Sarcasm Detection on Reddit comments

Lirida Papallazi

University of Milan

November 30, 2020

Outline

1. Purpose of the analysis
2. Analytical Background
3. Analysis
4. Results
5. Conclusion

The goal

This project focuses on the analysis of the comments made on the data-set containing the comments made on Reddit, with the aim of predicting the probability that a comment will receive a sarcastic answer, given its subreddit.

Natural Language Processing

NLP enables the computer to interact with humans in a natural manner.

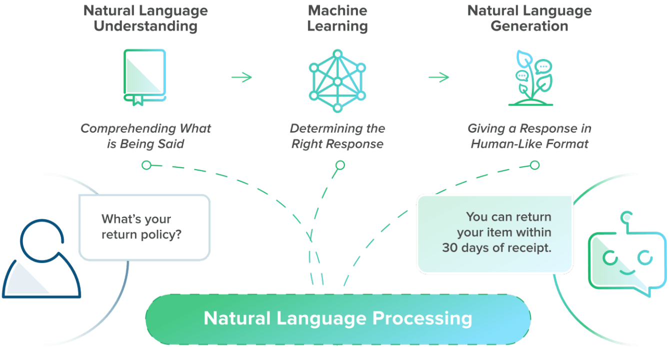

Natural Language Understanding
Machine Learning
Natural Language Generation
Comprehending What is Being Said
Determining the Right Response
Giving a Response in Human-Like Format
What's your return policy?
You can return your item within 30 days of receipt.
Natural Language Processing

Figure: Schema of the NLP logic

NLP Techniques

In order to apply the machine learning algorithms the textual data-set has to be pre-pocessed.

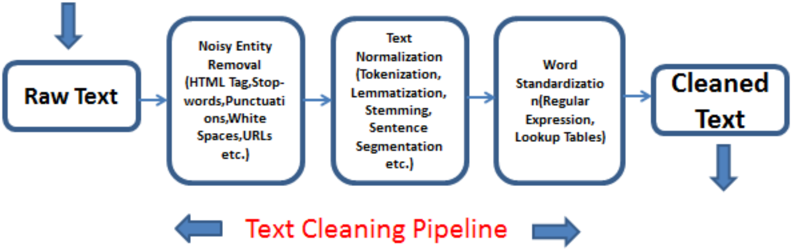

Raw Text
Noisy Entity Removal (HTML Tag,Stop-words,Punctuations,White Spaces,URLs etc.)
Text Normalization (Tokenization, Lemmatization, Stemming, Sentence Segmentation etc.)
Word Standardization(Regular Expression, Lookup Tables)
Cleaned Text
Text Cleaning Pipeline

Figure: Text cleaning pipeline

NLP Techniques (pt. 1)

Stopwords and Punctuation removal Stopwords are typically considered as noise in the text and therefore removed altogether with the punctuation.

Tokenization It is the process of breaking down a text paragraph into smaller chunks such as words or sentence.

NLP Techniques (pt. 2)

Parsing It identifies the structure of the syntax of a text and the dependency relationships between words.

Lemmatization and Stemming NLP uses Lemmatization and Stemming to transform the words back to their root. When the root to which the words are reconnected is the lemma, then we will talk about lemmatization. Stemming instead ”trims” the words to obtain the so called stems, which might not be always semantically correct.

Vectorization (pt. 1)

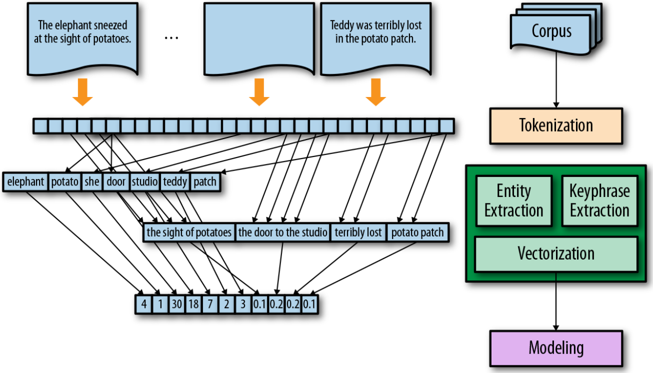


Figure: Process for Bag of Words

Vectorization (pt. 2)

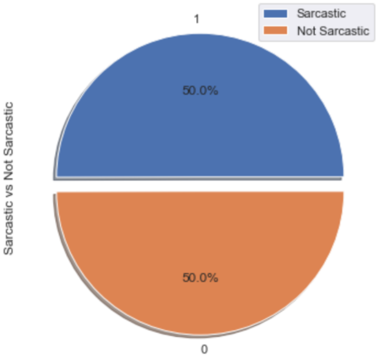
The application of machine learning algorithms requires that the training data is transformed in vectors, so that the computers can understand the data.

Bag of words Model It is used to keep track of the occurrence of each word;

TF-IDF It is intended to reflect how important a word is to a document in a collection or corpus.

The dataset

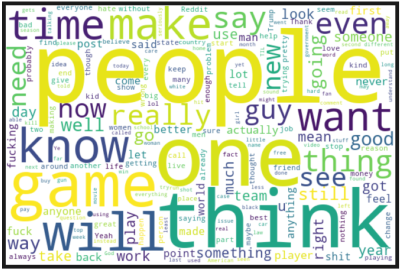
The the data-set is completely balanced in terms of distribution of sarcastic and non sarcastic comments, which means that it does not require any adaptation technique such as oversampling or under-sampling.



Pie chart of the distribution of sarcastic and non sarcastic comments

Wordclouds

A very nice visual representation of text data are Wordclouds, used to depict the most frequently used words in a document.



Wordcloud of parent comments that received a sarcastic answer

Text Classification: logics

Finally the classification model is built: this model will take the cleaned text of the parent comments as input and it will then come up with a prediction on whether the comment will receive a sarcastic or non sarcastic comment.
In particular 2 logics where used:

Alternative 1 Use two different Tf-Idf vectorizers for the parent comments and that of the subreddits;

Alternative 2 Use both the ”cleaned” version of the parent together with the subreddit name as input X, by simply joining the two texts.

Text Classification: results

Different learning algorithms were applied to the two alternative input data-sets and the following results were obtained.

Dataset	Naive Bayes	Logistic Regression	Random Forest	Best CV
Alternative 1	57.26%	59.97%	52.94%	NB = 57.46%
Alternative 2	56.82%	60.31%	51.20%	Logit = 57.27%

Conclusion

As it is possible to observe the results cannot be considered the best ones: the best accuracy achieved barely reaches the 60%. This might be a consequence of three major facts:

1. we are using an indirect approach to discover the probability of receiving a sarcastic answer and the labels actually refer to the comments. We proved that running the same algorithms on the comments leads to better performances;
2. analyzing some of the comments and their labels it is possible to note that some obviously sarcastic answers have not been labeled as such;
3. all the analysis was performed on a very small sample of the original dataset.

Final Considerations

Finally we tried to compute the probability of receiving a SARCASTIC answer of a comment taken directly from Reddit and with AskReddit as subreddit. The comment is "What is your dream job? (Serious)".

Dataset	Naive Bayes	Logistic Regression
Alternative 1	47.095%	44.36%
Alternative 2	25.71%	18.87%

The End