Udacity Machine Learning Engineer

YongKyung Oh

Project Proposal

Domain Background: Sentiment classification, detecting if a piece of text is positive or negative, is a common NLP task that is useful for understanding feedback in product reviews, user's opinions, etc. Sentiment can be expressed in natural language in both trivial and non-trivial ways. Sometimes sentiment lexicon words such as "Good" and "Bad" are explicitly mentioned in the text and make the task easier. However, very often it could be much more subtle than that.

Problem Statement: In this project, I will build two sentiment classifiers based on naïve Bayes and neural networks. I will use the movie review data and use the percentage of correct predictions as evaluation metric.

Datsets and Inputs: dev_text.txt contains reviews, one per line, and dev_label.txt contains their labels. I use these files for training and testing models, by splitting them. heldout_text.txt contains reviews that the models will be evaluated on the unknown label data. In this project, I will just show the performance of model on the dev data.

```
Train_text.head()
```

	label	text
0	pos	i love this movie it is great film that combin...
1	neg	eyes of the werewolf is a really bad movie th...
2	pos	seeing as the vote average was pretty low and ...
3	neg	david lynchs crude and crudely drawn take on s...
4	pos	before cujothere was lucky the devil dog in on...

```
Train_text.iloc[0]['text']
```

```
'i love this movie it is great film that combines english and indian cultures with feministtype issues such as
girls wanting to play sports that were previously reserved for men it shows the struggles of both an indian pe
rson wanting to break outside her cultural barriers and women wanting to break outside the gender restrictions
found in sports especially in england at the time i feel that the cultural struggles are more emphasized than
the other issuesbr br in contrast to the other comment i do not think this movie is anything like dirty dancin
g or any other such chick flick this move is loved by many types of people men and women young and old alike'
```

```
len(Train_text[Train_text['label']=='pos']), len(Train_text[Train_text['label']=='neg'])
```

```
(1000, 1000)
```

```
#Random sampling for train data
Train_text_pos = Train_text[Train_text['label']=='pos']
Train_text_neg = Train_text[Train_text['label']=='neg']
Train_text_sampling = Train_text.copy()
for i in range(10):
    np.random.shuffle(Train_text_pos.values)
    pos_sample = Train_text_pos.sample(n=100)
    Train_text_sampling = Train_text_sampling.append(pos_sample)

    np.random.shuffle(Train_text_neg.values)
    neg_sample = Train_text_neg.sample(n=100)
    Train_text_sampling = Train_text_sampling.append(neg_sample)
np.random.shuffle(Train_text_sampling.values)
```

```
len(Train_text_sampling), len(Train_text_pos), len(Train_text_neg)
```

```
(4000, 1000, 1000)
```

Dataset is balanced, but only have 2000 samples to train and validate. Due to that, I will conduct the resampling technique to improve the data-feeding to the model.

Solution Statement: I will build a RNN model for the sentiment classification. Specifically, I will build model using pytorch and torchtext. Pretrained word embedding will be used.

Benchmark model: There are two different benchmark models. First, naïve Bayes model will be the baseline model for comparison. Second, Word-to-Vector model will be compared with pretrained embedding model (Glove6B).

Evaluation metrics: For simplicity, the accuracy on the text and label will be the metric for model comparison.

Project Design

1. Preprocess: Train data is consist with 1000 pos-labeled review and 1000 neg-labeled review. Through preprocess, I divided sentence into tokens and remove uninformed text using nltk. Also, I removed all numeric text. After preprocess, dataset only contain useful tokens. Average token per sentence is 216.31 where min is 15 and max is 1654.

2. Resampling: The number of train data is not big. So, I implemented permutation resampling to select more samples. Through this step, I randomly select the same number of samples with original dev data. So, the total number of train data is 4000 and the train/validation ratio is different in each methods.

> A resampling-based method of inference—permutation tests—is often used when distributional assumptions are questionable or unmet. Not only are these methods useful for obvious departures from parametric assumptions (e.g., normality) and small sample sizes, but they are also more robust than their parametric counterparts in the presences of outliers and missing data.[i]

3. Strategy for data usage: After data clean-up, I split the train data into train set and validation set. I selected 20% of data. So, 1600 for train set and 400 for validation set. This data is used to evaluate the model. After that, I conducted random sampling for train set. Train set is not enough for each class. So, I shuffle the each class set and select random samples as follow. Through this process, I can get 3600 train data with 1800 pos-labled samples and 1800 neg-labeled samples. I use the validation for 400 samples.

4-1. Naïve Bayes model: Most of all, I build a vocabulary dictionary from both classes. I selected top 2000 words from each classes and use the unique 2500 words as dictionary. Main model is multinomial NB which is suggested by J. Rennie et al[ii]. (2003. Also, I applied 10-fold validation to select the best model.

> Naive Bayes classifier for multinomial models: The multinomial Naive Bayes classifier is suitable for classification with discrete features (e.g., word counts for text

classification). The multinomial distribution normally requires integer feature counts. However, in practice, fractional counts such as tf-idf may also work.[iii]

4-2. RNN model: Main model is bi-directional LSTM, which is appropriate for the sequential data such as text. Model only has two hidden layer with 100 nodes and each layer represent forward and backward characteristics. Output node is 2 and I use the logit for the output value. Rather than other activation function, I used logit and cross entropy to evaluate loss.

The basic idea of bidirectional recurrent neural nets (BRNNs) is to present each training sequence forwards and backwards to two separate recurrent nets, both of which are connected to the same output layer. This means that for every point in a given sequence, the BRNN has complete, sequential information about all points before and after it. Graves et al.(2005) have found that bidirectional networks are significantly more effective than unidirectional ones, and that LSTM is much faster to train than standard RNNs and MLPs, and also slightly more accurate.[iv]

Suggested approach is discussed in the following recent references.

- ✓ Xu, G., Meng, Y., Qiu, X., Yu, Z., & Wu, X. (2019). Sentiment analysis of comment texts based on BiLSTM. *Ieee Access*, *7*, 51522-51532.[v]
- ✓ Long, F., Zhou, K., & Ou, W. (2019). Sentiment analysis of text based on bidirectional LSTM with multi-head attention. *IEEE Access*, *7*, 141960-141969.[vi]
- ✓ Minaee, S., Azimi, E., & Abdolrashidi, A. (2019). Deep-sentiment: Sentiment analysis using ensemble of cnn and bi-lstm models. *arXiv preprint arXiv:1904.04206*.[vii]

[i] LaFleur, B. J., & Greevy, R. A. (2009). Introduction to Permutation and Resampling-Based Hypothesis Tests∗. *Journal of Clinical Child & Adolescent Psychology*, *38*(2), 286-294.

[ii] Rennie, J. D., Shih, L., Teevan, J., & Karger, D. R. (2003). Tackling the poor assumptions of naive bayes text classifiers. In *Proceedings of the 20th international conference on machine learning (ICML-03)* (pp. 616-623).

[iii] Cambridge, U. P. (2009). Introduction to information retrieval.

[iv] Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural networks*, *18*(5-6), 602-610.

[v] Xu, G., Meng, Y., Qiu, X., Yu, Z., & Wu, X. (2019). Sentiment analysis of comment texts based on BiLSTM. *Ieee Access*, *7*, 51522-51532.

[vi] Long, F., Zhou, K., & Ou, W. (2019). Sentiment analysis of text based on bidirectional LSTM with multi-head attention. *IEEE Access*, *7*, 141960-141969.

[vii] Minaee, S., Azimi, E., & Abdolrashidi, A. (2019). Deep-sentiment: Sentiment analysis using ensemble of cnn and bi-lstm models. *arXiv preprint arXiv:1904.04206*.