TOWARDS PERFORMANCE-MAXIMIZING NETWORK PRUNING VIA GLOBAL CHANNEL ATTENTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Network pruning has attracted increasing attention recently for its capability of transferring large-scale neural networks (e.g., CNNs) into resource-constrained devices. Such a transfer is typically achieved by removing redundant network parameters while retaining its generalization performance in a static or dynamic manner. Concretely, static pruning usually maintains a larger and fit-to-all (samples) compressed network by removing the same channels for all samples, while dynamic pruning can adaptively remove (more) different channels for different samples and obtain state-of-the-art performance along with a higher compression ratio. However, since the system has to preserve the complete network information for sample-specific pruning, dynamic pruning methods are usually not memory-efficient. In this paper, our interest is to explore a static alternative, dubbed GlobalPru, to conventional static pruning methods that can take into account both compression ratio and model performance maximization. Specifically, a novel channel attention-based learn-to-rank algorithm is proposed to learn the global channel attention of the network for various samples, wherein, each sample-specific channel saliency is forced to reach an agreement on the global ranking. Hence, all samples can empirically share the same pruning priority of channels to achieve channel pruning with minimal performance loss. Extensive experiments demonstrate that the proposed GlobalPru can achieve better performance than state-of-the-art static and dynamic pruning methods by significant margins.

1 INTRODUCTION

Convolutional neural networks (CNNs) have achieved great success in many visual recognition tasks including image classification He et al. (2016), object detection Ren et al. (2015), image segmentation Dai et al. (2016), etc. The success of CNNs is inseparable from an excessive number of parameters that are well organized to perform sophisticated computations, which conflicts with the increasing demand for deploying these resource-consuming applications on resource-limited devices.

Network pruning has been proposed to effectively reduce the deep model's resource cost without a significant accuracy drop. Unstructured pruning methods Han et al. (2015b;a) usually reach a higher compression rate, while relying on dedicated hardwares/libraries to achieve the actual effect. In contrast, structured pruning methods Li et al. (2016); He et al. (2017b) preserve the original convolutional structure and are more hardware-friendly. Considering the greater reduction in terms of floating-point operations (FLOPs) and hardware commonality, this research focuses on channel pruning. Existing methods perform channel pruning either statically or dynamically. Static pruning methods remove the same channels for all images Molchanov et al. (2019); Tang et al. (2020) while dynamic pruning removes different channels for different images Rao et al. (2018); Tang et al. (2021b). However, both existing static methods and dynamic methods have some limitations. Particularly, given the fact that the channel redundancy is highly sample-dependent, static pruning methods may remove some channels that are not redundant for certain images. Consequently, static methods refrain from a larger pruning rate to avoid a significant accuracy drop. To tackle the issue of image-specific redundant channels, dynamic pruning methods remove image-specific channels. In this way, dynamic methods achieve the state-of-the-art pruning ratio without significantly sacrificing performance. Despite the significant advantage, dynamic pruning usually requires preserving the full original model during inference, which restricts its practical deployment on resource-limited devices.

In this paper, we propose a new paradigm of static pruning method named GlobalPru. Although born from static, GlobalPru tackles the issue of image-specific redundant channels by making all images share the same ranking of channels with respect to redundancy. In other words, GlobalPru forces all images to agree on the same ranking of channel saliency (referred to as global channel ranking) to reduce the image-specific channel redundancy to the greatest extent possible. By removing channels with the lowest global rankings, GlobalPru can avoid the problem of existing static pruning methods, which remove more important channels while retaining less important channels for specific images with a high probability. More specifically, we first propose a novel global channel attention mechanism. Channel attention is local in the sense that the ranking of channels with respect to their importance is image-specific. Different from existing ones, our proposed global channel attention mechanism can identify the global channel ranking across all different samples in the training set, particularly through a learn-to-rank regularization. In detail, to make the static GlobalPru approach the maximum image-specific compression ratio of dynamic pruning as well as stabilize the training process, we first use a majority-voting-based strategy to specify the global ranking to make the static GlobalPru approach the maximum image-specific compression ratio of dynamic pruning. Then, given a certain ranking, all the image-specific channel rankings are forced to agree on the ranking via learn-to-rank regularization. When all the image-specific channel rankings are the same as the given ranking, the ranking becomes the global ranking. As a result of exposing global ranking for all images during training stages, GlobalPru can also avoid the disadvantage of existing dynamic pruning which needs to store the entire model for deciding image-specific channel ranking during inference and perform more efficient pruning on globally ordered channels.

Our contributions are summarized as follows:

- We propose GloablPru, a static network pruning method. GlobalPru tackles the issue of image-specific channel redundancy faced by existing static methods by learning a global ranking of channels w.r.t redundancy. GlobalPru produces a pruned network such that GlobalPru is a more memory-efficient solution than existing dynamic methods.

- To the best of our knowledge, we for the first time propose a global channel attention mechanism where all the images share the same ranking of channels w.r.t. importance. Instead of repeatedly computing image-specific channel rankings under existing local attention mechanisms, our proposed global attention enriches the representation capacity of models and therefore greatly improves the pruning efficiency.

- Extensive experimental results show that GlobalPru can achieve state-of-the-art performance with almost all popular convolution neural network architectures.

2 Related Work

2.1 Static Pruning & Dynamic Pruning

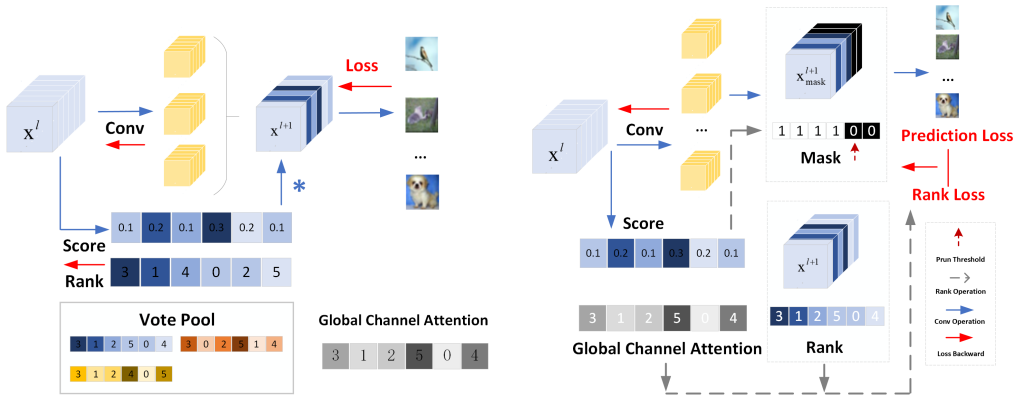
As the most traditional and classic model pruning method, static pruning shares a compact model among all different samples Wen et al. (2016); Liu et al. (2017a); Liebenwein et al. (2019); Molchanov et al. (2019); Tang et al. (2020). To be specific, static methods select pruning results through trade-offs on different samples, which leads to final compact models having limited representation capacity and thus suffering an obvious accuracy drop with large pruning rates. Recently, some works turn their attention to the pursuit of the ultimate pruning rate and focus on excavating sample-wise model redundancy, named dynamic pruning. Dynamic pruning generates different compact models for different samples Dong et al. (2017); Gao et al. (2018); Hua et al. (2019); Rao et al. (2018); Tang et al. (2021b). Actually, dynamic methods learn a path-decision module to find the optimal model path for each input during inference. For example, state-of-the-art work Liu et al. (2019) investigates a feature decay regularization to identify informative features for different samples, and therefore achieves an intermediate feature map to the model sparsity. Tang et al. (2021a) further improves the dynamic pruning efficiency by embedding the manifold information of all samples into the space of pruned networks. Despite dynamic methods achieve higher compression rate, most of them are not memory-efficient cause most of them requires to deploy the full model even in inference stage.

2.2 CHANNEL ATTENTION

Channel attention is similar to the scale coefficients added on feature channels to enhance the important features and also weaken the unimportant ones. It is usually implemented through extra channel attention modules, which can exploit the inter-channel relationship for input feature maps. An typical channel attention mechanism is SENet Hu et al. (2017), it uses the "Squeeze-and-Excitation" (SE) block which can be stacked together to adaptively recalibrate channel-wise feature responses on each convolutional layer by explicitly modelling inter-dependencies between channels. Li et al. (2019) proposes a dynamic selection mechanism in CNNs that allows each neuron to adaptively adjust its receptive field size based on multiple scales of input information, which is named "Selective Kernel" (SK) unit. By dynamically calculating channel attention on the different kernels, SK realizes parameter sharing and thus improves the model efficiency. In summary, nearly all the existing channel attention are local in the sense that the channel saliency is image-specific and can not identify a global channel attention over the entire dataset.

3 GLOBAL ATTENTION-BASED CHANNEL PRUNING

In this section, we will give a detailed problem formulation and theoretical principle of the proposed Global Channel Attention Pruning (GlobalPru) method. Specifically, GlobalPru includes two core pipelines, i.e., Global Channel Attention Election and Learn-to rank Regularization Pruning. As shown in Figure 1, for each layer, GlobalPru first explicitly elects the global channel attention that benefits most samples from the observations of the locally image-specific channel dependencies. After that, the proposed learn-to-rank algorithm forces the channels to be ordered towards the global channel attention simultaneously during the model training process. In this way, Globalpru could statically remove the most global redundancy approved by the majority of training data, while at the same time guaranteeing the network performance maximization.
As illustrated in Section 3.1, the GlobalPru is first formulated as a static strategy regularized by the channel-wise local saliency and the global rank loss, and can be optimized in an end-to-end manner. Then, the above-mentioned two core stages of GlobalPru are theoretically detailed in Section 3.2 and Section 3.3, respectively.



(a) Global channel attention. (b) Learn-to-rank regularization.

Figure 1: The overview of the training process of Channel Rank Network.

3.1 PROBLEM FORMULATION

Given a dataset with N labeled samples as $X = \{x_i\}_{i=1}^{N}$, $Y = \{y_i\}_{i=1}^{N}$. For a convolution neural network (CNN) with L convolution layers, $W^l \in R^{c^l \times c^{l-1} \times k \times k}$ denotes weight parameters of the l_{th} convolution layer, and Θ is the model parameters set. $F^l(x_i) \in R^{b \times c^l \times h \times w}$ is the output feature map of the l_{th} layer for data x_i, where b, c^l, h, w are the batch size, channel number, height and width of the output feature map, respectively. $\otimes$ stands convolution operation. And we have

$F^l(x_i) = Relu(BN(W^l \otimes F^{l-1}))$, where $Relu$ and BN represent the activation and normalization operation in CNN respectively. Finally, we use $f(X, \Theta)$ to represent the output of CNN over the input samples X.

Static pruning methods shares a compact network among all images, which can be formulated as:

$$\min_{\Theta} \sum_{i=1}^{N} \mathscr{L}\left(f(x_i, \Theta), y_i\right) + \lambda \cdot Norm(\Theta), \tag{1}$$

where $\mathscr{L}$ denotes the loss function and $Norm(\cdot)$ is the regularization term which is usually a human-designed criterion for sparse model. Finally, λ is used as a knob to strike the different trade-off between model accuracy and sparsity ratio.

Conversely, dynamic channel pruning methods discover effective sub-networks for each instance by incorporating different data with different model requirements into the consideration of pruning. To this end, most dynamic methods introduce small auxiliary modules to predict the saliencies of channels with given input data, and prune unimportant filters at run-time. Define a scoring module SE, for a specific input x_i, the channel saliencies on layer l can be computed as: $\pi^l(x_i) = SE^l(x_i) \in R^{c^l}$, where $\pi^l_j(x_i)$ indicates the relative importance score of the j_{th} channel. And a smaller π^l_j indicates that channel j is less important. In our work, the scoring module SE is the squeeze and excitation module Hu et al. (2017) and has been proved to be an efficient local channel attention indicator. Given a prune threshold ϵ^l, and $mask^l_j = 0$ to indicate that the j_{th} channel of layer l is redundant. Then, the $mask^l_j$ can be calculated as follows:

$$mask^l_j(x_i) = \begin{cases} 1 & \pi^l_j(x_i) \geq \epsilon^l \\ 0 & \pi^l_j(x_i) < \epsilon^l \end{cases} \tag{2}$$

Based on that, the actual channel saliencies and the output features of layer l and data x_i could be recomputed as:

$$\begin{aligned} \hat{\pi}^l(x_i) &= \pi^l(x_i) \cdot mask^l(x_i) \\ \hat{F}^l(x_i) &= Relu\left(BN\left(\left(mask^l(x_i) \cdot \hat{F}^{l-1}(x_i)\right) \otimes W^l\right)\right) \end{aligned} \tag{3}$$

A general dynamic pruning optimization could be formulated as follows:

$$\min_{\Theta} \sum_{i=1}^{N} \mathscr{L}\left(f(x_i, \Theta), y_i\right) + \lambda \sum_{l=1}^{L} ||\pi^l(x_i)|| \tag{4}$$

Most recent dynamic methods have achieved great compression rate by removing instance-specific model redundancy in run-time, however, the run-time path-finding mechanisms ask dynamic methods to deploy the heavy referenced model instead of the lightweight pruned model on edge devices in practice. Worse still, the deployment is often accompanied by additional path-addressing units. These unavoidable dependencies make dynamic pruning methods not memory and computation-efficient, thus without great applicability.

To take advantage the opposing static and dynamic pruning advantages and successfully achieve more efficient neural networks

To take advantage and tackle the disadvantages of dynamic pruning, we first investigate the local channel attention of different samples via a preheated scoring module SE. Then, the global channel attention can be identified through the **Majority Vote Mechanism** based on the investigated local attentions. Finally, by the novel **Learn-to-Rank** regularization, GlobalPru forces all image-specific channel rankings to converge to a certain global channel ranking. In this way, GlobalPru can efficiently prune a neural network in an orderly and interpretable manner.

Define T^l as the real channel importance order for layer l most voted by all input samples. And R^l is the current channel importance order computed by SE. And $\Phi(R_l, T_l)$ represents the channel importance ordering loss of layer l. While minimizing empirical risk and minimizing the channel score on each layer to improve model sparsity, GlobalPru also minimizes the summation of differences between the current channel ordering R^l and the target channel ordering T^l for each layer. After clarifying our goals, the optimization target could be formulated as:

$$\min_{\Theta} \sum_{i=1}^{N} \sum_{l=1}^{L} \mathscr{L}(f(x_i, \Theta), y_i) + \alpha \left\|\pi^l(x_i)\right\|_1 + \beta \sum_{l=1}^{L} \Phi(R^l, T^l), \tag{5}$$

where Φ is the learn-to-rank regularization defined in Eq. 6,7. For a particular channel j, T_j^l, R_j^l represent its real channel ranking index and current channel index, respectively. We cleverly use the previously defined channel score $|\pi_j^l|$ to measure the similarity between the current position of channel j on layer l and its target ranking position, and we have $P(R_j^l = T_j^l) = \frac{exp(\pi_j^l)}{\sum_{j=1^{c^l}} exp(\pi_j^l)}$. Given prior knowledge of the true ordering T_l, Φ is equal to:

$$\Phi(T^l, R^l) = -\log P(R^l = T^l | T^l) = -\log \frac{P(T^l | R^l = T^l) P(R^l = T^l)}{\sum_{R^l} P(R^l = T^l)} \tag{6}$$

Since T^l is the pre-defined true channel rank, and is independent with the the current channel rank, thus we have $P(T^l) = 1$ and $P(T^l | R^l = T^l) = 1$, then EQ. 6 can be further simplified as:

$$\Phi(T^l, R^l) = -\log \frac{P(R^l = T^l)}{\sum_{R^l} P(R^l = T^l)} = -\log \prod_{j=1}^{c^l} \frac{exp(\pi_j^l)}{\sum_{j=1}^{c^l} exp(\pi_j^l)} \tag{7}$$

3.2 Global Channel Attention

Lots of previous works have proven that the importance of layers Dong et al. (2020); Wang et al. (2018) or filters Tang et al. (2021b) are highly sample-dependent. Thus, the channel saliency of each layer for different samples can be totally different. In fact, natural images are actually a mixture of some intrinsic patterns such as color, edges, textures, etc., in different ratios. An important focus of computation vision is to search the more representative features of an image in a given visual task, and the enhancement of the most salient properties of images captured by convolution filters is enable to distinctly improve the task performance. To this end, we can introduce our conclusion of the problem background as:

- *The relative importance of high-level feature channels is image-specific as the responses of different areas in a certain image.*
- *The attention mechanism not only amplifies representations differences, but also can indicate model redundancy for an individual image.*
- *Channel attention is highly sample-related, which means the weakened channels are disordered and unpredictable across images.*

To implement the above-mentioned orderly channel pruning and therefore eliminate the dependency of dynamic pruning on repeat path-finding calculation, we propose to explore the "Global Channel Attention" for all different samples. As shown in Figure 1(a), our global channel attention is achieved by a majority vote mechanism. Specifically, a deep neural network with channel attention modules is first pre-warmed. Then, all image-specific local channel attentions are collected as the Electoral College to vote for a global channel attention, which would be the true rank in learn-to-rank regularization. In this way, GlobalPru can train pruned models with both dynamic sample awareness and unified channel importance.

3.3 Learn-to-rank Regularization

The learn-to-rank is explicitly modeled as a supervised learning task. However, the huge number of channels in deep neural networks brings unique challenges for how to enable converged training with the slightest additional overhead. The global channel ranking has given the observation to facilitate any image with the highest probability of removing the most redundant channels under the arbitrarily defined compression ratio. Thus, our target could be further formulated as a Maximum Likelihood Estimates (MLE) problem, which aims to find out a specific model parameter set Θ to maximize the joint probability distribution P of the observation "Global Channel Ranking". Specifically, we first regard the model parameters distribution as the likelihood function to infer a specific parameters set in high-dimension space which assigns the maximum probability to the known observation Tr. The joint probability distribution of channel ranking P is essentially the product of sampling densities p of each channel location in each training step. Thus, in addition to empirical risk minimization, GlobalPru trains a model distribution towards the probabilistic model which is most likely to generate the observed rank. As shown in Figure 1(b), and if the pruning rate p is pre-specified, the computation

cost could be further reduced by maximizing the probability that only the first $(1-p)c^l$ channels in T^l are ranked correctly. The proposed learn-to-rank regularization could be reformatted to Eq. 8, and:

$$\Phi(T^l, R^l) = -\sum_{j=1}^{(1-p)c^l} \left\{ \log exp(\pi_j^l) - \log \sum_{j=1}^{(1-p)c^l} exp(\pi_j^l) \right\} \tag{8}$$

4 EXPERIMENT

In this section, GlobalPru is empirically tested on the most representative neural network architectures: plain architecture, residual structure, and lightweight depthwise convolution networks to verify the effectiveness of the proposed "Global Attention Mechanism" and "Learn-To-Rank" method.

Datasets: The effectiveness of the proposed GlobalPru is verified on most popular image datasets of model pruning, including MiniImageNet, CIFAR-10, CIFAR100 and SVHN. Diverse datasets can help effectively demonstrate the broad applicability of our GlobalPru.

Models: We explore the effectiveness of GlobalPru on three mainstream neural network architectures includes: plain architecture - VGG-16, residual structure - ResNet with different depth, and lightweight depthwise convolution MobileNet-V2. These three representative architectures cover almost all the most popular convolutional networks currently.

Tasks: Most model pruning works only demonstrate the accuracy of the pruned models on the image classification task, and ignores the robustness of the pruned networks. To further demonstrate the applicability of our method, we also evaluate the performance of the pruned network against adversarial perturbations on the adversarial samples detection task.

Implementation Details: Standard data argumentation RandomSizedCrop and RandomHorizontalFlip are used in all datasets we used. The coefficient α to regulate the channel saliency score is set as 0.0001 in our work. And another coefficient β to balance the weight of the channel rank loss is also set as 0.0001 empirically. The stochastic gradient descent (SGD) is used for all training processes. Two pruning modes are tested in GlobalPru, named "Fixed" and "Mixed", respectively. Under the fixed mode, all layers use the same pruning rate with the Lottery Ticket of model pruning which corresponds to the highest test accuracy Frankle & Carbin (2019). For ResNet-18 on CIFAR10 in this work, the range of pruning rates which can achieve higher test accuracy than the original model is $[38.9\%, 100.0\%]$, and the optimal pruning rate corresponds to the highest test accuracy is 61.1% Under the mixed mode, the layer-wise pruning rates are set according to the sensitivity of each layer to model perturbation. Dong et al. (2020) has theoretically proved that the right sensitivity metric for model perturbation is the average Hessian trace. Align the value of the average Hessian trace of all layers with the optimal pruning rate, and the fluctuation of the pruning rate of each layer relative to the optimal pruning rate is set according to the standard deviation of the Hessian trace of this layer relative to the model average . The larger the Hessian trace value, the lower the pruning rate. All of the experiments are conducted on NVIDIA GeForce GPUs.

4.1 COMPARISON ON PLAIN ARCHITECTURE

The proposed method is first compared with the state-of-the-art network pruning algorithm on the most intuitive plain architecture, which is the initially successful way to construct a convolutional neural network. The typical plain network VGG-16 we used consists of a few building blocks (convolutional layers, activation layers, etc.) stacked in a vertical manner. A "Squeeze-and-Excitation" module is added following the convolutional layer to measure its relative channel saliency. Both static methods including Luo et al. (2017); Li et al. (2017); He et al. (2017a); Liu et al. (2017b), and the dynamic methods including Lin et al. (2017); Gao et al. (2019) are compared. All of the algorithms are tested on the Cifar-10 dataset and the comparison results are shown in Table 1. The pruning performance is reported over three main metrics: Memory Access Cost (MAC), FLoating point Operations (FLOPs), and Classification Accuracy Drop. When using the fixed pruning rate for all the computing layers, our method achieves 55.2% FLOPs decrease with a negligible accuracy drop of -0.21%. And when using the empirically mixed pruning rate for each layer, our method can prune more 1.5% FLOPs than the fixed mode, while with less accuracy degradation. The results show that our approach achieves the highest compression ratio on VGG-16 with competitive model

Table 1: Comparison of pruning performance on VGG-16 architecture and Cifar-10 dataset. Our method is compared with different channel pruning algorithms including both traditional static methods and sample-wise dynamic methods. "Dy?" means if the current method is a sample-wise dynamic method. "Fixed" and "Mixed" represent the above-mentioned fixed pruning mode and mixed pruning mode, respectively. "*" denotes the new static pruning paradigm, which approximates the maximum compression ratio of dynamic methods.

Method	Cite	Dy?	↓MAC	FLOPs(M)	↓FLOPs	-Acc
Baseline			0	626.4	0	0
LIWS	Wang et al. (2020)	Y	53.2%	184	70.6%	-0.8%
ThiNet	Luo et al. (2017)	N	50%	313	50%	-0.14%
L1-norm	Li et al. (2017)	N	34%	413.2	34%	-0.5%
CP	He et al. (2017a)	N	50%	313	50%	-0.32%
NS	Liu et al. (2017b)	N	51%	306.8	51%	-0.19%
RNP	Lin et al. (2017)	Y	50%	313	50%	-0.85%
FBS	Gao et al. (2019)	Y	50%	310.4	50.4%	-0.47%
GlobalPru(Fixed)	Ours	*	55%	280.87	55.2%	-0.21%
GlobalPru(Mixed)	Ours	*	56.7%	271.26	56.7%	-0.15%

Table 2: Comparison of pruning performance on Residual architecture: ResNet-20 architecture and Cifar-10 dataset.

Method	Cite	Dy?	FLOPs(M)	↓FLOPs	-Acc
Baseline			43.17	0	0
SFP	He et al. (2018)	N	24.95	42.2%	-1.39%
FPGM	He et al. (2019)	N	19.86	54%	-1.78%
DSA	Ning et al. (2020)	N	21.46	50.3%	-0.84%
Hinge	Li et al. (2020a)	N	23.53	45.5%	-0.38%
DHP	Li et al. (2020b)	N	20.80	51.8%	-0.68%
Maninp	Tang et al. (2021a)	Y	19.77	54.2%	-0.17%
FBS	Gao et al. (2019)	Y	19.97	53.1%	-1.25%
GlobalPru(Fixed)	Ours	*	15.77	60.84%	-0.14%
GlobalPru(Mixed)	Ours	*	16.07	60.1%	-0.26%

performance. Compared to the SOTA static pruning method, undoubtedly, our method wins an absolute advantage in pruning rate and only has a slight accuracy drop, i.e., -0.07% and -0.01 respectively for fixed and mixed modes, than state-of-the-art ThiNet. And when compared to the pioneering dynamic pruning method, our method gains and obviously less accuracy loss than the newest Liu et al. (2019), and outperforms Lin et al. (2017); Gao et al. (2019) in both compression rate and accuracy.

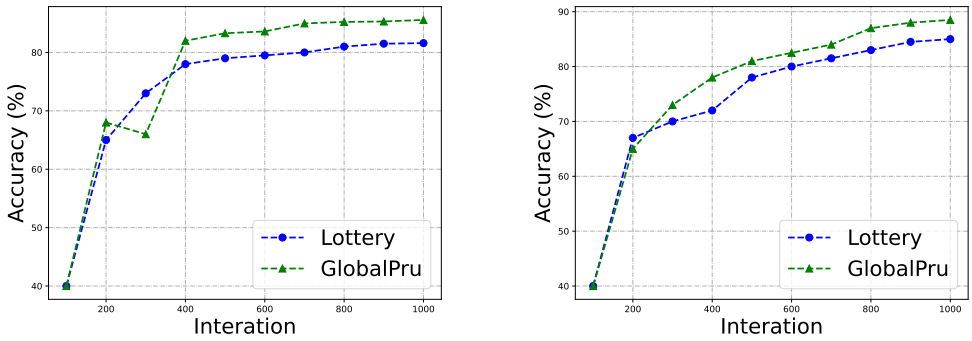
4.2 Comparison on Residual Architecture

We further explore the effectiveness of GlobalPru on the residual neural networks, where GlobalPru explores global attention inside residual blocks by adding the vanilla input feature representations to the output feature maps scaled by channel attention. To be specific, we introduce compression on each convolutional layer inside the residual block without changing the transfer of residual information.

Table 3: Comparison of pruning performance on DepthWise architecture: MobileNet-V2 architecture and Cifar-10 dataset.

Method	Cite	Dy?	FLOPs(M)	↓FLOPs	-Acc
Baseline			89.9	0	0
WM	Zhuang et al. (2018)	N	66.53	26%	-3.68%
DCP	Zhuang et al. (2018)	N	66.53	26%	-3.16%
NPPM	Gao et al. (2021)	N	47.65	47%	-3.1%
GlobalPru(Fixed)	Ours	*	47.47	47.2%	-0.69%
GlobalPru(Fixed)	Ours	*	35.83	60.14%	-0.55%

The compared methods include both SOTA static method He et al. (2018; 2019); Ning et al. (2020); Li et al. (2020a;b) and state-of-the-art dynamic pruning method Tang et al. (2021a) and Gao et al. (2019). The results are tested on ResNet-20 and Cifar-10. As shown in Table 2, the FLOPs, FLOPs decrease relative to the baseline and post pruning accuracy drop are reported. Our method under both fixed pruning rate and mixed pruning rate setting achieve the highest compression ratio at the same time with the most negligible accuracy drop. For example, even under the highest pruning rate 68% for all layers, the accuracy of our method still reaches 98.01% with only -0.34 degradation. Besides, we also compare the performance of GlobalPru with the classic three-stage "train-prune-fine-tune" static pruning at different pruning rates. According Figure **??**, we choose 40.1% and 61.1% as model pruning rates. They correspond to the point where the generalization performance is the highest and the point where the generalization accuracy starts to suffer, respectively. All of the comparisons show that GlobalPru can work efficiently on residual neural networks.



(a) Pruning rate is 40.1%.

(b) Pruning rate is 61.1%.

Figure 2: Comparison of pruning performance under different pruning rates on ResNet-20 and Cifar-10. "Lottery" represents an typical three-stage static pruning paradigm.

4.3 Comparison on DepthWise Architecture

Another popular model architecture using "Depthwise Separable Convolution" is also included in our experiment to cover another class of models in the spotlight. Also, the depthwise architecture is originally designed for resource-limited mobile devices and is already very light-weight. To validate whether our method can further compress these already compressed models, we compare it with the most widely used efficient MobileNet-V2. As shown in Table 3, while outperforming all methods in computing reduction, our model achieves the closest model accuracy to the baseline. The advantage of our method on lightweight DepthWise architecture is obvious which proves that our model enables further model compression in an almost lossless mode over current state-of-the-art methods.

Table 4: Comparison of pruning performance on MiniImageNet (M-ImageNet) and SVHN datasets with popular model pruning SOTAs.

Dataset	Method	Accuracy (%)		↓Size
		Top 1	Top 5	
M-ImageNet	PolarPrun	85.857	96.068	0.59
	SlimPrun	70.06	90.44	0.60
	GlobalPru	**89.227**	**96.97**	**0.60**
SVHN	PolarPrun	97.099	98.199	0.59
	SlimPrun	96.848	99.228	0.50
	GlobalPru	**97.543**	**99.3**	**0.60**

Table 5: Comparison of AUROC (%) of GlobalPru with typical adversarial detection methods on Fast Gradient Sign Method (FGSM)-based CIFAR100 and SVHN datasets .

AUROC (%) FGSM	Non Prune		Prune	
	MAHA	FBS	DPIC	**GlobalPru**
SVHN	99.63	99.95	99.96	**100.00**
CIFAR100	99.77	100.00	100.00	**100.00**

4.4 Generalization Verification

In addition to model accuracy, a good pruning method also needs to have good generalization and robustness, which is usually ignored in previous work. To show that GlobalPru has these favorable properties, we test the performance of GlobalPru on a more diverse vision dataset and extra visual tasks with added adversarial samples.

As shown in Table.4, we compare the performance of GlobalPru and two static pruning SOTAs on typical image datasets SVHN and MiniImageNet. SVHN contains large-scaled labeled data (over 600,000 digit images) for a more difficult, unsolved real-world problem. MiniImageNet, drawn from ImageNet, has complex samples and multi-classes, more suitable for prototyping and experimental research. Obviously, as a static alternative, GlobalPru achieves a higher model compression/task loss ratio than static pruning SOTA by removing the global redundancy of the model for all samples.

As shown in Table.5, we verify the robustness of our method by observing the performance of GlobalPru when faced with adversarial perturbations. It can be seen that GlobalPru exhibits extraordinary perturbation resistance, even surpassing some typical adversarial sample detection methods. We speculate that this is because the model redundancy removed by GlobalPru is selected by a majority vote of all samples, thereby weakening the effect of small image disturbances and making GlobalPru more robust than the previous pruning criterion only guided by task loss.

5 Conclution

To sum up, we propose a novel pruning method, i.e., the Channel Attention-based Learn-to-Rank Network, based on the shortcomings of current dynamic pruning methods, such as the need to save the complete model locally and repeat the forward computation. Our approach first explores the channel saliency rank of each sample, and then selects the most suitable channel rank supported by all current inputs through a majority voting mechanism. We define this channel rank as global channel attention. Next, we obtain a Channel Rank Network by proposing an efficient channel sorting algorithm to incorporate the knowledge of global channel attention into the training of the model, so as to quickly give an appropriate pruning response when new samples or sparse requirements arrive. While obtaining the advantages of dynamic pruning, our method avoids the defects of the original dynamic pruning method, and achieves better pruning performance than most state-of-the-art methods.

REFERENCES

Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3150–3158, 2016.

Xuanyi Dong, Junshi Huang, Yi Yang, and Shuicheng Yan. More is less: A more complicated network with less inference complexity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5840–5848, 2017.

Zhen Dong, Zhewei Yao, Daiyaan Arfeen, Amir Gholami, Michael W. Mahoney, and Kurt Keutzer. HAWQ-V2: hessian aware trace-weighted quantization of neural networks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Shangqian Gao, Feihu Huang, Weidong Cai, and Heng Huang. Network pruning via performance maximization. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pp. 9270–9280. Computer Vision Foundation / IEEE, 2021.

Xitong Gao, Yiren Zhao, Łukasz Dudziak, Robert Mullins, and Cheng-zhong Xu. Dynamic channel pruning: Feature boosting and suppression. *arXiv preprint arXiv:1810.05331*, 2018.

Xitong Gao, Yiren Zhao, Lukasz Dudziak, Robert D. Mullins, and Cheng-Zhong Xu. Dynamic channel pruning: Feature boosting and suppression. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015a.

Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28, 2015b.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Yang He, Guoliang Kang, Xuanyi Dong, Yanwei Fu, and Yi Yang. Soft filter pruning for accelerating deep convolutional neural networks. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pp. 2234–2240. ijcai.org, 2018.

Yang He, Ping Liu, Ziwei Wang, Zhilan Hu, and Yi Yang. Filter pruning via geometric median for deep convolutional neural networks acceleration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4340–4349, 2019.

Yihui He, Xiangyu Zhang, and Jian Sun. Channel pruning for accelerating very deep neural networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 1389–1397, 2017a.

Yihui He, Xiangyu Zhang, and Jian Sun. Channel pruning for accelerating very deep neural networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 1389–1397, 2017b.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. *CoRR*, abs/1709.01507, 2017.

Weizhe Hua, Yuan Zhou, Christopher M De Sa, Zhiru Zhang, and G Edward Suh. Channel gating neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*, 2016.

Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang. Selective kernel networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 510–519, 2019.

Yawei Li, Shuhang Gu, Christoph Mayer, Luc Van Gool, and Radu Timofte. Group sparsity: The hinge between filter pruning and decomposition for network compression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8018–8027, 2020a.

Yawei Li, Shuhang Gu, Kai Zhang, Luc Van Gool, and Radu Timofte. Dhp: Differentiable meta pruning via hypernetworks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16*, pp. 608–624. Springer, 2020b.

Lucas Liebenwein, Cenk Baykal, Harry Lang, Dan Feldman, and Daniela Rus. Provable filter pruning for efficient neural networks. In *International Conference on Learning Representations*, 2019.

Ji Lin, Yongming Rao, Jiwen Lu, and Jie Zhou. Runtime neural pruning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 2178–2188, 2017.

Chuanjian Liu, Yunhe Wang, Kai Han, Chunjing Xu, and Chang Xu. Learning instance-wise sparsity for accelerating deep models. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pp. 3001–3007. ijcai.org, 2019.

Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang. Learning efficient convolutional networks through network slimming. In *Proceedings of the IEEE international conference on computer vision*, pp. 2736–2744, 2017a.

Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang. Learning efficient convolutional networks through network slimming. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pp. 2755–2763. IEEE Computer Society, 2017b.

Jian-Hao Luo, Jianxin Wu, and Weiyao Lin. Thinet: A filter level pruning method for deep neural network compression. In *Proceedings of the IEEE international conference on computer vision*, pp. 5058–5066, 2017.

Pavlo Molchanov, Arun Mallya, Stephen Tyree, Iuri Frosio, and Jan Kautz. Importance estimation for neural network pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11264–11272, 2019.

Xuefei Ning, Tianchen Zhao, Wenshuo Li, Peng Lei, Yu Wang, and Huazhong Yang. Dsa: More efficient budgeted pruning via differentiable sparsity allocation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pp. 592–607. Springer, 2020.

Yongming Rao, Jiwen Lu, Ji Lin, and Jie Zhou. Runtime network routing for efficient image classification. *IEEE transactions on pattern analysis and machine intelligence*, 41(10):2291–2304, 2018.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

Yehui Tang, Yunhe Wang, Yixing Xu, Dacheng Tao, Chunjing Xu, Chao Xu, and Chang Xu. Scop: Scientific control for reliable neural network pruning. *Advances in Neural Information Processing Systems*, 33:10936–10947, 2020.

Yehui Tang, Yunhe Wang, Yixing Xu, Yiping Deng, Chao Xu, Dacheng Tao, and Chang Xu. Manifold regularized dynamic network pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5018–5028, 2021a.

Yehui Tang, Yunhe Wang, Yixing Xu, Yiping Deng, Chao Xu, Dacheng Tao, and Chang Xu. Manifold regularized dynamic network pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5018–5028, 2021b.

Xin Wang, Fisher Yu, Zi-Yi Dou, Trevor Darrell, and Joseph E. Gonzalez. Skipnet: Learning dynamic routing in convolutional networks. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIII*, volume 11217, pp. 420–436, 2018.

Yulong Wang, Xiaolu Zhang, Xiaolin Hu, Bo Zhang, and Hang Su. Dynamic network pruning with interpretable layerwise channel selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 6299–6306, 2020.

Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. *Advances in neural information processing systems*, 29, 2016.

Zhuangwei Zhuang, Mingkui Tan, Bohan Zhuang, Jing Liu, Yong Guo, Qingyao Wu, Junzhou Huang, and Jin-Hui Zhu. Discrimination-aware channel pruning for deep neural networks. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 883–894, 2018.