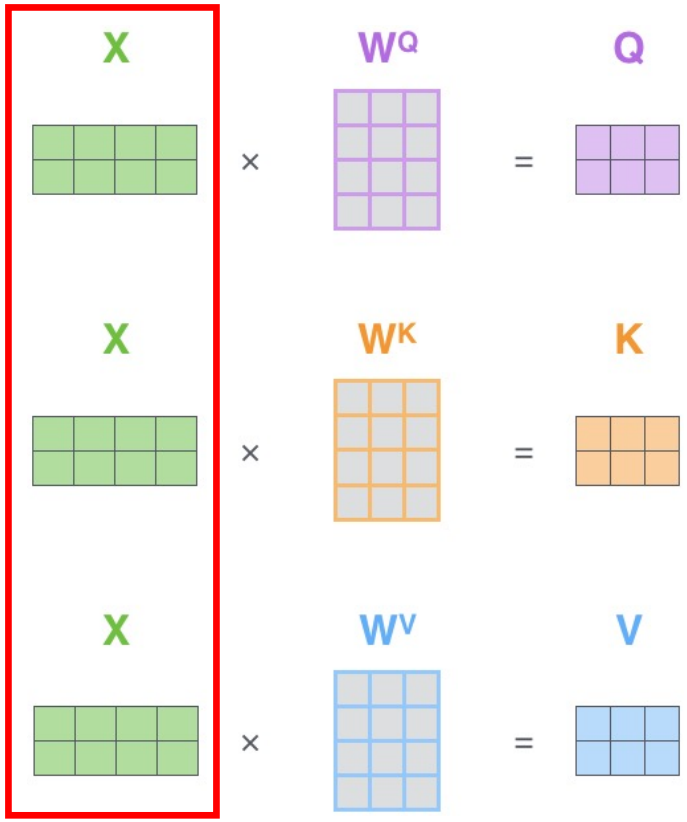


Architecture of Transformer Encoder

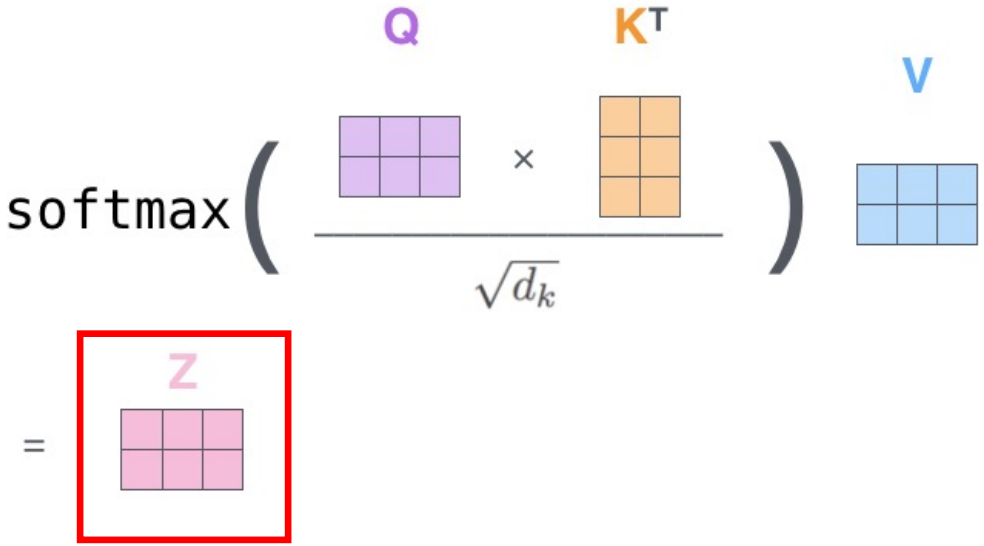
Input: Word embedding vectors
Output: New vector representations

Self-Attention in Matrix Calculation

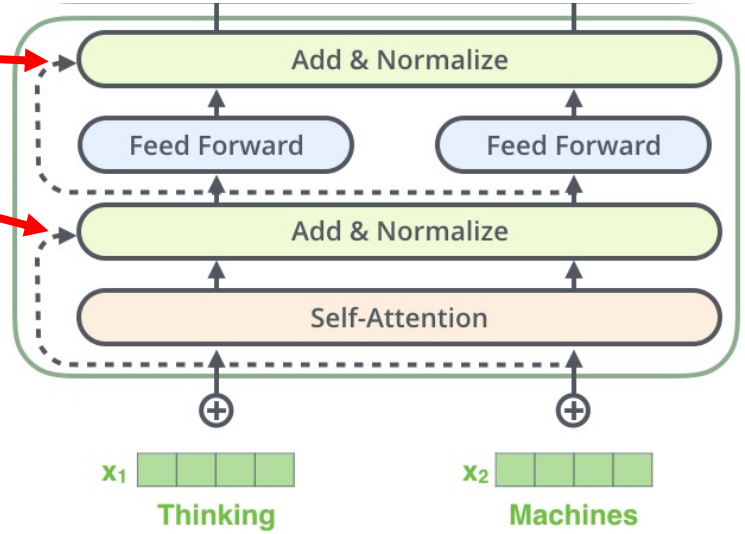
- First step to calculate ***Q, K, V***
  - Rows: ***words of a sentence***
  - Columns: ***hidden_dims***



- Second step to calculate the output ***Z***
  - Rows: ***words of a sentence***
  - Columns: ***hidden_dims***



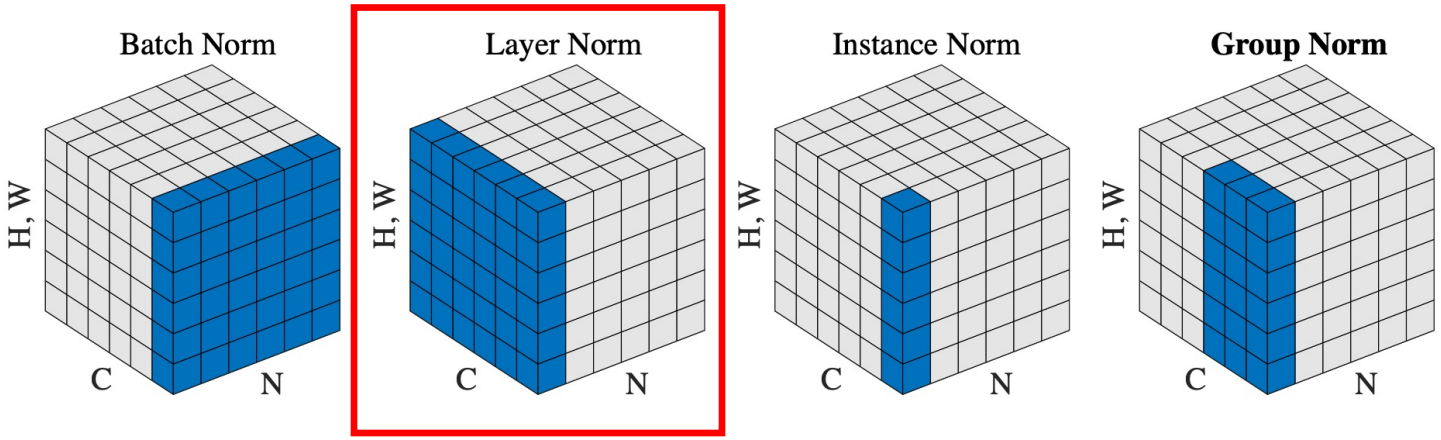
Layer Normalization [1]



- Motivations
  - 1. Dynamic length
  - 2. Different meaning of the same position
- Formula

$$y = \frac{x - \mathrm{E}[x]}{\sqrt{\mathrm{Var}[x] + \epsilon}} * \gamma + \beta$$

- Comparison between four normalizations [2]



[1] Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.

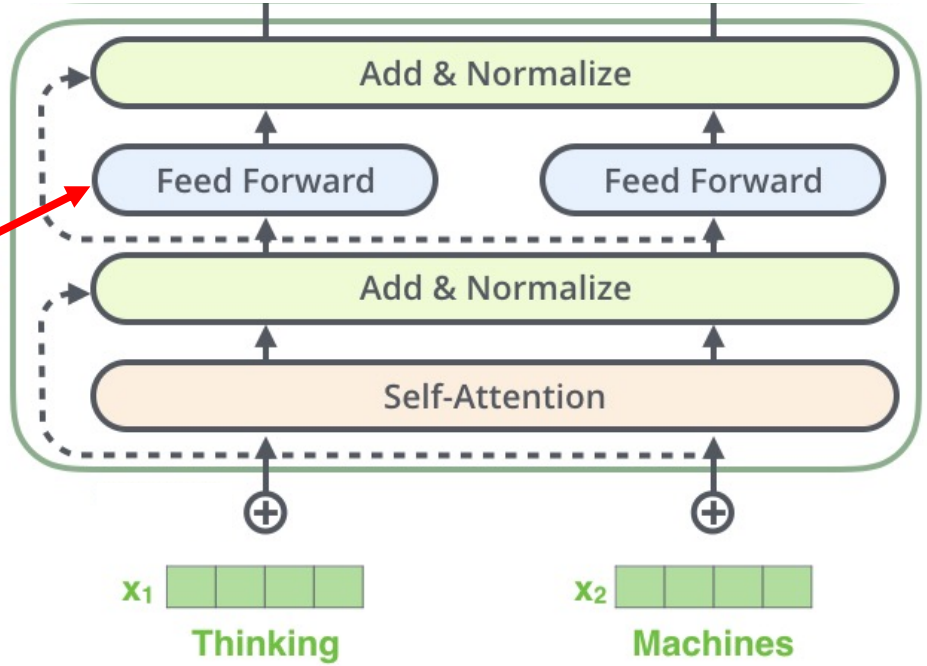
[2] Wu, Y., & He, K. (2018). Group normalization. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 3-19).

Feed-Forward Networks

- Architecture
  - Two linear transformations with a ReLU in between
  - Dimension of input and output =512
  - Dimension of inner-layer = 512*4

- Formula

$$\mathrm{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$



Other Transformer/BERT NLP Models

- **XLNet (CMU + Google AI)**
  - Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., & Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems* (pp. 5754-5764).
- **ALBERT (Google Language)**
  - Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942.
- **RoBERTa (Facebook AI)**
  - Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.
- **Transformer-XL (CMU + Google Brain)**
  - Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2019). Transformer-xl: Attentive language models beyond a fixed-length context. arXiv preprint arXiv:1901.02860.
- **ERNIE (Baidu)**
  - Sun, Y., Wang, S., Li, Y., Feng, S., Chen, X., Zhang, H., ... & Wu, H. (2019). Ernie: Enhanced representation through knowledge integration. *arXiv preprint arXiv:1904.09223*.
- **GPT-2 (OpenAI)**
  - Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI Blog, 1(8), 9.

Novel CV Applications using Transformers/BERT [1]

🏞️ Transformer for Image Synthesis - 🔗 Esser et al. (2020)

🚨 Transformer for Multi-Object Tracking - 🔗 Sun et al. (2020)

🎶 Transformer for Music Generation - 🔗 Hsiao et al. (2021)

💃 Transformer for Dance Generation with Music - 🔗 Huang et al. (2021)

🔮 Transformer for 3D Object Detection - 🔗 Bhattacharyya et al. (2021)

🗿 Transformer for Point-Cloud Processing - 🔗 Guo et al. (2020)

⏰ Transformer for Time-Series Forecasting - 🔗 Lim et al. (2020)

👁️ Transformer for Vision-Language Modeling - 🔗 Zhang et al. (2021)

🛣️ Transformer for Lane Shape Prediction - 🔗 Liu et al. (2020)

🌠 Transformer for End-to-End Object Detection - 🔗 Zhu et al. (2021)

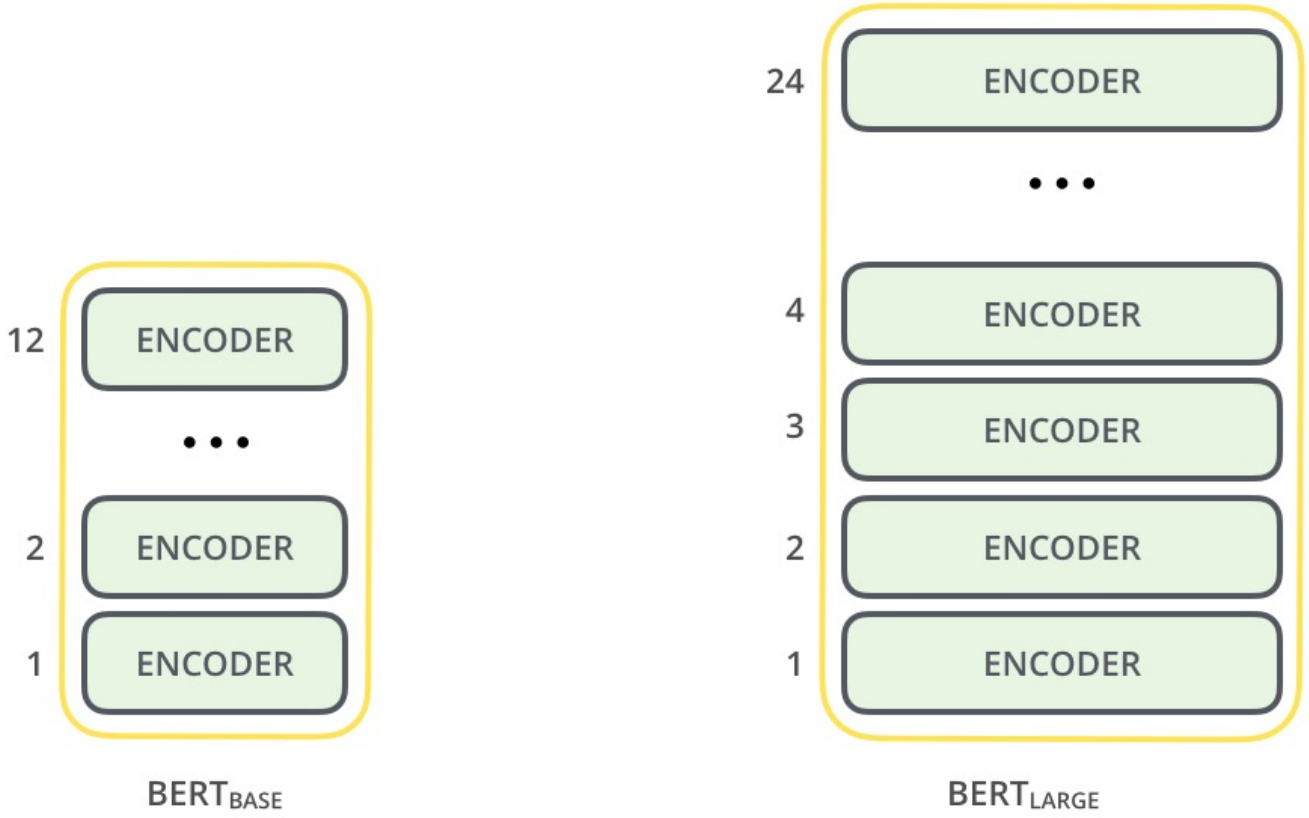
[1] https://paperswithcode.com/newsletter/3

Agenda

- Motivation
- Neural Network Pruning
- BERT Architecture
- **Structural Pruning vs. Sparse Pruning**

$\text{BERT}_{\text{Base}}$ *vs.* $\text{BERT}_{\text{Large}}$

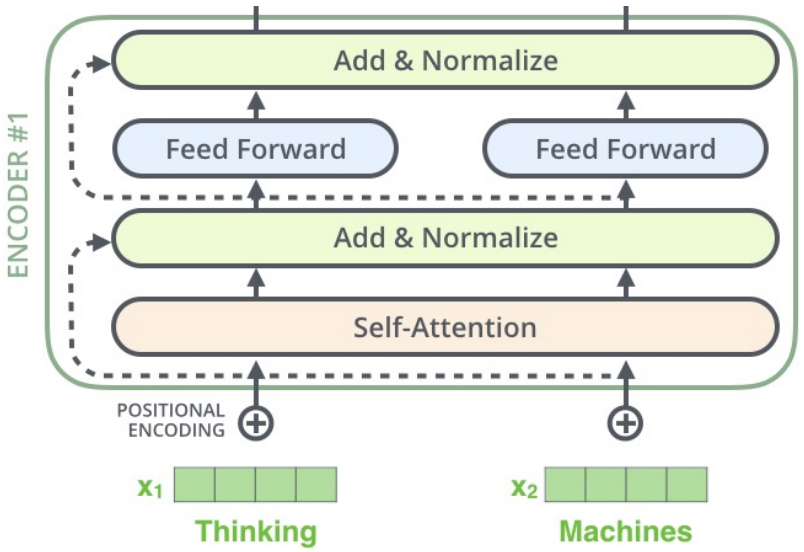
- $\text{BERT}_{\text{Base}}$
  - #parameters = **110M**, #encoders = **12**, #dimension = **768**, #head = **12**
  - #FLOPs = 123M * sentence_length * #Batch



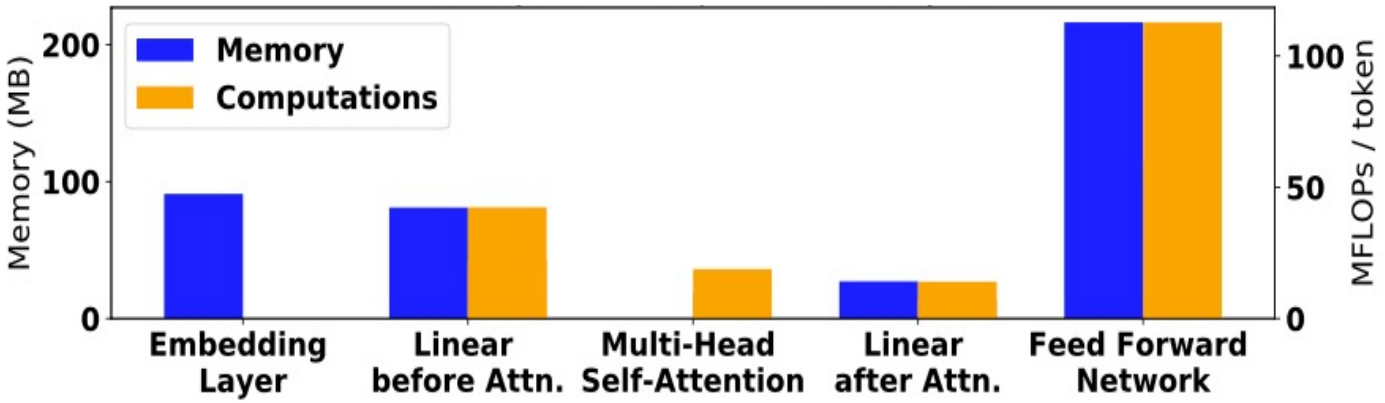
Comparison between $\text{BERT}_{\text{Base}}$ and $\text{BERT}_{\text{Large}}$

Parameter / FLOPs Distributions

- Memory (#parameters) is dominated by **embedding** and **linear layers**
- FLOPs is dominated by **linear layers**



Architecture of BERT (One Encoder Here)



Memory and computation analysis [1]

[1] BGanesh, P., etc. (2020). Compressing large-scale transformer-based models: A case study on bert. arXiv preprint arXiv:2002.11985.

Preliminary: Knowledge Distillation

- Goal
  - Inputs: teacher model (**large, well-trained**), student model (**small**), data
  - Output: **compressed student** with similar behaviors of teacher
- How? Minimize the **difference between the outputs** of teacher and student

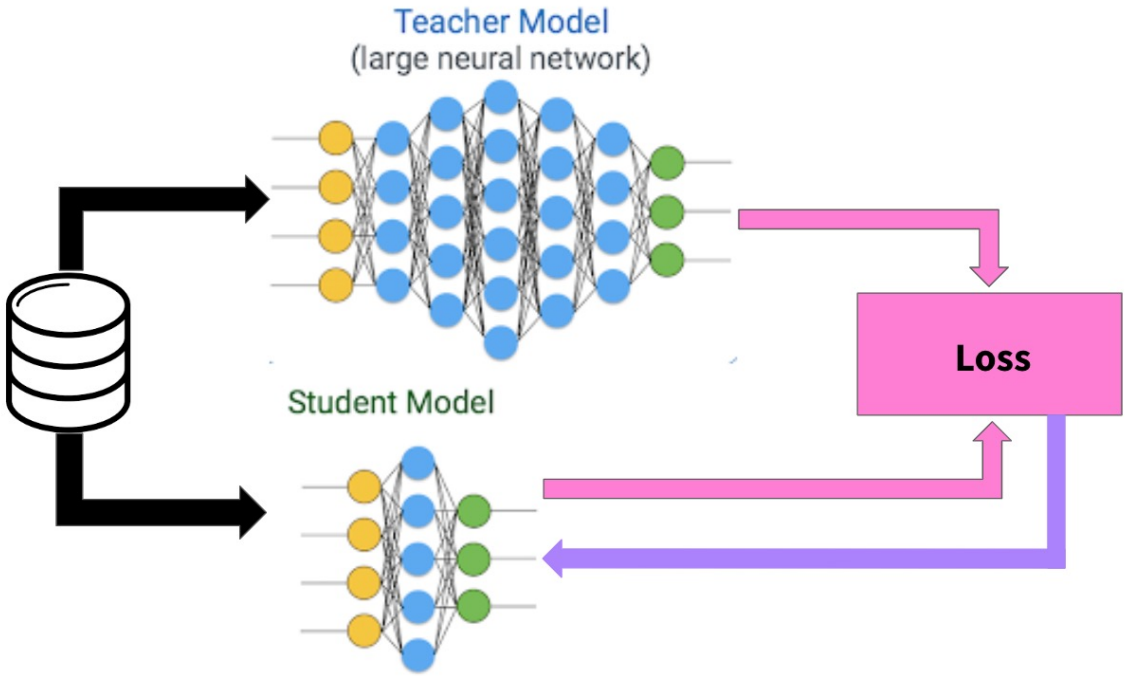
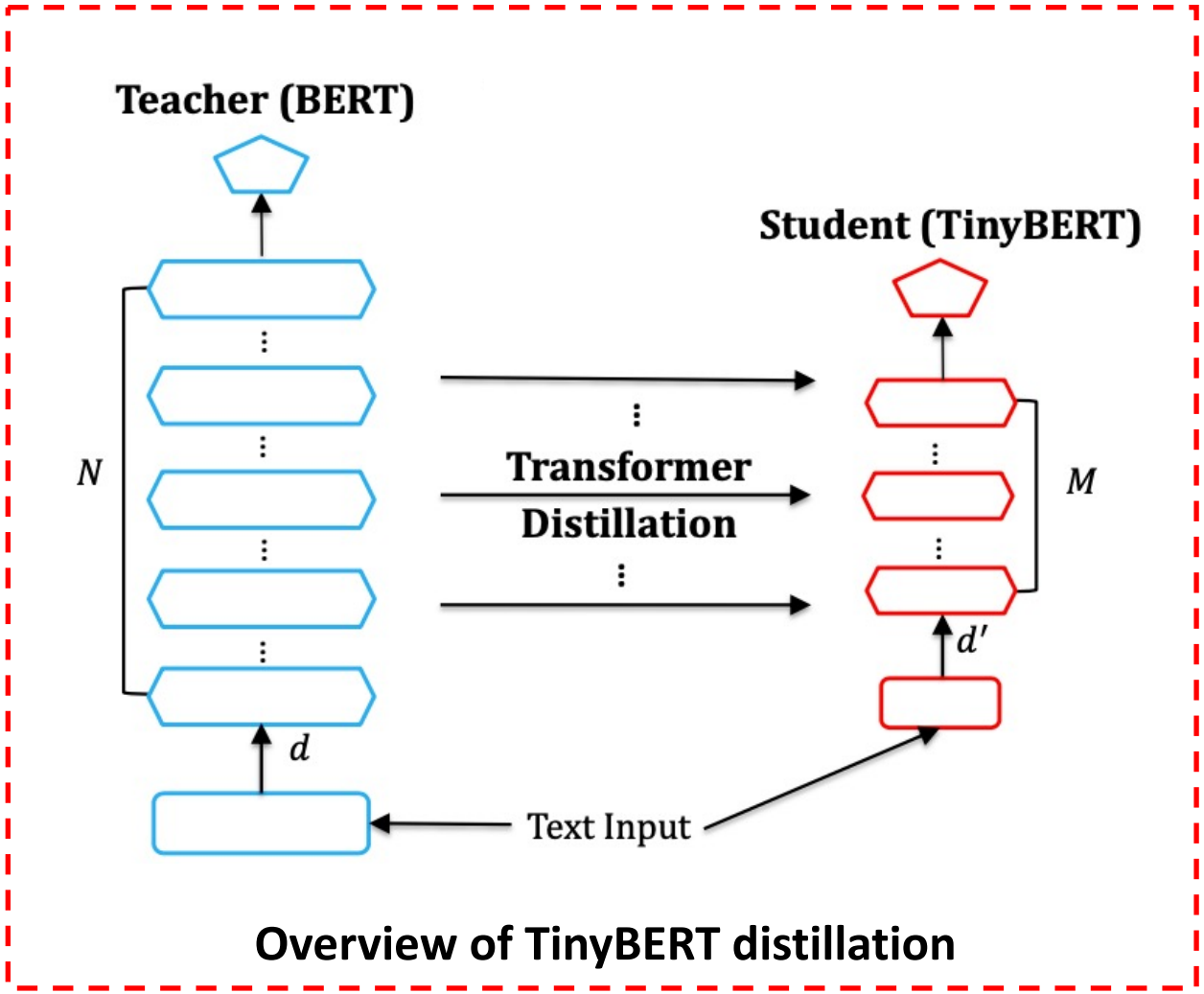


Illustration of knowledge distillation [1]

[1] https://towardsdatascience.com/knowledge-distillation-simplified-dd4973dbc764

TinyBERT: Distilling BERT for Natural Language Understanding [1]

- Knowledge distillation of the Transformer-based models
- A two-stage learning framework
- Results
  - **> 96%** the performance of teachers
  - **7.5x smaller** and **9.4x faster** on inference

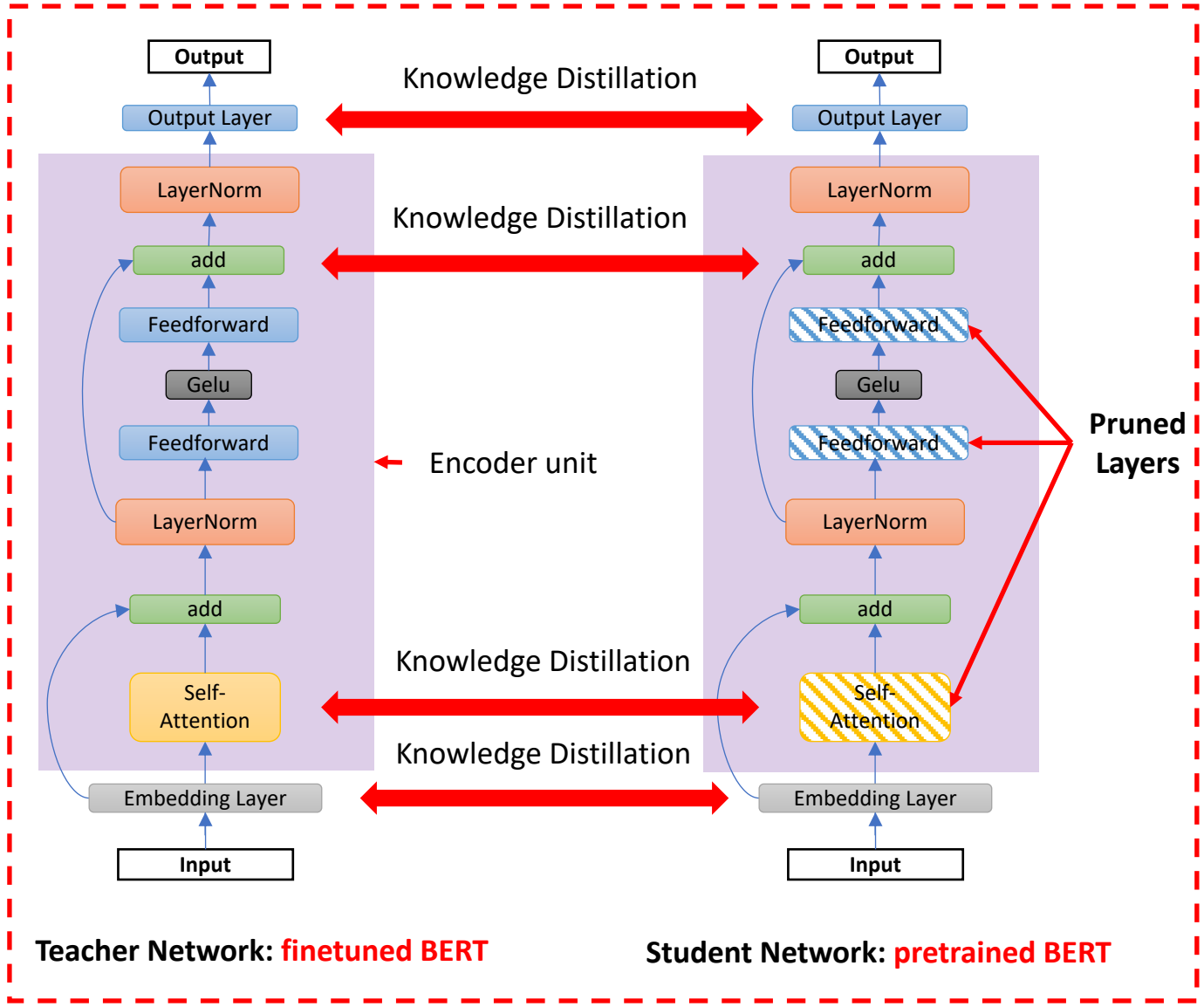


Overview of TinyBERT distillation

[1] Huawei Noah's Ark Lab. Tinybert: Distilling bert for natural language understanding. *EMNLP 2020.*

SparseBERT: Knowledge-Aware Sparse Pruning [1]

- Motivation: gap of sparse pruning in NLP
- Pruning while distillation
- Achieved SOTA
  - Compression ratio = **x20**
  - But only **1.4% performance drop**
- Sparse pruning
  - *Trending and with promising future*



[1] Dongkuan Xu, Ian En-Hsu Yen, Jinxi Zhao, Zhibin Xiao. Rethinking Network Pruning -- under the Pre-train and Fine-tune Paradigm. *NAACL 2021.*

All The Ways You Can Compress BERT [1]

- Literatures

Paper	Prune	Factor	Distill	W. Sharing	Quant.	Pre-train	Downstream
Compressing BERT: Studying the Effects of Weight Pruning on Transfer Learning	☑					☑	☑
Are Sixteen Heads Really Better than One?	☑						☑
Pruning a BERT-based Question Answering Model	☑						☑

- Experimental Results

Paper	Reduction	Of	Speed-up	Accuracy?	Comments
Compressing BERT: Studying the Effects of Weight Pruning on Transfer Learning	30%	params	?	Same	Some interesting ablation experiments and fine-tuning analysis
Are Sixteen Heads Really Better than One?	50-60%	attn heads	1.2x	Same	
Pruning a BERT-based Question Answering Model	50%	attn Heads + FF	2x	-1.5 F1	

[1] http://mitchgordon.me/machine/learning/2019/11/18/all-the-ways-to-compress-BERT.html

Q & A

Web: www.personal.psu.edu/dux19/

Email: dux19@psu.edu