Week 9 – other transformers models

EGCO467 Natural Language and Speech Processing

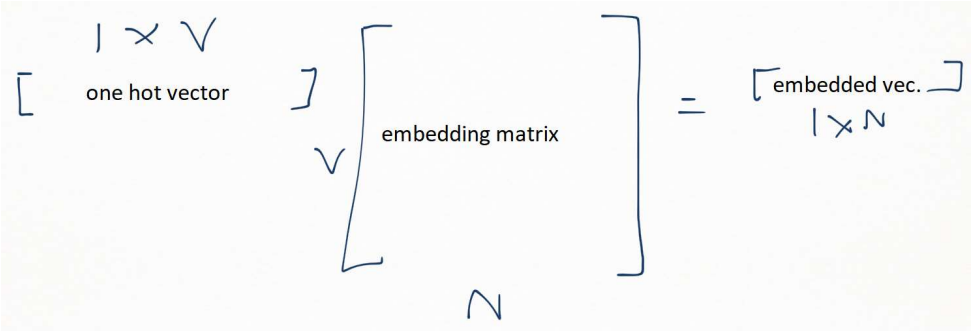
Other transformer models

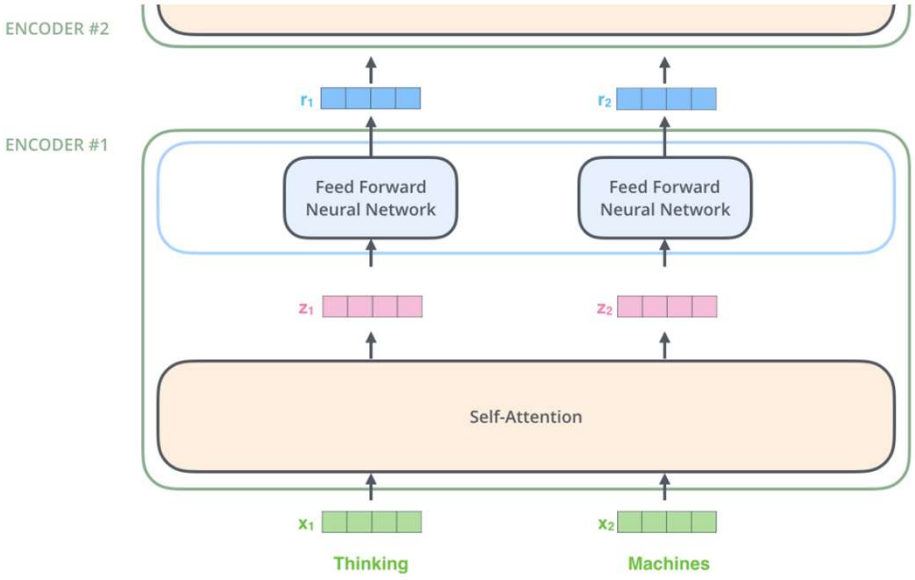
- AlBERT
  - [1] Lan, Zhenzhong, et al. "Albert: A lite bert for self-supervised learning of language representations." *arXiv preprint arXiv:1909.11942* (2019).
- DistilBERT
  - [2] Sanh, Victor, et al. "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter." *arXiv preprint arXiv:1910.01108* (2019).
- Longformer
  - [3] Beltagy, Iz, Matthew E. Peters, and Arman Cohan. "Longformer: The long-document transformer." *arXiv preprint arXiv:2004.05150* (2020).
- T5
  - [4] Raffel, Colin, et al. "Exploring the limits of transfer learning with a unified text-to-text transformer." *arXiv preprint arXiv:1910.10683* (2019).

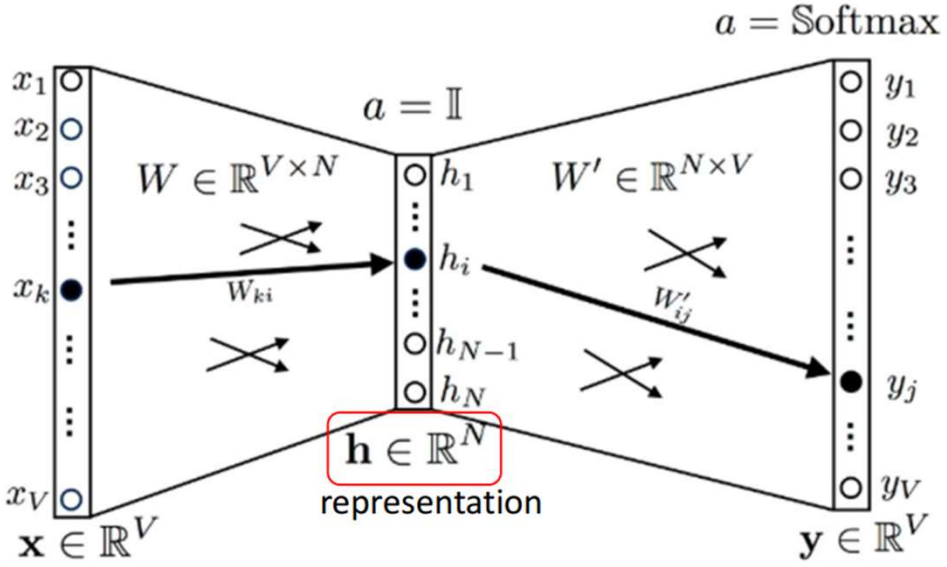
AlBERT

Parameter Reduction

- Untie hidden size m from Embedding dim N. Make m!=N

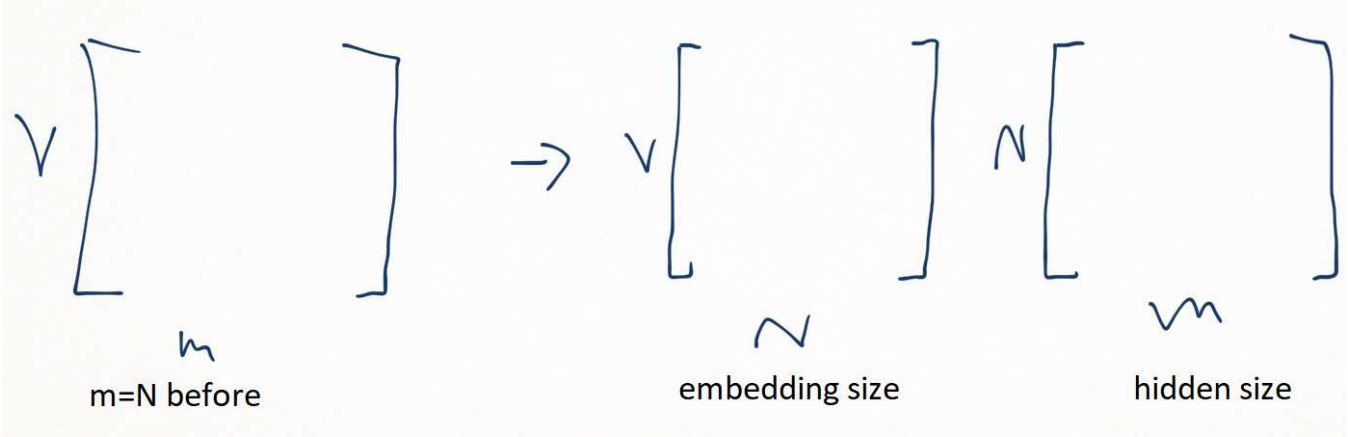






Factor the embedding matrix

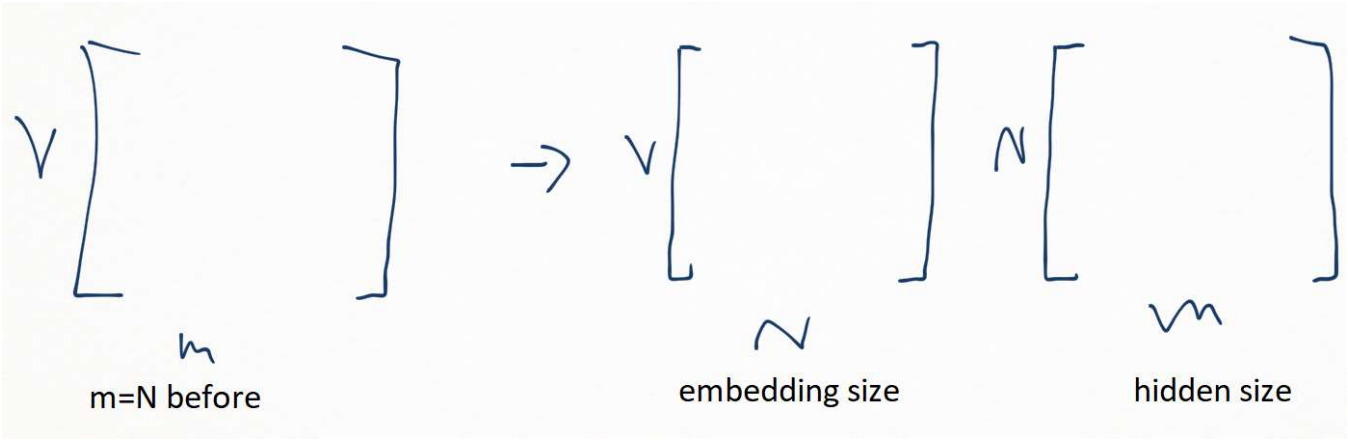
- Not projecting the one-hot vectors directly into the hidden space of size m
- Project them into a lower dimensional embedding space of size E first
- The product it to the hidden space.



O(V*m) => O(V*N + N*m)

Factor the embedding matrix

- V = 30,000
- m = 4096 (default in Huggingface's Albert)
- N = 300
- O(V*m) = 122,880,000
- O(V*N + N*m) = 10,228,800



Cross-layer parameter sharing

- Every layers use the same parameter (both attention and feedforward)

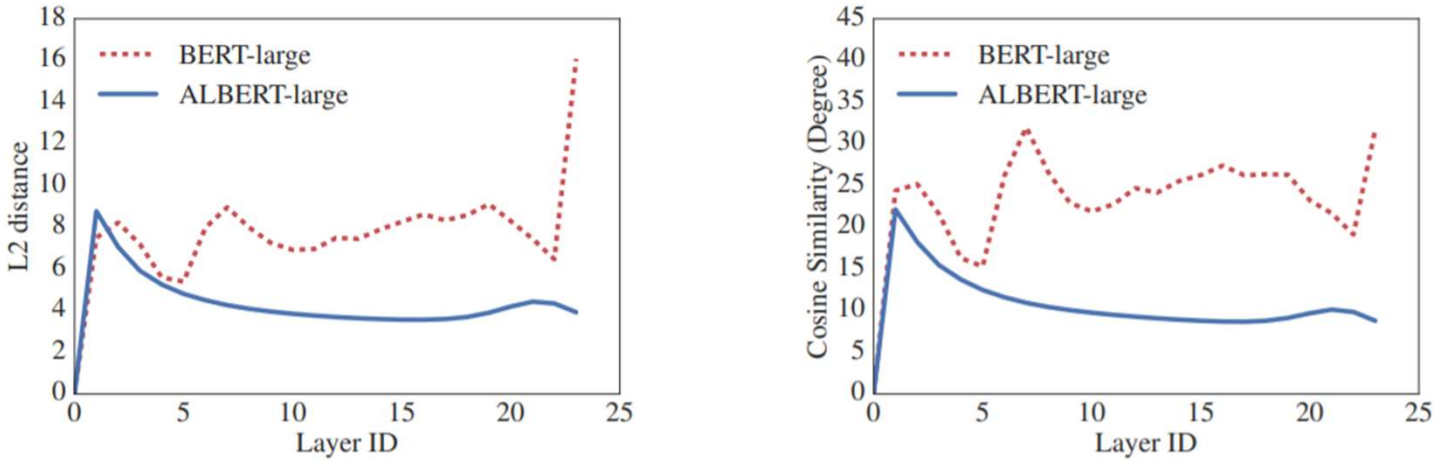
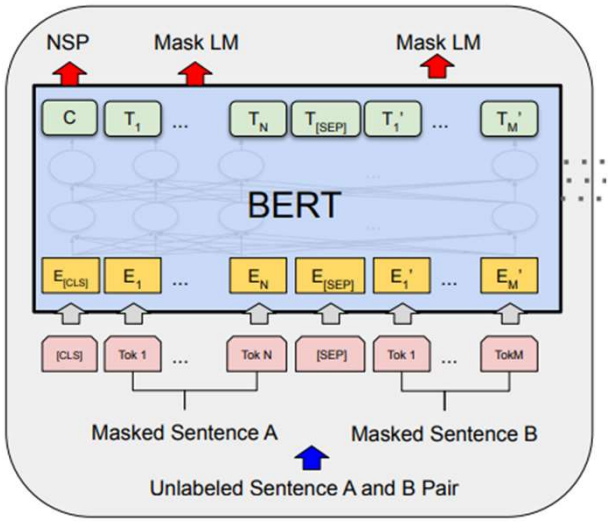


Figure 1: The L2 distances and cosine similarity (in terms of degree) of the input and output embedding of each layer for BERT-large and ALBERT-large.

NSP -> SOP

- NSP: B follows A, or B is some random other sentence
- SOP: order is natural (A then B) or order had been reversed (B then A)

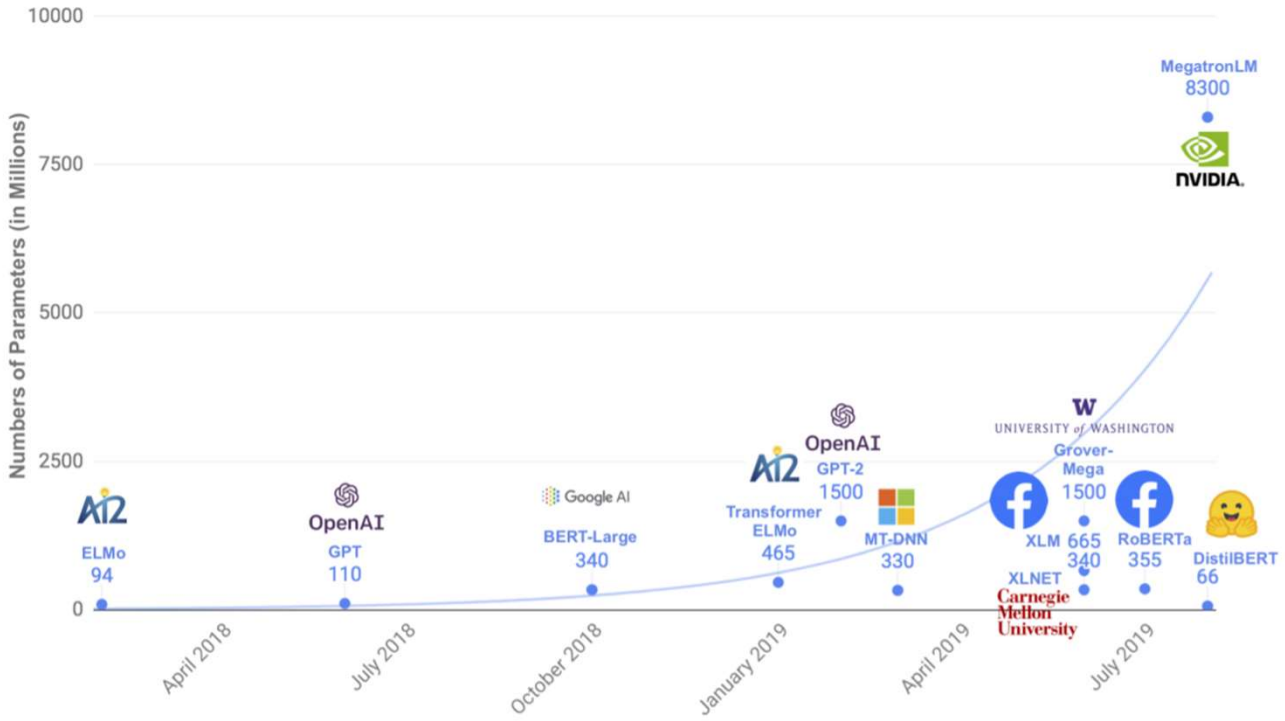


Bert vs. Albert

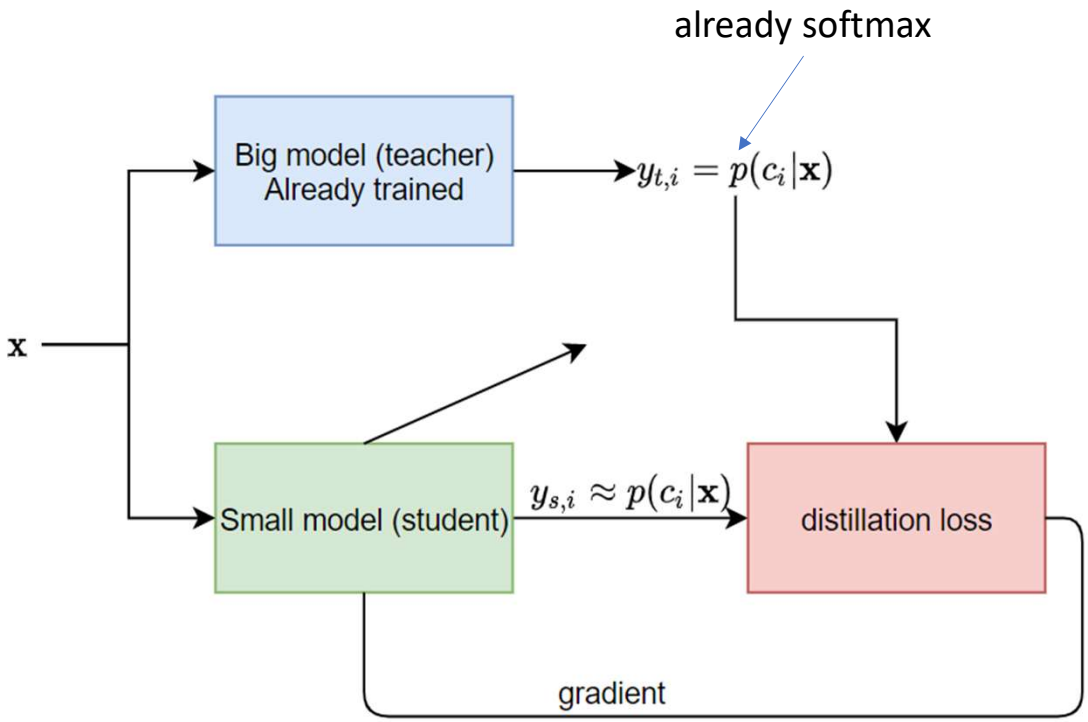
Model		Parameters	Layers	Hidden	Embedding	Parameter-sharing
BERT	base	108M	12	768	768	False
	large	334M	24	1024	1024	False
ALBERT	base	12M	12	768	128	True
	large	18M	24	1024	128	True
	xlarge	60M	24	2048	128	True
	xxlarge	235M	12	4096	128	True

DistilBERT

Distilbert


10000
7500
5000
2500
0
Numbers of Parameters (in Millions)
April 2018
July 2018
October 2018
January 2019
April 2019
July 2019
ELMo
94
OpenAI
GPT
110
Google AI
BERT-Large
340
Transformer
ELMo
465
OpenAI
GPT-2
1500
MT-DNN
330
UNIVERSITY of WASHINGTON
Grover-
Mega
1500
XLM 665
340
XLNET
Carnegie
Mellon
University
RoBERTa
355
DistilBERT
66
MegatronLM
8300
NVIDIA.

Knowledge distillation



Distillation Loss

- p(x) = probability distribution of teacher
- q(x) = probability distribution of student
- Loss (nn.KLDivLoss in Pytorch):

$$KL(p||q) = \mathbb{E}_p(log(\frac{p}{q})) = \sum_i p_i * log(p_i) - \sum_i p_i * log(q_i)$$

Total Loss

- Distillation loss (previous slide)
- Normal MLM loss (Bert)

$$\max_{\theta} \quad \log p_\theta(\bar{\mathbf{x}} \mid \hat{\mathbf{x}}) \approx \sum_{t=1}^{T} m_t \log p_\theta(x_t \mid \hat{\mathbf{x}}) = \sum_{t=1}^{T} m_t \log \frac{\exp\left(H_\theta(\hat{\mathbf{x}})_t^\top e(x_t)\right)}{\sum_{x'} \exp\left(H_\theta(\hat{\mathbf{x}})_t^\top e(x')\right)},$$

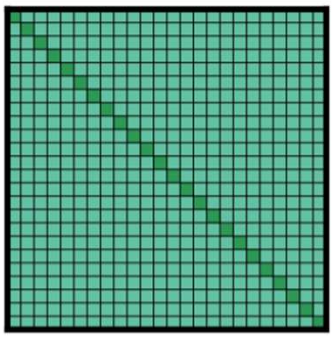
- Cosine embedding loss

$$\text{loss}(\mathrm{x}, \mathrm{y}) = \begin{cases} 1 - \cos(\mathrm{x}_1, \mathrm{x}_2), & \text{if } \mathrm{y} = 1 \\ \max(0, \cos(\mathrm{x}_1, \mathrm{x}_2) - \text{margin}), & \text{if } \mathrm{y} = -1 \end{cases}$$

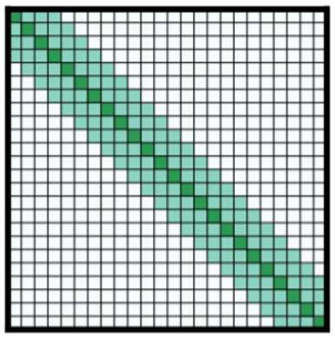
Longformer

Longformer

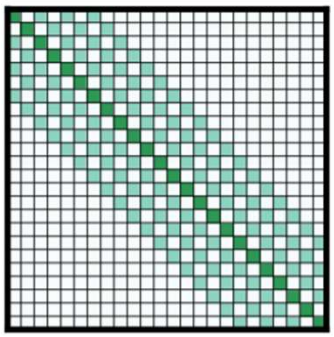
- full attention is N2
- maximum length of sequence usually capped at 512
- the farther apart a pair of tokens are, the less likely the pair is important



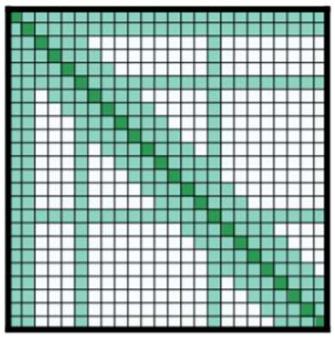
(a) Full n^2 attention



(b) Sliding window attention

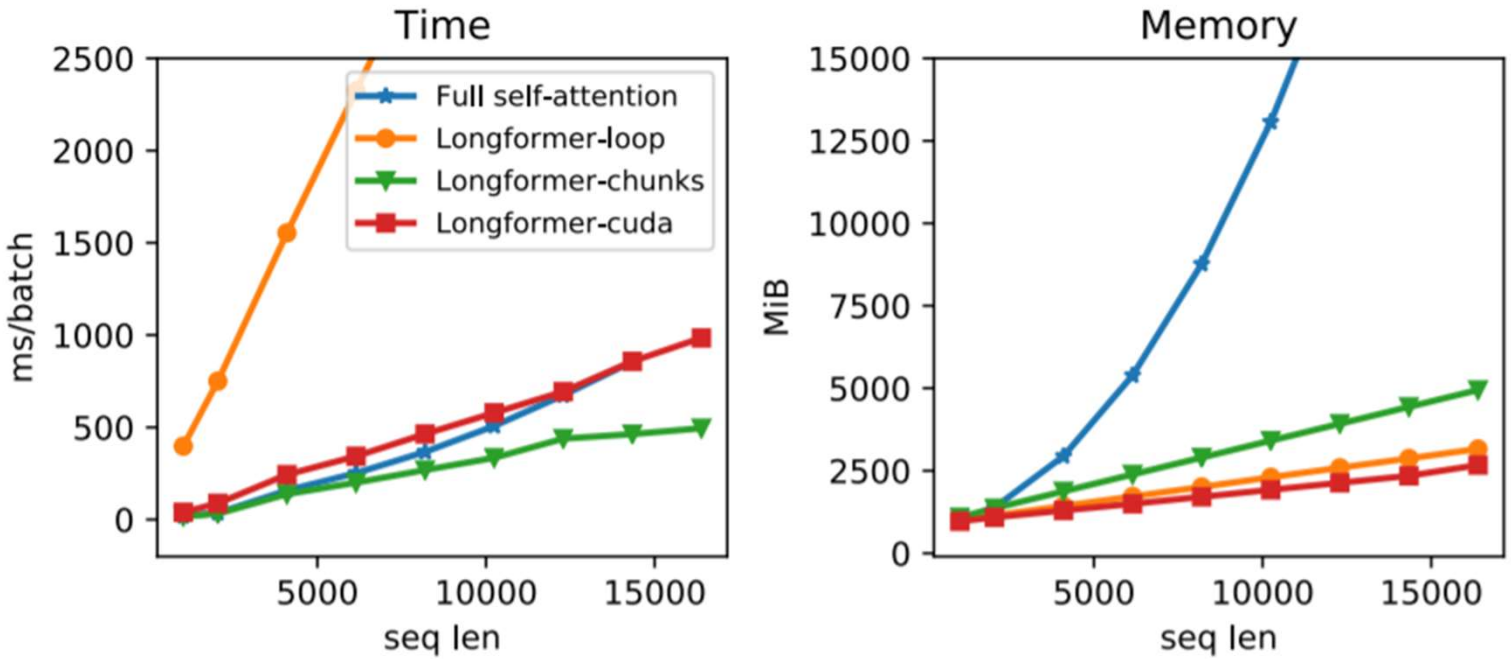


(c) Dilated sliding window



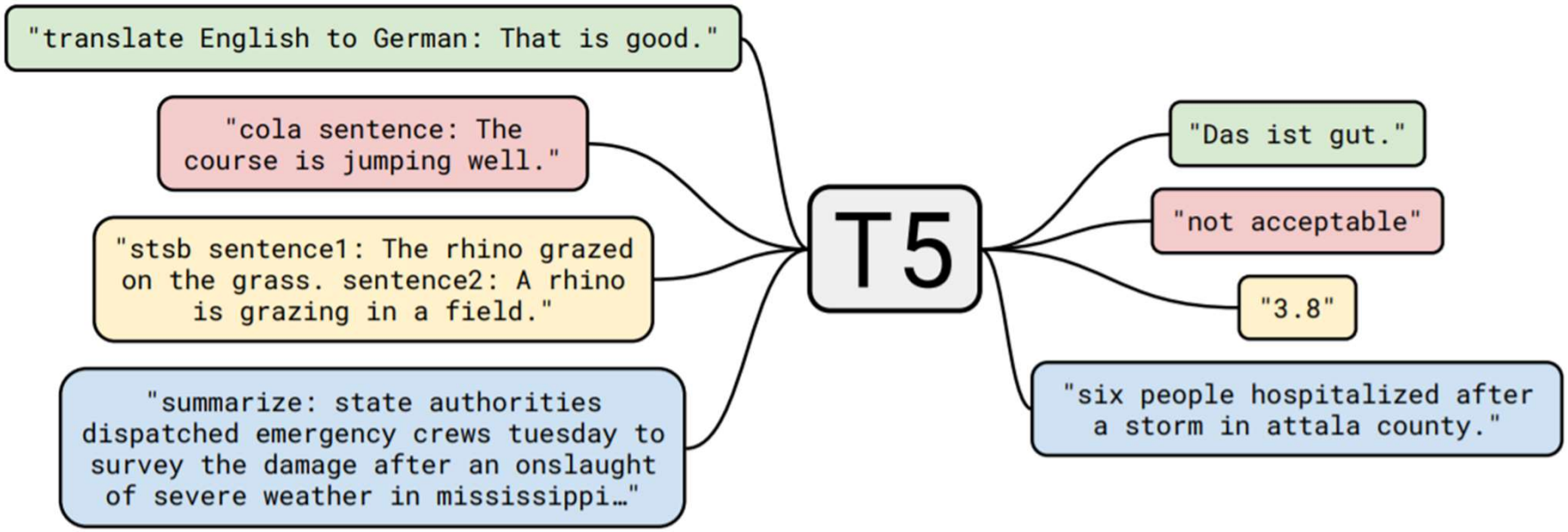
(d) Global+sliding window

Longformer


Time
ms/batch
2500
2000
1500
1000
500
0
5000
10000
15000
seq len
Full self-attention
Longformer-loop
Longformer-chunks
Longformer-cuda
Memory
MiB
15000
12500
10000
7500
5000
2500
0
5000
10000
15000
seq len

T5

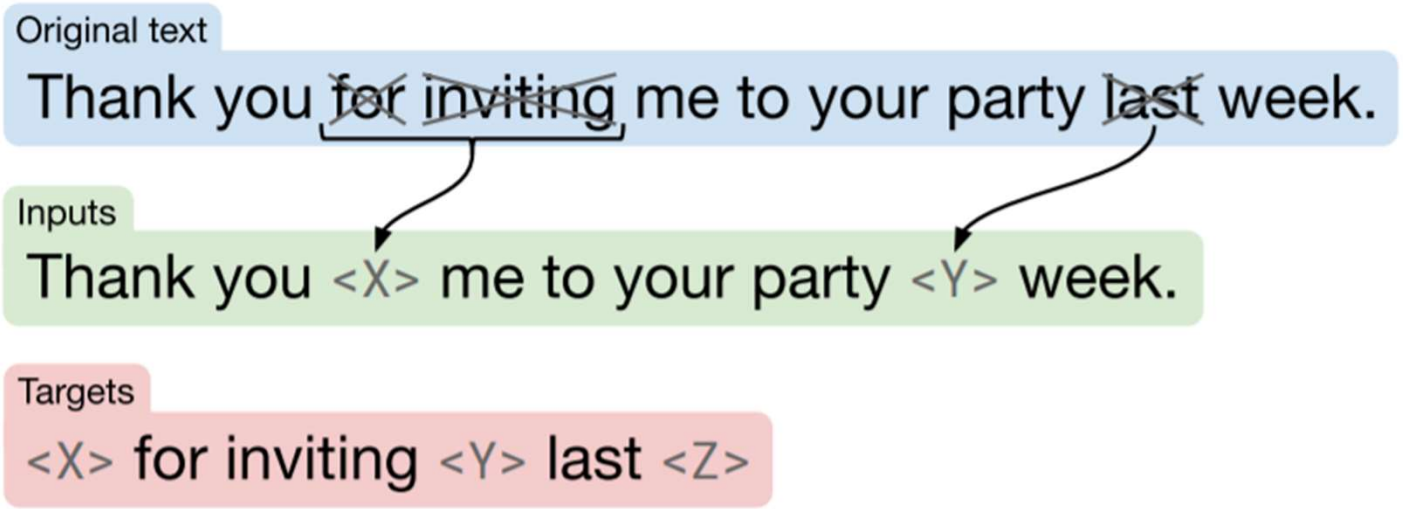
T5 (Text-to-Text Transfer Transformer)



The Colossal Clean Crawled Corpus (cleaned common crawl)

- We only retained lines that ended in a terminal punctuation mark (i.e. a period, exclamation mark, question mark, or end quotation mark).
- We discarded any page with fewer than 5 sentences and only retained lines that contained at least 3 words.
- We removed any page that contained any word on the "List of Dirty, Naughty, Obscene or Otherwise Bad Words".[6]
- Many of the scraped pages contained warnings stating that Javascript should be enabled so we removed any line with the word Javascript.
- Some pages had placeholder "lorem ipsum" text; we removed any page where the phrase "lorem ipsum" appeared.
- Some pages inadvertently contained code. Since the curly bracket "{" appears in many programming languages (such as Javascript, widely used on the web) but not in natural text, we removed any pages that contained a curly bracket.
- To deduplicate the data set, we discarded all but one of any three-sentence span occurring more than once in the data set.

pretraining objective



WangchanBERTa

WangchanBERTa

- Lowphansirikul, Lalita, et al. "WangchanBERTa: Pretraining transformer-based Thai Language Models." *arXiv preprint arXiv:2101.09635* (2021).
- RoBERTa pretrained on Thai text about 70GB size
- DGX-1 server with 8x V100 gpus trained for 125 days

Wongnai with WangchanBERTa

Project Topic 1

- Extract location from tweets

1. Mine Twitter for a certain hashtag.
2. Extract location from tweets.
3. (Extra) visualize on map.

Project Topic 2

- Extract sentiments from tweets

1. Extract top k most common positive words
2. Extract top k most common negative words
3. Overall sentiment (positive, negative, neutral) of tweets with this hashtag

Twitter API

- https://developer.twitter.com/en/docs/twitter-api