Mini-projects - Project I

EE-559: Deep Learning

May 22, 2020

Name: Portmann Cédric **Sciper: 257168**

Name: Gilbert Camille **Sciper: 251724**

Name: Brunoro Guilain **Sciper: 246898**

Contents

1 Project 1 - Classification, weight sharing, auxiliary losses

1.1 Introduction

The goal of this project is to implement a deep network such that, given as input a series of 2×14×14 tensor, corresponding to pairs of 14×14 grayscale images. The developed models predict for each pair if the first digit is lesser or equal to the second. The training and test set are 1000 pairs each.

Name	Tensor dimension	Type	Content
train_input	$N \times 2 \times 14 \times 14$	float32	Images
train_target	N	int64	Class to predict $\in \{0, 1\}$
train_classes	$N \times 2$	int64	Classes of the two digits $\in \{0, \ldots, 9\}$
test_input	$N \times 2 \times 14 \times 14$	float32	Images
test_target	N	int64	Class to predict $\in \{0, 1\}$
test_classes	$N \times 2$	int64	Classes of the two digits $\in \{0, \ldots, 9\}$

1.2 Models

All models were trained with the following parameters:

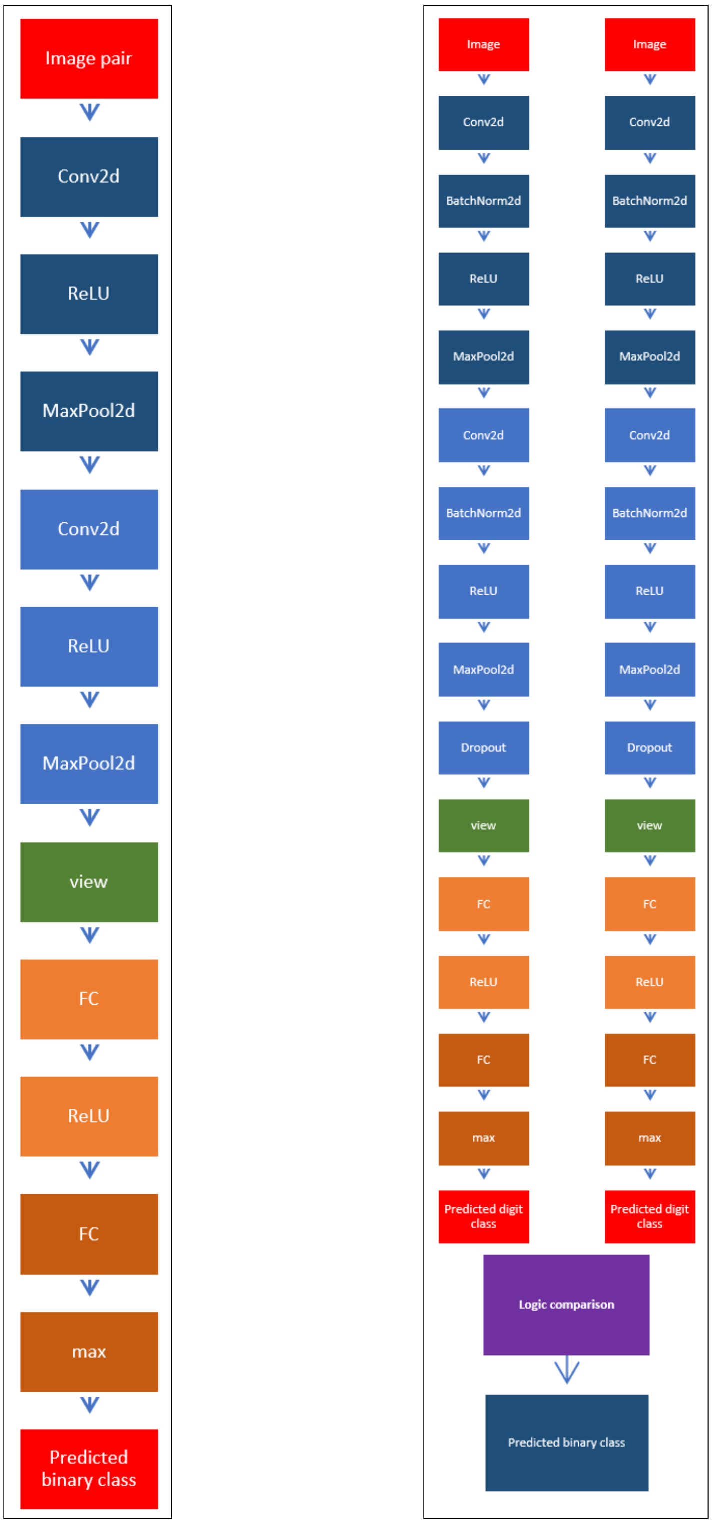
1. mini batch size = 100
2. hidden layers = 128
3. epochs = 25
4. learning rate = 0.001
5. training/validation split = 80%/20%

1.2.1 SimpleCNN

SimpleCNN is a convolutional neural network that makes a binary prediction based on two input channels, each with one image. The model is trained directly on the binary class to predict the boolean value that is actually of interest. SimpleCNN consist of two convolutional layers and two fully connected layers. In conjunction with the convolution layers, pooling layers (MaxPool2d) were used to reduce the resolution of the network from the previous input layer, leading to fewer parameters in lower layers. This compression results in faster computation and helps prevent over-fitting of the network.

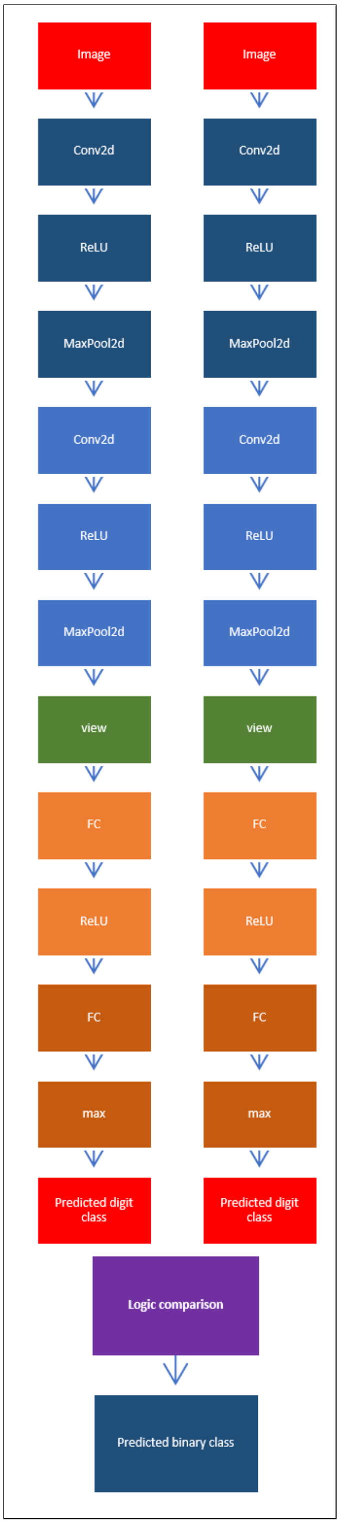
1.2.2 AdvancedCNN

AdvancedCNN is a convolutional neural network that makes a prediction of ten possible digit classes based on an image loaded into the input channel. This approach differs from SimpleCNN because it makes use of an auxiliary loss and AdvancedCNN is therefore expected to outperform SimpleCNN. Additional logic is subsequently required to compare which of the two images contains the larger value in order to predict the boolean value that is actually of interest. The model is trained on the digit classes and predicts the handwritten digit of an input image. AdvancedCNN consist of two convolutional layers and two fully connected layers. In conjunction with the convolution layers, pooling layers were used. Furthermore a dropout layer was added to counteract the tendency of AdvancedCNN to overfitting training data, resulting in a random bunch of nodes within the network that are not updated during training of the model. AdvancedCNN also contains batch normalization to ensure that each minibatch that goes through the network has a mean centered around zero with a variance of 1.



(a) SimpleCNN (b) AdvancedCNN

Figure 1: These graphs represent the implementations of the SimpleCNN and AdvancedCNN models. An implementation consists of a respective input, a model and an output stage.



(a) SiameseCNN

Figure 2: This graph represent the implementation of SiameseCNN. Weight sharing has no influence on the structure.

1.2.3 SiameseCNN without weightsharing

SiameseCNN is a convolutional neural network that makes a prediction of ten possible digit classes based on image pairs that were loaded into input channels. Each of the two channels feeds into a corresponding branch leading to a shared structure. This structure is similar to the AdvancedCNN without the Dropout layer. Each branch has its proper neural network with independent weights. CrossEntropy loss is used on each branch and combined in a weighted sum. We would like to mention that MarginRanking loss could also have been an alternative as it is suitable for our purpose of ranking the inputs. The comparator then aims to classify the output of the two branches 10 classes output into the 2 classes superior/inferior criterion.

1.2.4 SiameseCNN with weightsharing

SiameseCNN with weight sharing is a convolutional neural network that makes a prediction of ten possible digit classes based on one input channel containing an image. The global structure stays exactly the same as with SiameseCNN but the two branches share the exact same parameters i.e. weights. This weight sharing aims to link the two networks for learning purposes.

2 Discussion

The best performance was achieved with the AdvancedCNN in combination with a subsequent comparison logic. This was to be expected since the training of this model took the digit classes for each image pair into account, i.e. exploited the auxiliary loss. The binary prediction, as required in the problem definition, is implemented as a logic comparison. SiameseCNN has better accuracy than SimpleCNN by taking the two channels separately as input. SiameseCNN with weight sharing achieve a better performance than without weight sharing, due to the advantage of learning on both branches of the architecture at the same time. We conclude and statistically confirm that AdvancedCNN in combination with a subsequent comparison logic outperformed all other models.

2.1 Performance evaluation

Model	**Training**	**Validation**	**Testing**
SimpleCNN	88.78% +/- 2.32%	74.92% +/- 2.83%	78.80% +/- 5.11%
AdvancedCNN	98.45% +/- 0.32%	97.12% +/- 0.80%	97.67% +/- 0.68%
SiameseCNN	93.80% +/- 0.74%	91.34% +/- 1.57%	91.42% +/- 1.18%
SiameseCNN with weight sharing	94.26% +/- 0.79%	91.72% +/- 1.75%	93.48% +/- 1.31%

Table 1: The numbers indicate ”mean +/- standard deviation” for each model.