DL-GLEANING: AN APPROACH FOR IMPROVING INFERENCE SPEED AND ACCURACY

HyunYong Lee & Byung-Tak Lee
Energy System Research Section
Honam Research Center
Electronics Telecommunications Research Institute (ETRI)
Korea
{hyunyonglee,bytelee}@etri.re.kr

ABSTRACT

Improving inference speed and accuracy is one of the main objectives of current deep learning-related research. In this paper, we introduce our approach using middle output layer for this purpose. From the feasibility study using Inception-v4, we found that our approach has potential to reduce the average inference time while increasing the inference accuracy.

1 INTRODUCTION

One general way for improving inference accuracy of a deep neural network (DNN) is to use a large number of middle or hidden layers. One challenging issue of such a DNN is that inference takes long time and requires much memory and computing resources. Powerful and dedicated hardwares such as GPUs may be used for accelerating inference. However, in resource-limited environments (e.g., embedded devices), such powerful hardwares may not be available.

One common approach for improving inference speed is to transform a given DNN. For example, a DNN can be transformed to a shallower and wider one (Romero et al., 2015; Hinton et al., 2014) and then parallelism can be exploited. Or, the trained weights and biases can be banalized to make computation simple (Lin et al., 2016; Courbariaux et al., 2015; Courbariaux et al., 2016; Kim et al., 2015), particularly in hardware implementation.

Instead of transforming a DNN, we use additional middle output layer to accelerate inference. The rationale behind this approach is to complete inference as early as possible at middle output layer using proper criteria. A feasibility study using Inception-v4 shows that the use of two middle output layers can reduce the average inference time by around 34% while increasing Top1 and Top5 inference accuracy by 4.79% and 1.42% in the ideal case.

In the rest of this paper, we first describe the basic concept of our research and then examine the feasibility of using an additional middle output layer in achieving our goal. Then, we discuss potential approaches for achieving the goal. Finally, we conclude this paper.

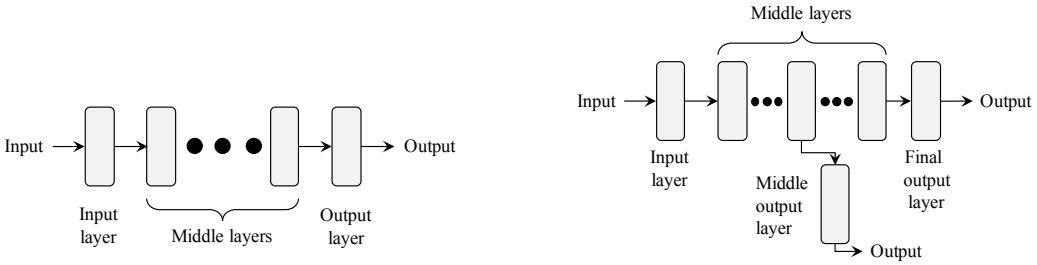
2 DEEP LEARNING-GLEANING

2.1 BASIC CONCEPT

Figure 1 shows a typical DNN and a proposed approach. In our approach, one or more middle output layer can be added to existing middle layers. Inference may be able to be completed at middle output layers if some criteria are satisfied. We will discuss the criteria later.

2.2 FEASIBILITY STUDY

To examine the feasibility of using the middle output layer in improving inference speed and accuracy, we conduct experiments using Inception-v4 model. We use two middle output layers, mid_A and mid_B. Mid_A and mid_B are located at four and eight inception blocks ahead of the final output



(a) Typical Deep Neural Network

(b) Proposed Deep Neural Network

Figure 1: Conceptual illustration of the proposed approach.

Table 1: Experiment result

	Final output layer	Proposed (w/ mid_A)	Proposed (w/ mid_A and mid_B)
Top1	80.23% (28.41ms)	83.20% (20.4ms)	85.02% (19.59ms)
Top5	95.2% (28.41ms)	96.17% (18.7ms)	96.62% (18.39ms)

layer. After training the model using ILSVRC 2012 train data set, we measure inference correctness of the middle output layers and the final output layer for each image of ILSVRC 2012 validation data set (including 50,000 images). We also measure the average inference time by the middle output layer and the final output layer. Our experiment machine has Intel i7-6700K 4GHz processor, 65GB main memory, and GeForce GTX Titan X GPU card.

Table 1 shows the experimental results including the cases w/ mid_A and w/ both middle output layers. The numbers inside the parenthesis indicate the average inference time. For the purpose of feasibility study, for *Proposed*, we assume that an inference for each image is completed at one middle output layer if that middle output layer infers the given image correctly, which means that *Proposed* is the ideal case. One interesting observation is that middle output layers correctly infer some images that are wrongly inferred by the final output layer. The use of mid_A increases Top1 and Top5 inference accuracy by 2.97% and 0.97%, respectively. The use of both middle output layers increases Top1 ad Top5 inference accuracy by 4.79% and 1.42%, respectively. This means that mid_B still have a chance to correctly infer some images that are missed by other output layers. Because of this observation, we call our approach *DL-gleaning*. Another observation is that *Proposed* reduces the average inference time by completing most inference at middle output layers. For example, the use of mid_A reduces the average inference time for Top5 inference by 34% by completing around 93% of inferences at mid_A.

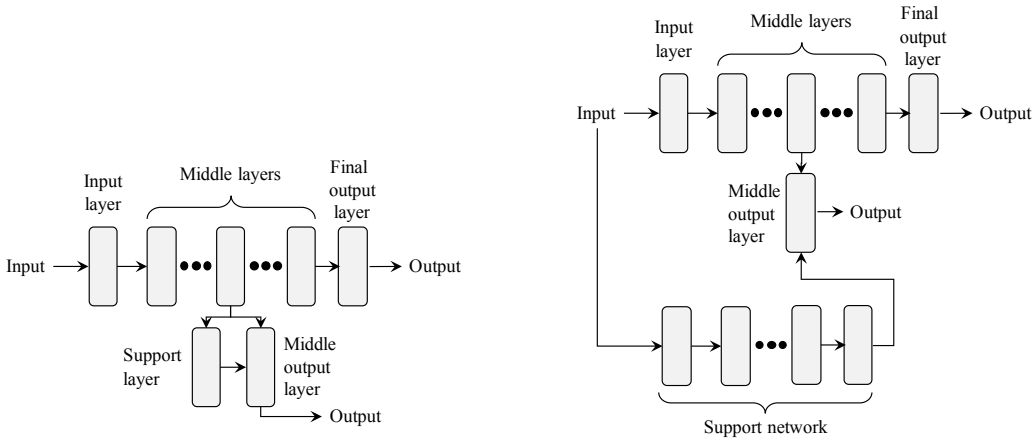
In summary, DL-gleaning has the potential to reduce inference time while improving inference accuracy (without increasing the number of DNN layers).

2.3 Approaches

Given the feasibility of DL-gleaning, we need to find good criteria for early inference at the middle output layer to fully exploit the potential benefits of DL-gleaning.

2.3.1 What we tried

We first used class probabilities produced by the middle output layer as criterion. For example, if the highest class probability exceeds a pre-defined threshold, an inference is completed. However, we found that the class probability-based approach is not feasible because Top1/Top5 inference accuracy (depending on class probabilities) of the middle output layer is worse than that of the final output layer. As a result, this approach reduces the inference time while degrading inference accuracy.



(a) Using a support layer

(b) Using a support network

Figure 2: Using a supporting information.

2.3.2 WHAT WE ARE TRYING

Instead of using the output of the middle output layer, we use additional information (from a support layer or a support network) coupled with the middle output layer (Figure 2). The output of the support layer or network indicates whether a given image can be inferred correctly by the middle output layer or not. Therefore, the number of outputs of the support layer or network is two (i.e., yes or no). The support layer or network is trained using the inference results of the trained middle output layer. The rationale behind this approach is to selectively infer images that are likely to be inferred correctly by the middle output layer at the middle output layer. Currently, we are trying to find d proper architecture of the support layer or network and a good way to train them.

2.4 APPLICATION SCENARIOS

DL-gleaning can be realized within a machine so as to achieve the two potential benefits of DL-gleaning.

Another and more interesting application is that DL-gleaning is realized across machines. For example, a part of DNN (e.g., up to the middle output layer) is implemented in a resource-limited edge devices such as sensor and gateway while remaining part of DNN is implemented in a powerful server. Potential benefits of this application scenario may include i) edge devices do not need to be as powerful as a dedicated central server and ii) most inference may be able to be handled by edge devices. This is our final deployment scenario using DL-gleaning.

3 CONCLUSION

Improving inference speed is important particularly when device resources are limited. In this paper, we briefly discuss and examine the feasibility of middle output layer-based approach. Currently, we are trying to improve DL-gleaning by using a more than one middle output layer. We also plan to realize DL-gleaning using mobile GPU and low-end/mid-range FPGA.

ACKNOWLEDGMENTS

This work was supported by Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government. [17ZK1210, Realtime Management of Renewable Energy System and Regional Energy Industry Advancement Project]

REFERENCES

Romero, Adriana, Ballas, Nicolas, Kahou, Samira Ebrahimi, Chassang, Antoine, Gatta, Carlo, and Bengio, Yoshua. Fitnets: Hints for thin deep nets. In *International Conference on Learning Representations*, 2015.

Hinton, Geoffrey, Vinyals, Oriol, and Dean, Jeff. Distilling the knowledge in a neural network. In *Deep Learning and Representation Learning Workshop, NIPS*, 2014.

Lin, Zhouhan, Courbariaux, Matthieu, Memisevic, Roland, and Bengio Yoshua. Neural networks with few multiplications. In *International Conference on Learning Representations*, 2016.

Kim, Minje and Smaragdis, Paris. Bitwise neural networks. In *International Conference on Machine Learning*, 2015.

Courbariaux, Matthieu, Bengio, Joshua, and David, Jean-Pierre. BinaryConnect: Training deep neural networks with binary weights during propagations. *arXiv:1511.00363*, 2015.

Courbariaux, Matthieu, Hubara, Itay, Soudry, Daniel, El-Yaniv, Ran, and Bengio, Yoshua. Binarized neural networks: Training neural networks with weights and activations constrained to +1 or -1. *arXiv:1602.02830*, 2016.