SCIENCE CHINA
Information Sciences

• LETTER •

October 2018, Vol. 61 109101:1–109101:3
https://doi.org/10.1007/s11432-018-9497-0

GlanceNets – efficient convolutional neural networks with adaptive hard example mining

Hanqing SUN & Yanwei PANG*

School of Electrical and Information Engineering, Tianjin University, Tianjin 300072*, China*

Received 27 February 2018/Revised 10 May 2018/Accepted 30 May 2018/Published online 3 September 2018

Citation Sun H Q, Pang Y W. GlanceNets – efficient convolutional neural networks with adaptive hard example mining. Sci China Inf Sci, 2018, 61(10): 109101, https://doi.org/10.1007/s11432-018-9497-0

Dear editor,

Deep convolutional neural networks (CNNs) have become the dominant approach in various computer vision tasks such as image classification [1–4]. Despite the success of CNNs, it is impeded to deploy such deep CNN models in real-time tasks due to high computational complexity. To address the problem, we propose GlanceNets with several bypasses (Figure 1). In modern CNNs, it is believed that shallow layers provide lower-level features, whereas deep layers correspond to higher-level features. However, it is not always necessary to classify a sample with the highest-level feature. In many cases, easy samples can be correctly classified with low-level features, just as one can recognize common items at a glance. Such observation is the key motivation of proposed GlanceNets in this study.

A bypass in proposed GlanceNets simply consists of batch normalization (BN), rectified linear unit (ReLU), global average pooling, fully connected (FC), and softmax (Figure 1). In training stage, losses of predictions are provided in all of the bypasses and are optimized jointly. Inspired by existing hard example mining methods [5] and Focal-Loss [6], an online hard example mining strategy with hard example weight function (green boxes in Figure 1) is designed to improve both accuracy and speed. A threshold learning method (blue boxes in Figure 1) is proposed to avoid threshold search procedures as used in existing work. In inference stage, predictions are given bypass-by-bypass. And if any bypass classifies a sample with much confidence, the prediction becomes final and subsequent computation costs are saved for the sample. In this way, the average runtime of GlanceNets becomes less than that of the backbone CNN.

Hard example weight function. The online hard example mining strategy is expected to enable GlanceNets not only to early classify easy examples in preceding bypasses, but also to focus on the hard samples (that is, incorrectly classified ones) in the subsequent part. We carry out the goal by designing a novel hard example weight function, which is denoted as w.

In the i-th bypass, a hard example weight function of a sample is

$$w_i = \sigma\left(\frac{\sum_{j=1}^{i-1} I_{\mathrm{ic}}\left[\boldsymbol{y}^{\mathrm{T}}\left(1-\boldsymbol{p}_j\right)\right]}{\sum_{j=1}^{i-1} I_{\mathrm{c}}\left[\boldsymbol{y}^{\mathrm{T}}\boldsymbol{p}_j\right]}-1\right)+\frac{1}{2}, \quad (1)$$

where $\sigma(\cdot)$ denotes the sigmoid function, $\boldsymbol{y}$ is the one-hot column vector of the ground truth label, $\boldsymbol{p}_j$ is the column prediction vector of the j-th bypass, and $I[\cdot]$ is the indicator function that indicates whether the prediction $\boldsymbol{p}$ is correct (denoted as I_{c}) or not (denoted as I_{ic}).

The proposed hard example weight function is used as a coefficient of cross entropy loss in GlanceNets, thus the final loss term becomes

$$J_{\mathrm{HEM},i} = w_i \cdot H, \quad (2)$$

* Corresponding author (email: pyw@tju.edu.cn)

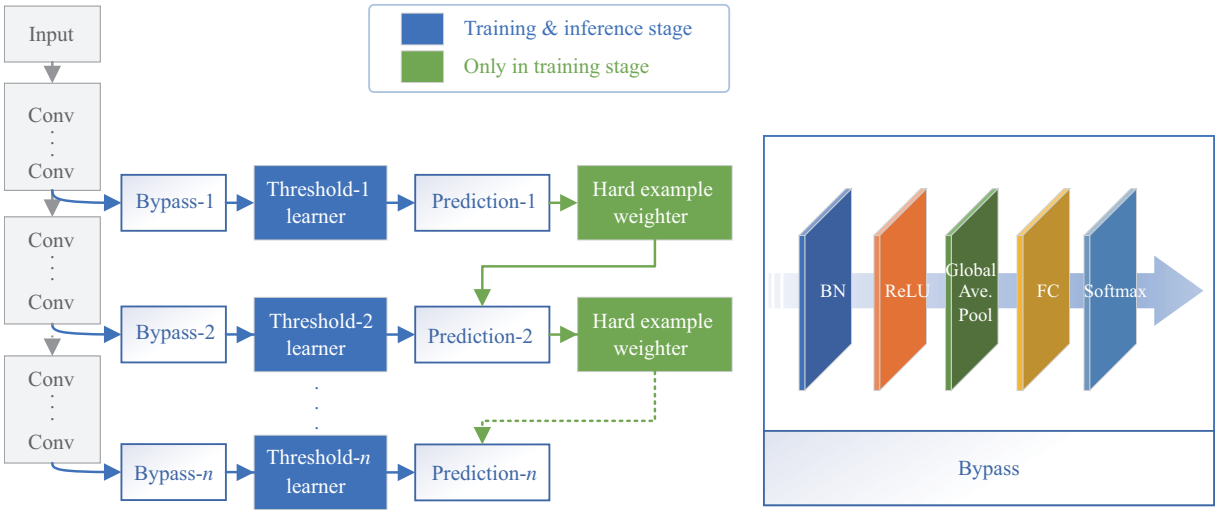

Figure 1 (Color online) A GlanceNet consists of a backbone CNN architecture and several bypasses, in which the proposed online hard example mining method (green boxes) and threshold learning method (blue boxes) are applied. Detail compositions of a bypass are shown on the right.

where H is the widely used cross entropy loss of Softmax in CNN models.

The comparison experiments of training GlanceNets with or without the proposed hard example weight function are conducted on CIFAR10 datasets to verify its effect (with $\alpha = 0.5$). The results demonstrate that the proposed hard example weight function can improve both speed (from 1.54 times faster to 1.69 times faster than baseline DenseNet [1] in inference time) and accuracy (from 89.12% to 89.84%). GlanceNets are able to classify more samples in preceding bypasses with the help of proposed hard example weight function. That fact can be observed obviously in the comparison experiments where the first bypass predicts around 15% more samples after exploiting the hard example weight function, thus more average runtime is saved.

Threshold learning approach. To measure the confidence of a prediction without ground truth labels in inference stage, following [7], the entropy of a prediction is used as the confidence metric. The definition of a prediction entropy is

$$H\left(\boldsymbol{q}\right) = -\sum_{i \in \mathcal{C}} q_i \ln q_i, \tag{3}$$

where $\boldsymbol{q}$ is the prediction vector (e.g., softmax outputs), q_i denotes the i-th element in the vector $\boldsymbol{q}$, and $\mathcal{C}$ is the set of all classes in a dataset. The lower the entropy becomes, the more confident the prediction is. If the entropy of a prediction is lower than the learned threshold H_{T}, the prediction becomes final and subsequent computation will not take place.

Thresholds have to be selected manually and empirically for every single bypass with a stochastic exhaustive search procedure in traditional methods [7]. To avoid the complication, a threshold learning method is proposed in this study to set the threshold H_{T} adaptively with one single hyper-parameter α, which represents a leverage between speed and accuracy. In this way, thresholds for all bypasses can be adaptively learned while training the CNN with a least computational cost.

For the i-th bypass in a GlanceNet, the threshold learning loss term $J_{\mathrm{T},i}$ is

$$\begin{aligned} J_{\mathrm{T},i} = &\left(1-\alpha\right) \left\| \max\left\{ I_{\mathrm{c}}\left[H\left(\boldsymbol{p}_i\right) \right] \right\} - H_{\mathrm{T},i} \right\|_2 \\ &+ \alpha \left\| \min\left\{ I_{\mathrm{ic}}\left[H\left(\boldsymbol{p}_i\right) \right] \right\} - H_{\mathrm{T},i} \right\|_2, \end{aligned} \tag{4}$$

where $H_{\mathrm{T},i}$ is the learned adaptive threshold of the i-th bypass, $\max\{I_{\mathrm{c}}[H(\boldsymbol{p}_i)]\}$ is the maximum information entropy within the correct predictions, $\min\{I_{\mathrm{ic}}[H(\boldsymbol{p}_i)]\}$ is the minimum information entropy within the incorrect predictions, and $\|\cdot\|_2$ denotes L2-norm.

Eq. (4) suggests that α controls the balance of distances from the threshold $H_{\mathrm{T},i}$ to the minimum information entropy of incorrect predictions and to the maximum entropy of the correct ones. The larger α is set, the smaller will the learned $H_{\mathrm{T},i}$ be, thus fewer samples will exit at the i-th bypass. GlanceNets hold a threshold learner for each bypass (Figure 1), providing an adaptive threshold for each bypass.

The hyper-parameter α, which is proposed to enable adaptive threshold learning in GlanceNets, is an important parameter that affects the efficiency of GlanceNets. As described above, the proportion of early classified samples lessens and more prediction results are given by subsequent bypasses with the increase of α.

The accuracy, speed, and threshold of GlanceNets with various α are further experimented on CIFAR10 dataset. The results show that: the larger α is, the lower the learned thresholds become, thus the overall accuracy becomes higher. On the contrary, smaller α can lead to faster speed. In practice, α should be picked according to the characteristic of the task and the backbone CNN architecture. Considering the trade-off between accuracy and speed, we suggest the value of α being set to 0.5–1.0.

Results and discussion. The quantitative comparison experiments of GlanceNets ($\alpha = 0.9$) and original DenseNet [1] are performed on CIFAR10 dataset. The proposed GlanceNets achieve an accuracy of 92.36% with computation complexity (measured in the number of floating-point multiplication-adds, i.e., FLOPs) reduced from 276 MFLOPs to 186 MFLOPs (30.18% reduced), which means 1.43 times faster than the baseline DenseNet.

The main contributions of this study are as follows:

(1) An efficient CNN framework, GlanceNets. The proposed GlanceNets are able to early classify a number of samples in the added bypasses, thus avoiding redundant computation of those samples. When more samples are early classified, more runtime can be reduced.

(2) Online hard example mining strategy with hard example weight function. The hard example weight function is designed to make each part of GlanceNets focus more on hard examples, leading to an overall classification accuracy raise.

(3) Threshold learning method as a term in loss function. The threshold learning method based on information entropy, the prediction confidence metric, can learn all thresholds adaptively, which avoids the complication of exhaustive search procedures.

(4) Compatibility with mini-batch SGD methods. The entire GlanceNets framework is compatible with widely used mini-batch SGD methods in modern CNNs, which means one can easy reuse numerous CNN software frameworks when implementing GlanceNets. Moreover, GlanceNets can be trained in an end-to-end fashion, which means it only requires a minor revision of an existing CNN backbone architecture.

In the perspective of back-propagation algorithm, bypass architectures as used in GlanceNets have benefits for backward propagation of gradients in training stage [7, 8]. As a bonus advantage, GlanceNets alleviate the gradient vanishing problem with the help of losses on the bypasses, thus suite training very deep CNNs. The experimental results have demonstrated that proposed GlanceNets have the capability of raising the efficiency of a CNN. Furthermore, GlanceNets can be used in conjunction with existing CNN pruning and compression methods. With the combination of diverse acceleration methods, the runtime can be further saved on the basis of GlanceNets. Besides, GlanceNets, as a framework, could be further utilized in semantic segmentation and object detection tasks. Considering the characteristics of various computer vision tasks, how to improve GlanceNets in those tasks remains a piece of future work.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 61632081) and National Program of Key Basic Research Project (973 Program) (Grant No. 2014CB340400).

References

1 Huang G, Liu Z, van der Maaten L, et al. Densely connected convolutional networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 2261–2269
2 Pang Y W, Sun M L, Jiang X H, et al. Convolution in convolution for network in network. IEEE Trans Neural Netw Learn Syst, 2018, 29: 1587–1597
3 Cao J L, Pang Y W, Li X L, et al. Randomly translational activation inspired by the input distributions of ReLU. Neurocomputing, 2018, 275: 859–868
4 Pang Y W, Cao J L, Li X L. Cascade learning by optimally partitioning. IEEE Trans Cybern, 2017, 47: 4148–4161
5 Shrivastava A, Gupta A, Girshick R. Training region-based object detectors with online hard example mining. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, 2016. 761–769
6 Lin T Y, Goyal P, Girshick R B, et al. Focal loss for dense object detection. In: Proceedings of IEEE Conference on Computer Vision, Venice, 2017
7 Teerapittayanon S, McDanel B, Kung H T. BranchyNet: fast inference via early exiting from deep neural networks. In: Proceedings of International Conference on Pattern Recognition, Cancun, 2016. 2464–2469
8 Lee C Y, Xie S N, Gallagher P, et al. Deeply-supervised nets. In: Proceedings of International Conference on Artificial Intelligence and Statistics, San Diego, 2015. 562–570