An Architecture Combining Convolutional Neural Network and Support Vector Machine for Image Recognition

Haotian Shi
Research School of Computer Science, Australian National University
u6158063@anu.edu.au

Abstract. Image recognition is an important and on-going research subject in machine learning. There has been many algorithms and methods that could achieve acceptable testing accuracy on image classification tasks. In this paper, the model of the combination of a convolutional neural network and a support vector machine was constructed and trained on the MNIST dataset and the Fashion-MNIST dataset, with comparison of a normal convolutional neural network and a multilayer perceptron network. The results showed that the CNN-SVM model could achieve high testing accuracy on the two datasets. To further the research on the heuristic pattern reduction method, the multilayer perceptron network was also trained on the datasets that had applied heuristic pattern reduction. The results showed that heuristic pattern reduction reduced the training time although it failed to improve the generalisation of the network.

Keywords: convolutional neural network, support vector machine, image recognition, heuristic pattern reduction

1 Introduction

Image recognition, in the context of machine learning, is the ability of computer to identify objects, places, people, writing and actions in images. It has been used to perform a large number of machine-based visual tasks, such as marking the content of images with meta-tags, performing image content search and guiding autonomous robots, self-driving cars and accident avoidance systems. Handwritten digit recognition is a challenging problem that has been intensely studied for many years in the field of image recognition. Numerous results have been achieved by researchers who have used different algorithms, such as neural networks (NNs), support vector machines (SVMs), k-nearest-neighbours (KNNs) and convolutional neural networks (CNNs).

One dataset used in this paper is Modified National Institute of Standards and Technology database, usually known as MNIST, which is a large database of handwritten digits that is commonly used for training various machine learning models. MNIST has a training set of 60,000 examples, and a test set of 10,000 examples of the handwritten digits 0–9. The images contain grey levels as a result of the anti-aliasing technique used by the normalization algorithm, and are centred in a 28 x 28 image by computing the centre of mass of the pixels, and translating the image so as to position this point at the centre of the 28 x 28 field. Another dataset used is Fashion-MNIST, which is a dataset of Zalando's article images—consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes. It shares the same image size and structure of training and testing splits with MNIST.

CNNs are the current state-of-the-art model architecture for image classification tasks. A CNN is a multi-layer neural network that can be viewed as the composition of two parts: an automatic feature extractor and a trainable classifier. It applies a series of filters to the raw pixel data of an image to extract and learns higher-level features, which the model can then use for classification. One key factor in the success of an image recognition system is feature extraction, which is the CNNs good at. Szarvas et al. (2005) research the automatically optimised features learned by the CNN on pedestrian detection, and find that the combination of CNN and SVM could generate the highest testing accuracy. Mori et al. (2005) trained the convolutional spiking neural network with time domain encoding the schemes using various fragments of images. The outputs of each layer in the network were sent to the SVM as features. The result shows that the SVM could obtain 100% face recognition rate on the 600 images of 20 people. Inspired by these works, Niu and Suen (2012) propose a hybrid CNN–SVM model for handwritten digit recognition. This model automatically retrieves features based on the CNN architecture, and recognizes the unknown pattern using the SVM recognizer. It could achieve high testing accuracy on MNIST dataset.

In this paper, a CNN-SVM model was implemented and tested on both MNIST and Fashion-MNIST dataset. In order to evaluate the improvement of its performance, a normal CNN model and a multilayer perceptron (MLP) network were constructed and tested on the two datasets. Gedeon (1992) proposes heuristic pattern reduction (HPR) method which could reduce the number of training patterns to avoid overtraining the neural network. This method had been proved to be quite effective on some datasets. To investigate whether HPR could improve the generalisation of MLP, the MLP model was also trained on the two datasets that had been processed by HPR method. All the results of the models on different datasets are compared in the last section of this paper.

2 Method

2.1 MNIST and Fashion-MNIST Dataset

MNIST and Fashion-MNIST datasets were used in paper. Both of them have 60,000 patterns for training and 10,000 patterns for testing. MNIST consists of images of handwritten digits labelled from 0 to 9 while Fashion-MNIST consists of images of clothes and shoes in ten labelled classes. All the patterns have been formatted to 28x28-pixel monochrome images, which makes it possible to change the dataset without modifying any part of the models.

Uniform Manifold Approximation and Projection (UMAP) is a dimension reduction technique that can be used for visualisation, but also for general non-linear dimension reduction. 70,000 patterns of each dataset, including training set and testing set, were projected to a two-dimensional figure and labelled with different colours shown below.

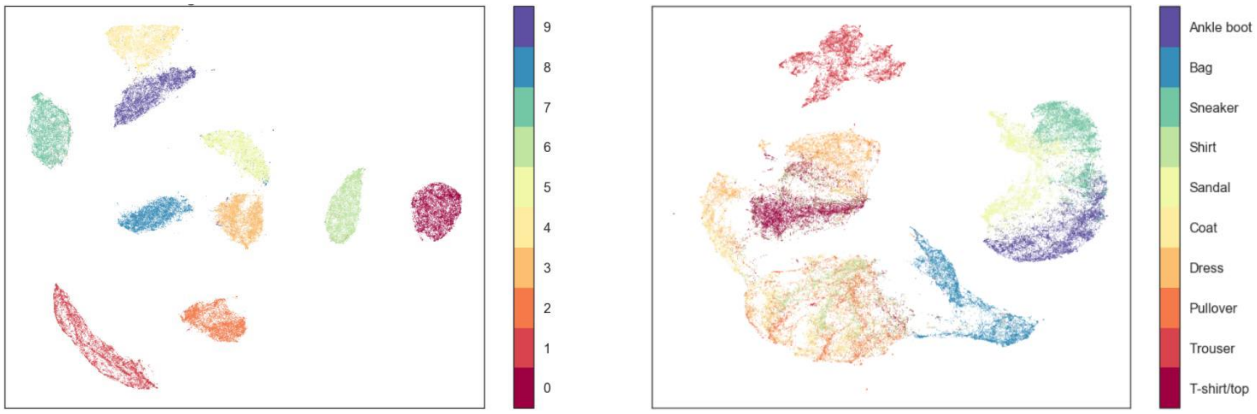

Fig. 1. MNIST digits embedded using UMAP (left) and Fashion-MNIST embedded using UMAP (right)

It can be concluded from the figures that the MNIST dataset might be easy to obtain high accuracy scores for the models while the Fashion-MNIST might not. Consequently, the performance of the models could be reflected better on the Fashion-MNIST dataset, which is also the additional experiment taken on CNN-SVM model than the work of Niu and Suen.

2.2 CNN Model Architecture

A CNN is typically composed of a stack of convolutional modules that perform feature extraction. Each module consists of a convolutional layer followed by a pooling layer. The last convolutional module is followed by one or more dense layers that perform classification. The final dense layer in a CNN contains a single node for each target class in the model (all the possible classes the model may predict), with a softmax activation function to generate a value between 0 and 1 for each node. We can interpret the softmax values for a given image as relative measurements of how likely it is that the image falls into each target class. When taking the log of that value, the value increases (and is negative), which is the opposite of what we want, so we simply negate the answer. Consequently, the Negative Log Likelihood (NLL) loss was selected as the loss function. The internal formula for the loss is as follows:

$$L_i = -\log\left(\frac{e^{f_{yi}}}{\sum_j e^{f_j}}\right) \tag{1}$$

where f is a vector that computed from the forward propagation of the network and i indexes the output neurons.

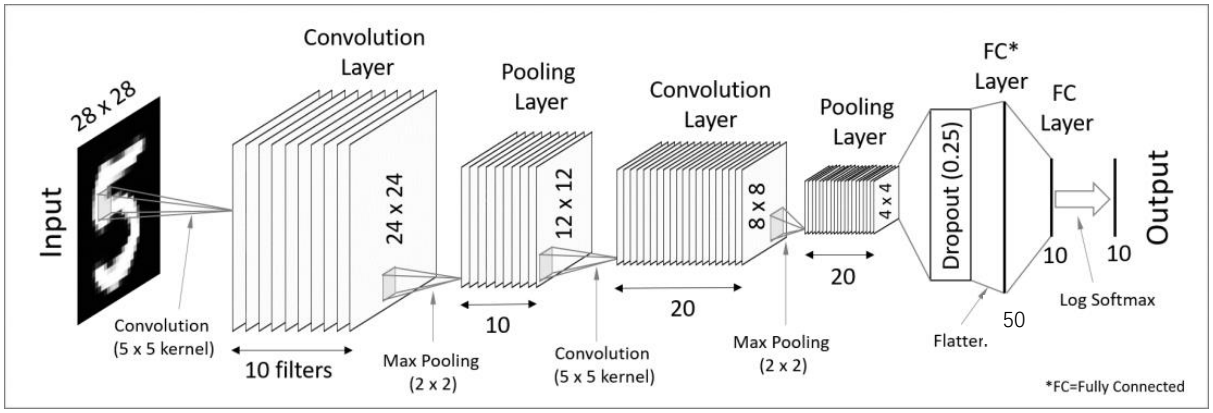

Fig. 2. Architecture of the CNN model

The CNN takes a 28 x 28 pixels grayscale image as input. The first convolutional layer applies 5x5 filters (extracting 5x5-pixel sub-regions), with ReLU activation function. Then performing max pooling with a 2x2 filter and stride of 2 (which specifies that pooled regions do not overlap). The second convolutional layer and pooling layer is similar to the first one, except the different of the number of filters. After dropout regularization with the rate of 0.25, there are two fully connected dense layers which classify the input to 10 target classes.

2.3 CNN-SVM Model Architecture

The architecture of the CNN-SVM model was constructed by replacing the last output layer of the CNN model with an SVM classifier. The outputs of the final dense layer in the CNN are ten values between 0 and 1 computed by the softmax activation function. The input of the activation function is the linear combination of the outputs from the previous hidden layer with trainable weights, plus a bias term. The output values of the hidden layer not only make sense to the CNN model, but also can be treated as input features for other classifiers.

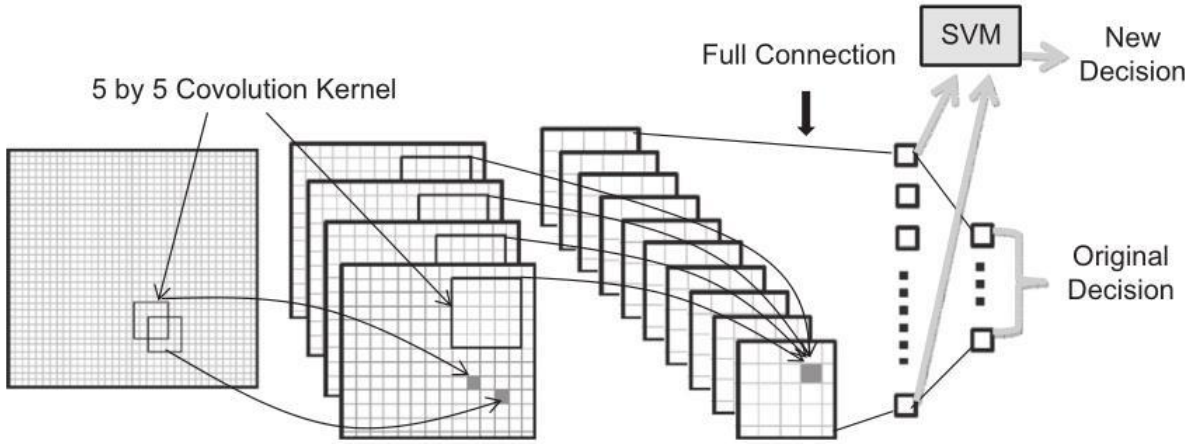

Fig. 3. Architecture of the CNN-SVM model

The SVM takes the outputs from the hidden layer as a new feature vector for training. Once the SVM classifier finishes training, it will be used to perform classification tasks.

2.4 MLP Model Architecture

The MLP model contains two hidden layers with ReLU activation function. After comparing the performance of the MLP with different number of hidden layer neurons, the number was set to 256. Dropout of keep probability 0.5 is used for regularization after each activation function. The loss function is cross-entropy loss. Adam was selected as the optimisation algorithm to update the model parameters based on the computed gradients. The classification accuracy would be taken into comparison with the CNN model and CNN-SVM model.

2.5 Heuristic Pattern Reduction

According to Gedeon, Wong and Harris (1995), the data set for HPR was selected based on the percentage of loss during training using each of the original data set. The patterns were then sorted in ascending order and the new training set was formed using every second sorted instance. By doing this we can remove half of the good and noisy data. To further investigate the effect of HPR, the MLP model will be trained on the two datasets which have been processed by HPR.

2.6 Performance Evaluation

The goal of a classification model is to learn patterns that generalise well for unseen data instead of just memorizing the data that it was shown during training. For multi-label classification model, accuracy is a good method to evaluate the model performance.

Accuracy (ACC) is calculated as the number of all correct predictions divided by the total number of the dataset, which is the classification accuracy for the test dataset. The best accuracy is 1.0, whereas the worst is 0.0. It can also be calculated by 1 – error rate.

$$ACC = \frac{\mathrm{M}}{\mathrm{N}} \tag{2}$$

where M is the number of correct predictions and N is the total number of patterns.

3 Results and Discussion

3.1 Results on MNIST Dataset

The three models were trained on the MNIST dataset and tested on 10,000 patterns. The number of epoch should be adjusted for each model in order to achieve its better performance.

Table 1. Accuracy of different classifiers on MNIST dataset

Classifier	MLP	CNN	CNN-SVM
Accuracy (%)	96.81	99.18	99.17

The results showed that both CNN and CNN-SVM performed much better than MLP on this image classification task. Both of the models could achieve relatively high accuracy on this dataset, which showed that the MNIST patterns are very easy to classify. However, CNN-SVM didn't show significant performance improvement than CNN on this dataset, instead, their accuracy was almost the same.

3.2 Results on Fashion-MNIST Dataset

When training the models on the Fashion-MNIST dataset, the models and parameters didn't need to be modified.

Table 2. Accuracy of different classifiers on Fashion-MNIST dataset

Classifier	MLP	CNN	CNN-SVM
Accuracy (%)	87.82	89.36	90.91

The results proved the discussion in Section 2.1 that Fashion-MNIST is much harder to classify than MNIST. All of the models didn't achieve very high accuracy as on MNIST. However, the performance of CNN-SVM was much better than CNN. It can be reasonably concluded that the combination of CNN and SVM could improve the classification accuracy than only using CNN.

3.3 Results of Applying HPR

Each of the MNIST and Fashion-MNIST dataset has 60,000 patterns in the training set, so the HPR should be modified to avoid too long training time even using GPU. The batch size was set to 10 and HPR would remove 10 patterns each time. The training set reduced to half of its original size after processing.

Table 3. Accuracy of MLP on different dataset

Dataset	MNIST	MNIST (HPR)	Fashion-MNIST	Fashion-MNIST (HPR)
Accuracy (%)	96.81	96.64	87.82	87.27

It can be seen from the table that the HPR method failed to improve the generalisation of the MLP on MNIST and Fashion-MNIST. However, the testing accuracy didn't have significant decrease and the training time reduced by half. Consequently, HPR is worth to try for reducing training time without significant decrease on accuracy although its positive effects don't hold in general.

3.4 Comparison with Other's Work

Maji and Malik (2009) suggest that with improved features a low complexity classifier, in particular an additive-kernel SVM, can achieve state of the art performance. The additive-kernel SVM was tested on the MNIST and USPS dataset. This approach achieves an error of 0.79% on the MNIST dataset, so the classification accuracy of their SVM is 99.21%. It can be seen that the accuracy of CNN model and CNN-SVM model have little difference with the additive-kernel SVM. In my perspective, although the MNIST dataset is very popular in image recognition research, it is too easy to achieve high testing accuracy and hard to measure the performance of different models.

Dufourq and Bassett (2017) propose Evolutionary Deep Networks (EDEN), a computationally efficient neuro-evolutionary algorithm which interfaces to any deep neural network platform. EDEN evolves simple yet successful architectures built from embedding, 1D and 2D convolutional, max pooling and fully connected layers along with their hyperparameters. EDEN was tested on 7 datasets, including MNIST and Fashion-MNIST. The testing accuracy is 98.4 ± 0.3 on MNIST and 90.6 ± 0.5. We can see that CNN-SVM model has similar accuracy with EDEN on Fashion-MNIST but higher accuracy on MNIST. In addition, I think Fashion-MNIST could be a potential replacement to MNIST because it can do better on evaluating the performance of models.

4 Conclusion and Future Work

In this paper, I investigated the performance of CNN-SVM model on the MNIST dataset and the Fashion-MNIST dataset, with comparison with a CNN model and a MLP model. All of the models could achieve relatively high testing accuracy on the MNIST dataset but the CNN-SVM model achieved the highest accuracy on the Fashion-MNIST dataset. By comparing with other's work, the combination of CNN and SVM is proved to be a good architecture on the two datasets. To extend the research on HPR in the previous paper, MLP was trained on the two datasets which had been processed by HPR. The results showed that HPR didn't improve the testing accuracy of MLP, so the improvement on generalisation of HPR doesn't hold in general, just as Gedeon mentioned in his paper. However, HPR could reduce the training time significantly without large accuracy decrease on some datasets. Consequently, this method is still worth to try in the future.

I'm currently planning to modify the CNN-SVM model and test it on more datasets in order to further investigate the performance and generalisation of CNN-SVM model. In the future, I will try to apply some pre-processing methods in the training sets and see whether they can achieve better performance than the model in this paper.

References

Dufourq, E., & Bassett, B. (2017). *EDEN: Evolutionary Deep Networks for Efficient Machine Learning.* Retrieved from arXiv.org e-Print archive: https://arxiv.org/abs/1709.09161

Gedeon, T., Wong, P., & Harris, D. (1995). Balancing bias and variance: Network topology and pattern set reduction techniques. *IWANN 1995: From Natural to Artificial Neural Computation*, pp. 551-558.

Maji, S., & Malik, J. (2009). *Fast and Accurate Digit Classification.* EECS Department, University of California, Berkeley. Retrieved from http://www2.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-159.pdf

Mori, K., Matsugu, M., & Suzuki, T. (2005). Face recognition using SVM fed with intermediate output of CNN for face detection. *Proceedings of the IAPR Conference on Machine Vision Applications*, pp. 410-413.

Niu, X., & Suen, C. (2012). A novel hybrid CNN–SVM classifier for recognizing handwritten digits. *Pattern Recognition, 45*, pp. 1318-1325.

Szarvas, M., Yoshizawa, A., Yamamoto, M., & Ogata, J. (2005). Pedestrian detection with convolutional neural networks. *Proceedings of the IEEE on Intelligent Vehicles Symposium*, pp. 224–229.