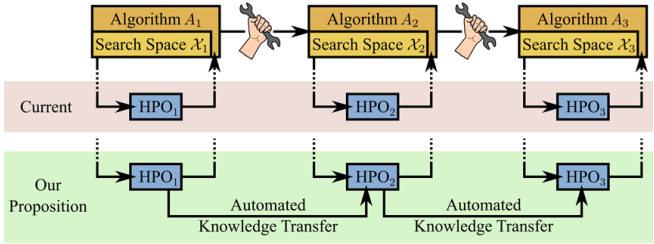
Hyperparameter Transfer Across Developer Adjustments

Danny Stoll[1], **Jörg K.H. Franke**[1], **Diane Wagner**[1], **Simon Selg**[1] & **Frank Hutter**[1,2]

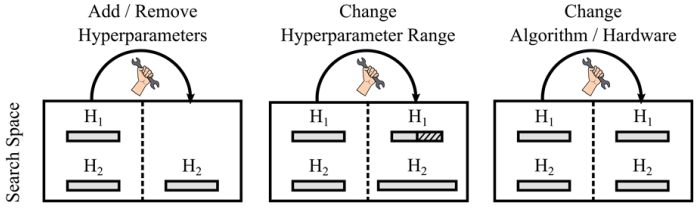
[1]University of Freiburg [2]Bosch Center for Artificial Intelligence

In a Nutshell

- Developer of ML algorithms often perform adjustments like adding dropout.
- **Idea:** Meta learn across developer adjustments, not across tasks.
- We apply the across adjustments idea to hyperparameter optimization (HPO).



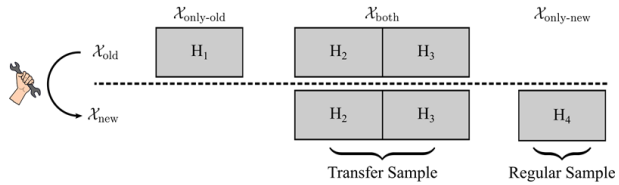
Type of Developer Adjustments



Benchmarks

FCN-A	Increase #units-per-layer 16×; Double #epochs; Fix batch size hyperparameter
FCN-B	Introduce per-layer choice of activation function; Change learning rate schedule from constant to cosine decay
NAS-A	Add 3x3 average pooling as choice of operation to each edge
NAS-B	Add node to cell template (adds 3 hyperparameters)
XGB-A	Expose four booster hyperparameters
XGB-B	Change four unexposed booster hyperparameter values
SVM-A	Change kernel; Remove hyperparameter for old kernel; Introduce hyperparameter for new kernel
SVM-B	Increase range for cost hyperparameter

Baseline Algorithms



- **No Transfer:** Tree-Structured Parzen Estimator (TPE) algorithm (Bergstra et al., 2011).
- **Only Optimize New Hyperparameters:** Use best values for hyperparameters in $\mathcal{X}_{\text{both}}$.
- **Drop Unimportant Hyperparameters:** Drop the 50% most unimportant hyperparameters in $\mathcal{X}_{\text{both}}$. Use fANOVA (Hutter et al., 2014) to measure importance.
- **Best First:** Use only optimize new for only the first evaluation.
- **Transfer TPE:** Use TPE model over $\mathcal{X}_{\text{both}}$ combined with random samples over $\mathcal{X}_{\text{only-new}}$.

Experimental Setup

- We measure the speedup to reach a reference objective over TPE without transfer.
- We look at different budgets for the old and new HPO (10, 20, and 40 respectively).
- We measure the speedup across repetitions for each task, and then aggregate across tasks for each benchmark.

Results

- Best first, transfer TPE, and their combination lead to large average **speedups up to 1.2-2.9x**.
- Only optimize new and drop unimportant never reach the objective in 20-70% of cases.