
CS 224N: Assignment 3

PETER888@STANFORD.EDU

SATURDAY 24TH FEBRUARY, 2018

Problem 1. A window into NER (30 points)

1.1 (a) Understanding NER (5 points, written)

1.1.1 i) Ambiguous Examples (2 points)

Answer:

1. "Walt Disney has just bought Fox for 52bn", here "Walt Disney" and "Fox" are ambiguous type. "Walt Disney" can stand for person, the founder of Walt Disney Company, or can stand for "Walt Disney" organization, the company.
2. "What does the Fox say", here "Fox" is ambiguous type. "Fox" can stand for the animal, or can stand for "Fox" organization, the company.

1.1.2 ii) Why use features (1 point)

Answer:

Words can have multiple means depending on its context. We need predict named entity labels from their context as well.

1.1.3 iii) Feature examples (2 points)

Answer:

1. Adjacent Verb. For example "Rose said", here Rose should be a person, not the flower rose, because it's adjacent verb "said"
2. Location. If "Chase" happening on highway, that's normally the verb chase, instead of the bank "Chase"

1.2 (b) Computational complexity (5 points, written)

1.2.1 i) Dimensions (2 points)

Answer:

$$\mathbf{e}^{(t)} \in \mathbb{R}^{1 \times ((2\omega+1) \times D)}, \mathbf{W} \in \mathbb{R}^{(2\omega+1) \times H}, \mathbf{U} \in \mathbb{R}^{H \times C}$$

1.2.2 ii) Complexity (3 point)

Answer:

For each step $\mathbf{e}^{(t)}$ requires $O((2\omega+1) \times D)$ computation, $\mathbf{h}^{(t)}$ requires $O((2\omega+1) \times D \times H)$ computation, and $\hat{\mathbf{y}}^{(t)}$ requires $O(H \times C)$ computation.

In total, requires $O(T \times ((2\omega+1) \times D + (2\omega+1) \times D \times H + H \times C))$ computation. Ignore the constants and $D, H \gg C$, the computation complexity of whole sentence is $O(T \times \omega \times D \times H)$

1.3 (c) Implement model(15 points, code)

Answer: See code: ~/q1_window.py.

1.4 (d) Analyze the predictions (5 points, written)

1.4.1 i) Report best F_1 score (1 point)

Answer: Get best F_1 score 0.84.

Token-level confusion matrix:					
go/gu	PER	ORG	LOC	MISC	O
PER	2935.00	38.00	90.00	16.00	70.00
ORG	135.00	1659.00	120.00	51.00	127.00
LOC	36.00	118.00	1863.00	14.00	63.00
MISC	45.00	52.00	56.00	999.00	116.00
O	36.00	43.00	20.00	23.00	42637.00

Explanation: The major errors which we can see from the confusion matrix are predict ORG as PER, O or LOC, and still errors are predict LOC to ORG, MISC as O.

1.4.2 ii) Describe limitation (4 points)

Answer:

Limitation 1. Window based prediction is limited by it's window size, human can easy understand "Sixflags Discovery Kingdom" should be an organization based on the coming word "located", but our model only have window size 2, so it can't do well on window longer than that.

Limitation 2. Context and surrounding tags influence. Humans are easily predict "Kingdom" should related with "Discovery" and should connected to each other in same entity name, and should be ORG

x : Sixflags Discovery Kingdom is located at northern California
y': PER ORG LOC O O O O LOC

Problem 2. Recurrent neural nets for NER (40 points)

2.1 (a) Computational complexity (4 points, written)

2.1.1 i) How many more (1 point)

Answer:

2.1.2 ii) Complexity (3 point)

Answer:

2.2 (b) F_1 score (2 points, written)

2.2.1 i) When CE cost and F_1 decreasing at same time (1 point)

Answer:

2.2.2 ii) Why not F_1 (1 point)

Answer:

2.3 (c) RNN cell (5 points, code)

2.4 (d) RNN model (8 points, code/written)

2.4.1 i) Loss and Gradient Update (3 points, written)

Answer:

2.4.2 ii) (5 points, code)

2.5 (e) More RNN model (12 points, code)

2.6 (e) Train RNN model (3 points, code)

Problem 3. Grooving with GRUs (30 points)

3.1 (a) Modeling latching behavior (4 points, written)

3.1.1 i) RNN cell values (1 point)

Answer:

3.1.2 ii) GRU cell values (3 points)

Answer:

3.2 (b) Modeling toggling behavior (6 points, written)

3.2.1 i) 1D RNN (3 points)

Answer:

3.2.2 ii) GRU cell values (3 points)

Answer:

3.3 (c) GRU cell (6 points, code)

3.4 (d) Learn dynamics (6 points, code)

3.5 (e) Analyze graphs (5 points, written)

Answer:

It's fine if GRU gradients are still below 5.0 without gradient clipping; just make conclusions based on what you see in your

graphs. A few things to consider:

- Are gradients increasing or decreasing for the GRU with and without gradient-clipping? Is this expected?
- Does loss improve or stay constant for the RNN vs. GRU? Does gradient-clipping affect the loss? Why might this be the case?

3.6 (f) Train GRU (3 points, code)