Answer to CS224n Assignment 3

Pengfei Gao, Fanny Yang, Hao Yin *

Problem 1

(a) i Stanford just offered me a big fellowship. [Stanford here may mean an ORG (Stanford university) or a PER]
Go Giant! [Giant may mean an ORG(SF Giant) or a PER (named Giant) or null (a huge creature)]

ii We may need context as an important information in inferring NER. For example, if you say George Stanford is a great man, then Stanford is a PER; if you say Stanford University, then Stanford is an ORG.

iii The proceeding word of the target, e.g., if the proceeding word is 'a', 'the' or 'at', 'in'.
The first letter of that word is capital or not.

*{pfgao, fanfyang, yinh}@stanford.edu. Each member contributes equally, and names are put in alphabetic order.

(b) i $\boldsymbol{e} \in \mathbb{R}^{1\times(2w+1)D}$, $W \in \mathbb{R}^{(2w+1)D\times H}$, $U \in \mathbb{R}^{H\times C}$.

ii For each time step, forming $\boldsymbol{e}$ takes $O((2w+1)D)$, computing $\boldsymbol{h}$ takes $O((2w+1)DH)$, then computing $\hat{\boldsymbol{y}}$ takes $O(HC)$. Therefore, for each time step, the total time for predicting label complexity is $O(H((2w+1)D+C))$. Now for a sentence of length T, the total time complexity is

$$O(HT((2w+1)D+C)).$$

(c) See my code submission as well as `window_predictions.conll`.

(d) i The best development entity-level F_1 score is 0.84, and the corresponding confusion matrix is:

```
DEBUG:Token-level confusion matrix:
go\gu          PER            ORG            LOC            MISC           O
PER            2920.00        57.00          51.00          21.00          100.00
ORG            122.00         1677.00        102.00         53.00          138.00
LOC            34.00          105.00         1875.00        29.00          51.00
MISC           40.00          72.00          27.00          1018.00        111.00
O              30.00          47.00          12.00          29.00          42641.00
```

Figure 1: Token-level confusion matrix

According to the confusion matrix, it is easy to mistakenly predict a PER as a ORG, predict ORG as LOC, and predict LOC as ORG.

ii One limitation is the prediction of the current word is not related with the prediction of the proceeding word (only related with the representation of the proceeding word). This will make mistakes like the following:

```
x : May 14 Practice at Lord 's
y*: O   O  O        O  LOC  LOC
y': O   O  O        O  LOC  O
```

Figure 2: Error1

where it predict 's to O while the true label LOC follows from the label of the proceeding word "Lord", which is LOC.

Another limitation is window size can only be limited, i.e., one can only handle windows of moderate size, not of arbitrary length. By fixing a small size of window, one might loose some important context. Example is the following:

```
x : SOCCER - MILAN 'S LENTINI MOVES TO ATALANTA .
y*: O      O ORG   O  PER     O     O  ORG      O
y': O      O ORG   O  O       O     O  O        O
```

Figure 3: Error2

In predicting LENTINI, window of size 1 can only see 'S and MOVES, but the most important context in this sentence is SOCCER (from which we will know that LENTINI is the soccer player)

Problem 2

(a) i $W_h \in \mathbb{R}^{H\times H}$, $W_x \in \mathbb{R}^{D\times H}$, $\boldsymbol{b}_1 \in \mathbb{R}^H$, $U \in \mathbb{R}^{H\times C}$, $\boldsymbol{b}_2 \in \mathbb{R}^C$ There are $H^2 + DH + H + HC + C$ parameters for RNN. Compared to $(2w+1)D + (2w+1)DH + HC$ parameters for forward NN.

ii To calculate next state cost $O(H(H+D))$.Calculate softmax cost $O(HC)$. Therefore, total cost is $O(HT(H+D) + HC)$.

(b) i Since cross-entropy loss only has a effect on the true label's negative log-likelihood. For a single sample, the cross-entropy loss is equal to the negative log-likelihood of classifying it to be true label. The loss property is decreasing with a smaller and smaller gradient as p goes up. Suppose we have 99 correctly predicted samples of which the probability of classifying true label is 0.5001; 1 incorrectly predicted sample of which the probability of classifying correct label is 0.0001. If we tune our model parameter, we could make the incorrect sample to have a 0.4999 probability for true label while the 99 correct samples to have a 0.4999 probability as well. In this case, cross-entropy loss will drop tremendously but the F1 score actually drops to zero.

ii First, precision and recall are discontinuous measures, which cannot be optimized by gradient methods. Second, the harmonic mean makes it hard to calculate gradient.

(c) code

(d) i The loss and gradient after $t = T$ is uselessly added into total objective if there is no masking. Masking multiplies zero to these useless terms to make sure they have no effect.

(e) code

(f) running code and deliver results

(g) i Limitation 1: RNN only conveys information from left to right. This means if a prediction needs information from the right side, RNN will fail. See the following prediction result.

```
 x : Estes , whose only win came at ...
y*: PER   O O     O    O   O ...
y': O     O O     O    O   O ...
```

Limitation 2: When sentences are too long, the previous information may get lost. See the following incorrect example. Elections are mentioned in the early sentence, but "Clintons" is not correctly classified. This may be caused by the information loss in RNN.

```
 x : ... that could completely turn the election around are new findings in the
         Whitewater scandal that would damage the Clintons .
y*: ...O        O  O   LOC        O       O    O     O      O   PER      O
y': ...O        O  O   MISC       O       O    O     O      O   MISC     O
```

ii Solution for 1: We can use bi-directed RNN which can convey information from both ways.
Solution for 2: We can use gated version of RNN, like GRU or LSTM.

Problem 3

(a) i $w_h = 1$, $u_h = 1$, $b_h = 0$.

ii $w_z = 1$, $u_z = 0$, $u_h = 1$, and then w_h can be any number.

(b) i We show that the parameter values that realizes this behavior do not exist.
Suppose $h^{(t-1)} = 0$, then $x^{(t)} = 1$ must lead to $h^{(t)} = 1$ while $x^{(t)} = 0$ must lead to $h^{(t)} = 0$, thus

$$\begin{aligned} u_h + b_h &> 0, \\ b_h &\leq 0, \end{aligned}$$

from which we know that $u_h > 0$. Now suppose $h^{(t-1)} = 1$, then $x^{(t)} = 1$ must lead to $h^{(t)} = 0$ while $x^{(t)} = 0$ must lead to $h^{(t)} = 1$, thus

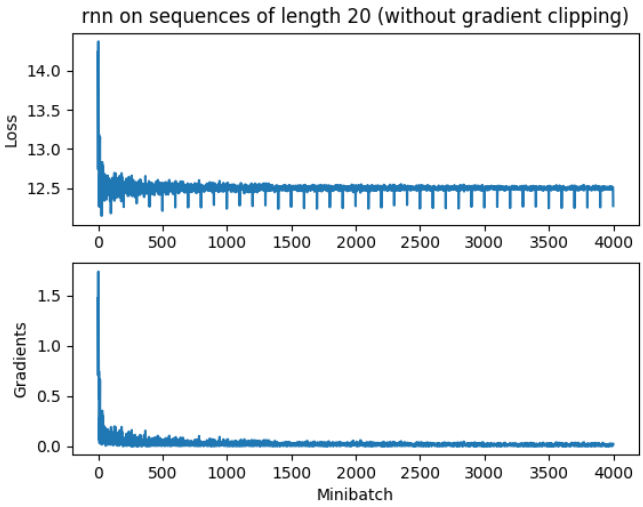
$$\begin{aligned} u_h + w_h + b_h &\leq 0, \\ w_h + b_h &> 0, \end{aligned}$$

from which we know that $u_h < 0$. Therefore, we must have $u_h > 0$ and $u_h < 0$, which is a contradiction.

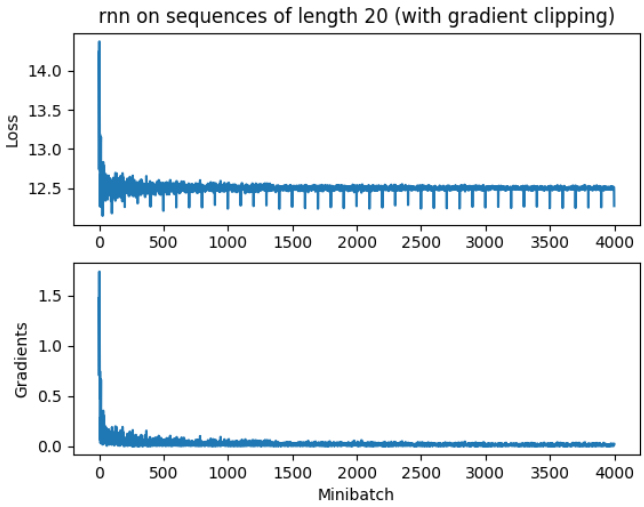
ii $b_r = 1$, $w_z = 1$, $u_z = -1$, $w_h = -1$, $u_h = 1$.

(c) See my code submission

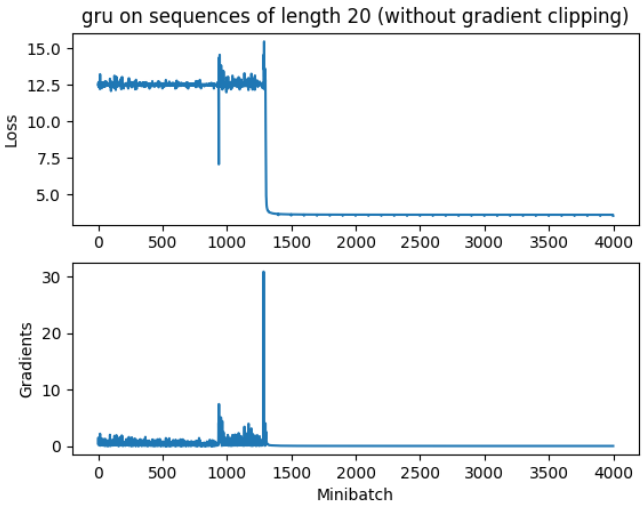
(d) Learning curves are as below:



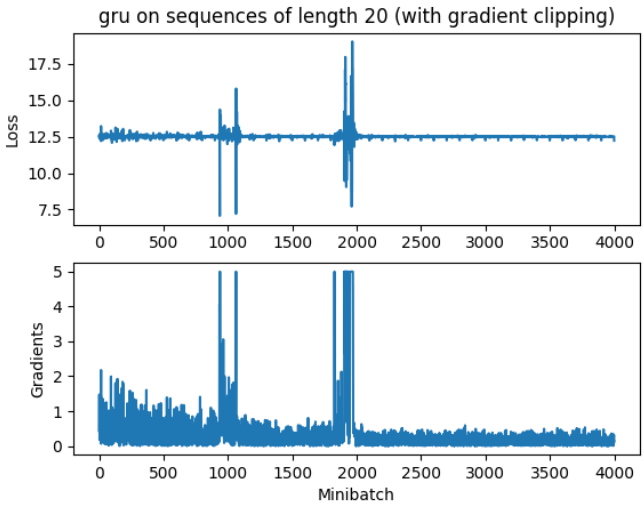
RNN without clip



RNN with clip



GRU without clip



GRU with clip

Figure 4: Learning curves

(e) i RNN experiences vanishing gradients, and neither model experience exploding gradients. The gradient clipping does not help.

ii The GRU model works better. The reason is that GRU model does not experience vanishing gradient, thus any word in the sentence can provide information in back-propagation (parameter training).

(f) See code submission as well as `window_predictions.conll`.