Fake-News Classification

Sujit Rai
M.Tech, Indian Institute of Technology Ropar

1 Preprocessing

The sentence and justification columns were extracted from the dataset. Both these columns were then concatenated with "[sep]" token as the separator present between them. The tokens were extracted from the sentences and justification using nltk tokenizer. All the individual words were then converted into 50-Dimensional Glove [Pennington *et al.*, 2014] word vectors, which were obtained from the glove model trained on 27B twitter dataset. Each of the word vectors were further concatenated with 10-dimensional emotion lexicon obtained from emolex, resulting in 60-dimenstional word vector. The intuition being, variation in emotions in sentence and justification might be useful in classification task. Stop words were found to be helpful experimentally therefore stop words weren't eliminated during initial preprocessing. The meta data containing the number of counts were extracted and padded with zeros to form a 60-dimensional word vector which was further appended at end of each sequence. The maximum sequence length present in the training dataset was close to 1300 whereas most of the samples had a sequence length lying in the range 100-150. Therefore, the max sequence length was chosen to be 200 thus resulting in 10143 samples for training, 1248 samples for validation and 1274 samples for testing.

2 LSTM

Initially, the experiments were performed using LSTM. Here, the LSTM architecture consisted of 4 lstm cell with 128-dimensional hidden layer. The activation function used was leaky relu at all layers except the last layer. The loss function used was sigmoid cross entropy loss, which was minimized using adam optimizer. The choice of hyper-parameters and the results are represented in Table 1

Task	Binary	Binary	Binary	Multi
Features	Glove (S+J) (no stop words)	Glove (S+J) (stop words)	Glove (S+J+M) (stop words)	Glove (S+J+M) (stop words)
Learning Rate	0.0005	0.0001	0.0001	0.0001
Batch Size	32	32	32	32
Dropout Keep Probability	0.75	0.75	0.75	0.75
Epoch	10	10	10	10
Val Accuracy	0.53	0.55	0.56	0.22
Test Accuracy	0.56	0.57	0.56	0.19

Table 1: Here S,J and M represents statement, justification and metadata. "no stop words" indicates that the stop words were eliminated during the preprocessing step.

3 Self-Attention

Since, LSTMs are sequential model, All the word vectors have to be passed in the model sequentially and also the output of LSTM is not capable of incorporating all the long-term task specific information. Therefore, Self-attention mechanism was experimented, In self-attention mechanism the sequences are processed parallely and the processing of the sequences are bi-directional. The model experimented with was inspired from transformer architecture [Vaswani *et al.*, 2017]. Since, cosine distance between 2 glove word vectors was a measure of similarity between the 2 corresponding words therefore, Self-attention mechanism was a suitable choice in terms of efficiency. The choice of hyper-parameters with the results are represented in Table 2

Task	Binary	Multi
Features	Glove (S+J+M) (stop words)	Glove (S+J+M) (stop words)
Learning Rate	0.0001	0.0001
Batch Size	32	32
Dropout Keep Probability	0.8	0.8
Epoch	10	10
Val Accuracy	0.57	0.25
Test Accuracy	0.60	0.24

Table 2: Here S,J and M represents statement, justification and metadata. "no stop words" indicates that the stop words were eliminated during the preprocessing step.

4 BERT

BERT [Devlin *et al.*, 2018] is an extention of transformer architecture and has achieved state of the results in various language translation and Classification tasks. BERT architectures are initially trained in an unsupervised manner on datasets such as wikipedia and then its finetuned on respective classification tasks. During the initial unsupervised training, bert architectures are trained to predict whether the 2 pairs of input sentences related to each other. Through this training it learns the relation between 2 given sentences. Therefore, BERT architecture was considered as the appropriate architecture for finetuning on Fake-news classification task. The implementation of BERT was taken from the official code [Google Colaboratory,] from google. The choice of hyper-parameters and the results are represented in Table 3

Task	Binary	Multi
Features	BERT (S+J) (stop words)	BERT (S+J) (stop words)
Learning Rate	2e-5	2e-5
Batch Size	32	32
Epoch	3	3
Test Accuracy	0.265	0.625

Table 3: Here S,J and M represents statement, justification and metadata. "no stop words" indicates that the stop words were eliminated during the preprocessing step.

References

[Devlin *et al.*, 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[Google Colaboratory,] Google Colaboratory. Predicting Movie Reviews with BERT. `https://colab.research.google.com/github/google-research/bert/blob/master/predicting_movie_reviews_with_bert_on_tf_hub.ipynb`.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.