

Finding Sarcasm in Reddit Postings: A Deep Learning Approach

Nick Guo, Ruchir Shah
{nickguo, ruchirfs}@stanford.edu

Predicting

Abstract

We use the recently published Self-Annotated Reddit Corpus (SARC) with a deep learning neural network to classify sarcastic statements and evaluate against baseline Bag of N-grams and Naive Bayes methods.

Introduction

Developing truly conversational speech agents - who can understand all the unique intricacies of the human language - remains one of the largest pending NLP problems of our time. Humans regularly use sarcasm as an important part of day-to-day conversation when venting, arguing, or even engaging in humorous banter with friends. For an agent to truly be conversational, detection of sarcasm is a must.

Related Work

The vast majority of studies implemented so far have used Twitter data as a medium, with distant supervision available in the form of tweets with the hashtag `*#sarcasm`.

Sarcasm = <Speaker, Hearer, Context, Utterance, Literal Words, Intended Words>

Data

Earlier this year, an enormous corpus of textual sarcasm was published containing 1.3 million sarcastic statements made by Reddit users (Khodak 2017). In particular, this data set takes advantage of a Reddit norm to include the symbol `/s` after a sarcastic statement. The corpus has information on the sarcastic or non-sarcastic post as well as information on all other posts in the thread. We worked with ~385k comments from `r/politics` divided into a training set of 305k, and test set of 80k comments. In both sets, about 3% of the comments are sarcastic.

Features

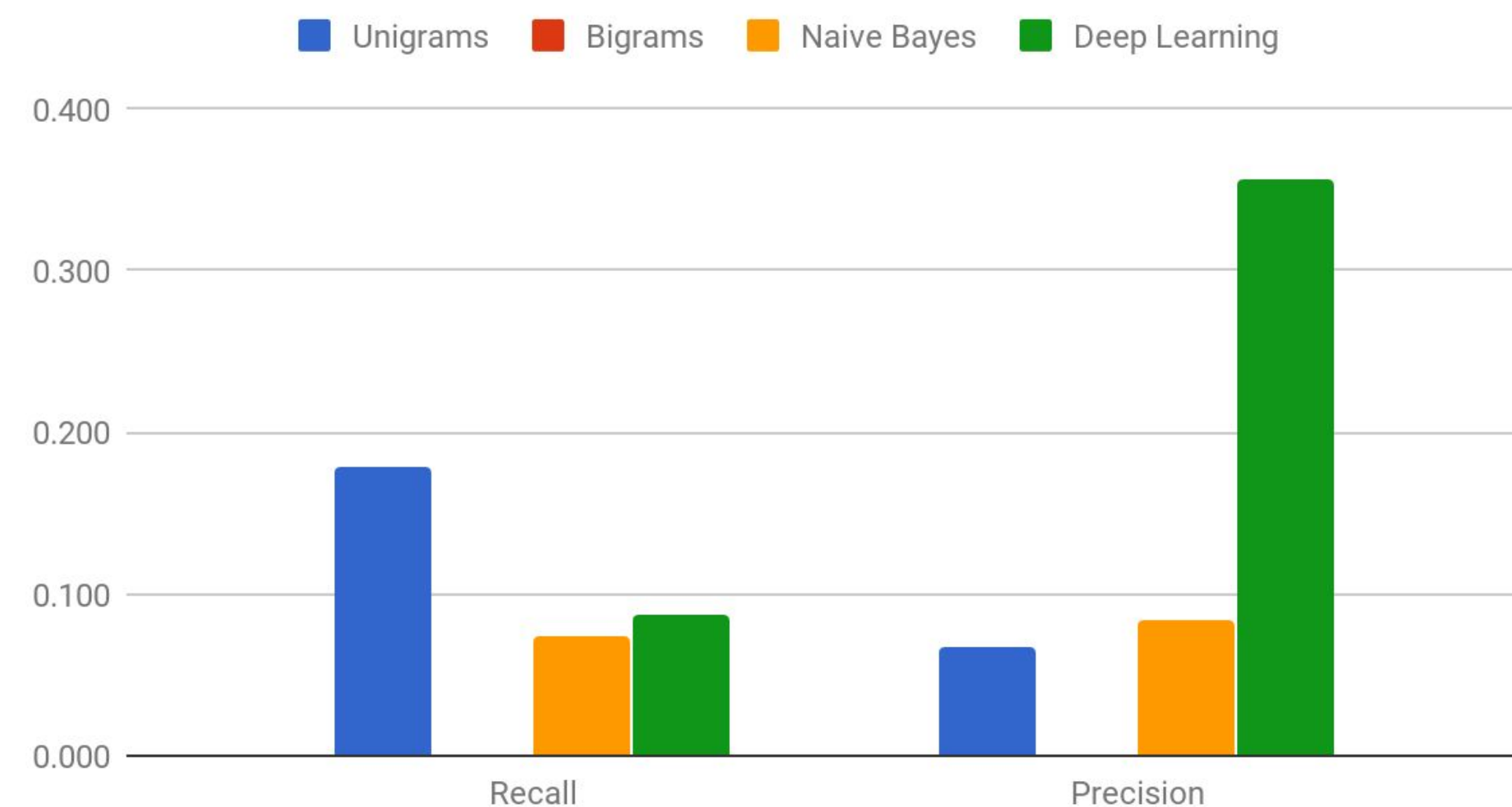
Raw input data:

Text from comment to classify
Text from previous comments in the chain

Derived features

Sentiment (scalar value from NLTK's `SentimentIntensityAnalyzer`)
Presence of Intensifiers (binary value to check for words such as 'really')
Avg. Word Embedding of the comment (vector to represent the text)
Embedding Delta between comment and its context
Absolute count of each of 25 Part-of-Speech tags
Lexical Density: (nouns + adjectives + verbs + adverbs) / all words

Precision & Recall on Test Data



Sample Comment:

- 1) "You're right, this election is totally different!"
- 2) [youre, right, election, totally, different] (with stemming): [your, right, elect, total, differ]
- 3) [-0.40301099, -0.51847959, ..., 0.07148877]

Neural Network Model

Fully connected with 2 hidden layers of 300 nodes and 4 nodes. ReLU function for activation in hidden layers, and sigmoid in output layer. Trained for 10 epochs.

We choose 300 neurons in the first layer because our feature vectors are just over 400 in size. We then added a second hidden layer to account for potentially different kinds of sarcasm (we saw better results with the second hidden than without).

References

Aditya Joshi, Pushpak Bhattacharya, Mark Carman. 2016. Automatic Sarcasm Detection: A Survey
Bjarke Felbo, Alan Mislove, Anders Sogaard, Iyad Rahwan, Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm.
Byron C. Wallace, Do Kook Choe, Laura Kertz and Eugene Charniak. 2014. Humans Require Context to Infer Ironic Intent (so Computers Probably do, too)
Chun-Che Peng, Mohammad Lakis, Jan Wei Pan. 2015. Detecting Sarcasm in Text: An Obvious Solution to a Trivial Problem.
David Bamman and Noah A. Smith. 2015. Contextualized Sarcasm Detection on Twitter
Mikhail Khodak, Nikunj Saunshi, Kiran Vodrahalli. 2017. A Large Self-Annotated Corpus for Sarcasm
Silvio Amir, Byron C. Wallace, Hao Lyu, Paula Carvalho, Mario J. Silva. 2016. Modelling Context with User Embeddings for Sarcasm Detection in Social Media

Train Results

	Error	Recall	Precision	F1 Score
Unigrams	0.111	0.269	0.087	0.131
Bigrams	0.031	0.000	0.000	0.000
Naive Bayes	0.056	0.207	0.168	0.185
Deep Learning	0.028	0.189	0.696	0.297

Test Results

	Error	Recall	Precision	F1 Score
Unigrams	0.098	0.179	0.068	0.098
Bigrams	0.030	0.000	0.000	0.000
Naive Bayes	0.052	0.074	0.084	0.079
Deep Learning	0.032	0.087	0.356	0.140

Note: We report results on the unbalanced data set which more closely resembles the real world, where the majority of statements people make are not sarcastic. Inputs to the deep learning model were not stemmed, while inputs to the others were stemmed.

Discussion

The neural net model is able to classify sarcasm with significantly higher precision than the other models are able to. We attribute the stronger performance of the neural net to its ability to learn from the meaning of the comment since the other models only look at words without context.

We create a model that is able to predict sarcasm with high precision and not necessarily high recall as there could be so many variations from comment to comment, and not even humans detect every piece of sarcasm perfectly.

Future Work

1. Run our model across the entire data set, not just `r/politics`
2. Create more effective model for representing context and meaning
3. Explore more advanced deep learning techniques and a generative model to move towards creation of sarcasm, not just classification