

Hide menu

LSTMs and Named Entity Recognition

- ✓

**Video:** Week Introduction  
1 min
- ✓

**Video:** RNNs and Vanishing Gradients  
6 min
- ✓

**Reading:** RNNs and Vanishing Gradients  
6 min
- ✓

**Reading:** (Optional) Intro to optimization in deep learning: Gradient Descent  
10 min
- ✓

**Lab:** Vanishing Gradients  
15 min
- ✓

**Video:** Introduction to LSTMs  
4 min
- ✓

**Reading:** Introduction to LSTMs  
3 min
- ✓

**Video:** LSTM Architecture  
3 min
- ✓

**Reading:** LSTM Architecture  
4 min
- ✓

**Video:** Introduction to Named Entity Recognition  
3 min
- ✓

**Reading:** Introduction to Named Entity Recognition  
2 min
- ▶

**Video:** Training NERs: Data Processing  
4 min
- ⊞

**Reading:** Training NERs: Data Processing  
5 min
- ⊞

**Reading:** Long Short-Term Memory (Deep Learning Specialization C5)  
10 min
- ▶

**Video:** Computing Accuracy  
1 min
- ⊞

**Reading:** Computing Accuracy  
2 min
- ▶

**Video:** Week Conclusion  
32 sec

Lecture Notes (Optional)

Practice Quiz

Assignment: Named Entity Recognition (NER)

< > Week 2 > Training NERs: Data Processing

< Previous Next >

# Training NERs: Data Processing

Processing data is one of the most important tasks when training AI algorithms. For NER, you have to:

- Convert words and entity classes into arrays:
- Pad with tokens: Set sequence length to a certain number and use the <PAD> token to fill empty spaces
- Create a data generator:

Once you have that, you can assign each class a number, and each word a number.

Sharon flew to Miami last Friday.

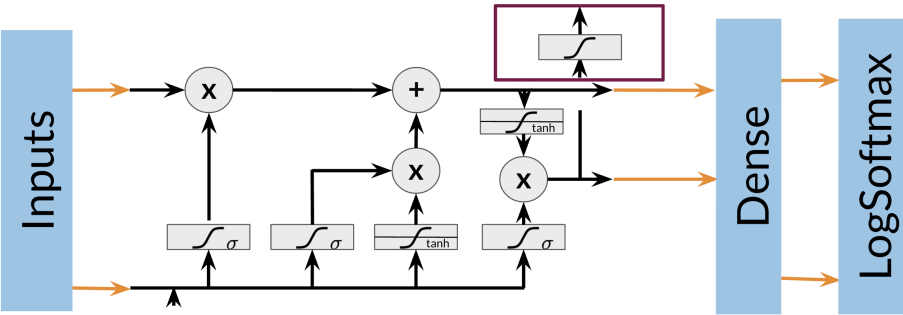
[ 4282, 853, 187, 5388, 2894, 7 ]

B-per O O B-geo O B-tim

Training an NER system:

1. Create a tensor for each input and its corresponding number
2. Put them in a batch ==> 64, 128, 256, 512 ...
3. Feed it into an LSTM unit
4. Run the output through a dense layer
5. Predict using a log softmax over K classes

Here is an example of the architecture:



Note that this is just one example of an NER system. You can have different architectures.