From Maxout to Channel-Out: Encoding Information on Sparse Pathways

Qi Wang and Joseph JaJa

Department of Electrical and Computer Engineering and,
University of Maryland Institute of Advanced Computer Studies
University of Maryland, College Park, MD, USA
{qwang37,joseph}@umiacs.umd.edu

Abstract. Motivated by an important insight from neural science that "functionality is determined by pathway", we propose a new deep network framework that encodes information on sparse pathways, called "channel-out network". We argue that the recent success of maxout networks can also be explained as its ability of encoding information on sparse pathways, while channel-out network does not only select pathways at training time but also at inference time. From a mathematical perspective, channel-out networks can represent a wider class of piecewise continuous functions, thereby endowing the network with more expressive power than that of maxout networks. We test our channel-out networks on several well-known image classification benchmarks, achieving new state-of-the-art performances on CIFAR-100 and STL-10.

Keywords: pathway selection, sparse pathway encoding, channel-out

1 Introduction

Many recent works on deep learning have focused on ways to regularize network behavior to avoid over-fitting. Dropout [1] has been widely accepted as an effective way for deep network regularization. Dropout was initially proposed to avoid co-adaptation of feature detectors, but it turns out it can also be regarded as an efficient ensemble model. The maxout network [2] is a newly proposed micro architecture of deep networks, which works well with the dropout technique. It sets the state-of-the-art performance on many popular image classification datasets. In retrospect, both methods follow the same approach: they restrict updates triggered by a training sample to affect only a sparse sub-graph of the network.

In this paper we provide a new insight into a possible reason for the success of maxout, namely that it partially takes advantage of what we call "sparse pathway encoding", a much more robust way of encoding categorical information than encoding by magnitudes. In sparse pathway encoding, the pathway selection itself carries significant amount of the categorical information. With a carefully designed scheme, the network can extract pattern-specific pathways during training time and recognize the correct pathway at inference time. Guided by

this principle, we propose a new type of network architectures called "channel-out networks". We run experiments with channel-out networks using several image classification benchmarks, showing competitive performances compared with state-of-the-art results. The channel-out network sets new state-of-the-art performance on two image classification datasets that are on the "harder" end of the spectrum - CIFAR-100 and STL-10 - demonstrating its potential to encode large amounts of information with higher level of complexity.

2 Review of Maxout Networks

The maxout network [2] is a recently proposed architecture that is significantly different from traditional networks in its activation style: the activation does not take a normal single-input-single-output form, but instead the maximum of several linear outputs. In [2], its advantage over normal differentiable activation functions (such as tanh) was attributed to its better approximation to exact model averaging, and the advantage over rectified linear (Relu) activation function was attributed to easier optimization at training time. Here we propose another insight of the power of the maxout network. The idea is motivated by a well-established principle in neural science: It is not the shape of the signal but the pathway along which the signal flows that determines the functionality of information processing [3]. The maxout node activates only one of the candidate input pathways, and the gradient is back-propagated only through that selected pathway, which means the information imposed by the training sample is encoded in a controlled sparse way. We call this behavior as "sparse pathway encoding". Note that although Relu networks also use sparse sub-networks, the pathway selection is less structured than that in maxout networks, which might be the reason that Relu networks are more vulnerable to over-fitting.

Although maxout networks encodes information sparsely, it does not infer sparsely, i.e. when doing inference every weight parameter effectively participates in computation. Since the power of deep network lies in the hierarchical feature structure, it is worthwhile to think about whether the sparse pathway encoding can also be arranged in a hierarchical way. In this paper we made our first attempt along this line by proposing a kind of network architecture called "channel-out networks", which is able to make active pathway selection at inference time.

3 The channel-Out networks

A channel-out network is characterized by channel-out groups (Figure 1). At the end of a typical linear layer (e.g. fully connected or convolutional layer), output nodes are arranged into groups, and for each group a special channel selection function is performed to decide which channel opens for further information flow. Only the activation of the selected channels are passed through, other channels are blocked off. When gradient is back-propagated through the channel-out layer, it only passes through the open channels selected during forward propagation.

Formally, we define a scalar/vector-valued channel selection function $\mathbf{f}(a_1, a_2, ..., a_k)$ which takes as input a vector of length k and outputs an index set of length l ($l < k$). Elements of the index set are selected from the domain $\{1, 2, ..., k\}$:

$$f_s(a_1, a_2, ..., a_k) \in \{1, 2, ..., k\}$$
$$s \; from \; 1 \; to \; l$$
$$\forall s \neq t, f_s(\cdot) \not\equiv f_t(\cdot)$$

Then with an input vector (typically the previous layer output) $\mathbf{a} = (a_1, a_2, ..., a_k) \in \mathcal{R}^k$, a channel-out group implements the following activation functions:

$$h_i = \mathbf{I}_{\{i \in \mathbf{f}(a_1, a_2, ..., a_k)\}} a_i \tag{1}$$

where $\mathbf{I}(\cdot)$ is the indicator function, i indexes the candidates in the channel-out group, a_i is the i^{th} candidate input, and h_i is the output (Figure 1). There are many possible choices of the channel selection function $f(\cdot)$. To ensure good performance, we require that the channel selection function possesses the following properties:

- The function must be piece-wise constant, and the piece-wise constant regions should not be too small. Intuitively, the function has to be "regular enough" to ensure robustness against the noise in the data.
- The pre-image size of each possible index output must be of almost the same size. In other words, each channel in the channel-out group should be equally likely to be selected as we process the training examples (so that the information capacity of the network is uniformly utilized).
- The computation cost for evaluating the function must be as low as possible.

Figure 2 compares a channel-out network with a maxout network. We can see that a channel-out network can actively select the pathway at a higher layer while maxout can't.

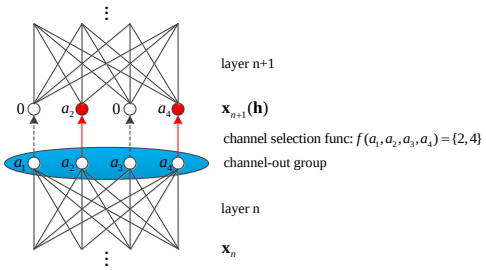

Fig. 1: Operation performed by a channel-out group

A side effect of enabling the network to do active pathway selection is the potential saving on computation power. As a concrete example, suppose all channel-out groups in a network are of size k and the channel selection function output

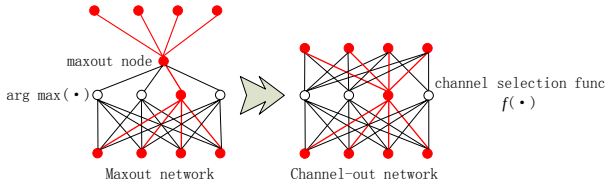

Fig. 2: Difference between maxout and channel-out: A maxout node is attached to a set of FIXED output links, resulting in same output pathway for different input pathways; A channel-out group is connected to a set of different output links, resulting in distinct output pathways.

is scalar. Consider a layer that's not the input or the output layer, with input dimension of m and output dimension of n. In forward propagation, since only m/k of the inputs are active, we can take advantage of index recording to reduce the computational cost to $1/k$ compared with a maxout layer of same size [1]. In back propagation, since both input and output active nodes are sparse, the computation can be reduced to $1/k^2$ of a full matrix computation. Maxout can also take advantage of sparsity of outputs to get $1/k$ computation reduction in back propagation, therefore channel-out training can be k times faster in back propagation also. Note that a channel-out layer and a maxout layer of same size (number of parameters) means that the number of channel-out groups of the channel-out layer is $1/k$ of the number of maxout nodes in the maxout layer.

To confirm that pathway encoding is indeed important in pattern recognition, we record the pathway selections of a well-trained channel-out model (with max(·) channel selection function) using the CIFAR-10 dataset. For ease of visualization and analysis, we set the size of channel-out groups to 2, so that we can use binary codes to represent the pathway selection. To better visualize the space of pathway patterns, we perform PCA analysis on the pathway pattern vectors and project them into the three dimensional space. Figures 3 shows the result. We can see that clusters have been well formed. Another interesting observation in our empirical study is that channel-out models with different initializations result in similar spatial class distributions in 3D PCA space, implying the robustness of pathway code as a feature.

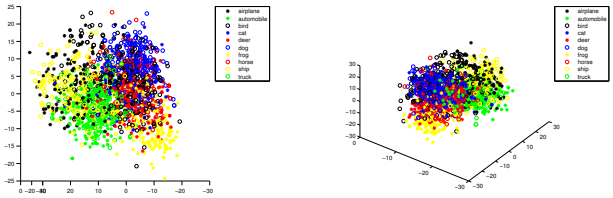

Fig. 3: 3D visualization of the pathway pattern: channel-out

[1] we found empirically that a channel-out networks and a maxout network with similar number of parameters will perform similarly in practice, so the premise for comparison is valid

4 Analysis of the channel-out network

In this section we give an intuitive explanation about why sparse pathway encoding works well in practice, especially when combined with dropout.

Recall that dropout, with each presentation of a training sample, samples a sub-network and encodes the information revealed by the training sample onto this sub-network. Since the sampling of data and sub-networks are independent processes, in a statistical sense the information provided by each training sample will eventually be "squeezed" into each of these sub-networks (third row of Figure 4). The advantage of such scheme, as has been pointed out in various papers [1, 4, 5], is that the same piece of information is encoded into many different representations, adding to the robustness at inference time. The side-effect, which has not been highlighted before, is that encoding conflicting pieces of information densely into sub-networks with small capacities causes interference problem. Data samples of different patterns (classes) attempt to build different, maybe highly conflicting network representations. When the sub-network is not large enough to hold all the information, opposite activations tend to cancel each other, resulting in ineffective encoding.

In contrast to dropout, sparse pathway encoding tends to encode each pattern onto one or a few specialized sub-networks. This is illustrated in the fourth row of Figure 4. Clearly, sparse pathway methods mitigate the interference problem caused by dropout. The problem with pure sparse pathway encoding is the under-utilization of network capacity. Patterns can be compactly encoded on to a small local sub-region of the network, leaving the rest of the network capacity unused. Finally, combining sparse pathway encoding and dropout can take advantage of the strengths of both methods to generate more efficient and accurate information encoding: the whole network will get used due to random sampling by dropout, while individual patterns are still compactly encoded onto certain local sub-networks, so that interference across patterns is much less severe. This is illustrated in the last row of Figure 4.

5 Benchmark Results

In this section we show the performance of the channel-out network on several image classification benchmarks. For all experiment results in this section, the channel selection function used is the $\max(\cdot)$ function. We run tests on CIFAR-10, CIFAR-100 [6] and STL-10 [7], significantly outperforming the state-of-the-art results on CIFAR-100 and STL-10.

Our implementation is built on top of the efficient convolution CUDA kernels developed by Alex Krizhevsky [6]. We got new state-of-the-art performance on CIFAR-100 (63.4%) and STL-10 (69.5%). Our result on CIFAR-10 (86.80%) does not beat sate-of-the-art, but is still competitive, and we believe that better results can be obtained if we spend more time on hyper-parameter tuning.

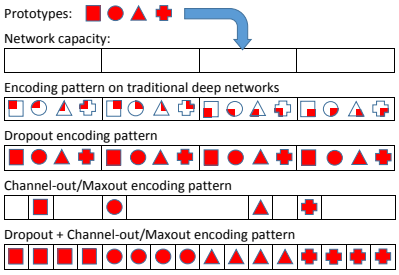

Fig. 4: Information encoding patterns. Each bin in the network capacity box represents a certain size sub-network. Dropout tends to encode all patterns to each capacity bin, resulting in efficient use of network capacity but high level of interference; Sparse pathway methods tend to encode each pattern to a specific sparse sub-network, resulting in least interference but waste of network capacity; The best approach is the combination of the two schemes.

5.1 CIFAR-10 [6]

The network used for CIFAR-10 experiment consists of 3 convolutional channel-out layers, followed by a fully connected channel-out layer, and then the softmax layer. The best model has 64-192-192 filters for the corresponding convolutional layers, and 1210 nodes in the fully connected layer. Dropout regularization is applied. No data augmentation is used.

The result along with the best CIFAR-10 results in the literature are shown in Table 1 (results with no data augmentation). The channel-out network performs a bit worse than the state-of-the-art set by maxout network, but is better than any of the other previous methods as far as we know. We believe that the channel-out performance could be further improved if we use a larger network and the hyper-parameters are better tuned.

Method	Precision
Maxout+Dropout [2]	88.32%
Channel-out+Dropout	**86.80%**
CNN+Spearmint [8]	85.02%
Stochastic Pooling [9]	84.87%

Table 1: Best methods on CIFAR-10

5.2 CIFAR-100 [6]

The CIFAR-100 dataset is similar to CIFAR-10, but with 100 classes. The channel-out network tuned for CIFAR-100 has similar achitecture as that for CIFAR-10. Images are pre-whitened and when presented to the network each time, they are horizontally flipped with probability 0.5. The test set precision was 63.41%, improving the current state-of-the-art by nearly 2 percentage points. Table 2 shows the best results on CIFAR-100.

Method	Precision
Channel-out+Dropout	**63.41%**
Maxout+Dropout [2]	61.43%
Stochastic Pooling [9]	57.49%
Learned Pooling [10]	56.29%

Table 2: Best methods on CIFAR-100

Motivated by the better performance on CIFAR-100 over CIFAR-10, we performed another experiment to test the assumption that channel-out networks might be better at encoding more variant patterns. We extract 10, 20, 50, 100 classes from the original dataset to form 4 classification tasks. We train a channel-out and a maxout network with similar number of parameters for each of the four tasks. We can see from Figure 5 that channel-out performs better on tasks with more classes. We used smaller networks in this experiment for quick test.

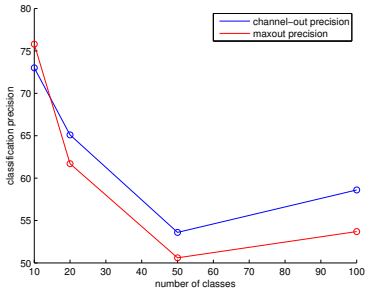

Fig. 5: Comparison of channel-out and maxout on 4 tasks of different difficult levels: channel-out does better on tasks with more classes.

5.3 STL-10 [7]

STL-10 is also a 10-class small images dataset, but with more variant patterns and background clutters than CIFAR-10. The network constructed is similar to that for CIFAR-10. Whitening and flipping are applied to data. Our method improves the current state-of-the-art by 5%, as is shown in Table 3.

Method	Precision
Channel-out+Dropout	**69.5%**
Hierarchical Matching Pursuit [11]	64.5%
Discriminative Learning of SPN [12]	62.3%

Table 3: Best methods on STL-10

6 Conclusions

We have introduced the concept of sparse pathway encoding and argued that this can be a robust and efficient way for encoding categorical information in a deep

network. Using sparse pathway encoding, the interference between conflicting patterns is mitigated, and therefore when combined with dropout, the network can utilize the network capacity in a more effective way. Along this direction we have proposed a novel class of deep networks, the channel-out networks. Our experiments show that channel-out networks perform very well on image classification tasks, especially for the harder tasks with more complex patterns.

7 Acknowledgements

Upon finishing this work, we found that a recent work from the IDSIA lab [13] proposed a similar model as the max(·) version of the channel-out network. Our work was independently developed, and provides a different perspective to explain its success - "sparse pathway encoding", which we believe to be a promising general direction that future research should pay attention to. New and state-of-the-art results are also important contributions of this paper.

References

1. Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.R.: Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580 (2012)
2. Goodfellow, I.J., Warde-Farley, D., Mirza, M., Courville, A., Bengio, Y.: Maxout networks. arXiv preprint arXiv:1302.4389 (2013)
3. Kandel, E.R., Schwartz, J.H., Jessell, T.M., et al.: Principles of neural science. Volume 4. McGraw-Hill New York (2000)
4. Srivastava, N.: Improving neural networks with dropout. PhD thesis, University of Toronto (2013)
5. Wan, L., Zeiler, M., Zhang, S., Cun, Y.L., Fergus, R.: Regularization of neural networks using dropconnect. In: Proceedings of the 30th International Conference on Machine Learning (ICML-13). (2013) 1058–1066
6. Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems 25. (2012) 1106–1114
7. Coates, A., Ng, A.Y., Lee, H.: An analysis of single-layer networks in unsupervised feature learning. In: International Conference on Artificial Intelligence and Statistics. (2011) 215–223
8. Snoek, J., Larochelle, H., Adams, R.P.: Practical Bayesian optimization of machine learning algorithms. arXiv preprint arXiv:1206.2944 (2012)
9. Zeiler, M.D., Fergus, R.: Stochastic pooling for regularization of deep convolutional neural networks. arXiv preprint arXiv:1301.3557 (2013)
10. Malinowski, M., Fritz, M.: Learning smooth pooling regions for visual recognition. In: the British Machine Vision Conference. (2013)
11. Bo, L., Ren, X., Fox, D.: Unsupervised feature learning for rgb-d based object recognition. ISER, June (2012)
12. Gens, R., Domingos, P.: Discriminative learning of sum-product networks. In: Advances in Neural Information Processing Systems. (2012) 3248–3256
13. Srivastava, R.K., Masci, J., Kazerounian, S., Gomez, F., Schmidhuber, J.: Compete to compute. technical report (2013)