Natural Language Processing

Deep Learning — Units 5 & 6

Dr. Jon Krohn
jon@untapt.com

Slides available at jonkrohn.com/talks

November 3rd, 2018

Outline

Outline

Outline

Outline

Outline

Outline

Outline

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Take-Home Exercise: VGGNet

Talk through the purpose of every line in the [VGGNet notebook], including all of the following terms:

- ReLU
- cross-entropy
- epoch
- parameters
- hyperparams
- SGD
- learning rate
- batch size
- Adam
- dropout
- batchnorm
- input layer
- dense/FC layer
- convolutional
- max-pooling
- flatten
- softmax layer

Outline

Outline

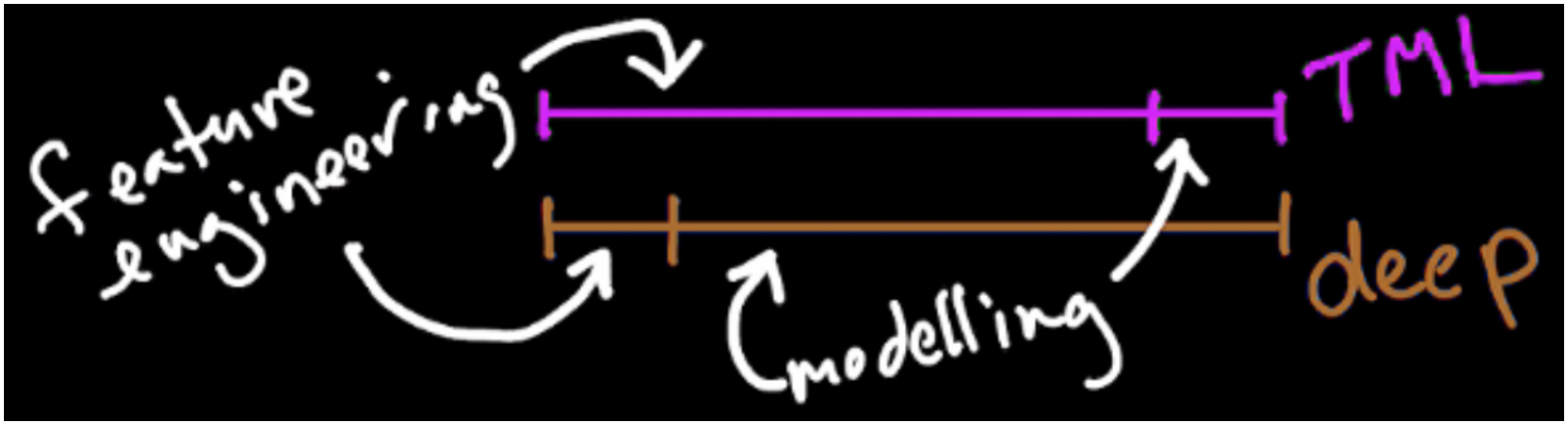
Two Core Concepts

1. Deep Learning
2. Natural Language Processing (NLP)

Two Core Concepts

1. Deep Learning
2. Natural Language Processing (NLP)

TML vs Deep Learning



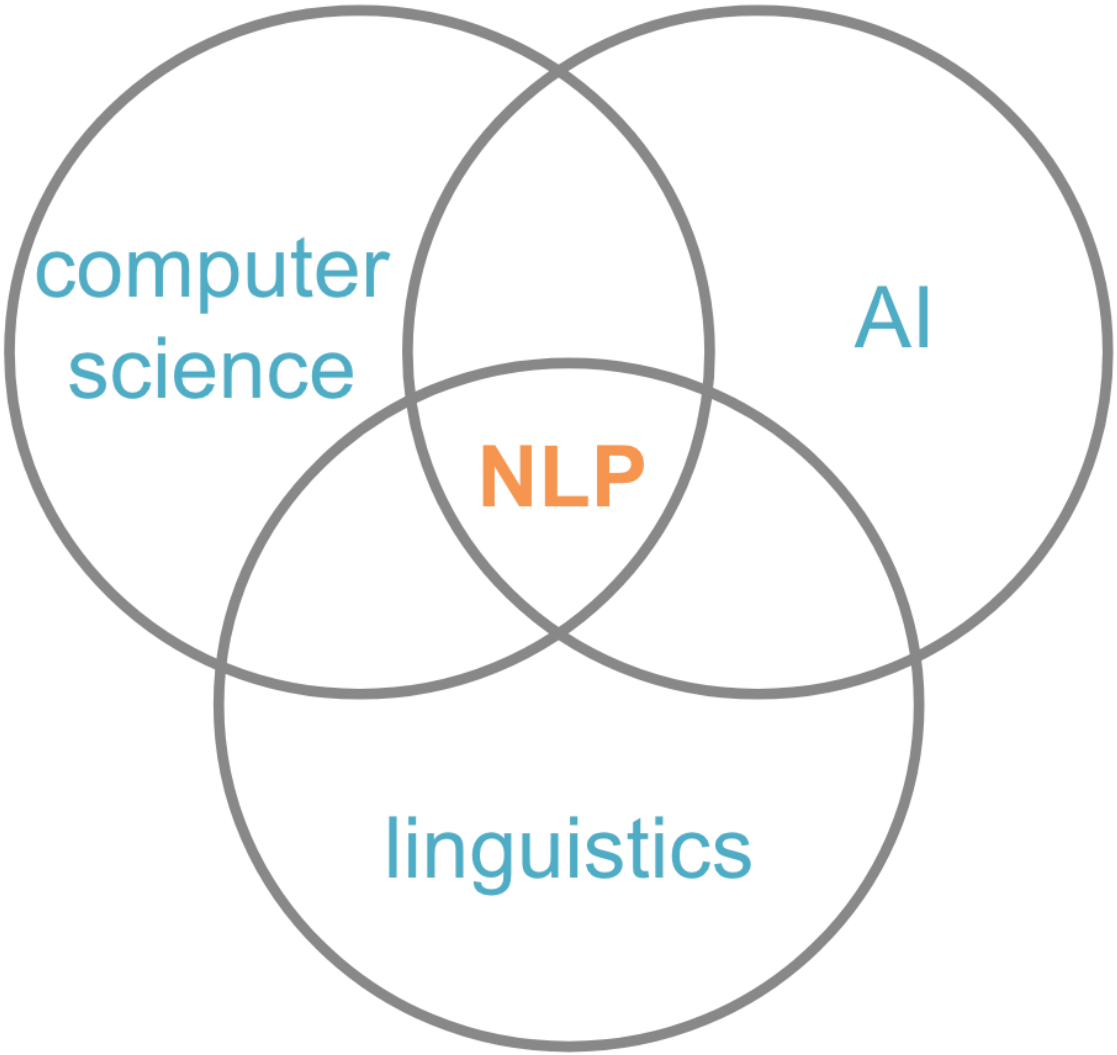
Two Core Concepts

1. Deep Learning
2. Natural Language Processing (NLP)

Two Core Concepts

1. Deep Learning
2. Natural Language Processing (NLP)

Natural Language Processing



Examples of NLP in Industry

- speech recognition (Echo, Siri, Cortana)
- search (typed into omnibox, spoken)
- classifying documents
- language translation
- chatbots

Examples of NLP in Industry

- speech recognition (Echo, Siri, Cortana)
- search (typed into omnibox, spoken)
- classifying documents
- language translation
- chatbots

Examples of NLP in Industry

- speech recognition (Echo, Siri, Cortana)
- search (typed into omnibox, spoken)
- classifying documents
- language translation
- chatbots

Examples of NLP in Industry

- speech recognition (Echo, Siri, Cortana)
- search (typed into omnibox, spoken)
- classifying documents
- language translation
- chatbots

Examples of NLP in Industry

- speech recognition (Echo, Siri, Cortana)
- search (typed into omnibox, spoken)
- classifying documents
- language translation
- chatbots

Outline

One-Hot Word Representations

word	The	cat	sat	on	the	mat.
the	1	0	0	0	1	0
cat	0	1	0	0	0	0
on	0	0	0	1	0	0
⋮						
n_{unique_words}						

Outline

Easy

- spell checking
- synonym suggestions
- keyword search

Easy

- spell checking
- synonym suggestions
- keyword search

Easy

- spell checking
- synonym suggestions
- keyword search

Intermediate

- reading level
- extracting information
- predicting next words
- classification
- sequence generation
- time-series analysis

Intermediate

- reading level
- extracting information
- predicting next words
- classification
- sequence generation
- time-series analysis

Intermediate

- reading level
- extracting information
- predicting next words
- classification
- sequence generation
- time-series analysis

Intermediate

- reading level
- extracting information
- predicting next words
- classification
- sequence generation
- time-series analysis

Intermediate

- reading level
- extracting information
- predicting next words
- classification
- sequence generation
- time-series analysis

Intermediate

- reading level
- extracting information
- predicting next words
- classification
- sequence generation
- time-series analysis

Complex

- machine translation
- question-answering
- chatbots

Complex

- machine translation
- question-answering
- chatbots

Complex

- machine translation
- question-answering
- chatbots

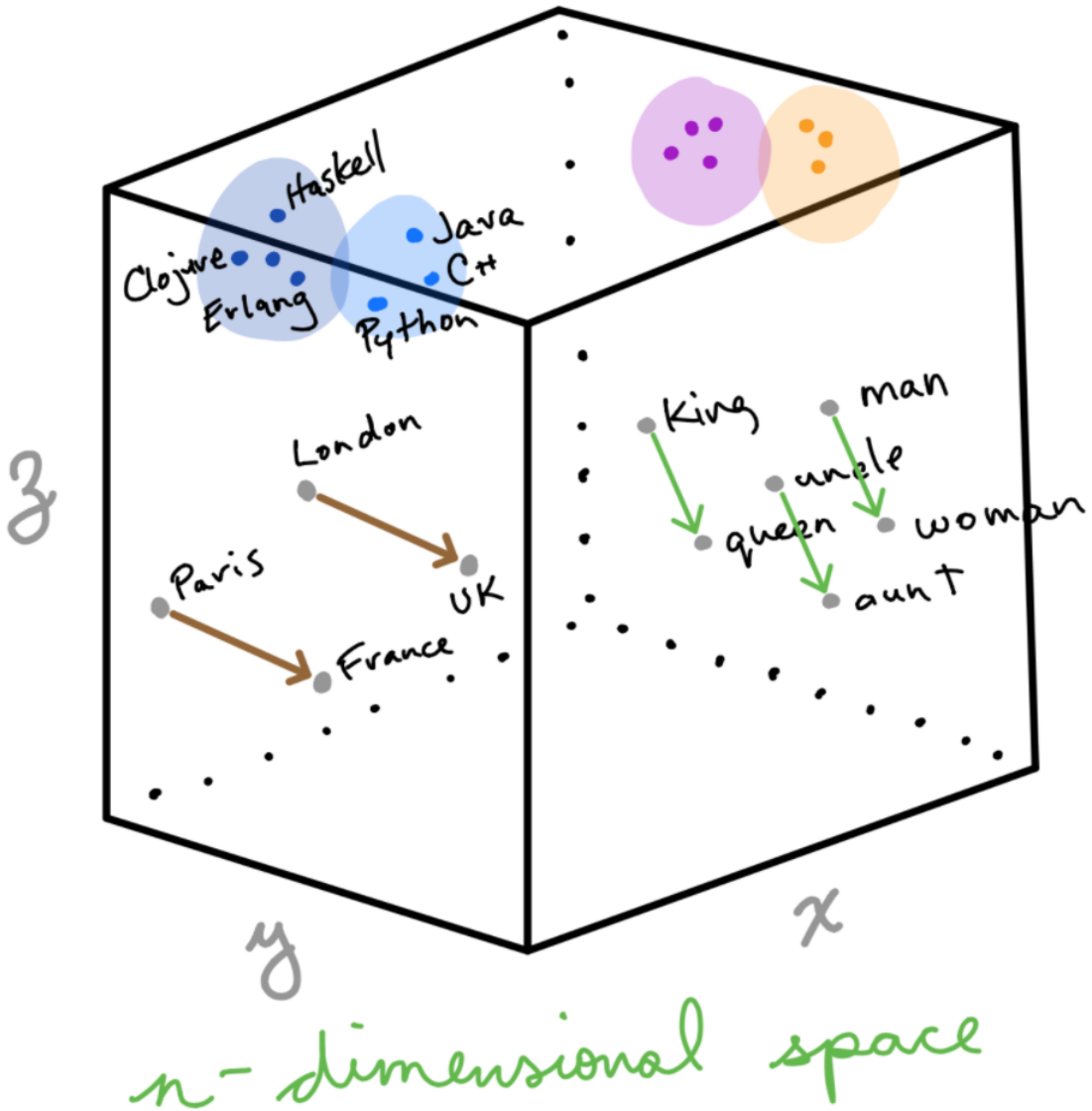
Outline

Outline

JR Firth (1957)

"You shall know a word by the company it keeps"

Vector Representations of Words



Word Vector Arithmetic

$$v_{king} - v_{man} + v_{woman} = v_{?}$$

$$v_{jeff\_bezos} - v_{amazon} + v_{facebook} = v_{?}$$

$$v_{windows} - v_{microsoft} + v_{google} = v_{?}$$

$$v_{cu} - v_{copper} + v_{gold} = v_{?}$$

[word2viz demo]

Word Representations

One-Hot	Vector-Based
lack nuance	extremely **nuanced**
handle new words poorly	seamlessly incorporate **new words**
subjective	**driven by** natural language **data**
laborious, manual taxonomies	fully-**automatic**
word similarity ignored	**word similarity** = closeness in space
unwieldy with large vocabulary	accommodate **large vocabularies**

Outline

JR Firth (1957)

"You shall know a word by the company it keeps"

Word Representations

	predicts	**relative strengths**
Skip-Gram (SG)	context given target	• small data set • rare words
CBOW	target given context	• many times faster • slightly better for frequent words

Evaluating Word Vectors

1. intrinsic
2. extrinsic

Evaluating Word Vectors

1. intrinsic
2. extrinsic

word2vec Hyperparameters

1. *n* dimensions
2. window size (SG ~10, CBOW ~5)
3. *n* iterations
4. data set size

word2vec Hyperparameters

1. *n* dimensions
2. window size (SG ~10, CBOW ~5)
3. *n* iterations
4. data set size

word2vec Hyperparameters

1. *n* dimensions
2. window size (SG ~10, CBOW ~5)
3. *n* iterations
4. data set size

word2vec Hyperparameters

1. *n* dimensions
2. window size (SG ~10, CBOW ~5)
3. *n* iterations
4. data set size

Outline

[*creating word vectors* notebook]

Outline

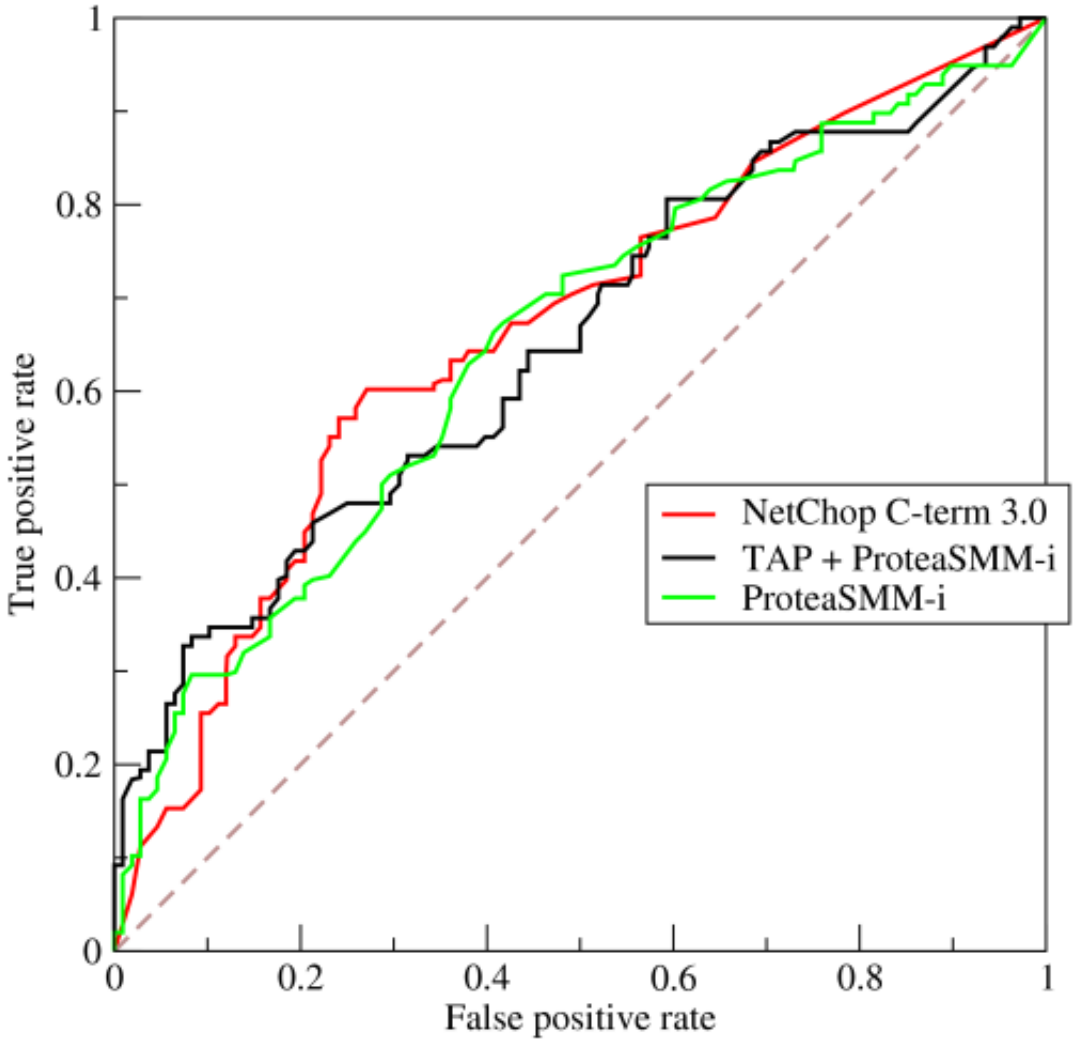
Outline

Best Practices for Preprocessing NLP Data

[*NL preprocessing best practices* notebook]

Outline

The Area Under the ROC Curve



Outline

Dense Net Classification

[*dense sentiment classifier* notebook]

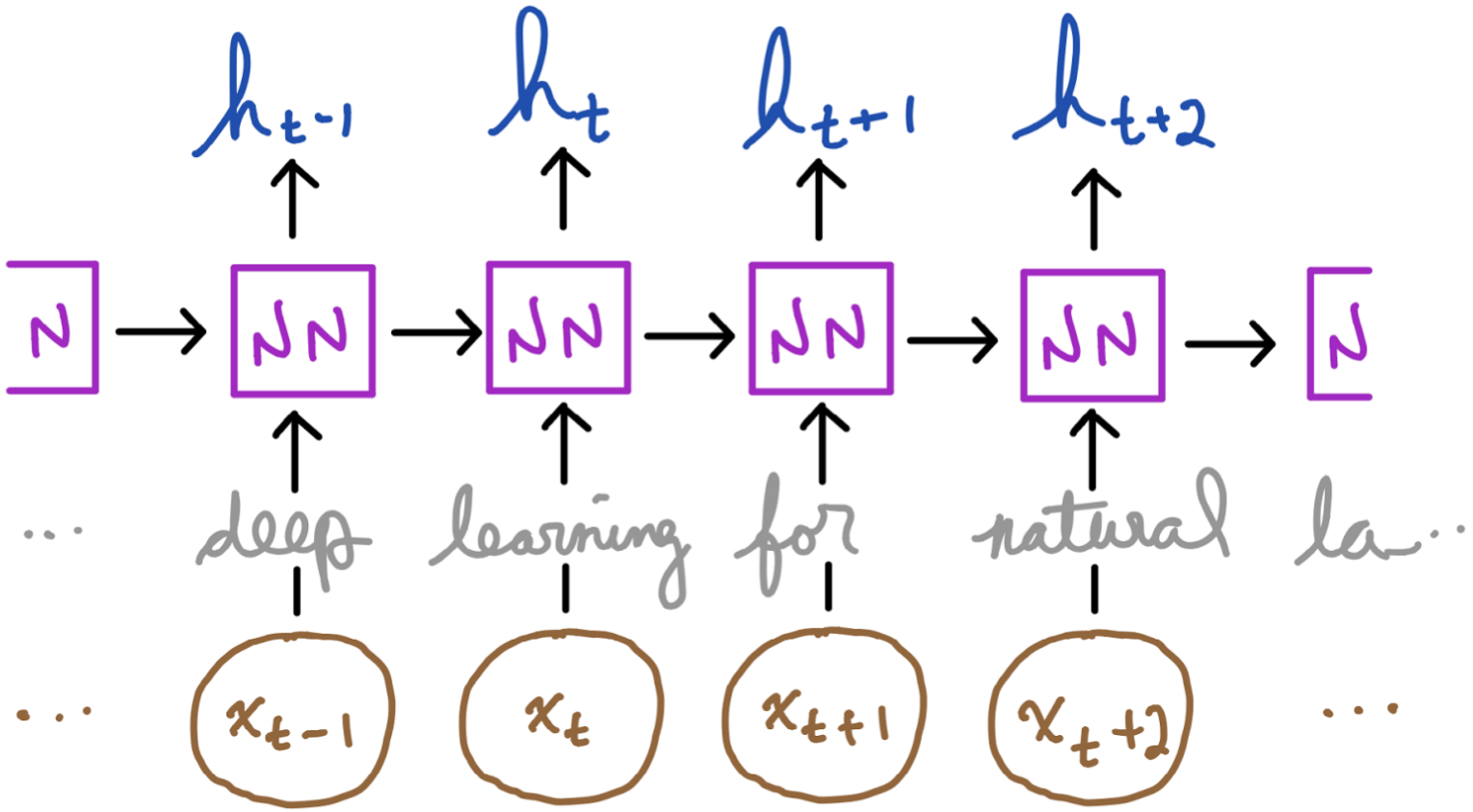
ConvNet Classification

[*convolutional sentiment classifier* notebook]

Outline

Outline

RNN Theory

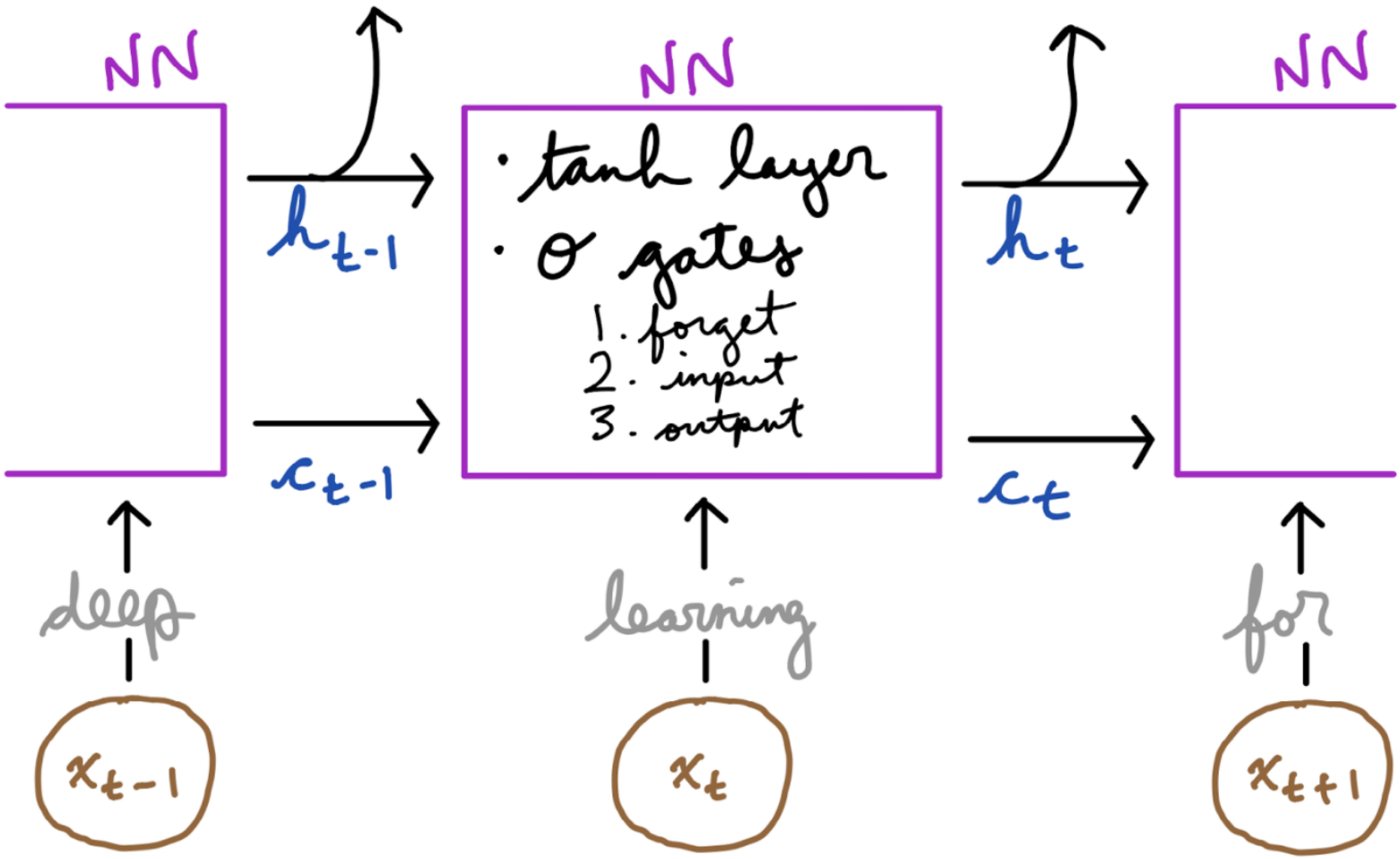


RNNs in Practice

[*rnn* notebook]

Outline

LSTM Theory



LSTMs in Practice

[*vanilla LSTM* and *GRU* notebooks]

Bi-Directional LSTMs

[*Bi-LSTM* notebook]

Stacked LSTMs

[*stacked LSTM* and *ye olde stackeroo* notebooks]

Outline

Parallel Network Architectures

[*multi-ConvNet* notebook]

Assessing

Your Deep Learning Project III

Assessing

Your Deep Learning Project III

1. split your data
   - training set (80% — for optimizing parameters)
   - validation set (10% — for hyperparameters)
   - test set (10% — don't touch yet!)
2. build and assess architecture
   - get above chance (simplifying problem, if necessary)
   - do existing performance benchmarks exist?
   - if not, use a simple architecture as benchmark
3. "teamwork makes the dream work" (?)

Assessing

Your Deep Learning Project III

1. split your data
   - training set (80% — for optimizing parameters)
   - validation set (10% — for hyperparameters)
   - test set (10% — don't touch yet!)
2. build and assess architecture
   - get above chance (simplifying problem, if necessary)
   - do existing performance benchmarks exist?
   - if not, use a simple architecture as benchmark
3. "teamwork makes the dream work" (?)

Assessing

Your Deep Learning Project III

1. split your data
   - training set (80% — for optimizing parameters)
   - validation set (10% — for hyperparameters)
   - test set (10% — don't touch yet!)
2. build and assess architecture
   - get above chance (simplifying problem, if necessary)
   - do existing performance benchmarks exist?
   - if not, use a simple architecture as benchmark
3. "teamwork makes the dream work" (?)

Assessing

Your Deep Learning Project III

1. split your data
   - training set (80% — for optimizing parameters)
   - validation set (10% — for hyperparameters)
   - test set (10% — don't touch yet!)
2. build and assess architecture
   - get above chance (simplifying problem, if necessary)
   - do existing performance benchmarks exist?
   - if not, use a simple architecture as benchmark
3. "teamwork makes the dream work" (?)

Assessing

Your Deep Learning Project III

1. split your data
   - training set (80% — for optimizing parameters)
   - validation set (10% — for hyperparameters)
   - test set (10% — don't touch yet!)
2. build and assess architecture
   - get above chance (simplifying problem, if necessary)
   - do existing performance benchmarks exist?
   - if not, use a simple architecture as benchmark
3. "teamwork makes the dream work" (?)

Assessing

Your Deep Learning Project III

1. split your data
   - training set (80% — for optimizing parameters)
   - validation set (10% — for hyperparameters)
   - test set (10% — don't touch yet!)
2. build and assess architecture
   - get above chance (simplifying problem, if necessary)
   - do existing performance benchmarks exist?
   - if not, use a simple architecture as benchmark
3. "teamwork makes the dream work" (?)

Assessing

Your Deep Learning Project III

1. split your data
   - training set (80% — for optimizing parameters)
   - validation set (10% — for hyperparameters)
   - test set (10% — don't touch yet!)
2. build and assess architecture
   - get above chance (simplifying problem, if necessary)
   - do existing performance benchmarks exist?
   - if not, use a simple architecture as benchmark
3. "teamwork makes the dream work" (?)

Assessing

Your Deep Learning Project III

1. split your data
   - training set (80% — for optimizing parameters)
   - validation set (10% — for hyperparameters)
   - test set (10% — don't touch yet!)
2. build and assess architecture
   - get above chance (simplifying problem, if necessary)
   - do existing performance benchmarks exist?
   - if not, use a simple architecture as benchmark
3. "teamwork makes the dream work" (?)

Assessing

Your Deep Learning Project III

1. split your data
   - training set (80% — for optimizing parameters)
   - validation set (10% — for hyperparameters)
   - test set (10% — don't touch yet!)
2. build and assess architecture
   - get above chance (simplifying problem, if necessary)
   - do existing performance benchmarks exist?
   - if not, use a simple architecture as benchmark
3. "teamwork makes the dream work" (?)

Up Next: TensorFlow