Insert Your Title Here∗

Insert Subtitle Here

FirstName Surname†  
 Department Name  
 Institution/University Name  
 City State Country  
 email@email.com

FirstName Surname  
 Department Name  
 Institution/University Name  
 City State Country  
 [email@email.com](mailto:email@email.com)

FirstName Surname  
 Department Name  
 Institution/University Name  
 City State Country  
 email@email.com

1 Application: Sentiment Analysis

Based on part of the dataset, we make an application for sentiment analysis. From the reviews and stars in the data we can infer the sentiment from the customers. By training the data in a proper way, we can build a model to detect the sentiment in a sentence of review.

The application is mainly implemented by LSTM with TorchText. When a sentence is input by user, the output will predict it as positive or negative one.

1.1 Data Preprocessing

First, we extract 100,000 lines of reviews and stars. We pick the reviews of 4 and 5 stars as positive, marked as 1, and the reviews of 1 and 2 stars as negative, marked as 0. These reviews and corresponding stars are saved in a CSV file, as the data source. Then We use TabularDataset to load this CSV file, and the field of it includes TEXT and LABEL. TEXT refers to the tokens in a review, and LABEL is 1 or 0.

1.2 Vocabulary Building

After loading the dataset, we should build a dictionary for the vocabulary in the dataset. Here glove embedding method is implemented. Embedding maps vocabulary to integers which can be represented by vector. In this way, we can represent the words with dense vector and build our dictionary. Then we save the dictionary into files for convenience.

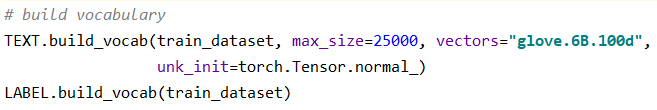


Figure 1: Building vocabulary.

1.3 LSTM Implementation

LSTM (Long short-term memory) is a special kind of RNN. It has a good performance in NLP because it takes the sequential relationship into account, which plays a significant role in analyzing tokens in sentence context. Especially, LSTM performs well when handling long sentences due to its capability of learning long term dependencies. It can also solve the gradient vanishing problem in RNN.

In the network, an embedding layer is first set to map the vocabulary size to a vector. Then a LSTM layer is set to map the vector to the hidden. In the end, a fully connected layer is applied to get the prediction result number we want. At the meanwhile, we use dropout as the regularization method. There is a bidirectional LSTM layer, which means one LSTM process the tokens in forward direction and another LSTM process in backward direction and the final hidden cells are concatenated.

In forward method, two arguments are passed, which are batch input and the number of tokens in each sequence of the batch. First we use embedding layer for the input and then we used pack\_padded\_sequence to get the new batch.

1.4 Get Model and Application

*1.4.1 Train Model.* We train LSTM model for 10 epochs with optimization strategy Adam. We can see that we get a train accuracy 98% and test accuracy 96% at last.

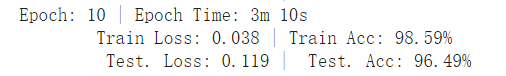


Figure 2: Training result.

*1.4.2 Application.* We use the model and the vocabulary built before to do sentiment prediction of the input sentence. Here some simple tests shown as the figure below. We can see that this application has a satisfactory effect in most cases.

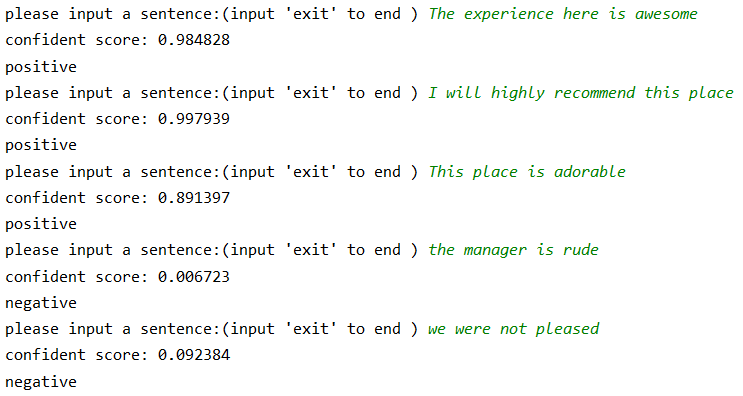


Figure 3: Some tests of the application.

Conference Name:ACM Woodstock conference

Conference Short Name:WOODSTOCK’18

Conference Location:El Paso, Texas USA

ISBN:978-1-4503-0000-0/18/06

Year:2018

Date:June

Copyright Year:2018

Copyright Statement:rightsretained

DOI:10.1145/1234567890

RRH: F. Surname et al.

Price:$15.00