## A BAYESIAN OPTIMIZATION WITH DEEP RANKING ENSEMBLES

Once the Deep Ensembles are trained, we aggregate the predictions for an input $x$ following the procedure explained in Section 3.2 to obtain $\mu(x), \sigma(x)$ and conditioning to a set of observations $\mathcal{D}_s$. For the sake of simplicity, we omit this conditioning in our notation. These outputs can be fed in several types of acquisition functions and decide for the next point $x$ to observe from the set of pending points to evaluate $\mathcal{X}$. Notice that the lower rank, the better the configuration, therefore we formulate the cast the acquisition function as a minimization problem. Specifically, we consider:

- **Average Rank**: $\alpha(x_j) = \mu(x_j)$
- **Lower Confidence Bound**: $\alpha(x_j) = \mu(x_j) - \beta \cdot \sigma(x_j)$
- **Expected Improvement**: $\alpha(x_j) = \int_r \max(0, \mu(x_k) - r)\, \mathcal{N}(r; \mu(x_j), \sigma(x_j))$

Where $\beta$ is a factor that trades of exploitation and exploration and $x_i$ is the best-observed configuration, i.e. $k = \arg\min_{i \in \{1,\dots,|\mathcal{D}_s|\}} y_i$ and $\mu(x_k)$ is the average rank predicted for that configuration. The previous formulation assumes a minimization, thus to choose the next query point you apply: $x = \arg\min_{x_j \in \mathcal{X}} \alpha(x_j)$.

**Algorithm 2:** Bayesian Optimization with DRE

**Input :** A prior distribution over datasets $p(\mathcal{D})$, initial observations $H = \{(x_1, y_1), ..., (x_N, y_N)\}$, pending points $\mathcal{X}$, number of BO iterations $K$, black-box function to optimize $f$

**Output:** Best observed configuration $x_*$

```
Train ensemble of MLP scorers following Algorithm 1 and prior p(D);
for j ← 1 to K do
    Suggest next candidate x = arg min_{x_j ∈ X} α(x_j, H) ;
    Observe response y = f(x) ;
    Update history H = H ∪ {(x, y)};
end
Return top performing configuration: arg min_{(x_i, y_i) ∈ H} y_i
```

## B EXPERIMENTAL SETUP FOR DEEP RANKING ENSEMBLES

**Meta-Feature Extractor** The DRE model has two configurable components: the meta-feature network and the scorers. The meta-feature extractor is a DeepSet with an architecture similar to the one used by Jomaa et al. (2021a). However, we used 2 fully connected layers with 32 neurons each for both $\phi$ and $\rho$ (Deep Set parameters) instead of 3 fully connected layers. The output size is set to 16 by default.

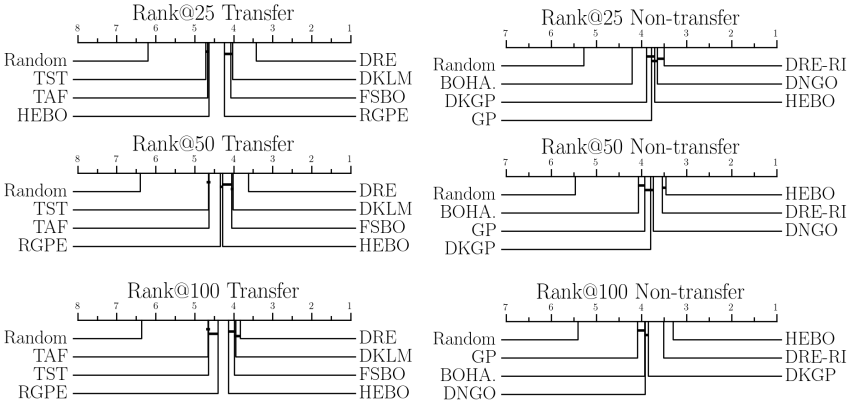
**Ensemble of Scorers** The ensemble of scorers is a group of 10 MLP (Multilayer Perceptrons) with identical architectures. Each neural network has 4 hidden layers and each hidden layer has 32 neurons. The neural networks are initialized independently and randomly (for DRE-RI) or warm-initialized with the meta-learned weights. The input size of each neural network is 16 (the dimesionality of the meta-features), plus the HP search space dimensionality. their output size is 1.

**Setup for Motivating Example.** For the creation of the Figure 2, we use as scorer network an MLP with 2 hidden layers and 10 neurons per layer. The meta-feature extractor has 4 layers and 10 neurons, and output dimensions equal to 10. The network is meta-trained for 1000 epochs, with batch size 10, learning rate 0.001, Adam Optimizer, and 10 models in the ensemble. For the meta-learning example, we do not fine-tune the networks, while we fine-tune the networks for the non-meta-learned example for 500 iterations.

## C ADDITIONAL PLOTS

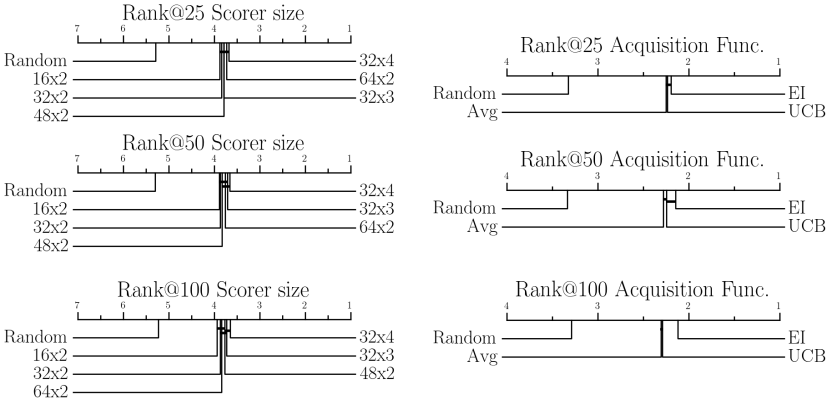
We present additional results on the critical difference diagrams for *i)* Transfer methods results (Figure 8a), *ii)* Non-Transfer (Figure 8b, *iii*) Scorer size (Figure 9a, *iv*) Acquisition Function (Figure

9b, *v* Ranking Loss (Figure 10a) and *vi* Meta-features (Figure 10b). These CD plots show the comparison of the performance at different number of trials (e.g. at 25 trials = Rank@25). The vertical lines connecting two methods indicate that their performances are not significantly different.




(a) Comparison vs. transfer methods (b) Comparison vs. non-transfer methods

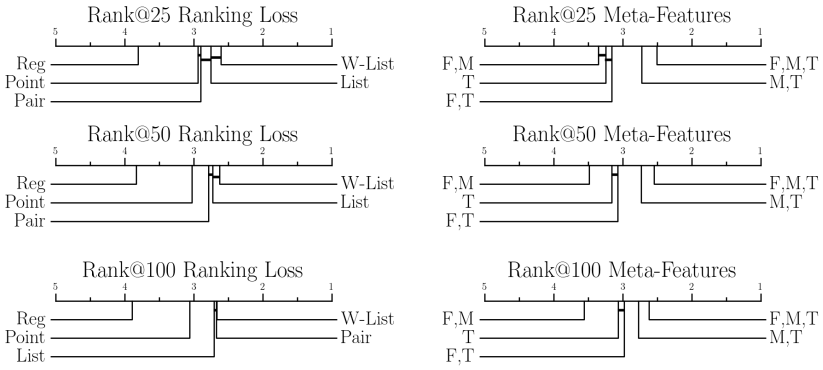
Figure 8: Critical Difference Diagram for a) Transfer and b) Non-transfer.




(a) Ablation of the DRE Scorer size

(b) Ablation of the Acquisition Function

Figure 9: Critical Difference Diagram for the results of the ablation of DRE hyperparameters in (a) and the choice of the acquisition function from Hypothesis 5 in (b).




(a) Ablation of Ranking Loss (b) Ablation of Meta-Features

Figure 10: Critical Difference Diagrams for the results of Hypothesis 3 in a) and Hypothesis 4 in b).

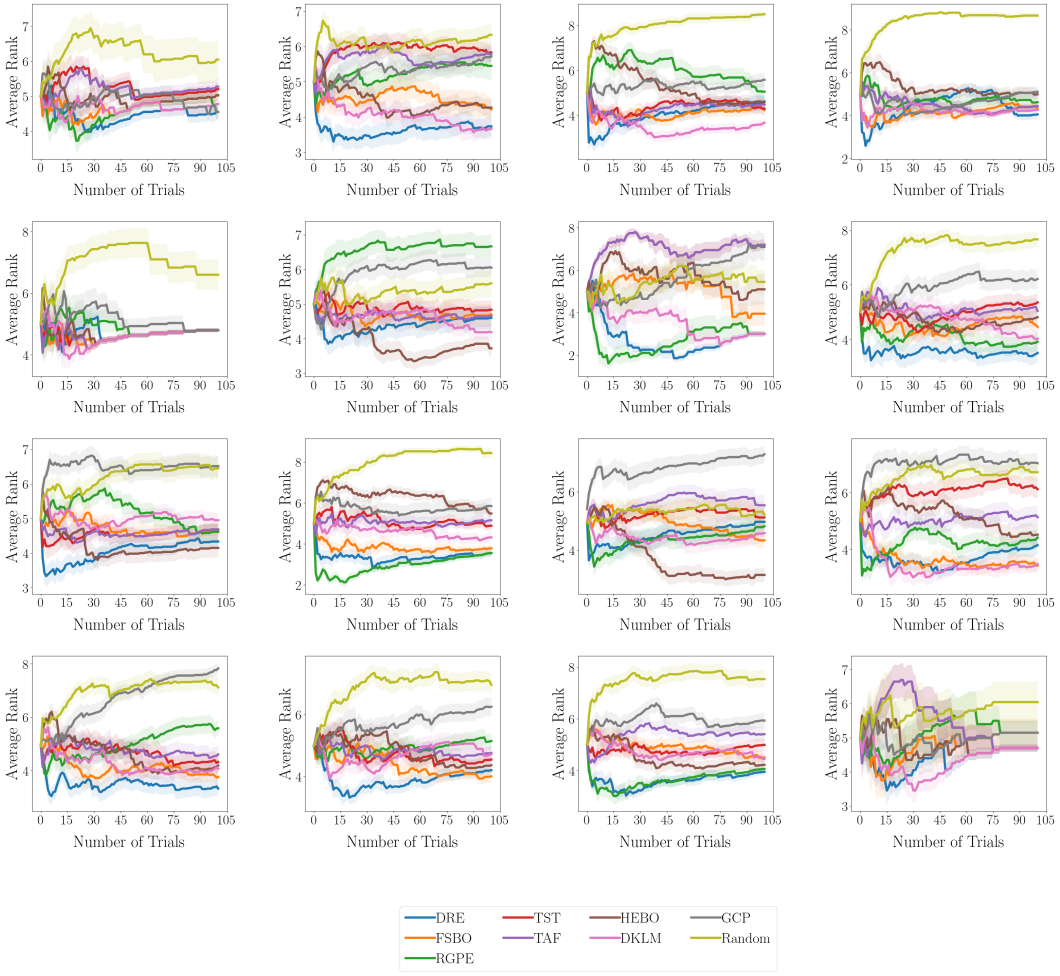



Figure 11: Average Rank per Search Space (Transfer Methods)

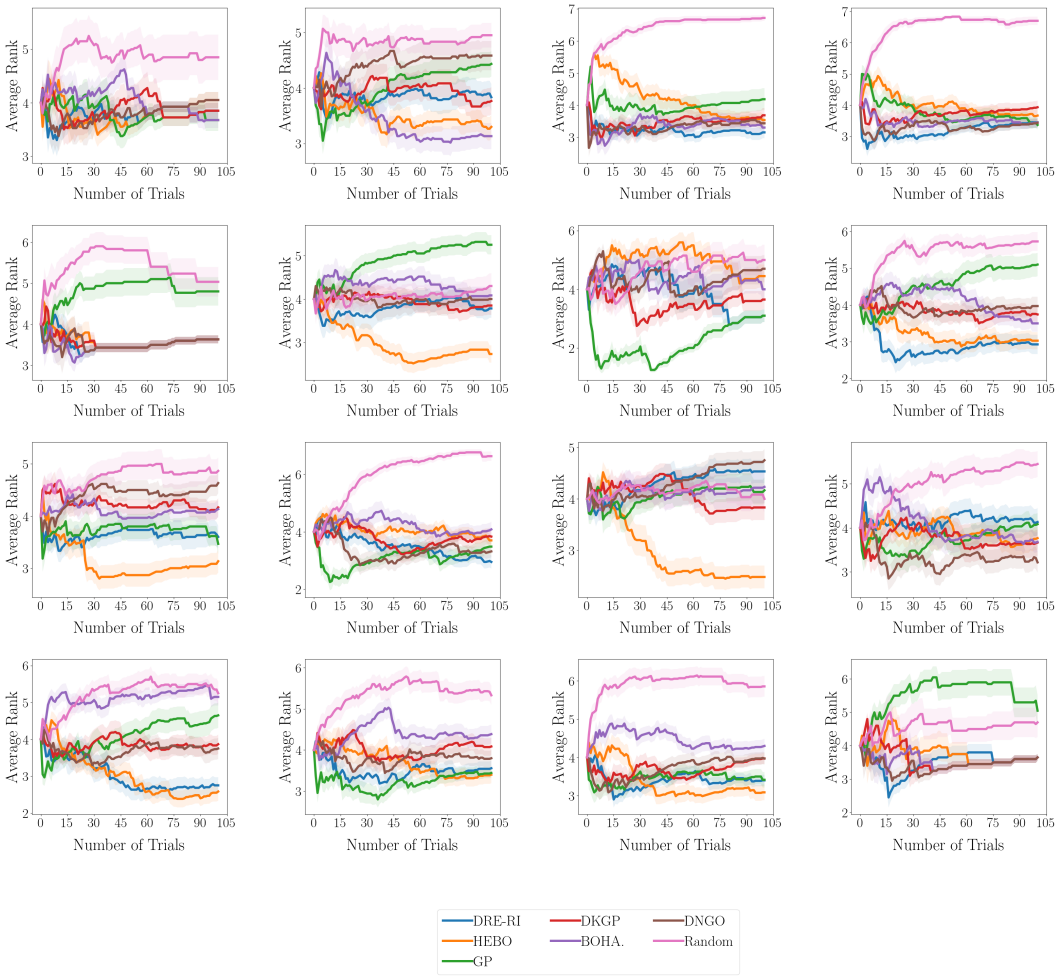



Figure 12: Average Rank per Search Space (Non-Transfer Methods)