BUILDING HIGH-LEVEL FEATURES USING LARGE SCALE UNSUPERVISED LEARNING

Quoc V. Le

Google Inc., USA

ABSTRACT

We consider the problem of building high-level, class-specific feature detectors from only unlabeled data. For example, is it possible to learn a face detector using only unlabeled images? To answer this, we train a deep sparse autoencoder on a large dataset of images (the model has 1 billion connections, the dataset has 10 million 200x200 pixel images downloaded from the Internet). We train this network using model parallelism and asynchronous SGD on a cluster with 1,000 machines (16,000 cores) for three days. Contrary to what appears to be a widely-held intuition, our experimental results reveal that it is possible to train a face detector without having to label images as containing a face or not. Control experiments show that this feature detector is robust not only to translation but also to scaling and out-of-plane rotation. We also find that the same network is sensitive to other high-level concepts such as cat faces and human bodies. Starting from these learned features, we trained our network to recognize 22,000 object categories from ImageNet and achieve a leap of 70% relative improvement over the previous state-of-the-art.

1. INTRODUCTION

The focus of this work is to build *high-level*, class-specific feature detectors from *unlabeled* images. For instance, we would like to understand if it is possible to build a face detector from only unlabeled images. This approach is inspired by the neuroscientific conjecture that there exist highly class-specific neurons in the human brain, generally and informally known as "grandmother neurons." The extent of class-specificity of neurons in the brain is an area of active investigation, but current experimental evidence suggests the possibility that some neurons in the temporal cortex are highly selective for object categories such as faces or hands [1], and perhaps even specific people [2].

Contemporary computer vision methodology typically emphasizes the role of *labeled* data to obtain these class-specific feature detectors. For example, to build a face detector, one needs a large collection of images labeled as containing faces, often with a bounding box around the face. The need for large labeled sets poses a significant challenge for problems where labeled data are rare. Although approaches that make use of inexpensive unlabeled data are often preferred, they have not been shown to work well for building high-level features.

This work investigates the feasibility of building high-level features from only *unlabeled* data. A positive answer to this question will give rise to two significant results. Practically, this provides an inexpensive way to develop features from unlabeled data. But perhaps more importantly, it answers an intriguing question as to whether the specificity of the "grandmother neuron" could possibly be learned from unlabeled data. Informally, this would suggest that it is at least in principle possible that a baby learns to group faces into one class because it has seen many of them and not because it is guided by supervision or rewards.

Unsupervised feature learning and deep learning have emerged as methodologies in machine learning for building features from *unlabeled* data. Using unlabeled data in the wild to learn features is the key idea behind the *self-taught learning* framework [3]. Successful feature learning algorithms and their applications can be found in recent literature using a variety of approaches such as RBMs [4], autoencoders [5, 6], sparse coding [7] and K-means [8]. So far, most of these algorithms have only succeeded in learning *low-level* features such as "edge" or "blob" detectors. Going beyond such simple features and capturing complex invariances is the topic of this work.

Recent studies observe that it is quite time intensive to train deep learning algorithms to yield state of the art results [9]. We conjecture that the long training time is partially responsible for the lack of high-level features reported in the literature.

We address this problem by scaling up the core components involved in training deep networks: the dataset, the model, and the computational resources. First, we use a large dataset generated by sampling random frames from random YouTube videos.[1] Our input data are 200x200 images, much larger than typical 32x32 images used in deep learning and unsupervised feature learning [11, 9, 12, 8]. Our model, a deep autoencoder with pooling and local contrast normalization, is scaled to these large images by using a large computer cluster. To support parallelism on this cluster, we use the idea of local receptive fields, e.g., [13, 12, 14]. This idea

[1] This is different from the work of [10] who trained their model on images from one class.

reduces communication costs between machines and thus allows model parallelism (parameters are distributed across machines). Asynchronous SGD is employed to support data parallelism. The model was trained in a distributed fashion on a cluster with 1,000 machines (16,000 cores) for three days.

Experimental results using classification and visualization confirm that it is indeed possible to build high-level features from unlabeled data. In particular, using a hold-out test set consisting of faces and distractors, we discover a feature that is highly selective for faces. This result is also validated by visualization via numerical optimization. Control experiments show that the learned detector is not only invariant to translation but also to out-of-plane rotation and scaling. Similar experiments reveal the network also learns the concepts of cat faces and human bodies. More details about our results and analyses are discussed in the full version of our paper [15].

The learned representations are also discriminative. Using the learned features, we obtain significant leaps in object recognition with ImageNet. For instance, on ImageNet with 22,000 categories, we achieved 15.8% accuracy, a relative improvement of 70% over the state-of-the-art. Note that, random guess achieves less than 0.005% accuracy for this dataset.

2. METHOD

Our training dataset is constructed by sampling frames from 10 million YouTube videos. To avoid duplicates, each video contributes only one image to the dataset. Each example is a color image with 200x200 pixels.

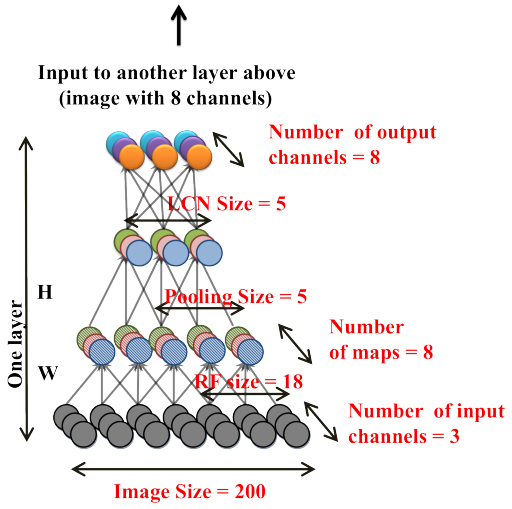

Fig. 1. The architecture and parameters in one layer of our network. The overall network replicates this structure three times. For simplicity, the images are in 1D.

Our algorithm is can be viewed as a sparse deep autoencoder with three important ingredients: local receptive fields, pooling and local contrast normalization. First, to scale the autoencoder to large images, we use a simple idea known as *local receptive fields* [16, 13, 10, 12]. This biologically inspired idea proposes that each feature in the autoencoder can connect only to a small region of the lower layer. Next, to achieve invariance to local deformations, we employ local L2 pooling [17, 18, 12] and local contrast normalization [19]. L2 pooling, in particular, allows the learning of invariant features [17, 12].

Our deep autoencoder is constructed by replicating three times the same stage composed of local filtering, local pooling and local contrast normalization. The output of one stage is the input to the next one and the overall model can be interpreted as a nine-layered network (see Figure 1). The first and second sublayers are often known as filtering (or simple) and pooling (or complex) respectively. The third sublayer performs local subtractive and divisive normalization and it is inspired by biological and computational models [20, 21, 19].[2]

As mentioned above, central to our approach is the use of local connectivity between neurons. In our experiments, the first sublayer has receptive fields of 18x18 pixels and the second sub-layer pools over 5x5 overlapping neighborhoods of features (i.e., pooling size). The neurons in the first sublayer connect to pixels in all input channels (or maps) whereas the neurons in the second sublayer connect to pixels of only one channel (or map). While the first sublayer outputs linear filter responses, the pooling layer outputs the square root of the sum of the squares of its inputs, and therefore, it is known as L2 pooling. Although we use local receptive fields, they are not convolutional: the parameters are not shared across different locations in the image (c.f. [16, 19, 10]). In addition to being more biologically plausible, unshared weights allow the learning of more invariances other than translational invariances [12].

In terms of scale, our network is perhaps one of the largest known networks to date. It has 1 billion trainable parameters, which is more than an order of magnitude larger than other large networks reported in literature, e.g., [9, 22] with around 10 million parameters. It is worth noting that our network is still tiny compared to the human visual cortex, which is 10^6 times larger in terms of the number of neurons and synapses [23].

The model was trained using approximately 1,000 machines using model parallelism (each model is splitted into 144 machines) and asynchronous SGD (with 5 model replicas communicating the parameters asynchronously to a central server of 256 machines). This optimization technique is described in detail in [24].

3. EXPERIMENTS

In this section, we first focus on analyzing learned representations in recognizing faces ("the face detector"). Results for

[2]The subtractive normalization removes the weighted average of neighboring neurons from the current neuron $g_{i,j,k} = h_{i,j,k} - \sum_{iuv} G_{uv} h_{i,j+u,i+v}$ The divisive normalization computes $y_{i,j,k} = g_{i,j,k} / \max\{c, (\sum_{iuv} G_{uv} g^2_{i,j+u,i+v})^{0.5}\}$, where c is set to be a small number, 0.01, to prevent numerical errors. G is a Gaussian weighting window. [19]

other concepts will follow.

The test set consists of 37,000 images sampled from two datasets: Labeled Faces In the Wild dataset [25] and ImageNet dataset [26]. There are 13,026 faces sampled from *non-aligned* Labeled Faces in The Wild.[3] The rest are distractor objects randomly sampled from ImageNet. These images are resized to fit the visible areas of the top neurons.

After training, we used this test set to measure the performance of each neuron in classifying faces against distractors. For each neuron, we found its maximum and minimum activation values, then picked 20 equally spaced thresholds in between. The reported accuracy is the best classification accuracy among 20 thresholds.

The best neuron in the network performs very well in recognizing faces, despite the fact that no supervisory signals were given during training. It achieves 81.7% accuracy in detecting faces. There are 13,026 faces in the test set, so guessing all negative only achieves 64.8%. The best neuron in a one-layered network only achieves 71% accuracy while best linear filter, selected among 100,000 filters sampled randomly from the training set, only achieves 74%.

We also use two visualization techniques to verify if the optimal stimulus of the neuron is indeed a face. The first method is visualizing the most responsive stimuli in the test set. Since the test set is large, this method can reliably detect near optimal stimuli of the tested neuron. The second approach is to perform numerical optimization to find the optimal stimulus [27, 28, 12]. In particular, we find the norm-bounded input x which maximizes the output f of the tested neuron, by solving:

$$x^* = \arg\min_x f(x; W, H), \text{ subject to } ||x||_2 = 1.$$

Here, $f(x; W, H)$ is the output of the tested neuron given learned parameters W, H and input x. In our experiments, this constraint optimization problem is solved by projected gradient descent with line search. Results, shown in Figure 2, confirm that the tested neuron learns the concept of faces.

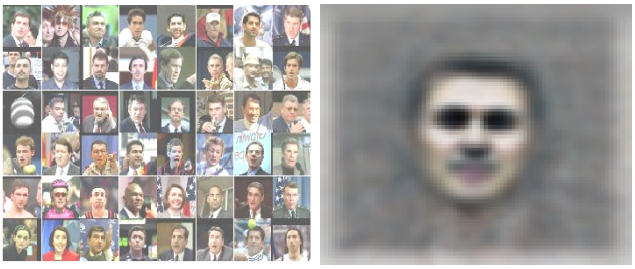

Fig. 2. Top: Top 48 stimuli of the best neuron from the test set. Bottom: The optimal stimulus according to numerical constraint optimization.

The network also learns other high-level concepts as well. In particular, similar visualization reveals that it also learns the concepts of cat faces, human bodies as well as other high-level concepts (see Figure 3).

[3] http://vis-www.cs.umass.edu/lfw/lfw.tgz

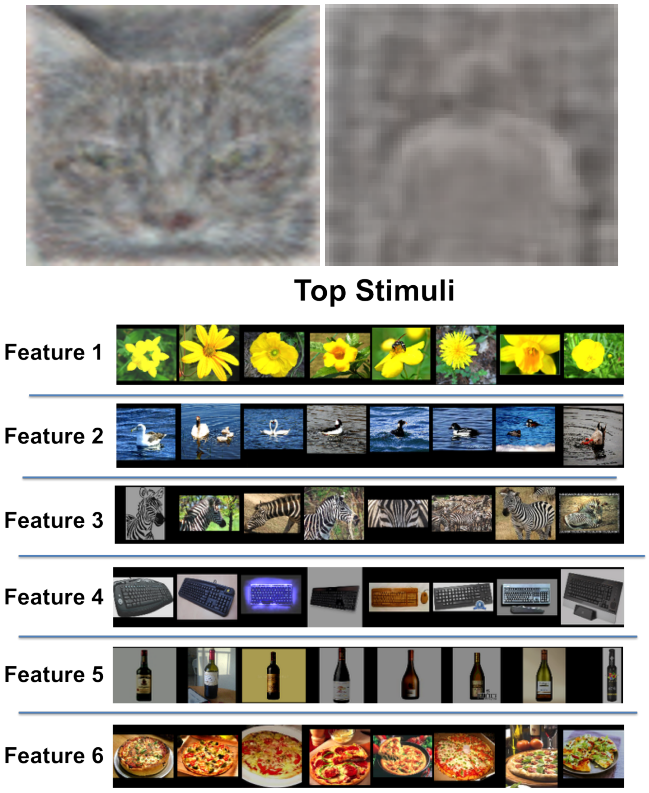

Fig. 3. Visualization of the cat face neuron (top left) and human body neuron (top right), and top stimuli for some of the neurons in the network (bottom).

We then applied the feature learning method to the task of recognizing objects in the ImageNet dataset [26]. Starting from a network that already learned features from YouTube and ImageNet images using the techniques described above, we then added one-versus-all logistic classifiers on top of the highest layer of this network. This method of initializing a network by unsupervised learning is also known as "unsupervised pretraining." During supervised learning with labeled ImageNet images, the parameters of lower layers and the logistic classifiers were both adjusted. This was done by first adjusting the logistic classifiers and then adjusting the entire network (also known as "fine-tuning"). As a control experiment, we also train a network starting with all random weights (i.e., without unsupervised pretraining: all parameters are initialized randomly and only adjusted by ImageNet labeled data).

We followed the experimental protocols specified by [31, 29], in which, the datasets are randomly split into two halves for training and validation. We report the performance on the validation set and compare against state-of-the-art baselines in Table 1. Note that the splits are not identical to previous work but validation set performances vary slightly across different splits.

The results show that our method, starting from scratch

Table 1. Summary of classification accuracies for our method and other state-of-the-art baselines on ImageNet.

Dataset version	2009 (∼9M images, ∼10K categories)	2011 (∼14M images, ∼22K categories)
State-of-the-art	16.7% [29]	9.3% [30]
Our method	16.1% (without unsupervised pretraining) **19.2%** (with unsupervised pretraining)	13.6% (without unsupervised pretraining) **15.8%** (with unsupervised pretraining)

(i.e., raw pixels), bests many state-of-the-art hand-engineered features. On ImageNet with 10K categories, our method yielded a 15% relative improvement over previous best published result. On ImageNet with 22K categories, it achieved a 70% relative improvement over the highest other result of which we are aware (including unpublished results known to the authors of [30]). Note, random guess achieves less than 0.005% accuracy for this dataset.

Acknowledgements: We thank Samy Bengio, Adam Coates, Tom Dean, Jia Deng, Mark Mao, Peter Norvig, Paul Tucker, Andrew Saxe, and Jon Shlens for helpful discussions.

4. REFERENCES

[1] R. Desimone, T. Albright, C. Gross, and C. Bruce, "Stimulus-selective properties of inferior temporal neurons in the macaque," *The Journal of Neuroscience*, 1984.

[2] R. Q. Quiroga, L. Reddy, G. Kreiman, C. Koch, and I. Fried, "Invariant visual representation by single neurons in the human brain," *Nature*, 2005.

[3] R. Raina, A. Battle, H. Lee, B. Packer, and A.Y. Ng, "Self-taught learning: Transfer learning from unlabelled data," in *ICML*, 2007.

[4] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, 2006.

[5] G. E. Hinton and R.R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, 2006.

[6] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layerwise training of deep networks," in *NIPS*, 2007.

[7] H. Lee, A. Battle, R. Raina, and Andrew Y. Ng, "Efficient sparse coding algorithms," in *NIPS*, 2007.

[8] A. Coates, H. Lee, and A. Y. Ng, "An analysis of single-layer networks in unsupervised feature learning," in *AISTATS 14*, 2011.

[9] D. C. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Deep big simple neural nets excel on handwritten digit recognition," *CoRR*, 2010.

[10] H. Lee, R. Grosse, R. Ranganath, and A.Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *ICML*, 2009.

[11] A. Krizhevsky, "Learning multiple layers of features from tiny images," Tech. Rep., University of Toronto, 2009.

[12] Q. V. Le, J. Ngiam, Z. Chen, D. Chia, P. W. Koh, and A. Y. Ng, "Tiled convolutional neural networks," in *NIPS*, 2010.

[13] R. Raina, A. Madhavan, and A. Y. Ng, "Large-scale deep unsupervised learning using graphics processors," in *ICML*, 2009.

[14] Q.V. Le, J. Ngiam, A. Coates, A. Lahiri, B. Prochnow, and A.Y. Ng, "On optimization methods for deep learning," in *ICML*, 2011.

[15] Q.V. Le, M.A. Ranzato, R. Monga, M. Devin, K. Chen, and A.Y. Ng G.S. Corrado, J. Dean, "Building high-level features using large scale unsupervised learning," in *ICML*, 2012.

[16] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient based learning applied to document recognition," *Proceeding of the IEEE*, 1998.

[17] A. Hyvärinen, J. Hurri, and P. O. Hoyer, *Natural Image Statistics*, Springer, 2009.

[18] K. Gregor and Y. LeCun, "Emergence of complex-like cells in a temporal product network with local receptive fields," *arXiv:1006.0448*, 2010.

[19] K. Jarrett, K. Kavukcuoglu, M.A. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?," in *ICCV*, 2009.

[20] N. Pinto, D. D. Cox, and J. J. DiCarlo, "Why is real-world visual object recognition hard?," *PLoS Computational Biology*, 2008.

[21] S. Lyu and E. P. Simoncelli, "Nonlinear image representation using divisive normalization," in *CVPR*, 2008.

[22] P. Sermanet and Y. LeCun, "Traffic sign recognition with multiscale convolutional neural networks," in *IJCNN*, 2011.

[23] B. Pakkenberg, D. P., L. Marner, M. J. Bundgaard, H. J. G. Gundersen, J. R. Nyengaard, and L. Regeur, "Aging and the human neocortex," *Experimental Gerontology*, 2003.

[24] J. Dean, G.S. Corrado, R. Monga, K. Chen, M. Devin, Q.V. Le, M.Z. Mao, M.A. Ranzato, A. Senior, P. Tucker, K. Yang, and A. Y. Ng, "Large scale distributed deep networks.," in *NIPS*, 2012.

[25] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Tech. Rep. 07-49, University of Massachusetts, Amherst, October 2007.

[26] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR*, 2009.

[27] P. Berkes and L. Wiskott, "Slow feature analysis yields a rich repertoire of complex cell properties," *Journal of Vision*, 2005.

[28] D. Erhan, Y. Bengio, A. Courville, and P. Vincent, "Visualizing higher-layer features of deep networks," Tech. Rep., University of Montreal, 2009.

[29] J. Sanchez and F. Perronnin, "High-dimensional signature compression for large-scale image-classification," in *CVPR*, 2011.

[30] J. Weston, S. Bengio, and N. Usunier, "Wsabie: Scaling up to large vocabulary image annotation," in *IJCAI*, 2011.

[31] J. Deng, A. Berg, K. Li, and L. Fei-Fei, "What does classifying more than 10,000 image categories tell us?," in *ECCV*, 2010.