Jon Krohn Follow
Nov 13, 2017 · 4 min read

Deep Learning for Natural Language Processing: Tutorials with Jupyter Notebooks

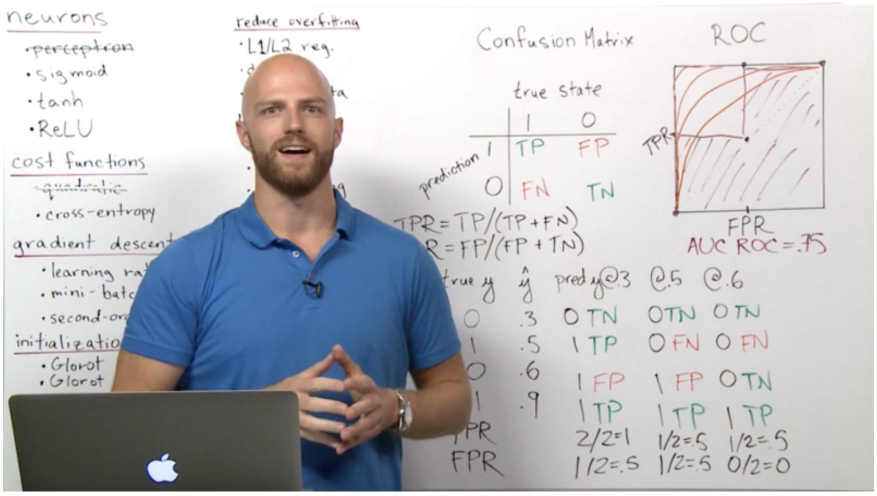
At untapt, all of our models involve Natural Language Processing (NLP) in one way or another. Our algorithms consider the natural, written language of our users' work experience and, based on real-world decisions that hiring managers have made, we can assign a probability that any given job applicant will be invited to interview for a given job opportunity.

A still from the intro to the "Deep Learning for NLP" video tutorials

With the breadth and nuance of natural language that job-seekers provide, **these are computationally complex problems. We have found deep learning approaches to be uniquely well-suited to solving them**. Deep learning algorithms:

1. trivially include millions of model parameters that are free to interact non-linearly;
2. can incorporate learning units designed specifically for use with sequential data, like natural language or financial time series data; and,
3. are typically far more efficient in production environments than traditional machine learning approaches for NLP.

To share my love of deep learning for NLP, I have created five hours of video tutorial content paired with hands-on Jupyter notebooks. **Following on from my acclaimed *Deep Learning with TensorFlow LiveLessons*, which introduced the fundamentals of artificial neural networks, my *Deep Learning for Natural Language Processing LiveLessons* similarly embrace interactivity and intuition, enabling you to rapidly develop a specialization in state-of-the-art NLP.**



A still from Lesson 3.2, where we calculate the area under the Receiver Operator Characteristic curve

These tutorials are for you if you'd like to learn how to:

- preprocess natural language data for use in machine learning applications;
- transform natural language into numerical representations (with word2vec);
- make predictions with deep learning models trained on natural language;
- apply advanced NLP approaches with Keras, the high-level TensorFlow API; or
- improve deep learning model performance by tuning hyperparameters.

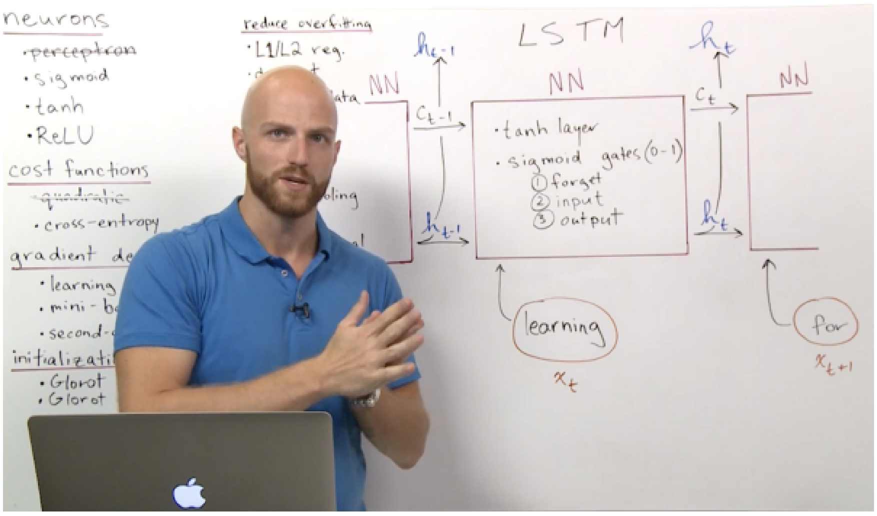
Below is a summary of the topics covered over the course of my five *Deep Learning for NLP* lessons (full breakdown detailed in my GitHub repository):

Lesson One: Introduction to Deep Learning for Natural Language Processing

- high-level overview of deep learning as it pertains to NLP specifically
- how to train deep learning models on an Nvidia GPU if you fancy quicker model-training times
- summarize the key concepts introduced in my Deep Learning with TensorFlow LiveLessons, which serve as a foundation for the material introduced in these NLP-focused LiveLessons

Lesson Two: Word Vectors

- leverage demos to enable an intuitive understanding of vector-space embeddings of words, i.e., nuanced, quantitative representations of word meaning
- discover resources for pre-trained word vectors as well as natural language data sets
- create your own vector-space embeddings with word2vec, including bespoke, interactive visualizations of the resulting word vectors



Lesson 4.3 introduces Long Short-Term Memory Units (LSTMs)

Lesson Three: Modelling Natural Language Data

- best practices for preprocessing natural language data
- calculating the ROC curve to evaluate the performance of classification models

- pair vector-space embedding with the fundamentals of deep learning introduced in my Deep Learning with TensorFlow LiveLessons to build dense and convolutional neural networks for classifying documents by their sentiment

Lesson Four: Recurrent Neural Networks

- provide an intuitive understanding of Recurrent Neural Networks (RNNs), which permit backpropagation through time over sequential data, such as natural language and financial time series data
- develop familiarity with the Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) varieties of RNNs which provide markedly more productive modelling of sequential data
- straightforwardly build LSTM and GRU deep learning architectures through the Keras high-level API

Lesson Five: Advanced Models

- discover Bidirectional-LSTMs, an especially potent LSTM variant, then leverage them in practice
- stack Bidirectional-LSTMs to enable deep learning networks to model increasingly abstract representations of language
- develop advanced data-modelling capabilities with non-sequential neural network architectures

. . .

In January, I'll be heading into the studio again to record my third set of lessons in this series; these will cover the exciting world of Deep Reinforcement Learning. To stay up-to-date, watch this space!

Jon Krohn is the Chief Data Scientist at untapt. He leads the Deep Learning Study Group from their Manhattan office and teaches his deep learning curriculum in-classroom at the NYC Data Science Academy.