Random Feature Attention

Hao Peng, Nikos Pappas, Dani Yogatama, Roy Schwartz, Noah Smith, Lingpeng Kong
Transformers

State-of-the-art results in many sequence modeling tasks

- Machine translation (Vaswani et al., 2017)
- Language modeling (Ott et al., 2018)
- Pretraining (Delvin et al., 2019)
Transformers

State-of-the-art results in many sequence modeling tasks

- Machine translation (Vaswani et al., 2017)
- Language modeling (Ott et al., 2018)
- Pretraining (Delvin et al., 2019)
- Reinforcement learning (Parisotto et al., 2019)
- Computer vision (Parmar et al., 2018; Dosovitskiy et al., 2020)
- Computational biology (Choromanski et al., 2020)

…
Attention

query

key

value

input
Attention

\[
\text{softmax} \quad \frac{\exp q^\top k_i}{\sum_j \exp q^\top k_j}
\]
Attention

\[
\sum_i \frac{\exp q^\top k_i}{\sum_j \exp q^\top k_j} v_i^\top
\]

query

key

value

input
Attention

\[ \begin{align*}
\text{softmax} & \quad \text{output} \\
\text{query} & \quad \text{key} \\
\text{value} & \quad \text{input}
\end{align*} \]

\[ \sum_i \frac{\exp q^\top k_i}{\sum_j \exp q^\top k_j} v_i^\top \]

time each step overall \( \mathcal{O}(N^3) \)

space

\( \mathcal{O}(N^2) \)
Attention Complexity: Seq2seq Decoding

causal attn.: $O(N)$

cross attn.: $O(N)$

length $N$
Attention Complexity: Seq2seq Decoding

- **target**: prefix \( N \) \( \rightarrow \) current \( N \)
- **source**: length \( N \)

- **causal attn.**: \( O(N) \)
- **cross attn.**: \( O(N) \)
Attention Complexity: Seq2seq Decoding

causal attn.: $O(N)$

cross attn.: $O(N)$

source

length $N$

target

prefix current

length $N$

time each step $O(N) + O(N)$

overall space $O(N^2)$

space $O(N)$
Overview

Transformers: quadratic overhead, limited in
• Character-level language modeling
• Document-level machine translation
• Speech
• ...
Overview

Transformers
• State-of-the-art results in many sequence modeling tasks
• Quadratic complexity, less well-suited for long sequences

This Work: Random Feature Attention
• Strong performance
• Scales linearly in sequence length
Random Fourier Features
Rahimi and Recht (2007)

Goal

\[ \exp q^\top k \approx \phi(q)^\top \phi(k) \]
Random Fourier Features
Rahimi and Recht (2007)

Goal
\[ \exp \mathbf{q}^\top \mathbf{k} \approx \phi(\mathbf{q})^\top \phi(\mathbf{k}) \]

Let
\[ \phi(\mathbf{x}) = \sqrt{1/D} \left[ \sin (\mathbf{w}_1^\top \mathbf{x}), \ldots, \sin (\mathbf{w}_D^\top \mathbf{x}), \cos (\mathbf{w}_1^\top \mathbf{x}), \ldots, \cos (\mathbf{w}_D^\top \mathbf{x}) \right]^\top \]

where
\[ \mathbf{w}_i \sim \mathcal{N}(0, 1) \]

Then
\[ \mathbb{E} \left[ \phi(\mathbf{q})^\top \phi(\mathbf{k}) \right] = \exp \mathbf{q}^\top \mathbf{k} \]

constant scalar depending on the norms of \( \mathbf{q} \) and \( \mathbf{k} \)
From Attention to Random Feature Attention

\[
\sum_i \frac{\exp q^\top k_i}{\sum_j \exp q^\top k_j} v_i^\top
\]

- \(q\) query
- \(k_i\) keys
- \(v_i\) values
From Attention to Random Feature Attention

\[
\sum_i \frac{\exp \mathbf{q}^\top \mathbf{k}_i}{\sum_j \exp \mathbf{q}^\top \mathbf{k}_j} \mathbf{v}_i^\top \\
\approx \sum_i \frac{\mathbf{\phi}(\mathbf{q})^\top \mathbf{\phi}(\mathbf{k}_i) \otimes \mathbf{v}_i}{\sum_j \mathbf{\phi}(\mathbf{q})^\top \mathbf{\phi}(\mathbf{k}_j)}
\]

\[\mathbb{E} \left[ \mathbf{\phi}(\mathbf{q})^\top \mathbf{\phi}(\mathbf{k}) \right] = \exp \mathbf{q}^\top \mathbf{k}\]

Random Fourier features
Rahimi and Recht (2007)
From Attention to Random Feature Attention

\[ \sum_i \frac{\exp q^\top k_i}{\sum_j \exp q^\top k_j} v_i^\top \]

\[ \approx \sum_i \frac{\phi(q)^\top \phi(k_i) \otimes v_i}{\sum_j \phi(q)^\top \phi(k_j)} \]

\[ = \frac{\phi(q)^\top \sum_i \phi(k_i) \otimes v_i}{\phi(q)^\top \sum_j \phi(k_j)} \]

\( q \) query

\( k_i \) keys

\( v_i \) values

Moving \( \phi(q) \) out of the sum
Random Feature Attention

\[ S = \sum_i \phi(k_i) \otimes v_i \]

\[ z = \sum_j \phi(k_j) \]

\[ \text{output} = \frac{S}{\frac{1}{N})} \]

\[ O(N) \]

\[ O(1) \]
Random Feature Attention

output = $\phi(q)^{\top} S / (\phi(q)^{\top} z)$

$S = \sum_i \phi(k_i) \otimes v_i$

$z = \sum_j \phi(k_j)$

$\mathcal{O}(1)$

$\mathcal{O}(N)$
Random Feature Attention

\[
\text{output} = \phi(q)^T S / (\phi(q)^T z)
\]

\[
S = \sum_i \phi(k_i) \otimes v_i
\]

\[
z = \sum_j \phi(k_j)
\]

per step: \(O(1)\)
overall: \(O(N)\)
Random Feature Attention

Construct $\phi$ such that:

$$
\sum_i \frac{\exp q^\top k_i}{\sum_j \exp q^\top k_j} v_i \approx \frac{\phi(q)^\top \sum_i \phi(k_i) \otimes v_i}{\phi(q)^\top \sum_j \phi(k_j)}
$$

• Linear time and constant space in decoding
• Drop-in substitute for softmax attention
• Suitable for finetuning applications
Random Feature Attention

Construct $\phi$ such that:

$$
\sum_i \frac{\exp q^T k_i}{\sum_j \exp q^T k_j} v_i^T \approx \frac{\phi(q)^T \sum_i \phi(k_i) \otimes v_i}{\phi(q)^T \sum_j \phi(k_j)}
$$

- Linear time and constant space in decoding
- Drop-in substitute for softmax attention
- Suitable for finetuning applications
- Size of feature map: 64 or 128
Random Feature Attention
Recurrent Updates

\[
S_t = S_{t-1} + \phi(k_t) \otimes v_t \\
z_t = z_{t-1} + \phi(k_t) \\
\text{output}_t = \phi(q_t)^\top S_t / \phi(q_t)^\top z_t
\]

Applications
- Language model
- Decoder self attention in a sequence-to-sequence model
Random Feature Attention
Recency Bias with Learned Gates

\[ S_t = \eta_t \cdot S_{t-1} + \phi(k_t) \otimes v_t \]
\[ z_t = \eta_t \cdot z_{t-1} + \phi(k_t) \]
\[ \text{output}_t = \phi(q_t) \top S_t / \phi(q_t) \top z_t \]

learned sigmoid gate
\[ \eta_t = \sigma(w \top x + b) \]
Random Feature Attention
Sequence-to-sequence decoding

encoder
feature map
sum over sequence...

source

$O(N)$
Random Feature Attention

Sequence-to-sequence decoding

output

\[ \phi(q_1) \]

causal attn.  
\[ \mathcal{O}(1) \]

\[ (\phi(k_1), v_1) \]

cross attn.  
\[ \mathcal{O}(1) \]
Random Feature Attention
Sequence-to-sequence decoding

causal attn. $O(1)$

$\phi(q_1) \rightarrow \phi(k_1), v_1$ ?

$S_1 \rightarrow z_1$

(output$_1$

$S_2 \rightarrow z_2$

(output$_2$

$\phi(q_2) \rightarrow \phi(k_2), v_2$

$O(1)$
cross attn.

S \rightarrow z$

$\mathcal{O}(1)$
Random Feature Attention
Sequence-to-sequence decoding

output_1
\phi(q_1) \uparrow
S_1 \quad z_1
(\phi(k_1), v_1)

output_2
\phi(q_2) \uparrow
S_2 \quad z_2
(\phi(k_2), v_2)

output_3
\phi(q_3) \uparrow
S_3 \quad z_3
(\phi(k_3), v_3)

 causal attn. \mathcal{O}(1)

 cross attn. \mathcal{O}(1)
Random Feature Attention

Sequence-to-sequence decoding

\[
\begin{align*}
\text{output}_1 & \quad \phi(q_1) \\
S_1 & \quad \phi(k_1), v_1 \\
\text{output}_2 & \quad \phi(q_2) \\
S_2 & \quad \phi(k_2), v_2 \\
\text{output}_3 & \quad \phi(q_3) \\
S_3 & \quad \phi(k_3), v_3 \\
\cdots & \\
S & \quad O(1) \text{ space} \\
& \quad O(N) \text{ time}
\end{align*}
\]
Experiments: Machine Translation

Dataset: WMT’14 (Bojar et al., 2014)
• EN->DE, 4.5M training instances
• EN->FR, 35.8M training instances

Implementation:
• Based on transformer base (Vaswani et al., 2017)
• Replace decoder causal and cross attention with random feature attention
• Random feature size: 64 causal, 128 cross
• Trained for up to 350K steps; beam size 4; average 10 checkpoints
Test set BLEU on WMT’14 EN->DE

- Baseline: 28.1
- RFA, w/o gate: 28
- RFA, w/ gate: 28.1

1.0x Speed
1.9x Speed
1.9x Speed

beam size 4, average 10 checkpoints
Test set BLEU on WMT’14 EN->FR

<table>
<thead>
<tr>
<th>Speed</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0x Speed</td>
<td>39</td>
</tr>
<tr>
<td>1.8x Speed</td>
<td>39.2</td>
</tr>
<tr>
<td>1.8x Speed</td>
<td>39</td>
</tr>
</tbody>
</table>

Baseline: 39
RFA, w/o gate: 39.2
RFA, w/ gate: 39
Experiments with Language Modeling

Dataset:
• WikiText-103 (Merity et al., 2016). 103M training data, 268K vocab size

Implementation:
• Based on Baevski and Auli, 2019
• Replace all self attention with random feature attention
• Random feature size: 64; context window 512, not “stateful”
• All models trained for 150K steps
Wikitext-103 test set perplexity (lower is better)

<table>
<thead>
<tr>
<th></th>
<th>Small</th>
<th>Big</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>34.5</td>
<td>26.2</td>
</tr>
<tr>
<td>RFA, w/o gate</td>
<td>35.7</td>
<td>27.5</td>
</tr>
<tr>
<td>RFA, w/ gate</td>
<td>32.7</td>
<td>25</td>
</tr>
</tbody>
</table>
Decoding Speed & Memory vs. Lengths

### Speed

- **Decoding Speed:** Tokens/s
- **Sequence Length:** $2^3$ to $2^{11}$

### Memory

- **Memory:** GB
- **Sequence Length:** $2^3$ to $2^{11}$

**Graphs:**
- **Softmax**
- **RFA**
Wrap-up

• RFA:
  • Linear complexity attention with random feature methods
  • Well-suited for tasks involving long sequences
  • Recurrent style update; intuitive ways to connect to gated RNNs

• Experiments:
  • Strong performance in language modeling and machine translation
  • 1.9x speed up in MT decoding; more for longer text
  • The only model that is competitive in both efficiency and accuracy in long text classification (Tay et al., 2021)
Wrap-up

• RFA:
  • Linear complexity attention with random feature methods
  • Well-suited for tasks involving long sequences
  • Recurrent style update; intuitive ways to connect to gated RNNs

• Experiments:
  • Strong performance in language modeling and machine translation
  • 1.9x speed up in MT decoding; more for longer text
  • The only model that is competitive in both efficiency and accuracy in long text classification (Tay et al., 2021)

• Notes:
  • Harder to achieve time saving when input is fully revealed: encoder, teacher-forcing training
  • Using 128/64 feature maps; smaller ones works with larger batches
Thank You!

collaborators

[QR codes for paper and code]

[Logos: Paul G. Allen School of Computer Science & Engineering, Allen Institute for Artificial Intelligence, The Hebrew University of Jerusalem, DeepMind]