Slides.pptx   Plan.docx

Sentiment Analysis {

[Movie Reviews]

< Recognizing the Polarity of Reviews Using Different NLP and ML Models, and Evaluating Their Quality >

}

Natural Language Processing

Sattam, Serry, Saleh, Ibrahim, Ali

00 {

[Introduction]

< Why this project? How to accomplish it? >

}

Motivation < /1 > {

< Movie ratings influence people's choices, and reviews could be sentimentally analyzed to provide these ratings>

}

Objective < /2 > {

< To construct different models using varied features to analyze movies' sentiment reviews>

}

Sattam, Serry, Saleh, Ibrahim, Ali

Slides.pptx

Plan.docx

Steps {

- Step 01 Labeled Data
- Step 02 Features
- Step 03 Models
- Step 04 Training & Testing

}

Procedures {

01 Planning

< Dataset & Models >

02 Preliminary Work

< Pre-Processing & Data-Exploration >

03 Experimentation

< Modeling & Results >

}

01 {

[Planning]

< What dataset did we train on? What models did we use? >

}

Dataset {

Reviews

< n = 50,000;>

Sentiments

< Positive; Negative>

Extra: Famous

< Rich Literature >

Missing: Meta-data

< Limited Scope >

}

Models {

<Features>

* Bag of Words: implement N-Gram
* TFIDF: implement N-Gram
* Word2Vec

<Algorithms>

* Multinomial Naïve Bayes
* Logistic Regression
* KNN
* RNN
* SVM
* LightGBM
* New: Average Document-Length

}

Sattam, Serry, Saleh, Ibrahim, Ali

02 {

[Preliminary Work]

< How did we pre-process the dataset? What did we find in it? >

}

Pre-Processing {

#Clean data before modeling

Normalize (Stemming; Turn_lowercase;)

Remove (HTML_tags; URLs; dates; special_characters; Stopwords;)

}

Data Exploration: Cloud {

Cloud of Words:

Unigrams are incompetent here

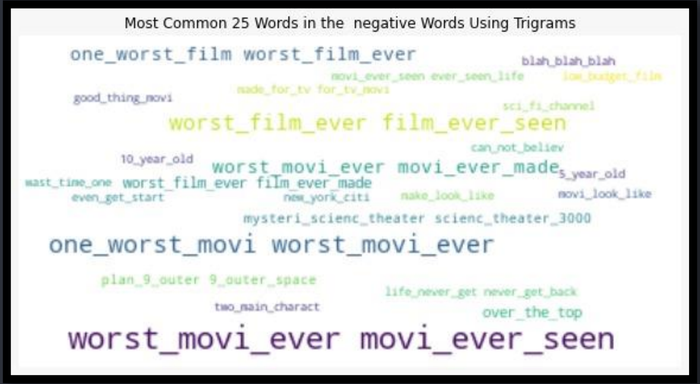
Most Common 50 Words in the Positive Data Frame

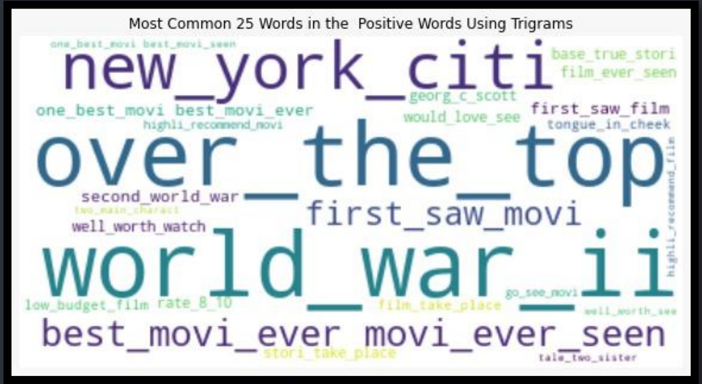
Most Common 50 Words in the Negative Data Frame

Cloud of Words:

Using Trigrams

}





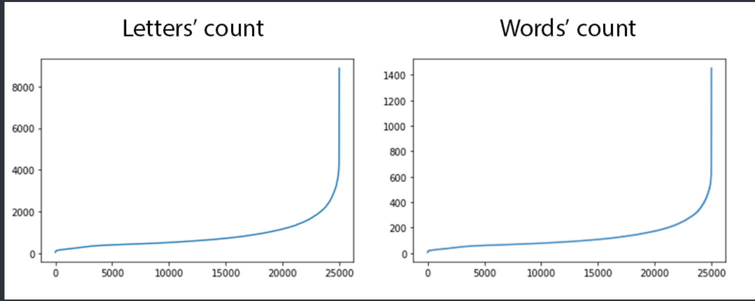
Sattam, Serry, Saleh, Ibrahim, Ali

Data Exploration: Language {

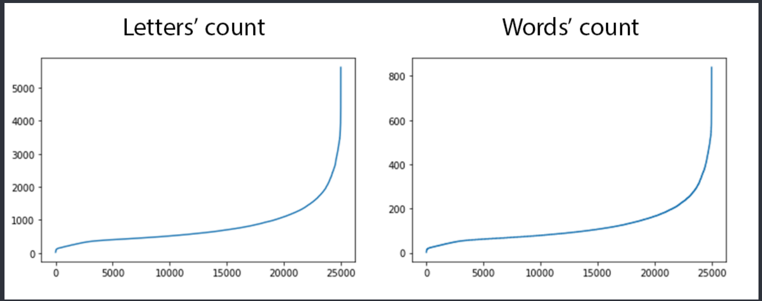
#First impression

The classes had different word count and review length

< Positive >



< Negative >



Characters 75%

< Positive Sentiments reach 75% more characters (8k vs. 5k) >

Words 60%

< Positive Sentiments reach 60% more words (1400 vs. 800) >

Unique 4%

< Positive Sentiment class has 4% more unique words (65k vs. 62.5k) >

Sattam, Serry, Saleh, Ibrahim, Ali

Hypothesis: Positive Sentiment is

More Novel

More Detailed

More Expressive

Highly Open to Experiences

Highly Extraverted

However…

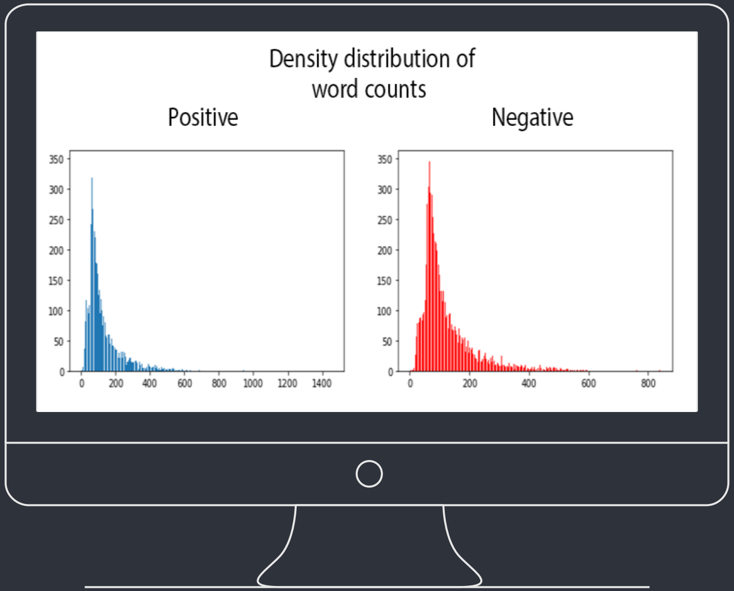
Exception: Outliers

{

Most reviews have a similar (average) number of words and characters.

}

}

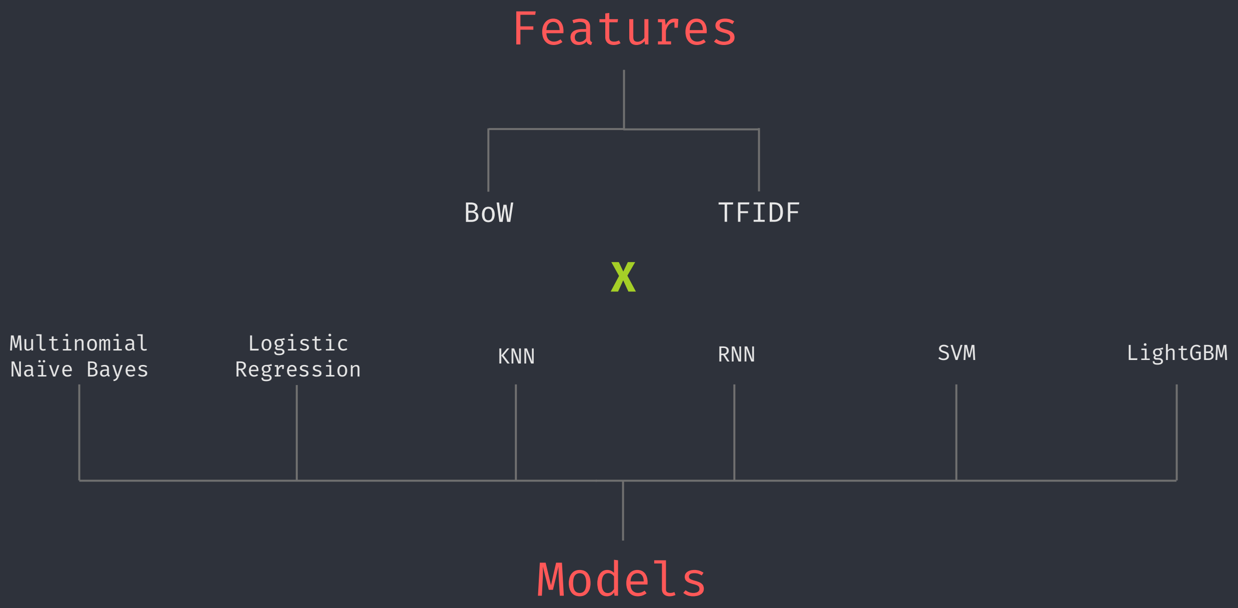


03 {

[Experimentation]

< How did we implement the models? How did they do? >

}


Features
BoW
TFIDF
X
Multinomial Naive Bayes
Logistic Regression
KNN
RNN
SVM
LightGBM
Models

Slides.pptx

Modelling.Py

Results: Unigram {

BoW	TFIDF	
86%	86%	Multinomial Naïve Bayes
88%	89%	Logistic Regression
64%	78%	K-Nearest Neighbors
81%	NA	Recurrent Neural Network
86%	89%	Support Vector Machine
85%	85%	Light Gradient Boosting Machine

}

Results: Bigram {

BoW	TFIDF	
88%	89%	Multinomial Naïve Bayes
87%	86%	Logistic Regression
NA	NA	K-Nearest Neighbors
NA	NA	Recurrent Neural Network
88%	89%	Support Vector Machine
79%	79%	Light Gradient Boosting Machine

}

Results: Trigram {

BoW	TFIDF	
81%	80%	Multinomial Naïve Bayes
75%	78%	Logistic Regression
NA	NA	K-Nearest Neighbors
NA	NA	Recurrent Neural Network
NA	NA	Support Vector Machine
65%	65%	Light Gradient Boosting Machine

}

Experiment: Average Document-Length {

Observation: The average length of positive and negative classes are different (77 vs. 82).

Methodology:

* Add a range around the average of each class resembling the class' trend, and classify the review according to where it lands.
* If outside of the ranges, then assign it randomly.

Results: ~50%; pure chance.

$$\lim_{x \to \infty} f(x) \approx 50$$

$$f(x) = \begin{cases} x \in [67,79], & x = T \\ x \in [80,92], & x = N \\ x = a(x) & \end{cases}$$

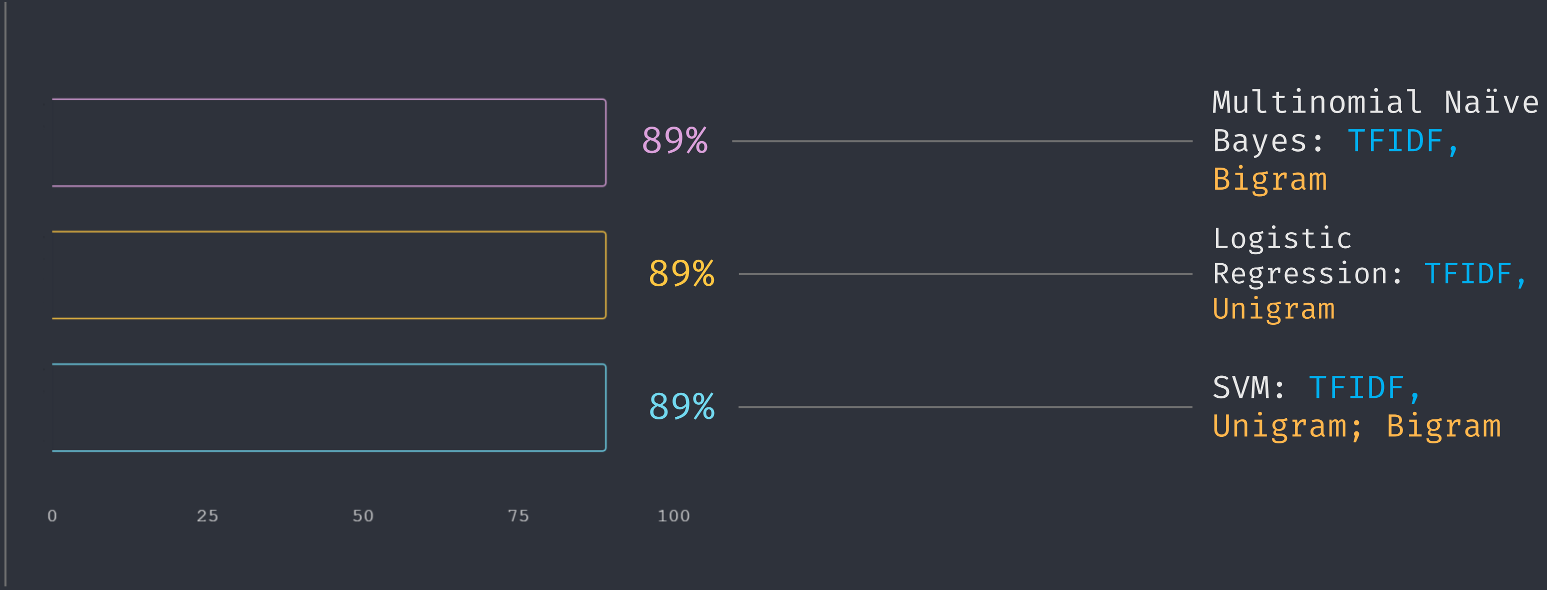
$$a(x) = \begin{cases} p(x|N) = 0.5 \\ p(x|N) = 0.5 \end{cases}$$

}

Slides.pptx

Modelling.Py

Ranking {



} The highest possible accuracy achieved by our multiple methods is: **89%**