Stance Detection on Online Twitter Data

Group 35

- Kushagr 2017062
- Preyansh 2017176

Problem Statement

1. Stance detection means to automatically determining from text whether the author is in favor of the given target, against the given target, or whether neither inference is likely.

2. The aim of the task is to test automatic systems in determining whether they can deduce the stance of the tweeter.

Data Set

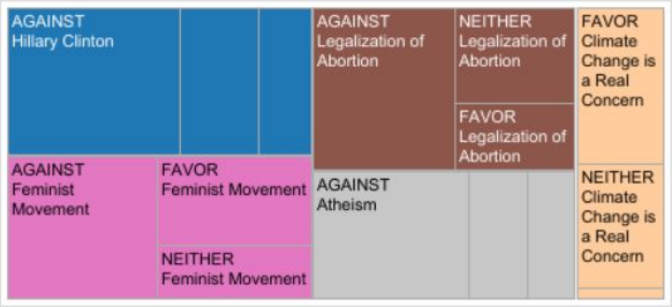
The Tweets that pertain to the following five topics :

- Atheism
- Climate change is a concern
- Feminist movement
- Hillary Clinton
- Legalization of abortion

Target	# total	# train	% of instances in Train			# test	% of instances in Test		
			favor	against	neither		favor	against	neither
Atheism	733	513	17.9	59.3	22.8	220	14.5	72.7	12.7
Climate Change	564	395	53.7	3.8	42.5	169	72.8	6.5	20.7
Feminist Movement	949	664	31.6	49.4	19.0	285	20.4	64.2	15.4
Hillary Clinton	983	689	17.1	57.0	25.8	295	15.3	58.3	26.4
Legal. Abortion	933	653	18.5	54.4	27.1	280	16.4	67.5	16.1
Total	4163	2914	25.8	47.9	26.3	1249	23.1	51.8	25.1

Targets

Stance by Target



Initial Approach- Feature Extraction

- We have extracted the tokens from each tweet by lowering all the words, removing the punctuation and stop words. For the Hashtags, we have removed the ’#’ symbol and then it is considered as a normal word.

- We observe that the feature vector length is quite long and so we tagged all the tokens using the nltk tagger, and among all tokens the tokens which are ’NN’, ’NNS’, ’NNP’, ’VB’, ’VBD’, ’VBJ’, ’VBN’, ’VBP’, ’VBZ’, ’JJ’, ’JJR’, ’JJS’ i.e. multiple forms of verbs, adjectives and nouns, are the only tokens which are there for a tweet. We add these tokens to our vocabulary.

- After building the vocabulary, we make feature vector for each of the tweets of length vocabulary size. The feature vector has corresponding indices for each of the words in the vocabulary.

Numbers of Features Used		Tokens	Types
2628	Hillary	11477 Hillary	3074 Hillary
2377	Abortion	11593 Abortion	2840 Abortion
2157	Atheism	9503 Atheism	2561 Atheism
2008	Climate	6563 Climate	2376 Climate
2643	feminism	12039 feminism	3110 feminism

Official Metric

$$F_{avg} = \frac{F_{favor} + F_{against}}{2}$$

$$\begin{aligned} F_{favor} &= \frac{2P_{favor}R_{favor}}{P_{favor}+R_{favor}} \\ F_{against} &= \frac{2P_{against}R_{against}}{P_{against}+R_{against}} \end{aligned}$$

Models on the extracted features

1. SVM Model
2. Random Forest Model
3. Logistic Regression Model
4. Gradient Boosting

SVM Model

Official Metric F1 Score : 0.615

Hillary	0.6372881355932203
Abortion	0.6
Atheism	0.6909090909090909
Climate	0.6686390532544378
feminism	0.5087719298245614

Gradient Boosting

Official Metric F1 Score: 0.67

Hillary	0.6338983050847458
Abortion	0.65
Atheism	0.7090909090909091
Climate	0.7218934911242604
feminism	0.6

Random Forest Model

Official Metric F1 Score: 0.66

Hillary	0.6135593220338983
Abortion	0.6321428571428571
Atheism	0.7045454545454546
Climate	0.6863905325443787
feminism	0.5894736842105263

Logistic Regression Model

Official Metric F1 Score 0.67

Hillary	0.6813559322033899
Abortion	0.65
Atheism	0.7318181818181818
Climate	0.6982248520710059
feminism	0.5789473684210527

Advanced Models

1. MLP + trainable embedding
2. MLP + glove embeddings
3. CNN + trainable embedding
4. CNN + glove

Embedding Layer

1. Embedding layer of input dim = Vocab size

2. Input seq length = 100 (maximum length, if less than this then padding is done) is added.

3. Output Layer Dimension = Dimension of each word embedding (100)

Architecture

1. MLP+TRAINING EMBEDDINGS - One Embedding Layer, Flattens its output, Pass through dense fully connected layer of 16 hidden neurons with sigmoid activation, Output Layer has 3 neurons applied with softmax.

2. MLP + GLOVE EMBEDDINGS - Pre trained Glove Vectors

Architecture

3. CNN + TRAINING EMBEDDINGS - One Embedding Layer, The second layer is a convolution layer with relu activation function, Flattens its output, Pass through dense fully connected layer of 16 hidden neurons with sigmoid activation, Output Layer has 3 neurons applied with softmax.

4. CNN + GLOVE EMBEDDINGS - Pre trained glove vectors

Results

1. Pre trained glove vectors tend to perform better than the training word vectors on our own.

2. Glove vectors capture the global context while forming the word embeddings.

MLP	0.62
MLP + Glove	0.64
CNN	0.63
CNN + Glove	0.65

Conclusion

1. The Advance Models have the official metric value of 0.65 whereas the feature extraction using POS tagger and then passing them through ML models (Multiclass Logistic OVR and Random Forest) have official metric value of 0.67.
2. This shows that the latter generalize better and feature extraction using POS tagger is quite discriminative.
3. The Advance Models though highly efficient on the train set tend to overfit being high capacity models.
4. Our first approach nears the winner of semeval 2016 task 6A (0.6782 n the official metric) and the benchmark is 0.69 set by authors.

Thank You