Cascade Bagging for Accuracy Prediction with Few Training Samples

Ruyi Zhang[1],Ziwei Yang[1], Zhi Yang[1], Xubo Yang[1], Lei Wang[3] and Zheyang Li[1,2]
[1]Hikvision Research Institute, [2]Zhejiang University
[3]University of Science and Technology of China
{zhangruyi5, yangziwei5, yangzhi13, yangxubo, lizheyang}@hikvision.com
wangl26@mail.ustc.edu.cn

Abstract

Accuracy predictors enable to estimate the final validation accuracy of the networks from their architectures. It can effectively assist in designing networks and improving efficiency of Neural Architecture Search(NAS). Because of the difficulties of obtaining data, the training of predictors is a few-shot learning problem which easily leads to overfitting. To alleviate this problem, we propose a cascade bagging algorithm(CBA) which consists of a two-level sub-predictor architecture to improve generalization. In addition, we propose a weak-supervised data augmentation strategy to enrich the dataset. Above all, our approach ranks the 3rd place in the Performance Prediction Track of CVPR2021 1st Lightweight NAS Challenge.

1. Introduction

Training a model to predict the final validation accuracy of a network from its architecture is great significant. Based on the predictor, the relationship between structure and accuracy can be analyzed deeply to assist in designing network. For NAS algorithms[10, 12], accuracy predictors can improve the efficiency of searching models due to its fast evaluation.

There exists excellent works on accuracy predictors [3, 8, 9], but they depend on amounts of computation overhead to obtain training data. Otherwise, their performance may drop dramatically due to overfitting problem[3]. Full training a network to obtain its accuracy is expensive. To reduce the cost, most methods provide proxy training schemes such as early stopping[15], dataset sampling[1] and lower resolution dataset, or using a proxy network with fewer filters and fewer cells[7, 14]. This is a tradeoff between label noise and cost. Few work has been to alleviate the overfitting in terms of improving predictor capabilities. One of the main difficulties is the huge gap between the input space and the training set, where the training sample distribution differs significantly from the test distribution, leaving the trained predictor with overfitting.

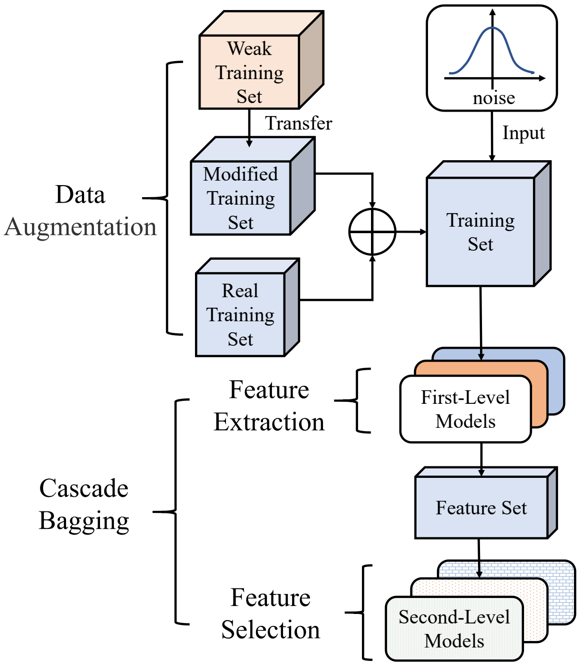

Figure 1. The overview framework of our method,consisting of data augmentation and cascade bagging algorithm.

In this paper, we design a ensemble algorithm CBA to jointly perform feature extraction and selection, thus reducing the overfitting of the predictor. The overfitting is further reduced by data transfer and the addition of noise. In the end, the validity of our method is verified using the dataset of CVPR2021 1st Lightweight NAS Challenge.

2. Method

As shown in Figure 1, we propose an ensemble method to tackle the overfitting problem via data augmentation, feature extraction and feature selection. More specifically, for

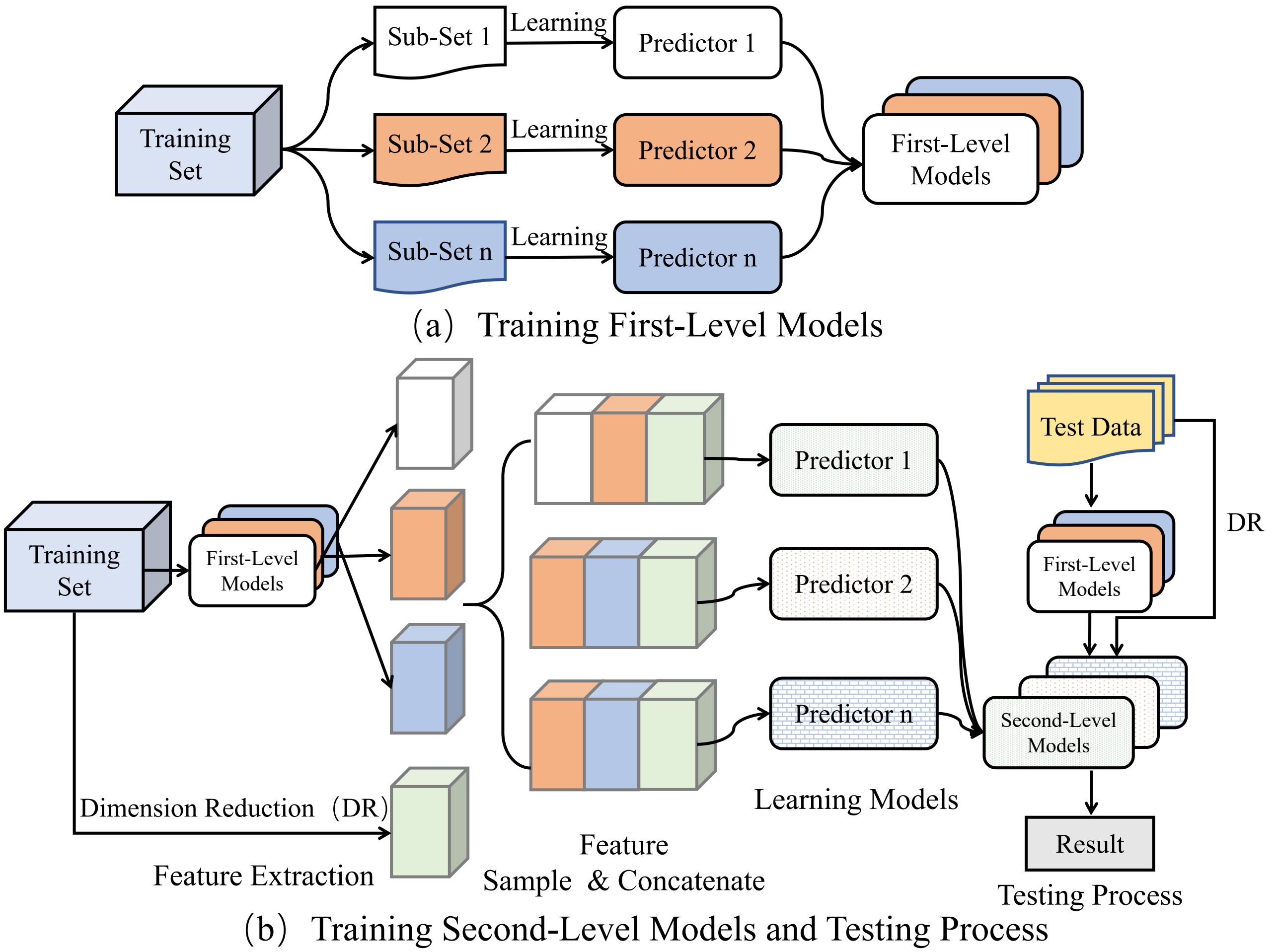

Figure 2. The training framework of the cascade bagging. (a) The training diagram of the first-level models. (b) The training diagram of the second-level models and testing process.

augmenting dataset, our method firstly transfers the accuracy obtained by proxy training scheme to the full training accuracy space. Secondly, to further expand the dataset, our method adds noise to the input space. The first-level models are trained by random sampling data with replacement to extract global features. In the end, CBA trains second-level models for feature selection, which means weighting and fusing the prediction results of first-level models from different perspective.

2.1. Data Augmentation

Our method augments the training set in terms of data transfer and addition of noise.

Giving two different datasets, one dataset is S_w, its weak label Y_w obtained by proxy training. The other dataset is S_r, which has a small amount data with full training true label Y_r. We want to modify Y_w of S_w to enlarge S_r. We firstly try to modify the weak label with the relationship of statistic information between Y_w and Y_r. More specifically, we divide dataset S_w into N groups based on accuracy, assuming the same bias between the weak and true labels for each group. Afterward, i-th weak label Y_{w_i} of a group will be modified with function as follows:

$$Y^{'}_{w_i} = Y_{w_i} + (E_r - E_w) + b \tag{1}$$

E_r and E_w are the mean values of Y_r and Y_w, respectively. b means modified bias, which is hyper-parameter. In practice, S_r is used for cross-validation to search hyper-parameters b and N.

Referring to the idea of Semi-supervised algorithms[5, 13], we further utilize the prediction results to correct weak labels with the following equation:

$$Y^{'}_{w_i} = \alpha * Y_{w_i} + (1 - \alpha) * Y_{P_i} \tag{2}$$

Y_{P_i} is the prediction result of the sample corresponding to Y_{w_i}. α denotes linear coefficient hyper-parameter. Since the predictor can be retrained after correcting the labels. Then another label correction can be made again. So it is a continuous iterative process until convergence.

Many studies[2, 13] have shown that adding noise to the input space can improve the generalization ability of the model. In this paper, we choose Gaussian noise to perturb the input features while keeping the corresponding label to generate training samples. The noise degree of the data is controlled by Signal Noise Ratio(SNR).

2.2. Cascade Bagging

This section introduces the details of cascade bagging on feature extraction and selection.

We extract network features effectively from different perspectives referring to the ideas of bagging and stacking[4, 6, 11]. As shown in Figure 2 (a), we firstly train a series of models by randomly data sampling with replacement. The predictions of these models are treated as global features of network.

We want to integrate the prediction results of the first-level models with adaptive weighting. But simply training one linear predictor to learn the weights of numerous global features is prone to overfitting. To alleviate this issue, our method trains a series of second-level models using different combinations of features. Specifically, as shown in the figure2(b), the input features of a second-level model is consisted of some global features by randomly sampling and the original feature downsampled by PCA. The main reason for inputting the original feature is to consider that different models may be good at predicting different instances. We need to input original features for adjusting the weights.

As shown on the right of Figure 2 (b), when predicting a test sample, the global features are first calculated using the first-level models and concatenated the reduced dimension features. The concatenated features input into corresponding second-level models to obtained the outputs. The outputs are averaged as the final prediction result.

3. Experiment

3.1. Dataset

The dataset of the competition contains 231 training samples. 200 samples of the training set have weak labels, which are obtained using a proxy training strategy. The other 31 samples with true label are full trained with more training steps and techniques. The aim of the competition is to train a full-trained accuracy predictor with above dataset.

3.2. Structure Encoder

The network of the competition dataset is sampled from the Mobilenet-like search space, where 16 blocks are searchable. The 16 blocks are connected to each other using sequence mode. The choices of each block range from [1,6] which means 6 (three choices of kernel size, two choices of expansion rate) different operations.

Based on the above space, we deign two forms of structure encoding, black-box and white-box. Black-box encodes a network with 16-dimensional features, and the value of the i-th dimension denotes the index of the operation selected of the i-th block. White-box encodes the network as a 2×16-dimensional tensor. Each block is represented by a two-dimensional vector and the two dimensional vectors of the i-th block represent the kernel size and expansion ratio of the i-th block.

3.3. Ablation Study

index	encode	group	bias	noise	RMSE
1	Black	-	-	w/o	0.252
2	Black	1	2.2	w/o	0.228
3	White	1	2.2	w/o	0.212
4	White	3	2.24,2.2,2.22	w/o	0.204
5	White	3	2.24,2.2,2.22	w	0.201

Table 1. Ablation study of data transferring with statistical information and input noise. ”encode” means encoding type of structure. ”group” and ”bias” means group number and value of $Y_{w_i} + (E_r - E_w) + b$ in eqn(1), respectively. ”noise” indicates whether noise is added to the input space.

As shown in Table 1,these experiments want to explored the impact of the data augmentation approach. All experiments base on Xgboost model. Experiment 1 only use 31 samples with true label. Compared to Experiment 1, Experiment 2 use more 200 modified samples by Eqn(1). The result of Experiment 2 has a huge improvement over Experiment 1, which shows the effectiveness of statistical information for data transferring. The performance of Experiment 5 with input noise is better than Experiment 4, indicating the validity of adding noise. In addition, we can see from the results of Experiment 2 and Experiment 3 that white-box is a better feature representation than black-box.

index	models for correction	iterations	RMSE
6	-	-	0.1857
7	SVM	1	0.1849
8	SVM	3	0.1846
9	SVM,KNN	3,1	0.1849
10	SVM,Xgboost,	3,1	0.1829

Table 2. Ablation study of data augmentation by correcting label with predictor. ”iterations” denotes number of iterative corrections. Such as the experimental 9, it means the iterations of SVM and KNN are 3 and 1 respectively.

The experiments in Table 2 all use bagging algorithm, which train 100 models with random data sampling. The experiments also utilize SVM replace with Xgboost. Because of the above changes, RMSE of Experiment 6 decreases from 0.201 in Experiment 5 to 0.1857. It shows that the bagging ensemble algorithm can significantly improve the generalization ability. We use the model trained in Experiment 6 to correct the weak labels using Eqn(2). The results of experiments 7 and 8 show that predictor correction labels are effective in reducing the variance and multiple correcting iterations also have some effect, respectively. Experiment 10 utilizes the SVM and the xgboost to correct the weak labels, and the results show that using more models may further improve performance. However, Experiment 9 shows that this improvement does not persist for all models. Label noise may be introduced due to poor performance of model.

index	algorithm	RMSE
11	Bagging	0.1829
12	Cascade bagging	0.1808

Table 3. Ablation study of cascade bagging

As shown in Table 3, these experiments want to prove validity of cascade bagging. We assigned Experiment10 from Table 2 as Experiment11 in Table 3 for intuitively analyzing. Experiment11 and Experiment12 only use different ensemble algorithms. More specifically, the cascade bagging of Experiment 12 used more 100 linear models to assigned the prediction weights of the models trained on Experiment 11. It can be seen that cascade bagging can utilize second-level models to further enhance the performance due to adaptive weighting and fusing the prediction result of first-level models.

4. Conclusion

We devised a method consisting of data augmentation and cascade bagging aiming to alleviate overfitting of accuracy prediction with few training samples. Combining the statistical information correction and model prediction correction for weak labels, we augmented training dataset with proxy training data and injecting noises to the data. We proposed a cascade bagging algorithm to further improve the generalization of the predictor. Finally, we proved the advantages of the above methods on the dataset provided by CVPR2021 1st Lightweight NAS Challenge.

References

[1] Fast bayesian optimization of machine learning hyperparameters on large datasets. 2016. 1

[2] Chris M Bishop. Training with noise is equivalent to tikhonov regularization. *Neural computation*, 7(1):108–116, 1995. 3

[3] Łukasz Dudziak, Thomas Chau, Mohamed S Abdelfattah, Royson Lee, Hyeji Kim, and Nicholas D Lane. Brp-nas: Prediction-based nas using gcns. *arXiv preprint arXiv:2007.08668*, 2020. 1

[4] B. Ghojogh and M. Crowley. The theory behind overfitting, cross validation, regularization, bagging, and boosting: Tutorial. 2019. 3

[5] Mfa Hady and F.Schwenker. Semi-supervised learning. 2006. 2

[6] H. C. Kim, S. Pang, H. M. Je, D. Kim, and S. Y. Bang. Support vector machine ensemble with bagging. *Pattern Recognition with Support Vector Machines, First International Workshop, SVM 2002, Niagara Falls, Canada, August 10, 2002, Proceedings*, 2002. 3

[7] H. Liu, K. Simonyan, and Y. Yang. Darts: Differentiable architecture search. 2018. 1

[8] Lizheng Ma, Jiaxu Cui, and Bo Yang. Deep neural architecture search with deep graph bayesian optimization. In *2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pages 500–507. IEEE, 2019. 1

[9] Han Shi, Renjie Pi, Hang Xu, Zhenguo Li, James T Kwok, and Tong Zhang. Bridging the gap between sample-based and one-shot neural architecture search with bonas. *arXiv preprint arXiv:1911.09336*, 2019. 1

[10] Yanan Sun, Xian Sun, Yuhan Fang, and Gary Yen. A new training protocol for performance predictors of evolutionary neural architecture search algorithms. *arXiv preprint arXiv:2008.13187*, 2020. 1

[11] B. Tang, Q. Chen, W. Xuan, and X. Wang. Reranking for stacking ensemble learning. 2010. 3

[12] Chen Wei, Chuang Niu, Yiping Tang, Yue Wang, Haihong Hu, and Jimin Liang. Npenas: Neural predictor guided evolution for neural architecture search. *arXiv preprint arXiv:2003.12857*, 2020. 1

[13] Q. Xie, M. T. Luong, E. Hovy, and Q. V. Le. Self-training with noisy student improves imagenet classification. 2019. 2, 3

[14] X. Zheng, R. Ji, Q. Wang, Q. Ye, Z. Li, Y. Tian, and Q. Tian. Rethinking performance estimation in neural architecture search. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[15] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le. Learning transferable architectures for scalable image recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1