

# CSC413 Assignment 1: A note about debugging

Debugging neural networks is challenging. Here are some guidelines for how to approach the coding portion of your assignment. Following these guidelines might appear time-consuming, but you will end up saving a lot of time by discovering bugs early.

## Question 1

Test each of the helper functions as you write them. We provided some test cases, so there shouldn't be anything unusual here.

## Question 3

Start by writing the `__init__` method. You may need to come back to this method when you write the `forward` method.

Then, work on the `forward` method. Before starting, I recommend commenting out the computation for `self.y` and make sure that this test code runs:

```
xs, ts = get_batch(train4grams, 0, 8, onehot=True) # get a batch of 8 data points
m = NumpyWordEmbModel()
m.forward(xs)
```

Now, write the `forward` method one line at a time. Every time you add a new line of code, rerun the test code and check that your code still runs. You might need to review the `__init__` method if some of your shapes were wrong.

Finally, write the `backward` method. When writing this method, make sure you know the shapes of the quantities that you're dealing with. You might want to write down (or print out) the shapes of `self.y`, `z_bar`, etc.

If you are uncertain about whether your `backward` method is correct, use the finite difference method (i.e. limit definition of the derivative) to check this code.

## Question 4

One classic technique for checking that a model is implemented correctly is to check if your model is capable of “overfitting” to one batch of data. In other words, take one, small minibatch of data (e.g. 5 data points), and **train the model only on that minibatch of data**. If your model is implemented correctly, then it should be able to overfit on this small dataset (i.e. reach ~100% accuracy relatively quickly, e.g. in several hundred iterations).

If you ask for debugging help on Q4(b), one of the first things we will ask is if you were successful in overfit on a batch of data.