Deep Learning in Opinion Detection

Guide: Ankan Mullick. Project Number: 22. Group Number: 10
Members: Anurag Bharadwaj, Arijit Panigrahy, Projjal Chanda, Siddhant Singh, Sourav Pal.

Introduction

Online news websites desire to develop engagement of users around the articles they publish. An important aspect of user engagement is commenting on the article. In order to facilitate meaningful conversation, an option might be to identify and highlight certain opinionated sentences in the article which might act as seed for discussion.

Previous Work

Paper referred: https://aclweb.org/anthology/W/W16/W16-4305.pdf

In this paper the authors extract an extensive set of features at the sentence level to classify a sentence as an opinion / fact using a binary Naive Bayes (NB) classifier.
Some features are:
Count of strong/weak polar words in a sentence, presence of pronouns, opinionated words like 'should', 'always', if' etc, dependencies for each sentence.
Proposed a graphical method (OP-D) to diversify the opinions extracted from an article.

Method

Using LSTMs in place of the Naive Bayes classifier to model longer dependencies.
Using a mix of features from the Stanford Parser output to be passed as the input to the LSTM cells. POS tags, binary features per word like- whether it is a pronoun or not, positivity/negativity score for the words.
Also used glove word vectors, dependency parse features embeddings as features for the LSTMs.

Training

We trained our models on 2 datasets:
1. The standard Multi-Perspective Question Answering (MPQA) dataset (contains 535 documents)
2. 120 news articles crawled from Yahoo news, each pertaining to some topic.

Trained models of RNN, LSTM, and CNN with Glove embedding word vectors and dependency based word embeddings by 5 fold cross validation.

The following shows the accuracy, precision, and recall for LSTM trained on GloVE vectors.

Activation	Batch Size	Dropout	Acc.	Precision	Recall
Sigmoid	100	0.2	0.79	0.78	0.86
Sigmoid	32	0.2	0.80	0.80	0.84
Sigmoid	32	0.1	0.79	0.79	0.83
Sigmoid	32	0.3	0.80	0.77	0.89
Sigmoid	32	0.4	0.79	0.78	0.86
Sigmoid	32	0.5	0.79	0.76	0.89

The following are the precision and recall values for top 3 and 5 sentences when articles are divided into different buckets based on opinion fraction.

	p,r @3 MPQA	**p,r @5 MPQA**	**p,r @3 Yahoo**	**p,r @5 Yahoo**
bucket1	0.36, 0.14	0.48, 0.18	0.38, 0.10	0.43, 0.16
	97/455		59/118	
bucket2	0.62, 0.20	0.63, 0.30	0.58, 0.13	0.58, 0.20
	163/455		27/118	
bucket3	0.87, 0.27	**0.86, 0.40**	0.75, 0.23	**0.76, 0.37**
	195/455		32/118	
fullData	0.67, 0.21	**0.70, 0.32**	0.53, 0.14	**0.55, 0.22**
		455/535		**118/120**

buckets for MPQA: (0,0.3], (0.3, 0.65], (0.65,1]
buckets for Yahoo: (0, 0.5], (0,5, 0.65], (0.65, 1]

Testing on Guardian dataset.

Crawled 50 articles each for 5 categories from Guardian, namely Sports, Tech, Finance, World, and Politics. For each category we took 10 articles each from 5 events like US Presidential election for Politics, and FIFA World Cup for sports.

We ran the LSTM with dependency features model (which we trained on MPQA dataset) on these 250 articles and found out opinion score for each of the sentences.
Below are the opinion fractions:

Category	Average Opinion Fraction
World	0.51
Politics	0.63
Finance	0.38
Technology	0.32
Sports	0.5

We took the top5 sentences from each article based on this score and **manually annotated** them as Opinion/Fact. We then found the precision for the top5 sentences (p@5) for different thresholds of the score being taken as opinion- 0.5, 0.6, 0.7, 0.8 and 0.9. The results have been tabulated here:

Section	P@5-50	P@5-60	P@5-70	P@5-80	P@5-90
Tech	0.4653061224	0.4612244898	0.4408163265	0.4	0.2612244898
World	0.71	0.705	0.695	0.685	0.6
Politics	0.788	0.788	0.788	0.775	0.682
Finance	0.698	0.689	0.675	0.64	0.52
Sports	0.876	0.868	0.851	0.804	0.617

We divided the articles into 3 buckets according to the number of opinionated sentences as a result of the manual annotation-

1. Having 0 or 1 opinionated sentences
2. Having 2 or 3 opinionated sentences
3. Having 4 or 5 opinionated sentences

Section	Num. articles in Bucket1	Num. articles in Bucket2	Num. articles in Bucket3
Tech	15/39	14/39	10/39
World	0/40	21/40	19/40
Politics	1/49	12/49	36/49
Finance	5/46	19/46	22/46
Sports	1/47	6/47	40/47

We then found out the precision @ 5 values for each of the different buckets, the results are tabulated here:

Section	Bucket1 P@5-50	Bucket2 P@5-50	Bucket3 P@5-50
Tech	0.12	0.517	0.86
World	0	0.514	0.926
Politics	0.2	0.533	0.889
Finance	0.15	0.568	0.909
Sports	0.2	0.467	0.955

Sentiment Scores

The sentiment scores have been calculated at two levels. The one at the sentence level , which are there in the folder Top_5_Senti, each csv has the articles top 5 sentences and the columns contain the following. Opinion score, Sentence, Opinion/Fact, Negative Sentiment Scores, Neutral Sentiment Scores, Positive Sentiment Scores, Compound Sentiment Scores The second level of sentiment analysis is at the sentence level where we report the average sentiment across all the sentences of an article. These are present in the link - https://docs.google.com/spreadsheets/d/1r_OaW40K4TR4xkIisbvaUCxNYLfnOAc-3bmVmfkV4Gc/edit#gid=1009567086

Future Work

We have built an opinion fact classifier using the models and calculate opinion fraction and opinionatedness for event specific and time bound datasets. With the sentiment scores calculated as above, we plan to see how sentiment of a person varies according to the change in opinion fraction of the articles. We also look to automatically find out a user's sentiment over time.