Efficient Deep Feature Learning and Extraction via StochasticNets

Mohammad Javad Shafiee, Parthipan Siva Paul Fieguth and Alexander Wong.

Presentation by Ivan Slobozhan
Embedded devices and DNN

- Low-power CPU
- Custom embedded CPU not flexible
- Addition costs
Way how to improve

• Synaptic formation is stochastic in nature
Random Graphs

• Common random graph representation: $G(n,p)$, where $0 < p < 1$.
• Generalized random graph model: $G(V, p_{ij}), p_{i,j} \in [0,1]$
Random Graphs

• Start with $n$ vertices $V = \{v_q | 1 \leq q \leq n\}$ and added $E = \{e_{ij} | 1 \leq i \leq n, 1 \leq j \leq n, i \neq j\}$.
**StochasticNets: Deep Neural Networks As Random Graph Realizations**

- $G(V, p(i \rightarrow j))$, where $V$ is a set of neurons $V = \{v_i | 1 \geq i \geq n\}$, with $v_i$ denoting the $i^{th}$ neuron and $n$ denoting the total number of neurons and $p(i \rightarrow j)$ is the probability that neural connection occurs between neuron $v_i$ and $v_j$. 
Constructing DNN as Random Graph

• Ensure the properties of the network architecture
FNN as a StochasticNet

- No neural connections between non-adjacent layers.
- No neural connections between neurons on the same layer.

\[ p_{i \rightarrow j | k \rightarrow h} = 0, \text{ when } i = j \mid |i - j| > \]
Realization of FFN as a StochasticNet

FFN

FFN Graph Realization
### Relationship to Other Methods

<table>
<thead>
<tr>
<th>Network Model</th>
<th>Network Formation</th>
<th>Training</th>
<th>Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dropout</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deep-Connect</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HashNet</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>StochasticNet</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Feature learning via Deep Convolutional Stochastic Nets

• Neural connectivity in the convolutional layers are arranged such that small spatially localized neural collections are connected to the same output neuron in the next layer.
Deep convolutional StochasticNet
Spatial neural connectivity models

\[ p(i \rightarrow j) = \begin{cases} 
    U(0, 1), j \in R_i \\
    0, \text{otherwise} 
\end{cases} \quad \text{Uniform} \]

\[ p(i \rightarrow j) = \begin{cases} 
    N(i, \sigma), j \in R_i \\
    0, \text{otherwise} 
\end{cases} \quad \text{Gaussian} \]
CIFAR-10 dataset

Gaussian

Uniform
STL-10

Gaussian

Uniform
Extraction time versus the number of neural connections
Classification using Uniform model

(a) SVHN

(b) STL-10
Classification using Gaussian model
Conclusion

• DNN could be effectively constructed by stochastic connectivity between neurons.
• StochasticNets reduce computation complexity
• Provide better or close to ordinary CNN results in feature extractions