

Supplementary

Dataset, code and experiment results are available at:
<https://codeload.github.com/ShenDezhou/FGC/zip/master>.

0. Dataset:

Movie meta-data, social network sentiments, and acting list. The following files are in dataset folder.

1. actor_dic.utf8
This file contains names of movie actors. I use a name dictionary for a lookup table before embedding the actor names in a movie.
2. movienamename files
movienamename_training.utf8 and movienamename_test.utf8 files contain the movie names collected, I use this as a reference movie dictionary.
3. fgc_test.utf8
fgc_training.utf8 and fgc_test.utf8 files are vectors representing movie metadata, sentiments and actor name lists.
fgc_training_states.utf8 and fgc_test_states.utf8 files store movie box-office (in 10K).
fgc_test_states_gold.utf8 is a preprocessed tag file generated from fgc_test_states.utf8 movie box-office file. This file stores binary classification of movie, if it is smaller than 263.5, tagged with A, else tagged with B.

The following files are in embedding folder.

1. weibo_coreembedding.npz
This file contains Sina Weibo social network shortest path with 8380 actors. It is a representation of actor in social network. It is compressed numpy format.
2. weibo_wembedding.npz
Social network measurement of actors, it is different representation for social network. It is compressed numpy format.

1. Environment

1.1 Hardware

Experiments are performed on a server with CPU of Intel Xeon CPU E5-2620 v4 @ 2.10GHz * 2 and GPU of NVIDIA GeForce GTX 1080 Ti GPU, the server has a total of 128GB memories and 11G GPU memories.

1.2 OS

Prepare a linux distribution os, e.g. CentOS Linux release 7.2.1511.

2. Library Requirement

2.1 Programming Environment

Firstly, installation of python 3.6+, NVIDIA CUDA10.0 are required.

2.2 Python Libraries

Secondly, python libraries need to be installed, install dependencies using command:

```
pip install Keras==2.2.4 numpy==1.16.3 scikit-learn==0.20.2  
scipy==1.2.0 sklearn-crfsuite==0.3.6 tensorflow-gpu==1.15.2,
```

full list as follows:

```
Keras==2.2.4, numpy==1.16.3, scikit-learn==0.20.2, scipy==1.2.0, sklearn-  
crfsuite==0.3.6, tensorflow-gpu==1.15.2.
```

3. Experimental Guideline

In total, two parts of code are provided. python code:

include CNN-LSTM and FC-GRU-CNN algorithm files.

- 1) CLSTM.py
- 2) FC-GRU-CNN.py

3.1 CNN-LSTM

Command parameters explained as follows: Full command as follows: python3 CLSTM.py

3.2 FC-GRU-CNN

Command parameters explained as follows: python3 FC-GRU-CNN.py

4. Experimental Results

Algorithm	Accuracy
C-LSTM[1]	0.5462
FC-GRU-CNN[this paper]	0.7500

5. Parameters

Parameters used to train FC-GRU-CNN model.

Parameter	Value
batch size	100
dense features	5
max actor names	225
total features	230
social media measurement dimensions	11
social network embedding dimensions	8380
FC Regularization	1e-4
FC kernel size layer 1	150
FC kernel size layer 2	100
FC kernel size layer 3	50
FC kernel size layer 4	100
FC kernel size layer 5	150
GRU Hidden size	150
GRU Bidirectional	True
CNN filter size	150
CNN kernel size	3
Max Pooling size	2
BatchNormalization momentum	0.99
dropout rate	0.2
learning rate	0.2
epochs	100
FC activation	softmax

6. Reference

[1] Zhou C, Sun C, Liu Z, et al. A C-LSTM neural network for text classification[EB/OL]. arXiv preprint arXiv:1511.08630, 2015.