Model Engineering – Dropout v BatchNorm

Explainable Machine Learning - Deep Learning Life Cycle

Jonas Amling   Baptiste Bony   Benedikt Marsiske

February 1, 2023

University of Bamberg

Table of contents

Research Question

Research Question

Our main Model Engineering challenges:

- Accuracy performance
- Prevent overfitting for the model to be generalizable to new data

Research Question: **Do dropout layers prevent overfitting ?**

In addition to Dropout, we tried another regularization technique, namely Batch Normalization.

Overfitting

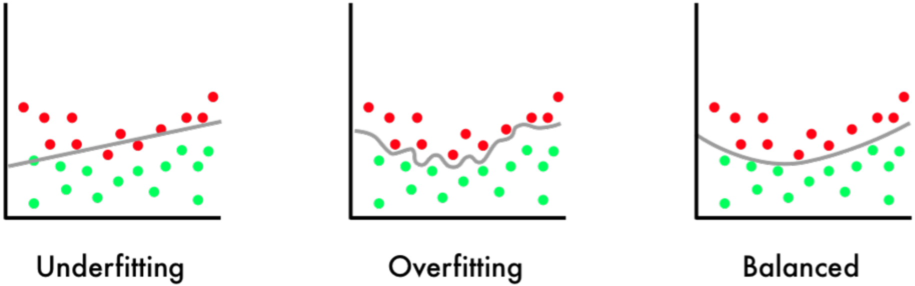

Figure 1: Visual explanation of Overfitting

Model Engineering Process

Overview of the different datasets

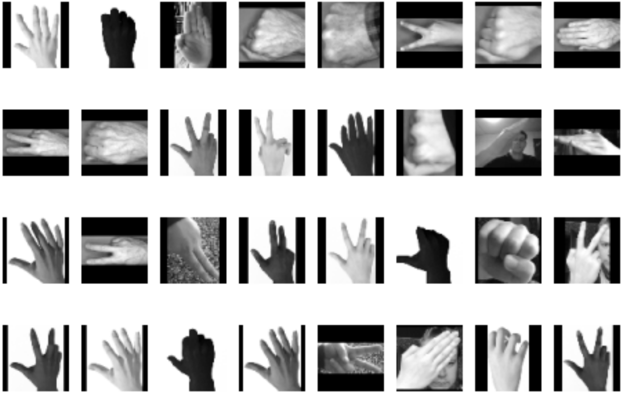

Figure 2: Sample of the Training Dataset

Overview of the different datasets

Figure 3: Sample of the Validation Dataset

Overview of the different datasets

Figure 4: Sample of the Testing Dataset

Our first model

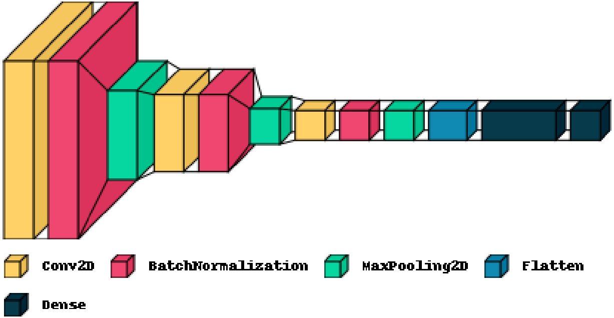

Figure 5: Visualization of our first model

VGG16 [1]

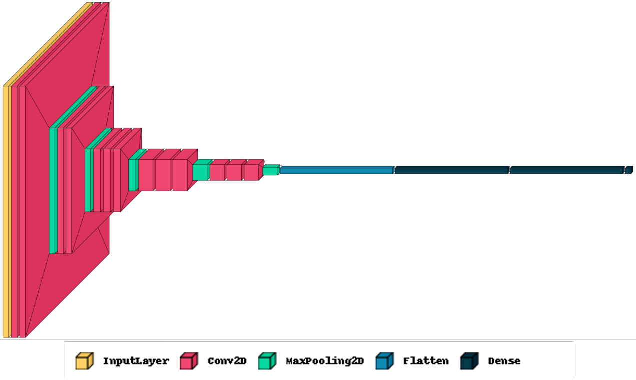

Figure 6: Visualization of the VGG16 model

Our custom model

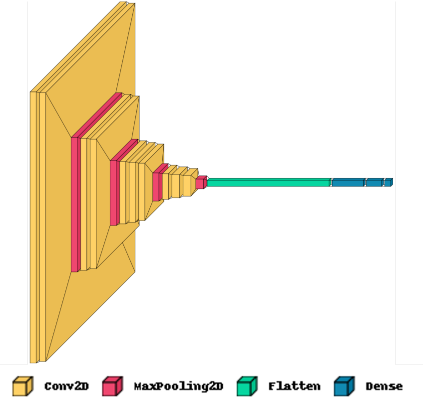

Figure 7: Visualization of our model

Potential Solution: Dropout

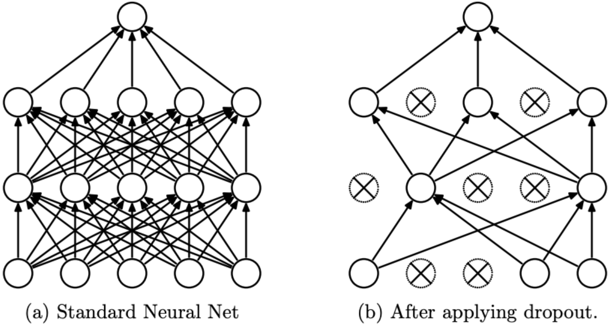

Figure 8: Scheme explaining the principle of Dropout Layers

Potential Solution: Batch Normalization

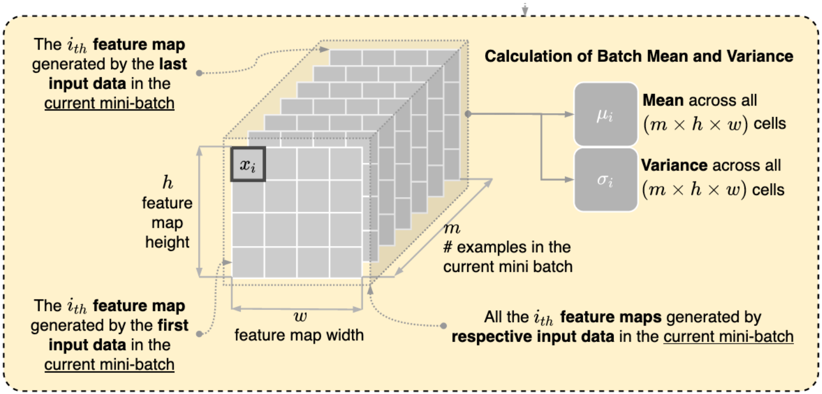

Figure 9: Scheme explaining the principle of Batch Normalization

Experiment

Model performance without any Regularization

Figure 10: Training v Validation Loss

Figure 11: Training v Validation Accuracy

Model performance without any Regularization

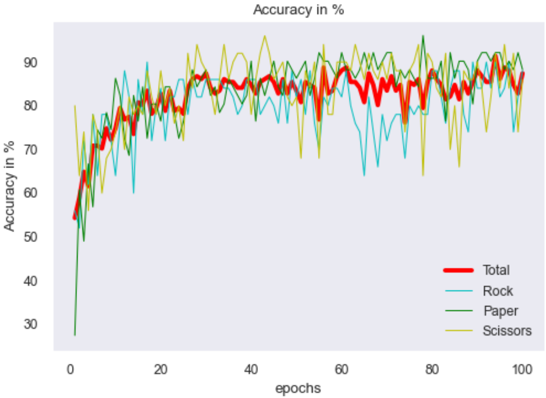

Figure 12: Validation Accuracy in detail

Model performance without any Regularization

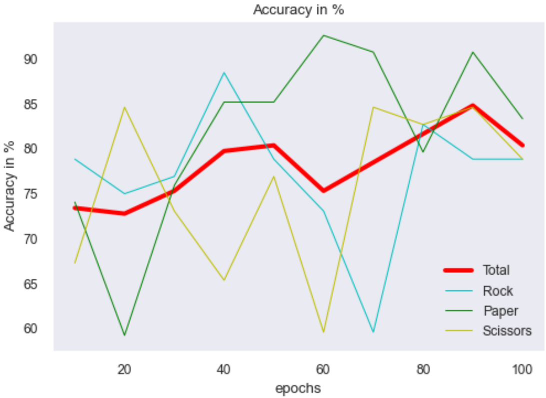

Figure 13: Testing Accuracy in detail

Model with Dropout

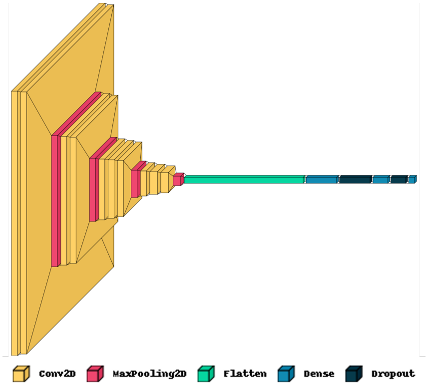

Figure 14: Visualization of our model with Dropout

Model performance with Dropout

Figure 15: Training v Validation Loss

Figure 16: Training v Validation Accuracy

Model performance with Dropout

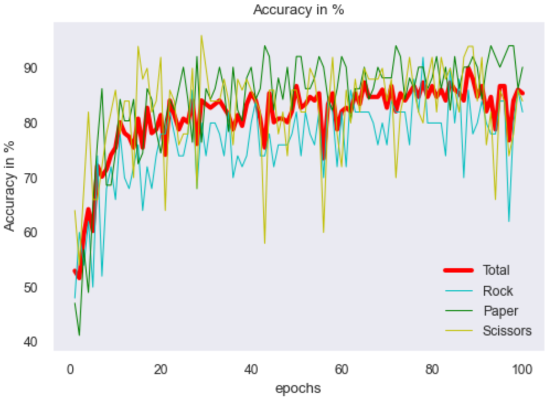

Figure 17: Validation Accuracy in detail

Model performance with Dropout

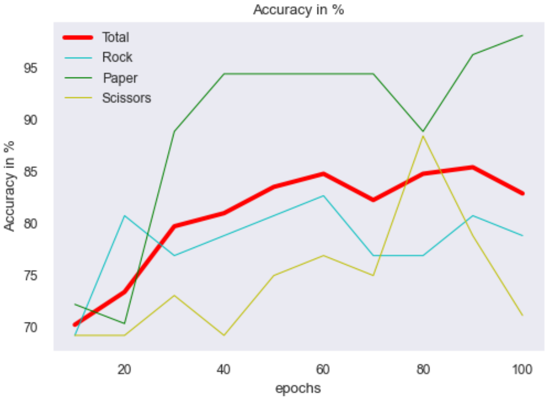

Figure 18: Testing Accuracy in detail

Model with Batch Normalization

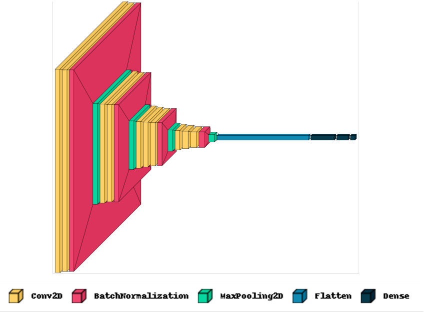

Figure 19: Visualization of our model with Batch Normalization

Model performance with Batch Normalization

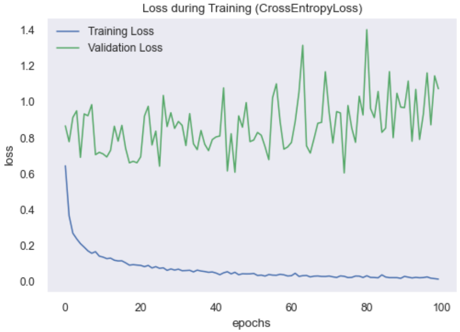

Figure 20: Training v Validation Loss

Figure 21: Training v Validation Accuracy

Model performance with Batch Normalization

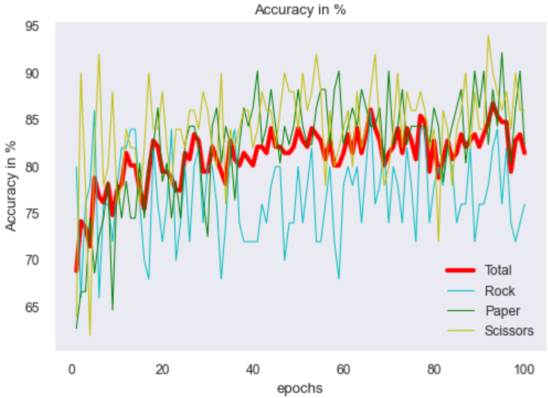

Figure 22: Validation Accuracy in detail

Model performance with Batch Normalization

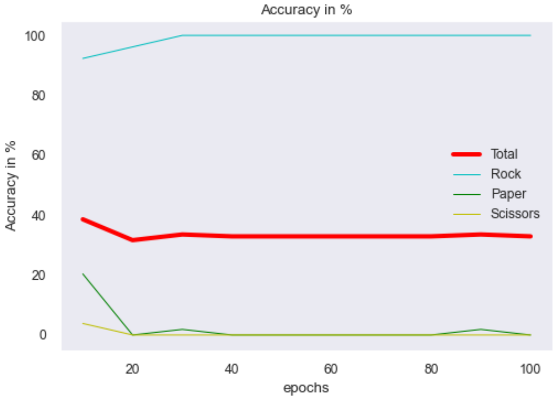

Figure 23: Testing Accuracy in detail

Model with both Dropout & Batch Normalization

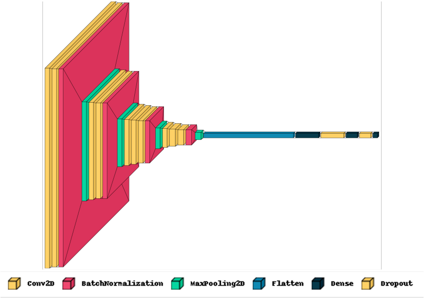

Figure 24: Visualization of our model with both Dropout & Batch Normalization

Model performance with both Dropout & Batch Normalization

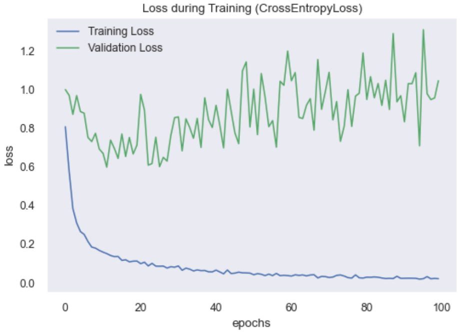

Figure 25: Training v Validation Loss

Figure 26: Training v Validation Accuracy

Model performance with both Dropout & Batch Normalization

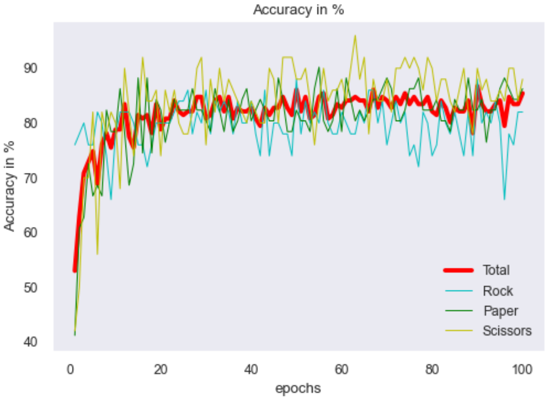

Figure 27: Validation Accuracy in detail

Model performance with both Dropout & Batch Normalization

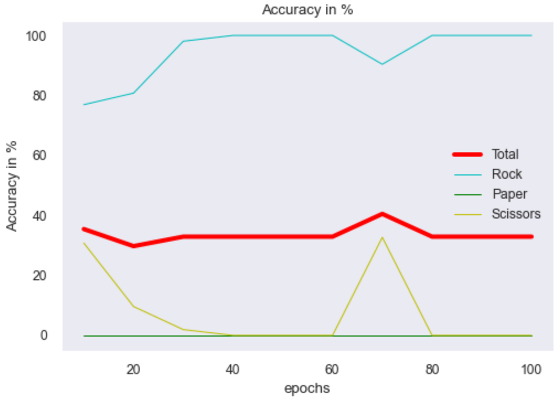

Figure 28: Testing Accuracy in detail

Comparison between the four cases

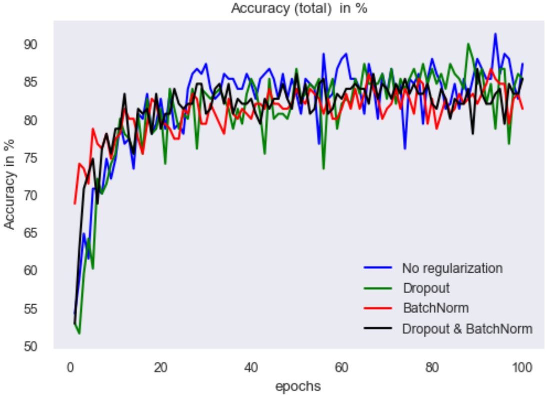

Figure 29: Comparison between all Validation Accuracies

Comparison between the four cases

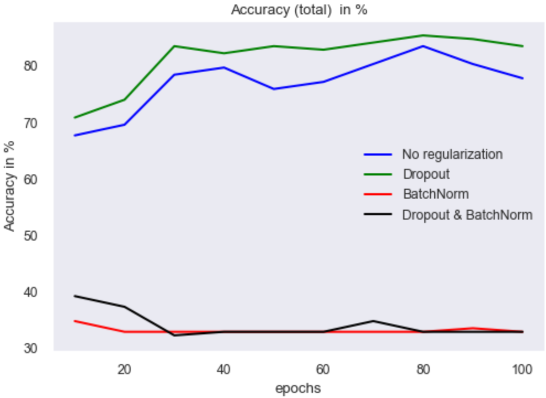

Figure 30: Comparison between all Testing Accuracies

Final results (with Dropout)

	Rock	Paper	Scissors	Total
Val Set	86,00%	90,20%	78,00%	84,77%
Test Set	78,85%	98,15%	71,15%	82,91%

Table 1: Accuracies on Val Set and Test Set for the Leaderboard

Thank you!

References i

K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014.