Using Representation from Autoencoder or Convolutional Neural Network to Classify Hand-Written Digit Pictures

Xiaodi Zhang[1]

[1] College of Engineering and Computer Science
Australian National University
ACT 2601 AUSTRALIA
u6368740@anu.edu.au

Abstract. Representation learning is to let computer automatically find the features and use these features to do some tasks. In this article, I use autoencoder and convolutional neural network (CNN) to discover the features from hand-written digits. Then I try to use the features they find to classify the digits. I find that both autoencoder and CNN are able to extract features, but only features from CNN can be used to classify the digits.

Keywords: features, autoencoder, CNN, pruning

1 Introduction

The way to choose the data features (or representations) is significant for the performance of machine learning (Bishop, 2006). Researchers have successfully applied many algorithms to representation learning, including supervised learning, semi-supervised learning, and unsupervised learning, and these algorithms can be used in various fields, such as speech recognition, object recognition, natural language processing, and so on (Bengio, Courville, & Vincent, 2013).

Autoencoder is one of unsupervised learning algorithms. It compresses data from input layer into a short code (features) and decompress this code to closely match the original data (Liou, Cheng, Liou, & Liou, 2014). CNN is another kind of representation learning which is inspired by biological process (Matsugu, Mori, Mitari, & Kaneda, 2003). It uses multiple hidden layers, typically consisting of convolutional layers, pooling layers, and fully connection layers. In CNN, convolutional layers and pooling layers are used to extract features.

According to the research from Gedeon (Gedeon, 1995), the quality of features extracted by autoencoder mainly depends on the number of hidden neurons: the more number of hidden neurons, the higher quality of features extracted by autoencoder. However, in his research, only one picture was used to train the autoencoder. To improve the capability of autoencoder, I used 30,000 different hand-written digit pictures to train the autoencoder, expecting to extract the common features of hand-written digit. I then use the features extracted by autoencoder and CNN to classify the hand-written digit. I find that both autoencoder and CNN can extract features successfully, but only the features from CNN is able to classify the digits.

2 Method

2.1 Image Processing

60,000 hand written digit pictures are downloaded from MNIST database. Each picture is transferred to a 1 * 784 array, and each number in the array is transferred ranging from 0 to 1. The pictures in the dataset is shuffled with random sequence.

2.2 Autoencoder

The autoencoder has 3 layers with 784 input neurons and 784 output neurons. The hidden layer uses Sigmoid activation function. The learning rate is 0.05, and the momentum is 0.99. Mean square loss function (MSELoss) is chosen to calculate the error between the decompressed picture and the original grey picture. SGD optimizer is used, and backpropagation method is used to train the neuron network.

2.3 Classifier

The classifier is constructed with one input layer, one output layer, and two hidden layers. Input layer has 98 neurons; the first hidden layer has 1000 neurons, the second layer has 1000 neurons, and the output layer has 10 neurons. Both two hidden layers uses ReLU activation function. The learning rate is 0.05, and the momentum is 0.99. Cross Entropy loss function is chosen to calculate the error between the prediction result and the real digit. SGD optimizer is used, and backpropagation method is used to train the neuron network.

2.4 CNN

The CNN is constructed with one input layer, two convolutional layers, two max pooing layers, and a fully connected layer. In the first convolutional layer, the input height is 1; the output channel is 16; the kernel size is 5; the stride is 1; and the padding is 2. In the second layer, the input height is 16; the out channel is 52; the kernel size is 5, the stride is 1, and the padding is 2. Both two convolutional layers uses ReLU activation function. The kernel size in pooling layers is 2. The learning rate is 0.001, and the batch size is 50. Cross Entropy loss function is chosen to calculate the error between the prediction result and the real digit. Adam optimizer is used, and backpropagation method is used to train the neuron network.

3 Results and Discussion

3.1 Autoencoder is Able to Extract Common Features from Hand-Written Digits.

First of all, 40,000 hand-written digit pictures with 28 by 28 pixels are chosen, and an autoencoder with 98 hidden neurons (compression ratio is 1:8) is created. A series preliminary experiments have been done to determine the best parameters. I find that to get a satisfied result, learning rate and momentum should be set as 0.05 and 0.99, respectively.

After autoencoder being trained by 40,000 epochs, another 10 hand-written digit pictures (from 0 to 9), which are not in training set, are picked to test the decompression result. According to Figure 1, we can find that even though autocoder has never seen these 10 pictures, it can still recognize the features of them and decompress features to well match the original pictures.

This result illustrates that the capability of a autoencoder is not limited to decompress a single picture. However, after being trained by a large number of different pictures with some common features, autoencoder can remember more than one features and decompress a series of different pictures. Comparing to Gedeon's paper (Gedeon, 1995), my work expands the capability of autoencoder, and find autoencoder is so powerful that it can decompress those pictures it has never seen.

Fig. 1. The decompression results from 10 different hand-written digit pictures. The decompression pictures are clear, which suggest that autoencoder has learnt the common features from hand-written digit pictures

3.2 The Quality of Decompressed Pictures Depends on the Number of Hidden Neurons.

Gedeon (Gedeon, 1995) found that while training the autoencoder by a single picture, pruning the hidden neurons leads to the increasing of loss value and low quality of decompressed picture. However, his work just limited in compressing one picture. I expand his work by using multiple pictures.

In my research, I found that the regulation discovered by Gedeon also exists in multi-picture problem. With the number of hidden neurons decreases, the loss value increases to a high level, and the quality of decompressed picture decreases

(Figure 2). I also found an interesting phenomenon that the loss value has a continuously violent fluctuation. The intensity depends on the number of hidden neurons: more hidden neurons can alleviate the intensity.

This experiment also illustrates that autoencoder is able to extract common features from hand-written digit pictures. The number of hidden neurons should not be more than 8, and the training epochs should be more than 20,000.

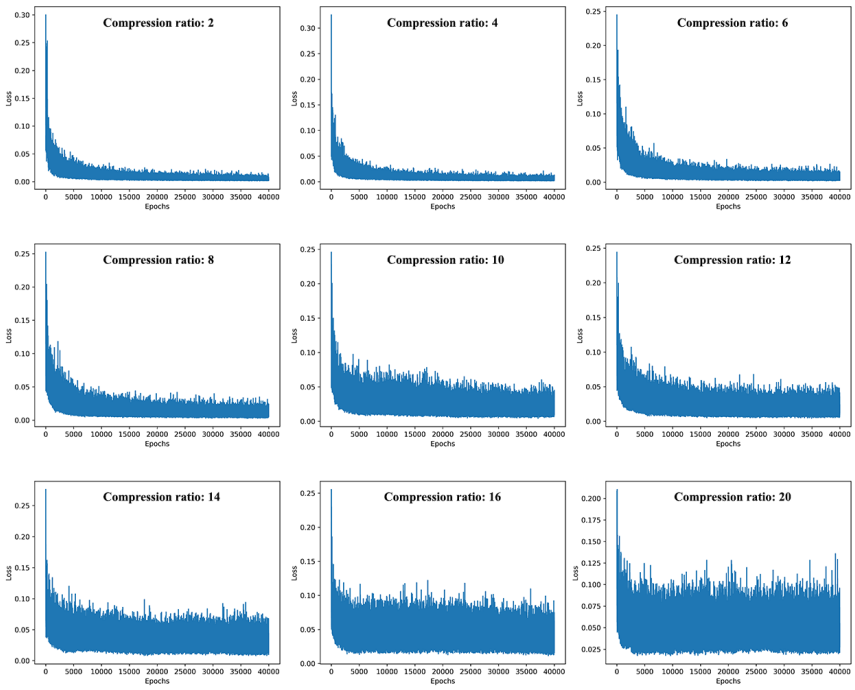

Fig. 2. The loss value from different compression ratio. With the compression ratio increases, the loss value soars to a high value, and the intensity of fluctuation increases to a high level.

3.3 The Features Extracted by Autoencoder is Unable to Classify the Digit

I have successfully extract common features from hand-written pictures, and I hope these features can be used in classify the hand-written digit. A classifier with 98 inputs, two hidden layers, and 10 outputs is constructed. Interestingly, no matter how I change the parameters, the classifier does not work at all. As we can see in Figure 3, the loss value continuously violently fluctuates and cannot decreases to a low level. This result suggests that the features extracted by autoencoder is unable to classify the digits.

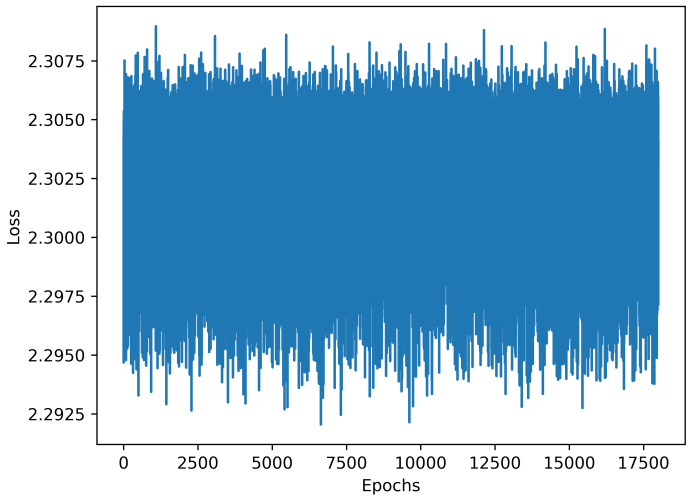

Fig. 3. The classifier (learning rate is 0.05, and momentum is 0.99) is trained by 18,000 features extracted from autoencoder. The loss value continuously keeps at a high level, suggesting these features cannot be used to classify the hand-written digits. No matter how to change parameters, the result is always the same.

3.4 Features Extracted by CNN Performs Excellent in Classification

Although autoencoder can extract common features from hand-written digit pictures, I am disappointed with the fact that these features cannot be used in classification. So, I try to use CNN to extract features and classify the pictures.

As we can see in Figure 4, only after being trained for 200 epochs, the accuracy of classification increases to more than 80%, which illustrates that the features extracted by CNN are useful to classification. The final accuracy of CNN reaches to 98%, which represents an excellent performance in recognize pictures and classification. According to this experiment, we can conclude that the features extracted by autoencoder and CNN are totally different, so the performances of them in classification is also different. Features extracted by CNN can be used in classification but features from autoencoder do not have this capability.

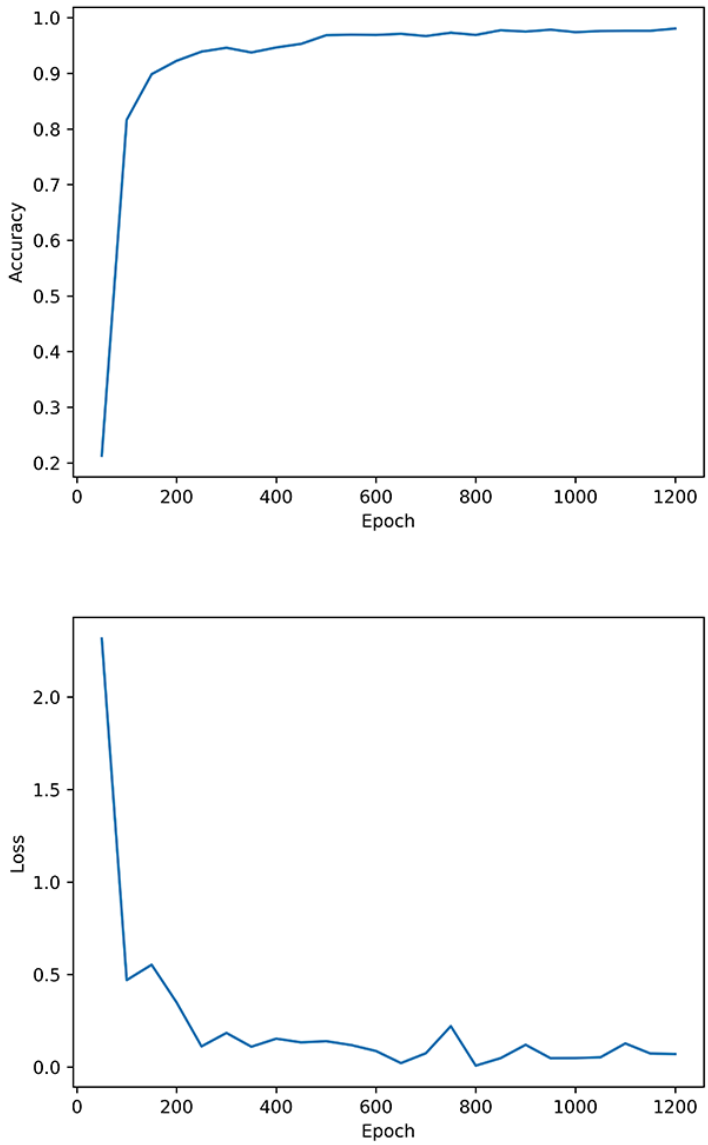

Fig. 4. The accuracy and loss value of classification. After being trained for 200 epochs, CNN can get a high quality of features and excellent performance in classification.

4 Conclusion and Future Work

I have shown that autoencoder can be trained by a series of different pictures with some common features. After being trained by 40,000 hand-written digit pictures, autoencoder can compress and decompress any other hand-written digit pictures, which tremendously expands the work did by Gedeon (Gedeon, 1995) who only let autoencoder be able to compress and decompress only one picture.

I have also found that while the hidden neuron being pruned, the quality of decompressed pictures decreases, and the loss value increases. This result is the same as the result of Gedeon's researches (Gedeon, 1995). However, the loss value is not a flatten line in my experiment, but continuously fluctuates in the training progress. This result illustrates that the

quality of decompressed pictures is not very stable when extracting common features from a large number of different pictures.

Previous results show the successful extraction of common features from hand-written digit pictures. I hope to use these features to classify them. A fully connected classifier with two hidden layers is applied, but the classifier does not work even though I try to use different parameters. This result suggests that the features extracted by autoencoder cannot be used to classification.

My further experiments have shown that CNN is another algorithm to extract common features from hand-written digit pictures. The features extracted by CNN are so powerful that they can be used to classification. The result shows that after being trained for 200 epochs, the extracted features are great enough to classify the pictures with high accuracy.

I finally conclude that the features learnt by autoencoder and CNN are much different. Features extracted by autoencoder can only be used to compress and decompress the pictures, but without capability to classify the pictures. On the other hand, features extracted by CNN can be used to classify, but without capability to decompress the pictures. Different representation learning algorithms can extract different features, and different features should be used in different tasks. So, it is necessary for us to take care of the way to choose the representation learning algorithm before doing some tasks.

The result of autoencoder and CNN seems great for us to understand, but we still have many works to do in the future. First of all, we still do not know why the features extracted by autoencoder cannot be used to classify the pictures, but CNN can do it. It is like a black box that no one know how autoencoder and CNN chooses the features. It is hard to explain the black box, but I hope it will come true in the near future. In addition, the training epochs for autoencoder are still too large, so it wastes amount of time. It is better to improve the autoencoder to speed it up.

In conclusion, the autoencoder can be used to a series of different pictures with common features. The quality of decompressed picture largely depends on the number of hidden neurons. Both autoencoder and CNN can extract features from hand-written digit pictures, but only features extracted from CNN can be used to classification. However, further works need to explain how autoencoder and CNN chooses features from pictures, and why the features from autoencoder cannot classify the pirctures.

References

1. Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, *35*(8), 1798-1828.
2. Everingham, M., Zisserman, A., Williams, C. K., Van Gool, L., Allan, M., Bishop, C. M., ... & Duffner, S. (2006). The 2005 pascal visual object classes challenge. In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment* (pp. 117-176). Springer, Berlin, Heidelberg.
3. Gedeon, T. D. (1995, November). Indicators of hidden neuron functionality: the weight matrix versus neuron behaviour. In *Artificial Neural Networks and Expert Systems, 1995. Proceedings., Second New Zealand International Two-Stream Conference on* (pp. 26-29). IEEE.
4. Liou, C. Y., Cheng, W. C., Liou, J. W., & Liou, D. R. (2014). Autoencoder for words. *Neurocomputing*, *139*, 84-96.
5. Matsugu, M., Mori, K., Mitari, Y., & Kaneda, Y. (2003). Subject independent facial expression recognition with robust face detection using a convolutional neural network. *Neural Networks*, *16*(5-6), 555-559.