Deep Learning for NLP with Pytorch

Author: Robert Guthrie

This tutorial will walk you through the key ideas of deep learning programming using Pytorch. Many of the concepts (such as the computation graph abstraction and autograd) are not unique to Pytorch and are relevant to any deep learning toolkit out there.

I am writing this tutorial to focus specifically on NLP for people who have never written code in any deep learning framework (e.g, TensorFlow, Theano, Keras, Dynet). It assumes working knowledge of core NLP problems: part-of-speech tagging, language modeling, etc. It also assumes familiarity with neural networks at the level of an intro AI class (such as one from the Russel and Norvig book). Usually, these courses cover the basic backpropagation algorithm on feed-forward neural networks, and make the point that they are chains of compositions of linearities and non-linearities. This tutorial aims to get you started writing deep learning code, given you have this prerequisite knowledge.

Note this is about *models*, not data. For all of the models, I just create a few test examples with small dimensionality so you can see how the weights change as it trains. If you have some real data you want to try, you should be able to rip out any of the models from this notebook and use them on it.

Introduction to PyTorch

Deep Learning with PyTorch

Word Embeddings: Encoding Lexical Semantics

Sequence Models and Long-Short Term Memory Networks

Advanced: Making Dynamic Decisions and the Bi-LSTM CRF

Previous   Next