Summaries "Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations"

Weituo Hao 109241801
Stony Brook University
Computer Science Department
Advanced Computer Vision

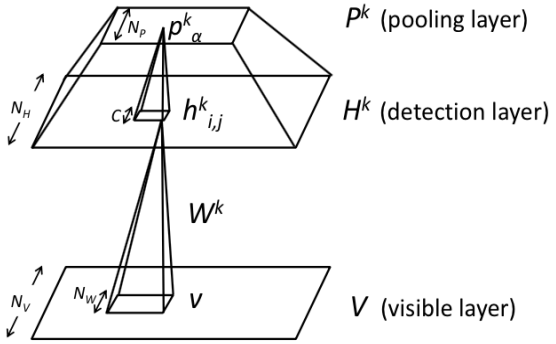
Contribution:
In this article, the author gives us an insightful method to scale unsupervised learning models such as deep belief networks to full-sized, high-dimensional images. The novel key for the algorithm is the probabilistic max-pooling.

Background and Challenges:
The author first introduces several learning models for deep networks and pays attention to deep belief networks (DBNs) particularly. Even though DBNs have successfully been used in controlling domains, there are two challenges claimed by the author. First, since the objects are high-dimensional images, the algorithm should remain computationally tractable when applying to large images. Second, representation should be invariant to local translations since objects may appear at same positions in images.

To address the challenges mentioned above, the writer presents the convolutional deep belief network contains two important keys. First, the algorithm learns features which are shared among all locations in an image. Second, probabilistic max-pooling as a novel technique allows large input to shrink to smaller higher-layer units is proposed.

Keys:
In the preliminaries, the paper describes Restricted Boltzmann Machines (RBM) and deep belief networks (DBNs), two commonly mentioned learning models. But the writer argues that RBMs and DBNs cannot be scaled to full images because weights used for detecting given feature must be learned separately resulting from 2-D structure of images.

After introducing convolutional RBM (CRBM), the writer proposes a new architecture consists of multi-layer CRBMs plus an operation called probabilistic max-pooling. It is similar to max-pooling which is used to allow higher-layer representations to be invariant to translations of the input. But the author claims that max-pooling is only useful for feed-forward architectures. Probabilistic max-pooling supports both top-down and bottom-up inference. The main idea is to add a pooling layer above the detection layer.



The detection layer and pooling layer both have K groups of units, and each group of the pooling layer has $N_p \times N_p$ binary units. The pooling layer shrinks the representation of the detection layer by a factor C. Note that the connection between detection units and pooling unit should obey following constraints: at most one of the detection units may be on, and the pooling unit is on if and only if a detection unit is on. So the energy function of this model is as follows:

$$E(\boldsymbol{v}, \boldsymbol{h}) = -\sum_k \sum_{i,j} (h^k_{i,j} (\widetilde{W}^k * v)_{i,j} + b_k h^k_{i,j}) - c \sum_{i,j} v_{i,j}$$

subj.to

$$\sum_{(i,j) \in \beta_\alpha} h^k_{i,j} \leq 1, \forall k, \alpha$$

To avoid being overcomplete, the author takes the sparsity regularization. In practice, block Gibbs sampling is replaced by mean-field method to approximate the posterior distribution.

Result:

In the result part, the author shows their model has the ability to learn hierarchical representations of natural image, and continues to evaluate their algorithm on Caltech-101 object classification task and claims to achieve 57.7% test accuracy using 15 training images per class, and 65.4% test accuracy using 30 training images per class. Then the performance of the model is evaluated on the MNIST handwritten digit classification task and the writer obtains 0.8 % test error. Lastly, the author also claims that their model can learn hierarchical object-part representations in an unsupervised setting and can tractably perform hierarchical probabilistic inference in full-sized images.

Summaries "Why Does Unsupervised Pre-training Help Deep Learning?"

Weituo Hao 109241801
Stony Brook University
Computer Science Department
Advanced Computer Vision

Contribution:
This article answers the question about how unsupervised pre-training works for deep architects learning. It is suggested that unsupervised learning leads the learning process towards basins of attraction of minima that support better generalization as a regularization.

Main claims:
The author explores how unsupervised pre-training helps more effective unsupervised deep architecture learning via extensive experimentations, and states two main claims. First, unsupervised pre-training serves as a special form of regularization which means minimizing variance and introducing bias towards configurations of the parameter space that are useful for unsupervised learning. Second, generalization effects due to pre-training do not vanish as the size of the labeled examples grows larger.

To support his points, the writer first states that in dealing with the strong dependencies of the parameters across layers, unsupervised pre-training acts as a regularizer. The regularization effect is a consequence of the pre-training procedure establishing an initialization point of the fine-tuning procedure inside a region of parameter space. Another point made by the author is that the effectiveness of the unsupervised learning is limited to the extent that learning P(**X**) is helpful in learning P(**Y**|**X**). He reviews several previous works and points out a common feature of those models. That is the data is usually first transformed in a new representation using unsupervised learning, and a supervised learning is stack on top used to classify the new representations. However, sharing parameters between unsupervised and supervised components can achieve good results. The author gives two examples. One example is Weston et al., 2008 and the other one is from Salakhutdinov and Hinton 2008. In addition, the writer also points out that early stopping is another kind of regularization by constraining the optimization procedure to a region of the parameter space that is close to the initial configuration of parameters.

Experiments:
To test the main claims, the author takes several experiments on deep belief networks and stacked denoising auto-encoders on three data sets: MNIST, InfiniteMNIST, and Shapeset. And the experiments are carried out to compare how

models with pre-training and without pre-training address the apparent local minimum.

Through the experiments, the writer claims that increasing the depth of the learning architect will increase the probability of finding bad apparent local minimum. Compared with random initialization seed, unsupervised pre-training is robust which means the error on average by unsupervised pre-training is lower than the random initialization. By feature vision, the author gives a reasonable explanation that the dynamics induced by pre-training can limit the region of the parameter space that is inaccessible for models without pre-training. Moreover, the author displays the model trajectories to illustrate that pre-training model appears to be more compacted than models without pre-training. Thus the latter one is more inclined to get local minimum. So the conclusion that pre-trained models can achieve better generalization that is robust to random initialization is given by the article.

Further experimental results:
The author emphasizes that even though a gradient-based optimization should end in the apparent local minimum of whatever basin of attraction started from, unsupervised pre-training will end up in a deeper apparent local minimum. And then 8 experiments are carried out to test different hypothesis and conclusions.

1 The first experiment rules out the possibility that pre-training provides a better conditioning process by larger weights. It turns out that unsupervised pre-training does provide better marginal conditioning but the difference is not significant to indicate the discrepancy between pre-trained and non-pre-trained results.

2 The second experiment concludes that unsupervised pre-training appears to be a special kind of regularizer because at a same training cost level, the pre-trained models yield a lower test cost than randomly initialized ones. Furthermore, when the layers are big enough, the unsupervised pre-training models obtain worse training errors but better generalization performance like regulariers in general.

3 The third experiment shows that unsupervised pre-training helps for larger layers and deeper networks but appears to hurt small networks. One possible explanation by the author is that only a small subset of input variations is relevant for predicting the class label. When the hidden layers are small, it is less likely that the transformations for predicting label class are obtained by unsupervised pre-training. One possible reason for this phenomenon is variation present in P(**X**) is less predictive of **Y** than random projections can be.

4 The former three experiments strengthen the regularization hypothesis, and the writer further refutes the optimization hypothesis in the fourth experiment. One problem in optimization is pointed out that the use of early stopping itself can be viewed as a kind of regularizer influences the training error greatly. This can verify the regularization hypothesis inversely.

5 In the fifth experiment, the author compares the regularization effect of unsupervised pre-training with that of L1 and L2, and concludes that for MNIST dataset, L1 and L2 regularization is as near as pre-training, but for InfiniteMNIST dataset, the optimal amount of L1 and L2 regularization is zero.

6 Then the author confirms that the effectiveness of unsupervised pre-training as a regularizer is maintained as the data set grows in the sixth experiment. Since additional capacity cannot be used without pre-training, and the starting point of the non-convex matters, it is not suitable to choose random starting points and continue to optimize.

7 In the seventh experiment, the author verifies the facts that early examples have greater influence and that pre-trained models seem to reduce this influence by comparing models with and without pre-training.

8 For the eighth experiment, the writer pre-trains the first three layers on MNIST and InfiniteMNIST data set respectively. For InfiniteMNIST data set, it is true that pre-training offers benefits for the first two layers, but not the first. As more layers are pre-trained, the models become better generalization. For MNIST data set, the final training error reduces as the pre-training layers increases.

Results:
All in all, the author makes a conclusion that unsupervised pre-training, as a regularizer that only influences the starting point of supervised training, has an effect that does not disappear with more data. However, all the conclusions are obtained based on the datasets used by the author. For other data sets, conclusions can be obtained only after the same experiments are carried out.