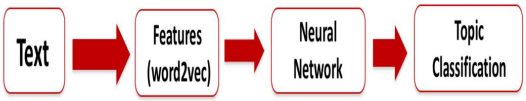
Topic Modeling using Neural Networks

Gobi Dasu ◆ Rahul Makhijani ◆ Michelle Song ◆ CS 230 Spring 2018

Introduction

- Topic detection has a wide range of applications in industry and life. For instance, for a task of company classification - given a description of SpaceX, the output category would be Aerospace.
- We decided to explore the question of which are the **best deep learning models** for topic classification given **mid-size data (~10-20K examples)**.

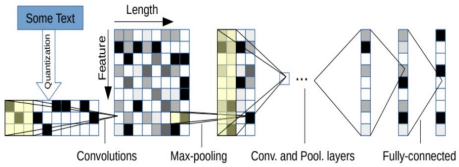


Dataset

- We are using the Reuters dataset, which has 14K examples (135 topic categories).
- Reuters data was collected from the Reuters newswire in 1987.
- We divide the data into a training set of 14,668 samples and a test sample of 6910 (for the baselines) and a dev/test sample each of 3455.
- A snippet of dataset:

HOUSTON OIL TRUST <HO> HALTS ROYALTIES HOUSTON, April 7 - Houston Oil Trust said there will be no royalty funds available for distribution to unit holders in April. It also said that based on recent independent petroleum engineers' estimates of Oct 31, 1986, there may be no amounts avialable for distribution the rest of the year. Category: *"earn"*

Architectures



A diagram of a CharLevel CNN

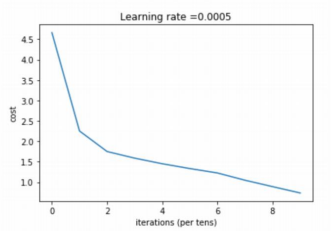
1. Character-level Convolutional Networks
We planned to use the same character level ConvNets that Zhang et. al. used. There are 9 layers deep with 6 convolutional layers and 3 fully-connected layers.

2. Basic ConvNet
The first comparison benchmark is a basic ConvNet. It is a basic two layer fully connected CNN. The non linear activation functions are RELU and softmax. We use 50 hidden states for the hidden layer, and minibatches of 100 samples.

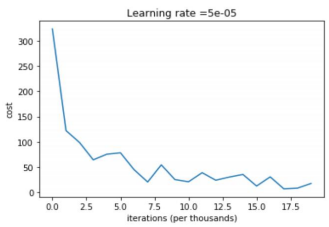
3. LSTM
We offer another comparison with a recurrent neural network model, particularly LSTM. We use a 100 state sequential model and dropout of 0.3 for the regularization. The final layer is softmax.

Results

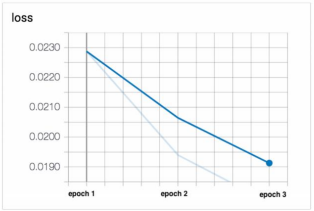
Cross-entropy Cost analysis of the Basic CNN as function of the iterations of Adam.



Cross-entropy Cost analysis of the Char Level CNN as function of the iterations of Adam.



Cross-entropy Cost analysis of the LSTM as function of the iterations of Adam.



Hyperparameter tuning

Hidden layer Size	Learning Rate	Dev Accuracy
50	0.00005	0.537482
50	0.0001	0.564399
50	0.005	0.514616
100	0.00005	0.541534
100	0.0001	0.561795
100	0.005	0.575398
150	0.00005	0.566136
150	0.0001	0.563821
150	0.005	0.575398

Table 1: Hyperparameter tuning for Basic CNN

Accuracy	CNN	Char- CNN	LSTM
Train	0.66	0.44	0.4
Test	0.53	0.50	0.66

Dropout Rate	Hidden layer Size	Epocs	Dev Accuracy
0	100	3	0.1716
0	100	5	0.348
0	200	3	0.5314
0	200	5	0.2124
0	300	3	0.0724
0	300	5	0.5876
0.4	100	3	0.1774
0.4	100	5	0.3797
0.4	200	3	0.6634
0.4	200	5	0.4654
0.4	300	3	0.6234
0.4	300	5	0.6208
0.8	100	3	0.2538
0.8	100	5	0.1922
0.8	200	3	0.1514
0.8	200	5	0.5899
0.8	300	3	0.1838
0.8	300	5	0.6394

Table 2: Hyperparameter tuning for LSTM