

Differentiable Model Selection for Ensemble Learning

DISCLAIMER: Summarized by AI

Problem they are trying to solve / Purpose of method

Goal:

To improve ensemble learning by creating a framework that **learns to select an optimal subset of pre-trained models (base learners)** for each input, instead of applying fixed aggregation rules across the entire ensemble.

Previous challenges:

- Traditional ensemble methods (e.g., majority voting, averaging) treat all base learners equally or use fixed, global strategies, ignoring input-specific performance differences.
- Selecting the most suitable base learners **per input** is difficult, especially in a differentiable, trainable way.
- Gradient-based model selection has emerged but often struggles with non-differentiable choices and local optima.

Why this method is needed:

- To leverage **error diversity** among base learners by selecting models dynamically based on the input features.
- To enable **differentiable and trainable model selection** that integrates seamlessly with modern deep learning pipelines.
- To improve ensemble performance **especially when individual base learners are weak or specialized**.

How does it differ from other methods?

Unique aspects:

- Proposes **end-to-end Combinatorial Ensemble Learning (e2e-CEL)**, which combines machine learning with combinatorial optimization to select subsets of base learners dynamically.
- Uses a **differentiable knapsack-based selection layer**, enabling gradient-based learning over discrete model selections.
- Unlike Super Learner or consensus-based aggregation (e.g., averaging, voting), e2e-CEL **learns per-input sub-ensemble selections** rather than global or static rules.
- Employs a **perturbed optimizer technique** (e.g., Gumbel-based smoothing) to maintain differentiability in the selection process.

Key distinction:

- It turns ensemble selection into an **optimization problem embedded within a neural network**, trained via backpropagation — **the first method to do this for input-conditioned ensemble selection**.

How the method works

Simple overview:

1. Each input goes through a **selection network** that scores all base learners.
2. The top-k base learners (sub-ensemble) are selected using a **differentiable knapsack solver**.
3. Their predictions are aggregated using a **differentiable soft voting** mechanism.
4. The entire system is trained end-to-end to minimize classification loss.

More detailed steps:

1. **Selection Net** ($g\theta$): Given input z , it predicts scores \hat{c} for each base learner.
2. **Sub-ensemble Selection**: Solve a knapsack-like problem to select top-k learners based on \hat{c} . The selection function $K(\hat{c})$ is smoothed using noise perturbation (based on Gumbel-max trick) to ensure differentiability.
3. **Prediction Aggregation**: The selected learners' softmax predictions are masked and summed. The result is normalized with softmax to output final class prediction.
4. **Training**: Only the selection net's parameters θ are optimized using backpropagation, with gradients flowing through the knapsack selection and aggregation layers.

Algorithmic innovation:

- Uses **Monte Carlo sampling of perturbations** to estimate gradients of the selection process.
- Selection and aggregation operations are constructed to be **gradient-friendly**, enabling learning despite discrete components.

Performance:

- Tested on MNIST, CIFAR-10, UTKFace, and FER2013 datasets.
- Outperforms traditional ensemble strategies (Super Learner, averaging, voting) by a significant margin.
- Especially powerful when base learners are **specialized and have complementary errors**.