Advanced Natural Language Processing

Assignment 2: Intent Classification with Feedforward Neural Network

Lucia Welther, Matrikelnummer: 835106

December 10, 2025

1 Training and Testing on Same Data

Using the entire dataset for both training and testing can lead to very high accuracy results. The model probably overfits, which means that it memorizes the training data rather than learning generalizable patterns. Therefore, keeping a test set separate from the training data would result in more realistic model performance.

2 ReLU and Sigmoid Advantages/Disadvantages

2.1 ReLU

Looking at Figure 1, you see that the ReLU function outputs zero for all negative inputs and increases linearly for positive inputs, while its derivative is a step function.

Some advantages of ReLU are that it is a simple $\max(0, z)$ operation and therefore computationally very effective and inexpensive. It has a constant gradient of 1 for $z > 0$, which helps to mitigate the vanishing gradient problem compared to activation functions such as sigmoid or tanh. ReLU is also fast in comparison to other activation functions.

On the other hand, ReLU has some disadvantages. The dying ReLU problem occurs when updates get stuck outputting 0 if $z \leq 0$ during training. Once the gradient becomes 0, the neuron can stop updating and may not recover.

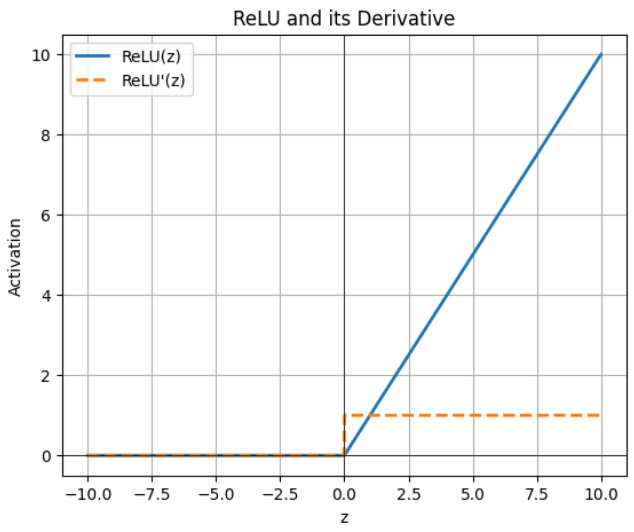


Figure 1: ReLU function and its derivative. The function outputs $\max(0, z)$, and the derivative is 1 for $z > 0$ and 0 otherwise.

2.2 Sigmoid Function

In Figure 2, you see the sigmoid function as a S-shaped curve with outputs reaching from 0 to 1, while its derivative forms a bell-shaped curve that peaks at $z = 0$ and vanishes for large $|z|$.

Unlike ReLU, sigmoid is continuous and therefore differentiable everywhere. This ensures stable gradients for small input values. Sigmoid also always outputs a value between $[0, 1]$, even when $z \leq 0$, making it well suited for modeling probabilities.

Sigmoid has the disadvantage of the vanishing gradient problem: when $|z|$ gets too large, the gradient gets very close to zero.This results in exponentially shrinking gradients during backpropagation, which slows down the learning in deeper networks.

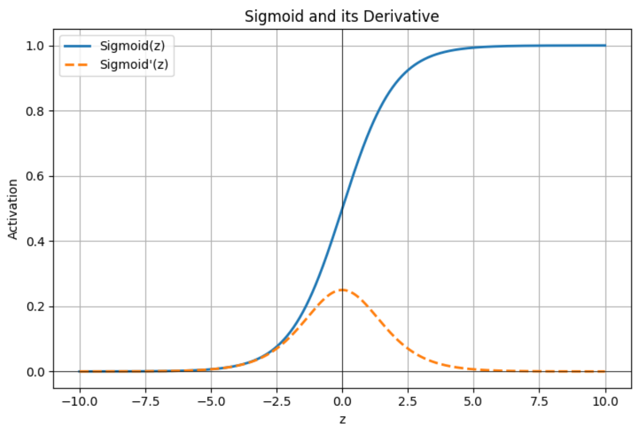
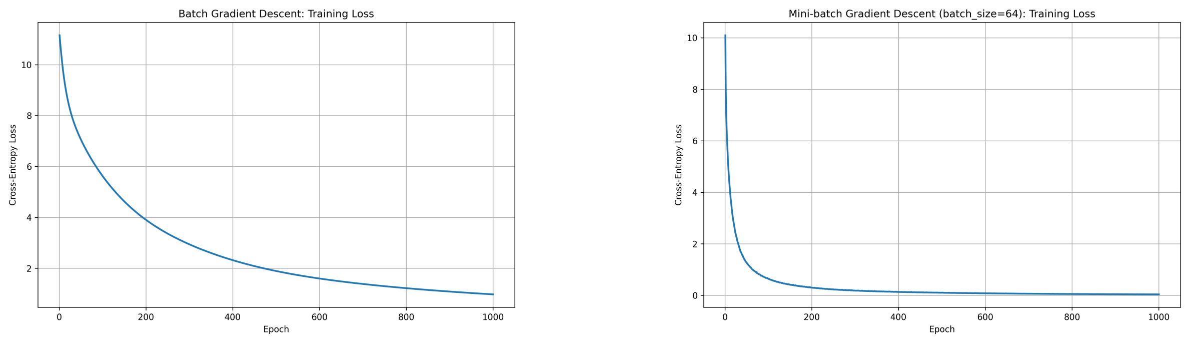


Figure 2: Sigmoid function and its derivative. The function outputs values in $(0, 1)$, and the derivative peaks at $z = 0$ and approaches zero for large $|z|$.

3 Batch and Mini-Batch Gradient Descent Loss



(a) Batch Gradient Descent

(b) Mini-batch Gradient Descent (batch_size=64)

Figure 3: Training loss over 1000 epochs for batch GD and mini-batch GD.

3.1 Observation: Batch GD

The loss decreases from more than 10 to 0.98. During the first epochs, the loss drops the most, followed by a smooth convergence. The accuracy starts at 12.2% and improves to 75.40%, which shows that the network was trained successfully.

3.2 Observation: Mini-Batch GD

With mini-batch training, the loss decreases even more dramatically than with Batch GD. From a loss of 10.0 it decreases to 0.04, where the first 100 epochs show the most drastic drop from 10.0 to 0.59, followed by a continued steady decrease throughout training. The accuracy, same as for batch GD, starts at 12.2%, representing random guessing. Mini-batch improves to 99.30%, which is significantly higher than the accuracy from batch GD. Mini-batch has an increased number of weight updates per epoch, which allows the network to learn more effectively.

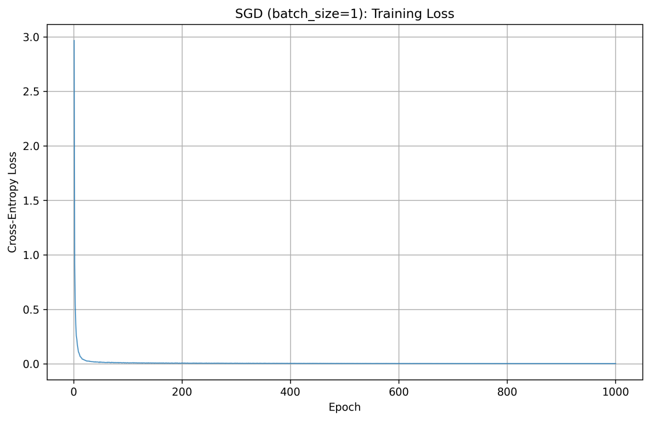


Figure 4: Training loss for SGD (batch_size=1) over 1000 epochs.

3.3 Observation: Stochastic GD

SGD shows the most drastic loss decrease within the first epochs compared to the other gradient descent methods. Starting with a loss of 3 and dropping to close to zero within the first few epochs, this indicates that the model overfits to the training set. The accuracy reaches 99.9%, which is unusually high and confirms overfitting. This is due to testing on the same data used for training.

4 GitHub Repository

GitHub Repository: `https://github.com/luzecode/aNLP_Assignment2.git`

Last commit hash: `Added Report and Jupyter Notebook for Plots, some adjustments to the Code`