PoNet: Pooling Network for Efficient Token Mixing in Long Sequences

Chao-Hong Tan[1*], Qian Chen[2], Wen Wang[2], Qinglin Zhang[2], Siqi Zheng[2], Zhen-Hua Ling[1]

[1] National Engineering Laboratory for Speech and Language Information Processing, University of Science and Technology of China [2] Speech Lab, Alibaba Group
* Work is done during the internship at Speech Lab, Alibaba Group.

ALIBABA DAMO ACADEMY

ICLR

Introduction

Motivation

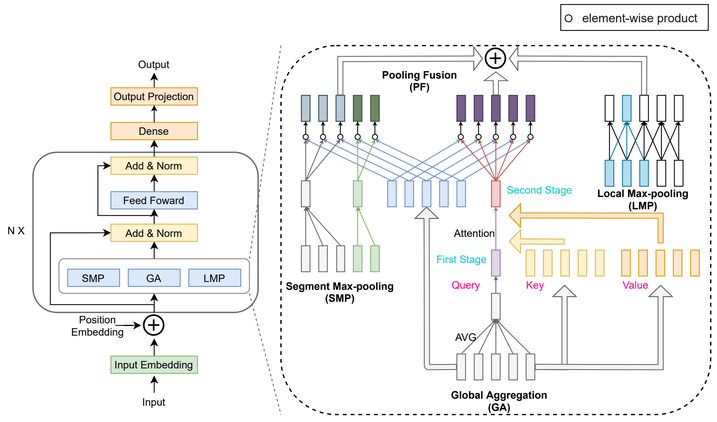
- The self-attention mechanism in Transformer has quadratic time and memory complexity with respect to the sequence length, hindering applications to long sequences
- Inspired by External Attention (EA) (Guo et al., 2021) which mainly used MLP and Softmax to capture sequence information - still slow due to Softmax Involving calculations of exponents

Contributions

- Proposed a novel **Pooling Network** (**PoNet**) for token mixing in long sequences with **linear complexity**
- **First** to explore the full potential of the simple **pooling mechanism** for token mixing and for modeling long-range dependencies
- **Long sequence modeling** capabilities - Long Range Arena benchmark
  - PoNet significantly outperforms Transformer by **+2.28 absolute** (+3.9% relative) on accuracy
  - Up to **9** times faster and memory usage **10** times smaller than Transformer on GPU
- **Transfer learning** capabilities - GLUE
  - PoNet-Base reaches **95.7%** of the accuracy of BERT-Base on GLUE

PoNet Architecture

PoNet models different levels of contextual information through a **multi-granularity pooling** block as a **drop-in replacement** for the self-attention sublayer in Transformer. It consists of three components, **global aggregation** (**GA**), **segment max-pooling** (**SMP**), and **local max-pooling** (**LMP**), which are then aggregated through **pooling fusion** (**PF**).



The right enlarged view shows multi-granularity pooling (GA, SMP, LMP) and pooling fusion.

- **GA**: Capture the most important **global information** for each token
  - **First stage (FS-GA)**: Average at the sequence level

$$g = \frac{1}{N}\sum_{n=1}^{N} h_{Q_{g_n}} \in \mathbb{R}^d$$

  - **Second stage (SS-GA)**: Cross-Attention, more accurate representation

$$g' = Attention\{g, H_{K_g}, H_{V_g}\}$$

- **SMP**: **Alleviate information loss** from compressing a long sequence into a single global token, and explore prior knowledge of **structure**
- **LMP**: Capture contextual information from neighbors for each token
- **PF**: GA and SMP values interact with tokens through **element-wise product** and all these three features are added up as the final output.

Long Sequence Modeling — LRA Benchmark

Accuracy

In group 3, our PoNet achieves best accuracy except on the Pathfinder task.

Model	ListOps(2K)	Text(4K)	Retrieval(4K)	Image(1K)	Pathfinder(1K)	AVG.
Transformer(1)	**36.37**	64.27	57.46	42.44	71.40	54.39
Longformer (1)	35.63	62.85	56.89	42.22	69.71	53.46
BigBird (1)	36.05	64.02	**59.29**	40.83	74.87	**55.01**
Performer (1)	18.01	**65.40**	53.82	**42.77**	**77.05**	51.41
Transformer(2)	**36.06**	61.54	**59.67**	**41.51**	80.38	**55.83**
Linear (2)	33.75	53.35	58.95	41.04	**83.69**	54.16
FNet (2)	35.33	**65.11**	59.61	38.67	77.80	55.30
Transformer(3)	37.10	65.02	79.35	38.20	**74.16**	58.77
Performer(3)	18.80	63.81	78.62	37.07	69.87	53.63
Reformer(3)	19.05	64.88	78.64	43.29	69.36	55.04
Linformer(3)	37.25	55.91	79.37	37.84	67.60	55.59
Nyströmformer(3)	37.15	65.52	79.56	41.58	70.94	58.95
FNet	37.40	62.52	76.94	35.55	FAIL	53.10
PoNet (Ours)	**37.80**	**69.82**	**80.35**	**46.88**	70.39	**61.05**

Results with **(1)** are cited from (Tay et al., 2021), with **(2)** are from (Lee-Thorp et al., 2021), with **(3)** are from (Xiong et al., 2021). We implement our **PoNet** and re-implement **FNet** (Lee-Thorp et al., 2021) based on the PyTorch codebase from (Xiong et al., 2021) and use the **same experimental configurations** to ensure a fair comparison.
For each group, the **best results** for each task and AVG are bold-faced.

Speed and Memory

PoNet is the second fastest model and consumes the second smallest memory.
The speedup from PoNet over Transformer **escalates** on longer input lengths.

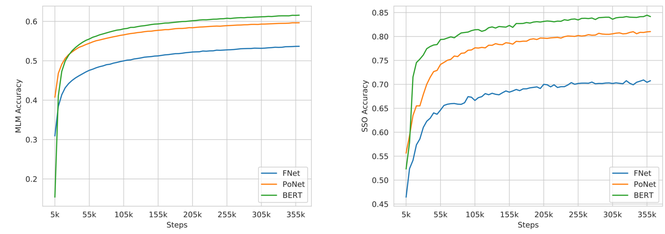
Seq. length	512	1024	2048	4096	8192	16384
	Training Speed (steps/s)↑					
Transformer	45.1	19.4	6.3	1.8	OOM	OOM
Performer	39.4(0.9x)	25.0(1.3x)	14.3(2.3x)	7.8(4.3x)	4.0	2.0
Nyströmformer	39.1(0.9x)	30.3(1.6x)	20.0(3.2x)	11.5(6.4x)	6.1	3.1
FNet	**83.4(1.8x)**	**61.3(3.1x)**	**38.1(6.0x)**	**21.4(11.9x)**	**11.0**	**5.4**
PoNet (Ours)	50.4(1.1x)	40.1(2.1x)	27.8(4.4x)	16.2(9.0x)	8.7	4.5
	Peak Memory Usage (GB)↓					
Transformer	1.4	2.5	6.7	23.8	OOM	OOM
Performer	1.5(1.1x)	2.1(0.8x)	3.1(0.5x)	5.4(0.2x)	9.8	18.7
Nyströmformer	1.2(0.8x)	1.5(0.6x)	1.9(0.3x)	2.8(0.1x)	4.5	8.2
FNet	**1.1(0.8x)**	**1.2(0.5x)**	**1.4(0.2x)**	**1.7(0.1x)**	**2.3**	**3.8**
PoNet (Ours)	**1.1(0.8x)**	1.3(0.5x)	1.7(0.2x)	2.4(0.1x)	3.6	6.5

Comparison of **GPU training speed** and **peak memory consumption** on various input sequence lengths on the LRA text classification task (using the same hyper-parameter setting for this task as in (Xiong et al., 2021)).
The **best results** are bold-faced with the second-best results underlined.

Transfer Learning

Pre-training Tasks

Pre-train with MLM and SSO (Sentence Structural Objective). PoNet achieves significantly better accuracy than FNet while slightly worse than BERT.



(a) MLM Accuracy (b) SSO Accuracy

MLM and SSO validation accuracy against the numbers of training steps from BERT-Base, FNet-Base, and PoNet-Base. MLM is used in BERT (Devlin et al., 2019). SSO is used in StructBERT (Wang et al., 2020).

GLUE Fine-tuning Results

PoNet reaches 95.7% of BERT accuracy and outperforms FNet by 4.5% rel.

Model	MNLI(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	AVG.
BERT-Base	81.35/80.98	88.89	88.01	91.17	47.66	87.83	86.66	69.31	**80.21**
FNet-Base	73.13/73.66	85.75	80.50	88.65	40.61	80.62	80.84	57.40	73.46
PoNet-Base (Ours)	76.99/77.21	87.55	84.33	89.22	45.36	84.57	81.76	64.26	76.80

GLUE Validation results. All models are uncased and pre-trained with the same configurations using **5GB data** (Wikitext-103 and BooksCorpus) with **340K steps**. We report the best GLUE results for each model from multiple hyper-parameter configurations. We report the mean of accuracy and F1 for QQP and MRPC, matthews correlations for CoLA, spearman correlations for STS-B, and accuracy for others. MNLI(m/mm) means match/mismatch splits.

When investigating the model capacity by pre-training with more data, PoNet-Base still reaches 96% of BERT-Base's accuracy as shown in Group (3).

Model	MNLI(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	AVG.
BERT-Base(1)	84/81	87	91	93	73	89	83	83	83.3
Linear-Base(1)	74/75	84	80	94	67	67	83	69	77.0
FNet-Base(1)	72/73	83	80	95	69	79	76	63	76.7
BERT-Base(2)	85/85	89.77	91.78	92.66	58.88	89.28	89.31	70.76	**83.52**
FNet-Base(2)	75/76	86.72	83.23	90.13	35.37	81.43	80.34	59.92	74.23
PoNet-Base(Ours)(2)	78/78	87.76	85.17	89.00	47.24	85.86	83.39	63.53	77.54
BERT-Base(3)	83/83	89.48	90.65	91.74	51.19	89.28	88.73	67.51	**81.63**
FNet-Base(3)	75/76	86.17	82.52	88.42	40.57	83.64	80.90	61.73	74.99
PoNet-Base(Ours)(3)	79/78	87.92	86.31	89.79	45.18	87.17	84.27	66.43	78.29

More GLUE Validation results. Results with **(1)** are from (Lee-Thorp et al., 2021). For BERT-Base **(2)** and FNet-Base **(2)**, we use the official checkpoints while PoNet-Base **(2)** **is pre-trained on 5GB data**, For a fair comparison on model capacity with more data, BERT-Base**(3)**, FNet-Base**(3)**, and PoNet-Base**(3)** are all pre-trained on the same **16GB data** (Wikipedia and BooksCorpus), trained with MLM+SSO tasks for **1M steps**. Note that BERT-Base(3) has a lower performance than the official BERT-Base(2), mainly due to the different batch size.

Ablation Analysis

Removing GA significantly degrades SST (Sentence Structure Task) and STS-B accuracy. Removing SMP or LMP degrades MLM and CoLA accuracy. Weakening SST loss weakens GA representation learning but strengthens SMP and LMP. All three pooling are important for the performance of PoNet.

Model	Pre-trained tasks MLM	Pre-trained tasks SST	Downstream tasks CoLA	Downstream tasks STS-B
PoNet(340K steps)	59.44	80.75	45.36	84.57
PoNet w/o SS-GA	59.33	76.92	46.18	78.38
PoNet w/o GA	56.64	74.36	49.51	64.61
PoNet w/o SMP	56.96	78.41	44.21	84.89
PoNet w/o LMP	56.53	80.27	41.44	85.55
PoNet w/o (SMP&LMP)	43.61	76.72	11.36	84.93
PoNet using $\mathcal{L}_{MN}$	62.53	79.28	50.91	75.32
PoNet using $\mathcal{L}_{OM}$	63.11	−−	51.26	69.83

L_{MN} denotes MLM and NSP loss. L_{OM} denotes only MLM loss. SST denotes NSP when using L_{MN} and the SSO task otherwise. All pre-training experiments run 340K steps with 5GB data.

Code Release

Mail: chtan@mail.ustc.edu.cn
Git: github.com/lxchtan/PoNet