Efficient Identification of Approximate Best Configuration of Training in Large Datasets

Silu Huang¹, Chi Wang², Bolin Ding³, Surajit Chaudhuri²

¹University of Illinois (UIUC)  ²Microsoft Research  ³Alibaba Group

Work done in Microsoft
Background: **Large Configuration Space**

Machine Learning Pipeline

- Data Loading
- Data/Feature Preprocessing
- Model Selection
- Hyperparameter Tuning
- Score Evaluation

So **MANY** choices
- Which transformation?
- Which model?
- Which hyperparameters?
Related Work: **AutoML**

**Machine Learning Pipeline**

- Data Loading
- Data/Feature Preprocessing
- Model Selection
- Hyperparameter Tuning
- Score Evaluation

**AutoML**

- Bayesian Optimization
- Meta Learning
- Successive-halving
Motivation: Large Scale Dataset

Even running one single configuration is time-consuming

Sampling ➔ Extrapolate over training set size?

Different perspective

Related works:

Bayesian ➔ Extrapolate across model configurations

Meta Learning ➔ Combined as our input

Successive-halving ➔ Heuristic, no guarantee
Problem: **Approximate Configuration Identification**

**Input**
- A set of candidate configurations

**Output**
- Best configuration
- Approximate best configuration
- Tolerance

\[ A(\text{best}) - A(\text{approximate best}) \leq \varepsilon \]
Outline

• CI-based Framework

• CI Estimator

• Scheduler

• Experimental Evaluation
Outline

- CI-based Framework
  - CI Estimator
  - Scheduler
- Experimental Evaluation
Preliminary Exp: Learning Curve

FlightDelay:
7.3M Records
630 Attributes
Preliminary Exp: Learning Curve

Naïve plateau estimator is error-prone

FlightDelay:
7.3M Records
630 Attributes
Proposal: CI-based framework

Each Probe: Lower Bound → Upper Bound

Non-Overlapping ⇒ Prune $C_2$

$C_1$ $C_2$

Large Small
Proposal: CI-based framework

$C_1$ $C_2$ $C_3$ $C_4$ $C_5$
Outline

• CI-based Framework

• CI Estimator

• Scheduler

• Experimental Evaluation
Single Configuration: CI Estimator

1. **Training Dataset**
   - Train Model: $A_{tn}$
   - Evaluate Model: $A_{te}$

2. **Sampled Training Dataset**
   - Training Accuracy: $A_{tn}$
   - Test Accuracy: $A_{te}$

3. **Test Dataset**
   - Sampled Test Dataset

4. **CI Estimator**
   - Upper Bound: $A_{tn}$
   - Lower Bound: $A_{te}$
Fitness Assumption:

\[ D \rightarrow H \]
\[ D' \rightarrow H' \]

Then, \( A(H, D) \geq A(H', D) \)

With Probability at least \((1 - \delta)\)
Outline

• CI-based Framework

• CI Estimator

• Scheduler

• Experimental Evaluation
Multi Configurations: Scheduler

What’s the Probe Sequence among all Configurations?

$C_1$, $C_2$, $C_3$, $C_4$, $C_5$
Scheduler: Lagrange Multiplier Method

Minimize: \( \sum t_i \)

Identify the Best Configuration

s.t.

\[ u_2 \leq l_1 \]
\[ u_3 \leq l_1 \]
[...]

\[ u_m \leq l_1 \]

Apply Lagrange Method

Two Conditions:

\[ \frac{dt_1}{dl_1} = -\left( \frac{dt_2}{du_2} + \cdots + \frac{dt_m}{du_m} \right) \]

\[ l_1 = u_2 = \cdots = u_m \]
Multi Configurations: Scheduler

**Optimal Solution**

\[
\frac{dt_1}{dl_1} = -\left( \frac{dt_2}{du_2} + \cdots + \frac{dt_n}{du_n} \right)
\]

\[l_1 = u_2 = \cdots = u_n\]

**Make a Guess on the best configuration \(C_1\)**

- If \(\frac{dt_1}{dl_1} > -\sum_{i=2}^{n} \frac{dt_i}{du_i}\), then
- **No**: Probe \(C_1\)
- **Yes**:
  - **Probe \(C_1^*\) with second highest upper bound**
  - Based on \(u_2 = \cdots = u_n\)

**Highest upper bound**
Outline

• CI-based Framework

• CI Estimator

• Scheduler

• Experimental Evaluation
## Datasets

| Dataset        | \( |D| \) | \( |F| \) | Origin                                      |
|----------------|--------|--------|---------------------------------------------|
| TwitterSentiment | 1.4M   | 9866   | Twitter, Stanford                           |
| FlightDelay     | 7.3M   | 630    | U.S. Department of Transportation           |
| NYCTaxi         | 10M    | 21     | NYC Taxi & Limousine Commission             |
| HEPMASS         | 10M    | 28     | UCI                                         |
| HIGGS           | 10.6M  | 28     | UCI                                         |
Experiments

• Experiment I: Efficiency Compared to Full Run
• Experiment II: Effectiveness Compared to Full Run
• Experiment III: Compare with Successive-Halving
ABC vs. Full-run: Efficiency Comparison
ABC vs. Full-run: Effectiveness Comparison

![Graph showing test accuracy comparison between Full-run and ABC across different datasets and configurations.](image-url)
ABC vs. Successive-halving: Efficiency & Effectiveness

- CI-based pruning
- Successive-halving
ABC vs. Successive-halving: $\epsilon$-Guarantee