LANGUAGE MODELING WITH GATED CONVOLUTIONAL NETWORKS
YANN N. DAUPHIN, ANGELA FAN, MICHAEL AULI AND DAVID GRANGIER
FACEBOOK AI RESEARCH
CS 546 Paper Presentation
Jinfeng Xiao
2/22/2018
Intro: Language Models

■ Full model:

\[ P(w_0, \ldots, w_N) = P(w_0) \prod_{i=1}^{N} P(w_i|w_0, \ldots, w_{i-1}) \]

■ n-gram model:

\[ P(w_i) = P(w_i|w_{i-n+1}, \ldots, w_{i-1}) \]

■ Hard to represent long-range dependencies, due to data sparsity
Intro: LSTM

http://colah.github.io/posts/2015-08-Understanding-LSTMs/
Intro: LSTM

- State-of-the-art neural network approach for language modeling
- + Can theoretically model arbitrarily long dependencies
- -- Not parallelizable; $O(N)$ operations

http://colah.github.io/posts/2015-08-Understanding-LSTMs/
Intro: CNN

- Predict the current word $y$ with previous words $x$ (i.e. context)
- Model long-term dependencies with $O(N/k)$ operations
This Paper: GCNN

- Gated Convolutional Neural Networks
- Each CNN layer is followed by a gating layer
- Allows parallelization over sequential tokens
- Reduces the latency to score a sentence by an order of magnitude
- Competitive performance on WikiText-103 and Google Billion Words benchmarks
Architecture

- Word Embedding +
- CNN +
- Gating
Architecture

- Word Embedding +
- CNN +
- Gating
Architecture

- Word Embedding +
- **CNN** +
- Gating

\[ E = D_{w_i} \]

\[ A = E \cdot W + b \]

\[ B = E \cdot V + c \]

\[ *: \text{Convolution operation} \]
Architecture
- Word Embedding +
- CNN +
- Gating

Learned parameters
Example: Convolution

\[
\begin{bmatrix}
a & b & c \\
d & e & f \\
g & h & i \\
\end{bmatrix}
\times
\begin{bmatrix}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9 \\
\end{bmatrix}
\]

\[2,2 = (i \cdot 1) + (h \cdot 2) + (g \cdot 3) + (f \cdot 4) + (e \cdot 5) + (d \cdot 6) + (c \cdot 7) + (b \cdot 8) + (a \cdot 9).\]

- “Average” over a small patch around an element
Architecture

- Word Embedding +
- CNN +
- Gating

\[
A = E \cdot W + b \\
B = E \cdot V + c \\
H_0 = A \odot \sigma(B) \\
Y = \text{softmax}(WH_L)
\]
Two Gating Mechanisms

- Gated linear units (GLU)
  \[ h_t(E) = (E \ast W + b) \otimes \sigma(E \ast V + c) \]

- Gated tanh units (GTU)
  \[ h_t(E) = \tanh(E \ast W + b) \otimes \sigma(E \ast V + c) \]
Evaluation Metric: Perplexity

- The perplexity of a discrete probability distribution $p$ is
  \[
  \frac{1}{e \cdot N} \sum_{i=1}^{N} -\log p(w_i|\ldots,w_{i-1})
  \]
- It measures how well our model matches the held out test data set.
- The smaller, the better.

https://en.wikipedia.org/wiki/Perplexity
Benchmark: Google Billion Word
Average Sequence Length = 20 Words

<table>
<thead>
<tr>
<th>Model</th>
<th>Test PPL</th>
<th>Hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sigmoid-RNN-2048 (Ji et al., 2015)</td>
<td>68.3</td>
<td>1 CPU</td>
</tr>
<tr>
<td>Interpolated KN 5.Gram (Chelba et al., 2013)</td>
<td>67.6</td>
<td>100 CPUs</td>
</tr>
<tr>
<td>Sparse Non-Negative Matrix LM (Shazeer et al., 2014)</td>
<td>52.9</td>
<td></td>
</tr>
<tr>
<td>RNN-1024 + MaxEnt 9 Gram Features (Chelba et al., 2013)</td>
<td>51.3</td>
<td>24 GPUs</td>
</tr>
<tr>
<td>LSTM-2048-512 (Jozeñowicz et al., 2016)</td>
<td>43.7</td>
<td>32 GPUs</td>
</tr>
<tr>
<td>2-layer LSTM-8192-1024 (Jozeñowicz et al., 2016)</td>
<td>30.6</td>
<td>32 GPUs</td>
</tr>
<tr>
<td>BIG GLSTM-G4 (Kuchaiev &amp; Ginsburg, 2017)</td>
<td>23.3*</td>
<td>8 GPUs</td>
</tr>
<tr>
<td>LSTM-2048 (Grave et al., 2016a)</td>
<td>43.9</td>
<td>1 GPU</td>
</tr>
<tr>
<td>2-layer LSTM-2048 (Grave et al., 2016a)</td>
<td>39.8</td>
<td>1 GPU</td>
</tr>
<tr>
<td>GCNN-13</td>
<td>38.1</td>
<td>1 GPU</td>
</tr>
<tr>
<td>GCNN-14 Bottleneck</td>
<td>31.9</td>
<td>8 GPUs</td>
</tr>
</tbody>
</table>

\[ \text{ReLU}(X) = X \otimes (X > 0) \]
GCNN Is Faster
On Google Billion Words
Benchmark: WikiText-103
Average Sequence Length = 4,000 Words

<table>
<thead>
<tr>
<th>Model</th>
<th>Test PPL</th>
<th>Hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM-1024 (Grave et al., 2016b)</td>
<td>48.7</td>
<td>1 GPU</td>
</tr>
<tr>
<td>GCNN-8</td>
<td>44.9</td>
<td>1 GPU</td>
</tr>
<tr>
<td>GCNN-14</td>
<td>37.2</td>
<td>4 GPUs</td>
</tr>
</tbody>
</table>
Short Context Size Suffices

Google Billion Word
Avg. Text Length = 20

Wiki-103
Avg. Text Length = 4,000
Summary

- GCNN: CNN + Gating
- Perplexity is comparable with the state-of-the-art LSTM
- GCNN converges faster and allows parallelization over sequential tokens
- The simpler linear gating (GLU) works better than LSTM-like tanh gating (GTU)