PyTorch

CSE 447 / 517
January 27th, 2022 (Week 4)

Logistics

- A3 is due tomorrow **(Friday, 1/28)**

- 447 Project Checkpoint 1 due tomorrow (**Friday, 1/28**)

Agenda

- Quiz 3 Solutions
- PyTorch Tutorial + Resources
- Q & A

Quiz 3 - Problem Setup

Suppose you have a neural language model (NLM) that takes in a bigram and predicts the next word in the sequence.

This (simple) language model has the following vocabulary:

0. the; 1. a; 2. mouse; 3. cat; 4. chased; 5. ate; 6. <STOP>

You want to find out what your model predicts as the next word, given the bigram "the cat".

Quiz 3 - Question 1

The embedding matrix for the NLM is:

```
[ 8  4 -4  7]
[-3  5  3  5]
[ 7  7 -6  6]
[ 9 -2 -1 -3]
[ 2  4  2  5]
[-2  1  4 -4]
[ 0 -3 -2  0]
```

where each row corresponds to a word in the ordered vocabulary given above. The model takes in the concatenation of the word embeddings for the two words in the bigram as its input. For example, the input for the bigram (a, a) would be [-3 5 3 5 -3 5 3 5]. What is the input for the bigram (the, cat)?

Quiz 3 - Question 1

The embedding matrix for the NLM is:

[8 4 -4 7]	**0. the;**
[-3 5 3 5]	1. a;
[7 7 -6 6]	2. mouse;
[9 -2 -1 -3]	**3. cat;**
[2 4 2 5]	4. chased;
[-2 1 4 -4]	5. ate;
[0 -3 -2 0]	6. <STOP>

where each row corresponds to a word in the ordered vocabulary given above. The model takes in the concatenation of the word embeddings for the two words in the bigram as its input. For example, the input for the bigram (a, a) would be [-3 5 3 5 -3 5 3 5]. What is the input for the bigram (the, cat)?

Quiz 3 - Question 1

The embedding matrix for the NLM is:

[8 4 -4 7]	**0. the;**
[-3 5 3 5]	1. a;
[7 7 -6 6]	2. mouse;
[9 -2 -1 -3]	**3. cat;**
[2 4 2 5]	4. chased;
[-2 1 4 -4]	5. ate;
[0 -3 -2 0]	6. <STOP>

… What is the input for the bigram (the, cat)?

[8 4 -4 7 9 -2 -1 -3]

Quiz 3 - Question 2

After passing the input vector through a feedforward layer, you get the hidden state vector [1 -1 0 -1 2 9 -3 6]. What would this vector be after running the hidden state through a ReLU nonlinearity?

Quiz 3 - Question 2

After passing the input vector through a feedforward layer, you get the hidden state vector [1 -1 0 -1 2 9 -3 6]. What would this vector be after running the hidden state through a ReLU nonlinearity?

Recall:

$$\mathrm{ReLU}(x) = max(0, x)$$

Quiz 3 - Question 2

After passing the input vector through a feedforward layer, you get the hidden state vector [1 -1 0 -1 2 9 -3 6]. What would this vector be after running the hidden state through a ReLU nonlinearity?

Recall:

$$\mathrm{ReLU}(x) = max(0, x)$$

Plug in our input vector, we have:

$$\mathrm{ReLU}([1, -1, 0, -1, 2, 9, -3, 6]) = [1, 0, 0, 0, 2, 9, 0, 6]$$

Quiz 3 - Question 3

Suppose you want to try multiple nonlinearities in the NLM -- what would the hidden state given in the previous question be if you used a tanh nonlinearity instead?

(You should round to the third decimal place for each element in the vector.)

Quiz 3 - Question 3

Suppose you want to try multiple nonlinearities in the NLM -- what would the hidden state given in the previous question be if you used a tanh nonlinearity instead?

(You should round to the third decimal place for each element in the vector.)

Apply tanh() element-wise:

$$\tanh([[1, -1, 0, -1, 2, 9, -3, 6]]) = [0.762, -0.762, 0, -0.762, 0.964, 0.999, -0.995, 0.999]$$

Quiz 3 - Question 4

After running the hidden state through a final feedforward layer to get a vector with the same dimension of the NLM's vocabulary, you get the following output (the logits): [-2 -0.8 0 1.3 3 2.5 2.2]. What is the probability distribution you get after applying the softmax operation to these logits?

(You should round to the third decimal place for each element in the resulting vector.)

Quiz 3 - Question 4

After running the hidden state through a final feedforward layer to get a vector with the same dimension of the NLM's vocabulary, you get the following output (the logits): [-2 -0.8 0 1.3 3 2.5 2.2]. What is the probability distribution you get after applying the softmax operation to these logits?

(You should round to the third decimal place for each element in the resulting vector.)

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Quiz 3 - Question 4

… [-2 -0.8 0 1.3 3 2.5 2.2]. What is the probability distribution you get after applying the softmax operation to these logits?

$$\begin{aligned}
&\text{softmax}([-2, -0.8, 0, 1.3, 3, 2.5, 2.2]) \\
&= [\frac{exp(-2)}{Z}, \frac{exp(-0.8)}{Z}, \frac{exp(0)}{Z}, \frac{exp(1.3)}{Z}, \frac{exp(3)}{Z}, \frac{exp(2.5)}{Z}, \frac{exp(2.2)}{Z}] \\
&= [0.003, 0.010, 0.022, 0.079, 0.432, 0.262, 0.194]
\end{aligned}$$

where

$Z = exp(-2) + exp(-0.8) + exp(0) + exp(1.3) + exp(3) + exp(2.5) + exp(2.2)$

Quiz 3 - Question 5

Based on the probability distribution obtained in the last problem and the ordered vocabulary for the NLM given earlier, what is the next word predicted by the NLM?

Quiz 3 - Question 5

Based on the probability distribution obtained in the last problem and the ordered vocabulary for the NLM given earlier, what is the next word predicted by the NLM?

0. the;	0.003
1. a;	0.010
2. mouse;	0.022
3. cat;	0.079
4. chased;	0.432
5. ate;	0.262
6. <STOP>	0.194

Quiz 3 - Question 5

Based on the probability distribution obtained in the last problem and the ordered vocabulary for the NLM given earlier, what is the next word predicted by the NLM?

0. the;	0.003
1. a;	0.010
2. mouse;	0.022
3. cat;	0.079
4. chased;	**0.432**
5. ate;	0.262
6. <STOP>	0.194

Credit:
https://github.com/ZhaofengWu/pytorch-nlp-tutorial

NumPy and PyTorch

NumPy Tutorial:
https://drive.google.com/file/d/1fSOOYiLYW6oSBCSbnDK2zgzhtFdMSKwm/view?usp=sharing

PyTorch Tutorial:
https://drive.google.com/file/d/1uzJCSaHmsibZqal31xHRNwO_7eVPiwRn/view?usp=sharing

Credit:
https://github.com/ZhaofengWu/pytorch-nlp-tutorial

NumPy and PyTorch (ext.)

Sentiment analysis Example:
https://drive.google.com/file/d/11naYkZPn15AjMYUpCZ-9ikUq4gdwhf4C/view?usp=sharing

Q & A