Reproducible and Efficient Benchmarks for Hyperparameter Optimization of Neural Machine Translation Systems

Xuan Zhang    Kevin Duh
Machine Translation

Architecture Hyperparameters:
- #layer
- #units/layer
- #embed

Training Hyperparameters:
- optimizer type
- learning rate
- batch size

Objectives

Training Accuracy:
- BLEU
- perplexity

Computational Cost:
- inference speed
- model size

Machine Translation

Architecture Hyperparameters:
- #layer
- #units/layer
- #embed

Training Hyperparameters:
- optimizer type
- learning rate
- batch size

⚠️ Exponential explosion of choices!

Objectives

Training Accuracy:
- BLEU
- perplexity

Computational Cost:
- inference speed
- model size

\[1\text{Vaswani, Ashish, et al, “Attention is all you need.” Advances in neural information processing systems. 2017.}\]
Machine Translation

Architecture Hyperparameters:
- #layer
- #units/layer
- #embed

Training Hyperparameters:
- optimizer type
- learning rate
- batch size

⚠️ Exponential explosion of choices!

Objectives

Training Accuracy:
- BLEU
- perplexity

Computational Cost:
- inference speed
- model size

⚠️ Difficult to optimize multiple objectives!

---

Machine Translation

Architecture Hyperparameters:
- #layer
- #units/layer
- #embed

Training Hyperparameters:
- optimizer type
- learning rate
- batch size

⚠️ Exponential explosion of choices!

⚠️ Require significant computational resources to train!

The Transformer model architecture.¹

Objectives

Training Accuracy:
- BLEU
- perplexity

Computational Cost:
- inference speed
- model size

⚠️ Difficult to optimize multiple objectives!

Let $\lambda$ be the hyperparameters of a ML algorithm with domain $\Lambda$, $L(\lambda, D_{\text{train}}, D_{\text{valid}})$ denote the loss of the ML algorithm, using hyperparameters $\lambda$ trained on $D_{\text{train}}$ and evaluated on $D_{\text{valid}}$.

The HPO problem is to find a configuration $\lambda^*$ that minimizes this loss:

$$\lambda^* \in \arg\min_{\lambda \in \Lambda} L(\lambda, D_{\text{train}}, D_{\text{valid}})$$
Hyperparameter Optimization (HPO)

Model-Free Optimization Methods:
- Grid Search
- Random Search
- Population-based methods
e.g. genetic algorithms, evolutionary algorithms --- CMA-ES

Sequential Model-Based Optimization Methods (SMBO):
- Bayesian Optimization
- Tree Parzen Estimator
Hyperparameter Optimization (HPO)

Model-Free Optimization Methods:
- Grid Search
- Random Search
- Population-based methods (e.g. genetic algorithms, evolutionary algorithms) --- CMA-ES

Sequential Model-Based Optimization Methods (SMBO):
- Bayesian Optimization
- Tree Parzen Estimator

This talk

Algorithms
Hyperparameter Optimization (HPO)

\( f(\lambda) \): model performance (e.g. BLEU)

\( \lambda \): hyperparameter configuration

SMBO Framework
Hyperparameter Optimization (HPO)

Hyperparameter Optimization (HPO)

\[ f(\lambda) \]

\( \lambda \): hyperparameter configuration

\( f(\lambda) \): model performance (e.g. BLEU)

⚠️ Each model takes days to weeks to train.
Hyperparameter Optimization (HPO)

Large search space.

Each model takes days to weeks to train.

\[ \lambda: \text{hyperparameter configuration} \]

\[ f(\lambda): \text{model performance (e.g. BLEU)} \]
HPO algorithms are expensive. It is not feasible to run too many experiments and compare different HPO algorithms on NMT tasks in practice.

Li and Talwalkar (2019)\(^1\): “Of the 12 papers published since 2018 at NeurIPS, ICML, and ICLR that introduce novel Neural Architecture Search methods, none are exactly reproducible.”

Enable **reproducible** Hyperparameter Optimization (HPO) research on Neural Machine Translation (NMT) tasks.
Contributions

– 01 –
DATASET

We release a benchmark dataset for comparing HPO methods on NMT models.

– 02 –
BENCHMARKS

We benchmark the performance of several HPO methods on both single-objective and multiobjective optimization on our dataset.

– 03 –
ALGORITHM

We propose a novel graph-based HPO method.
Contributions

- 01 – DATASET
  We release a benchmark dataset for comparing HPO methods on NMT models.

- 02 – BENCHMARKS
  We benchmark the performance of several HPO methods on both single-objective and multiobjective optimization on our dataset.

- 03 – ALGORITHM
  We propose a novel graph-based HPO method.
Dataset

Table-Lookup Framework

Procedure:
1. Train a large number of NMT systems with diverse hyperparameter configurations and record their performance.
2. Constrain HPO methods to sample from this finite set of models.
Procedure:
1. Train a large number of NMT systems with diverse hyperparameter configurations and record their performance.
2. Constrain HPO methods to sample from this finite set of models.

Benefits:
1. Allows HPO developers to simply lookup the performance of NMT systems without training them.
2. Reproducible and efficient HPO experiments.

Limitations:
Table needs to be large enough to cover the hyperparameter space.
**Dataset**

- **6 MT Corpora:**
  - large resource (WMT2019 Robustness): ja-en, en-ja (4M lines)
  - mid resource (TED Talks): zh-en, ru-en (170k lines)
  - low resource (IARPA MATERIAL): sw-en, so-en (24k lines)

**Search Space:** 2245 Transformers (1547 GPU days)

<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>

**Objectives:**

- BLEU, perplexity;
- decoding time, #updates, GPU memory, #model parameters
Dataset

Application

HPO Algorithm Selection

Develop a robust HPO algorithm

HPO A

HPO B

Target MT Data

MT Data a

MT Data b

MT Data c

Evaluate

Apply

1

2

3

Reproducible and Efficient Benchmarks for Hyperparameter Optimization of Neural Machine Translation Systems
Dataset

Application

HPO Algorithm Selection

Single-objective Optimization

Reproducible and Efficient Benchmarks for Hyperparameter Optimization of Neural Machine Translation Systems

19
Dataset

Application

HPO Algorithm Selection

Single-objective Optimization

Reproducible and Efficient Benchmarks for Hyperparameter Optimization of Neural Machine Translation Systems
Dataset

Application

HPO Algorithm Selection

Single-objective Optimization

fix the target BLEU

Dataset

 BLEU

#NMT models

RS
BO_EI_M
BO_EI_R
GB_EI_M
GB_EI_R
GB{EIF_M
GB{EIF_R

HPO Algorithm Selection

Single-objective Optimization

Dataset

BLEU

#NMT models

RS
BO_EI_M
BO_EI_R
GB_EI_M
GB_EI_R
GB{EIF_M
GB{EIF_R

Reproducible and Efficient Benchmarks for Hyperparameter Optimization of Neural Machine Translation Systems
Dataset

Application

HPO Algorithm Selection

Single-objective Optimization

Multiobjective Optimization

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reproducible and Efficient Benchmarks for Hyperparameter Optimization of Neural Machine Translation Systems</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reproducible and Efficient Benchmarks for Hyperparameter Optimization of Neural Machine Translation Systems</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reproducible and Efficient Benchmarks for Hyperparameter Optimization of Neural Machine Translation Systems</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
</tr>
</tbody>
</table>
Dataset

Hyperparameter Importance

Application

HPO Algorithm Selection

Single-objective Optimization  Multiobjective Optimization

Hyperparameter Analyses
Dataset

Model Ranking Correlation

Application

HPO Algorithm Selection

Single-objective Optimization

Multiobjective Optimization

Hyperparameter Analyses
Summary

We provide a tabular dataset for comparing HPO methods on NMT models.

- **Our benchmarks are reproducible.**
  Dataset and code are publicly available.

- **Our benchmarks are efficient.**
  One can perform multiple random trials of the same algorithm to test robustness.

😊 Feel free to utilize our dataset to develop your new HPO methods.
Reproducible and Efficient Benchmarks for Hyperparameter Optimization of Neural Machine Translation Systems

PAPER
https://www.mitpressjournals.org/doi/pdf/10.1162/tacl_a_00322

DATASET
https://github.com/Este1le/hpo_nmt

CODE
https://github.com/Este1le/gbopt