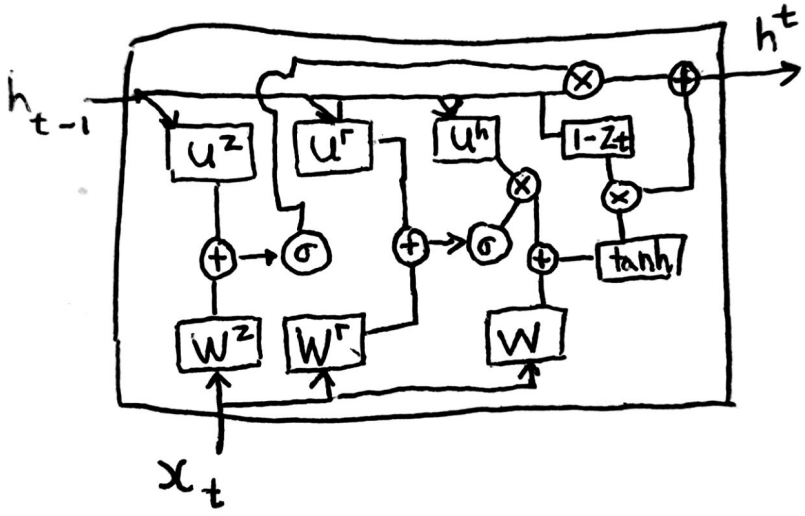
Deep Learning Homework 9

Shaun Toh 1002012

April 1, 2019

1 Task 1

1.1 GRU Cell diagram



1.2 Dimensions of weight matrices

$$W^z = (1 \times d_x)$$
$$W^r = (1 \times d_x)$$
$$W = (d_h \times d_x)$$
$$U^z = (1 \times d_h)$$
$$U^r = (1 \times d_h)$$
$$U^h = (d_h \times d_h)$$

1.3 GRU cell advantage

There appear to be slightly less matrix multiplications required in the GRU's pass through as compared to that of an LSTM. This means they are probably faster in training especially when stacked. They still retain the LSTM's ability to prevent exploding and vanishing gradients using the tanh and sigmoid functions. But an LSTM might be able to keep track of longer running states as opposed to a GRU because it has an entire vector of memory just for it, while the GRU's output is also its memory.

2 Task 2

2.1 Defense

One possible way to protect a deep learning system against adversarial attacks is to increase the complexity of the decision boundary that is drawn by the deep learning system. In order to prevent overfitting, this will also require an even larger number of data points for the model to be trained on, while greater computational power will be required to ensure that it can completed in a timely fashion. In addition to that, we will also need to train adversarial networks that run against models of the existing system to provide training samples of adversarial images. However, we should ensure the network first has sufficient capacity to account for such adversarial images, or it will very likely overfit them.

2.2 Attack Method

One way to attack this is still to use Basic Iterative Method since it is still a white box attack.

3 Task 3

3.1 minimising an objective without a lower bound

Minimising an objective without a lower bound means any adversarial images that you create that are intended to fool a neural net are allowed to stray very far from the original image. For example, by letting g_w become $-\infty$, you are essentially stating that if the adversarial net is able to make the class be predicted as $-\infty$ instead of 1, you minimise the objective function.

3.2 Capping the unbounded loss

Capping the unbounded loss creates artificial walls about the optimisation problem, preventing the losses from dropping below a certain point. The result of the wall creation is the need to optimise both the classifier's output value by adjusting the image, and also the need to reduce the amount of changes that are

placed into the image. To apply this to compound loss, we just need to remember to weight each loss and cap them accordingly if they are able to descend into negative infinity for losses.

4 Task 4

Since you have a 0.8 chance of getting any right at all in the hierarchical classifier, we then know that the likelihood of getting any class correct is 0.64, because both classifiers with 80% accuracy have to get their guesses correct to end in the correct class. Better accuracy is harder to achieve in the hierarchical classifier because to equal the 80% of the single classifier, both classifiers have to have 0.894 accuracy.