CLSP Seminar

Feb 28. 2020
Speaker 1

Xuan Zhang
Hyperparameter Optimization of Neural Machine Translation Systems

Xuan Zhang, Kevin Duh
Overview

0. Introduction to Hyperparameter Optimization (HPO)

I. A new benchmarking dataset for HPO of NMT systems

II. A new HPO method
Overview

0. Introduction to Hyperparameter Optimization (HPO)

I. A new benchmarking dataset for HPO of NMT systems

II. A new HPO method
0. Hyperparameter Optimization - Motivation

• Choosing effective hyperparameters is crucial for building strong NMT systems: *initial learning rate, batch size, etc.*
Choosing effective hyperparameters is crucial for building strong NMT systems: *initial learning rate, batch size, etc.*

Manual tuning is tedious and unreliable, especially when optimizing multiple objectives: *BLEU & decoding time.*
0. Hyperparameter Optimization - Approach

Goal: find a hyperparameter configuration with as few evaluations of $f(\cdot)$ as possible.
0. Hyperparameter Optimization - Challenges

• State-of-the-art NMT models require significant **computational resources** for training.

• NMT models usually have large hyperparameter **search space**.

-> It is expensive to compare HPO methods on NMT tasks.
Overview

0. Introduction to Hyperparameter Optimization (HPO)

I. A new **benchmarking dataset** for HPO of NMT systems

II. A new **HPO method**
I. Table-Lookup Datasets for NMT HPO

• First, we pretrain a large number of NMT systems covering a wide range of hyperparameter configurations, and record their performance metrics.

• Then, we constrain our HPO methods to sample from this finite set of models.
I. Table-Lookup Datasets for NMT HPO

• MT Data:

• Model: Transformer

• Hyperparameters:
  preprocessing configurations: bpe
  training settings: initial learning rate
  architecture designs: #layers, embed, #hidden, #heads in self-attention

• Objectives:
  translation accuracy: dev BLEU, dev perplexity
  computational cost: decoding time, training time, GPU memory for training, total number of model parameters
I. Table-Lookup Datasets for NMT HPO

<table>
<thead>
<tr>
<th>dataset</th>
<th>bpe (1k)</th>
<th>#layers</th>
<th>#embed</th>
<th>#hidden</th>
<th>#att_heads</th>
<th>init_lr $(10^{-4})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>zh, ru, ja, en</td>
<td>10, 30, 50</td>
<td>2, 4</td>
<td>256, 512, 1024</td>
<td>1024, 2048</td>
<td>8, 16</td>
<td>3, 6, 10</td>
</tr>
<tr>
<td>sw</td>
<td>1, 2, 4, 8, 16, 32</td>
<td>1, 2, 4, 6</td>
<td>256, 512, 1024</td>
<td>1024, 2048</td>
<td>8, 16</td>
<td>3, 6, 10</td>
</tr>
<tr>
<td>so</td>
<td>1, 2, 4, 8, 16, 32</td>
<td>1, 2, 4</td>
<td>256, 512, 1024</td>
<td>1024, 2048</td>
<td>8, 16</td>
<td>3, 6, 10</td>
</tr>
</tbody>
</table>

Table 1: Hyperparameter search space for the NMT systems.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#models</th>
<th>Best BLEU</th>
<th>bpe</th>
<th>#layers</th>
<th>#embed</th>
<th>#hidden</th>
<th>#att_heads</th>
<th>init_lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>zh-en</td>
<td>118</td>
<td>14.66</td>
<td>30k</td>
<td>4</td>
<td>512</td>
<td>1024</td>
<td>16</td>
<td>3e-4</td>
</tr>
<tr>
<td>ru-en</td>
<td>176</td>
<td>20.23</td>
<td>10k</td>
<td>4</td>
<td>256</td>
<td>2048</td>
<td>8</td>
<td>3e-4</td>
</tr>
<tr>
<td>ja-en</td>
<td>150</td>
<td>16.41</td>
<td>30k</td>
<td>4</td>
<td>512</td>
<td>2048</td>
<td>8</td>
<td>3e-4</td>
</tr>
<tr>
<td>en-ja</td>
<td>168</td>
<td>20.74</td>
<td>10k</td>
<td>4</td>
<td>1024</td>
<td>2048</td>
<td>8</td>
<td>3e-4</td>
</tr>
<tr>
<td>sw-en</td>
<td>767</td>
<td>26.09</td>
<td>1k</td>
<td>2</td>
<td>256</td>
<td>1024</td>
<td>8</td>
<td>6e-4</td>
</tr>
<tr>
<td>so-en</td>
<td>604</td>
<td>11.23</td>
<td>8k</td>
<td>2</td>
<td>512</td>
<td>1024</td>
<td>8</td>
<td>3e-4</td>
</tr>
</tbody>
</table>

Table 2: For each language pair, we report the number of NMT systems trained on it, the oracle best BLEU we obtained and its corresponding hyperparameter configuration.
Overview

0. Introduction to Hyperparameter Optimization (HPO)

I. A new *benchmarking dataset* for HPO of NMT systems

II. A new **HPO method**
II. Graph-Based Semi-Supervised Learning as a HPO Method

• **Semi-supervised learning:** utilize a handful of labeled data and a large amount of unlabeled data to improve prediction accuracy.

• **Graph-Based Semi-Supervised Learning (Zhu et al., 2003):** describes the structure of data with a graph.

  *vertex:* (data point, label) $\rightarrow$ (configuration, model performance)
  *edge weight:* similarity between vertices $\rightarrow$ similarity between configurations
  *smoothness:* neighbors connected by edges tend to have similar labels.

**Predict:** Labels can propagate throughout the graph.