Sentiment Analysis of Hindi Reviews

Team Number 9

- Anjali Singh, B.tech, CSE, 2nd sem
- Abhipsa Jena, B.tech, CSE, 4th sem
- Quanta Gupta, B.tech, CSE, 4th sem
- Tamanna Verma, B.tech, CSE, 4th sem
- Vidushi Gupta, B.tech, CSE, 6th sem

Name of Mentor: Dr. Rishabh Kaushal

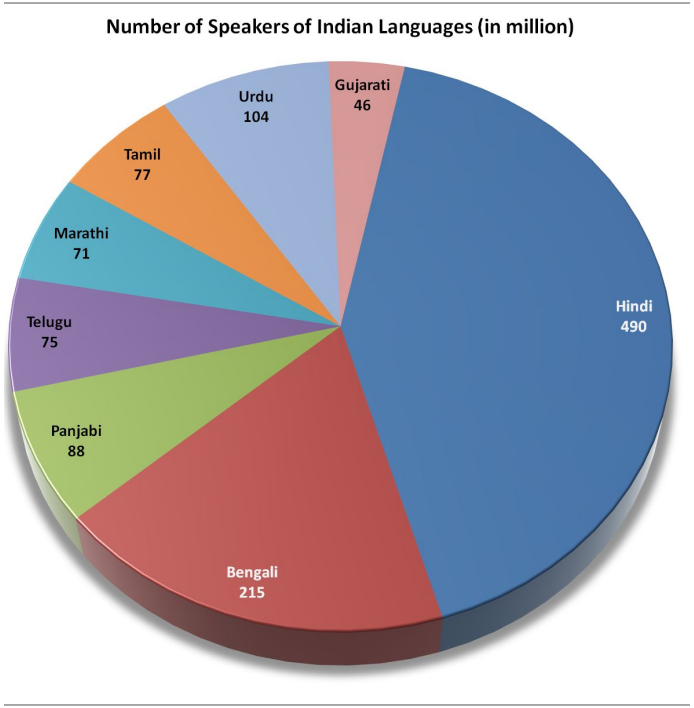
Motivation

According to the 2011 consensus, 43.6% of the Indian population can speak in **Hindi.**

Majority of these people are literate in only **Hindi** language.

Many reviewers also prefer reviewing bollywood movies in Hindi and most movie buffs prefer relying on the movie's native speakers' reviews while exploring movies of a foreign/regional language.

Thus, classifying the sentiment of a movie review written in Hindi into positive or negative can help bring the Bollywood 'movie buff' community closer.



Problem Statement

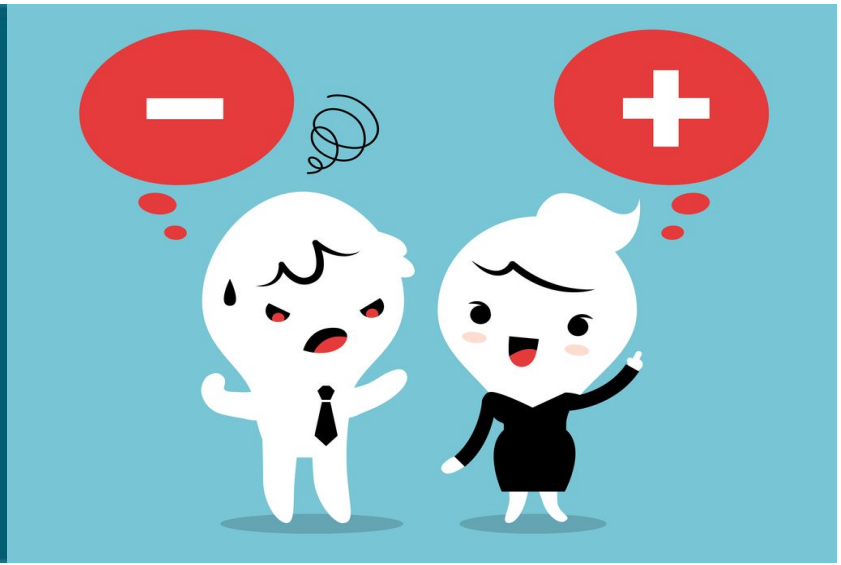
To predict the sentiment of the Hindi movie reviews after translation into English (using Google Translate API) using Natural Language Processing.

Input: A set of hindi movie reviews.

Output: Sentiment Analysis as

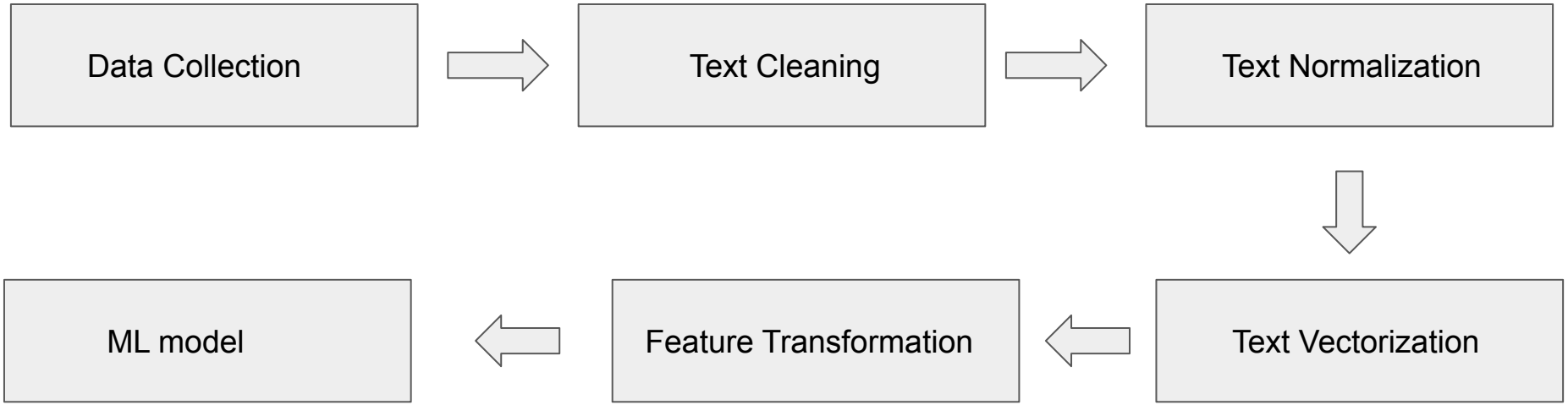
Pos=1, Neg=0

Sentiment analysis

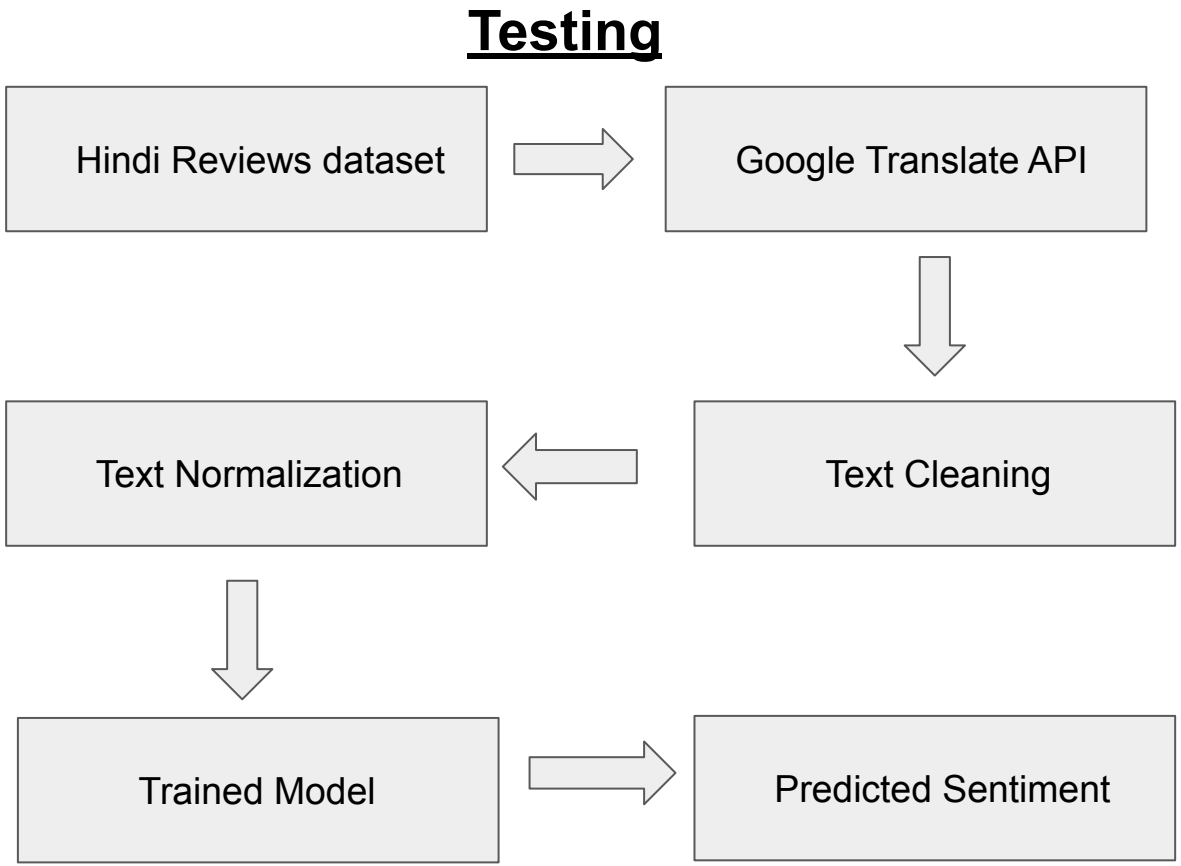


Flow of the Model:

Training



Flow of the Model:



Data Description

Source of Training Data:

https://www.kaggle.com/nltkdata/movie-review?select=movie_review.csv
https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews

Source of Testing Data:

https://github.com/shubham721/Sentiment-Analysis-On-Hindi-Reviews/blob/master/neg_hindi.txt
https://github.com/shubham721/Sentiment-Analysis-On-Hindi-Reviews/blob/master/pos_hindi.txt

Training Data:

```
In [6]: df
```

Out[6]:

	Unnamed: 0	review	sentiment
0	0	One of the other reviewers has mentioned that ...	positive
1	1	A wonderful little production. The...	positive
2	2	I thought this was a wonderful way to spend ti...	positive
3	3	Basically there's a family where a little boy ...	negative
4	4	Petter Mattei's "Love in the Time of Money" is...	positive
...	...	...	...
70995	70995	o . k , so the acting is not up to par , but i...	negative
70996	70996	but what about the script ?	negative
70997	70997	well , that's not up to par either .	negative
70998	70998	emmerich and screenwriting pal dean devlin see...	negative
70999	70999	the dialogue is banal (broderick looks at a l...	negative

71000 rows × 3 columns

In [148]:
```
df.info
```

Out[148]:
```
<bound method DataFrame.info of                                                  review  sentiment  length  \
0      One of the other reviewers has mentioned that ...        1.0    1761
1      A wonderful little production. <br /><br />The...        1.0     998
2      I thought this was a wonderful way to spend ti...        1.0     926
3      Basically there's a family where a little boy ...        0.0     748
4      Petter Mattei's "Love in the Time of Money" is...        1.0    1317
...                                                  ...        ...     ...
70995  o . k , so the acting is not up to par , but i...        0.0      74
70996                        but what about the script ?        0.0      27
70997               well , that's not up to par either .        0.0      36
70998  emmerich and screenwriting pal dean devlin see...        0.0     130
70999  the dialogue is banal ( broderick looks at a l...        0.0     202

                                    clean_review_sentence
0      One reviewer mentioned watching 1 Oz episode h...
1      A wonderful little production   The filming te...
2      I thought wonderful way spend time hot summer ...
3      Basically family little boy Jake think zombie ...
4      Petter Mattei Love Time Money visually stunnin...
...                                                  ...
70995                          k acting par never movie
70996                                            script
70997                                  well par either
70998  emmerich screenwriting pal dean devlin seem ca...
70999  dialogue banal oderick look lot fish utters lo...

[71000 rows x 4 columns]>
```

Testing Data:

Original dataset in hindi

	Unnamed: 0	Text	sentiment
0	0	इसी मुद्दे पर फ़िल्म बनाने का प्रयास सराहा जाना...	postive
1	1	उनकी भाव-भंगिमाओं में अनोखापन है	postive
2	2	जॉन अब्राहम एक्शन दृश्यों में यों भी अच्छे और ...	postive
3	3	दोस्तों को नायक की मौत पर कहानी का अंत एक त्रा...	negative
4	4	कहानी बड़ी होशियारी से अपनी अपील को बच्चों के ...	postive
...	...	...	...
793	848	यह लोकतंत्र के एक मजबूत स्तम्भ के दुरुपयोग का ...	negative
794	849	उनका चित्रण बहुत ही फॉल्स है	negative
795	850	दर्शकों से कनेक्ट नहीं कर पाई 'हसीना पारकर	negative
796	851	दूसरी ओर उड़िया पृष्ठभूमि पर बनी इस फिल्म में ...	negative
797	852	कमी है तो ऐसे प्रसंगों और दृश्यों की जहां वे व...	negative

798 rows x 3 columns

In [3]:

```
df.info
```

```
<bound method DataFrame.info of      Unnamed: 0                                               Text sentiment
0             0  इसी मुद्दे पर फ़िल्म बनाने का प्रयास सराहा जाना...   postive
1             1                    उनकी भाव-भंगिमाओं में अनोखापन है   postive
2             2  जॉन अब्राहम एक्शन दृश्यों में यों भी अच्छे और ...   postive
3             3  दोस्तों को नायक की मौत पर कहानी का अंत एक त्रा...  negative
4             4  कहानी बड़ी होशियारी से अपनी अपील को बच्चों के ...   postive
..          ...                                                  ...       ...
793         848  यह लोकतंत्र के एक मजबूत स्तम्भ के दुरुपयोग का ...  negative
794         849                        उनका चित्रण बहुत ही फॉल्स है  negative
795         850          दर्शकों से कनेक्ट नहीं कर पाई 'हसीना पारकर  negative
796         851  दूसरी ओर उड़िया पृष्ठभूमि पर बनी इस फिल्म में ...  negative
797         852  कमी है तो ऐसे प्रसंगों और दृश्यों की जहां वे व...  negative

[798 rows x 3 columns]>
```

Code for Google Translation API:

In [30]:
```
from googletrans import Translator


translator=Translator()
```

In [31]:
```
translations = {}

for column in df_copy.columns:
    # Unique elements of the column
    unique_elements = df_copy[column].unique()
    for element in unique_elements:
        # Adding all the translations to a dictionary (translations)
        translations[element] = translator.translate(element).text
translations
```

```
{'इसी मुद्दे पर फ़िल्म बनाने का प्रयास सराहा जाना चाहिए': 'An attempt to make a film on this issue should be appreciated',
 'उनकी भाव-भंगिमाओं में अनोखापन है': 'They have a weirdness',
 'जॉन अब्राहम एक्शन दृश्यों में यों भी अच्छे और विश्वसनीय लगते हैं।': 'John Abraham looks good and believable even in action sequences.',
 'दोस्तों को नायक की मौत पर कहानी का अंत एक त्रासद अंत लग सकता है': 'Friends may find the end of the story a tragic end on the death of the protagonist',
 'कहानी बड़ी होशियारी से अपनी अपील को बच्चों के मिजाज से उड़ान भरते हुए युवा व परिपक्\u200dव सोच-अप्रोच का विस्तार प्रदान करती है': 'The story cleverly lends its appeal to the mood of the young and mature, taking flight from the mood of the children.',
 'फिल्म का बैकग्राउंड स्कोर और गीत-संगीत उल्लेखनीय है। ': "The film's background score and song-music are notable.",
 'ऐसे में डांस उसको कैसे दूसरे शहरों में लेकर जाता है यह बखूबी दिखाया गया है': 'In this way, how the dance takes him to other cities is shown very well',
 ' दर्शकों को रिझाने या बहलाने के लिए इस फिल्म में कुछ भी नहीं है': 'There is nothing in this film to entice or entice the audience',
 'यह उन्\u200dहोंने जिस आधार पर कहा उसी वंशवाद के विरोध का ढोल राहुल गांधी पीट रहे हैं।': 'This is the basis on which he said that Rahul Gandhi is beating the drum of opposition to dynasty.',
 'लेकिन शायद इसका भी कोई असर फिल्म को संजोने में नहीं हो पाया': 'But perhaps it did not have any effect in saving the film.',
 '‘बाहुबली 2’ भव\u200dयता और विशालता में पहली से ज\u200dयादा बड़ी और चमकदार हो गई है': "'Bahubali 2' has become bigger and brighter than ever
```

Labelling of Data and Preprocessing

Training Data After preprocessing

```
In [166]: df.head()
```

Out[166]:

	review	sentiment	clean_review_sentence
0	One of the other reviewers has mentioned that ...	1.0	One reviewer mentioned watching 1 Oz episode h...
1	A wonderful little production. The...	1.0	A wonderful little production The filming te...
2	I thought this was a wonderful way to spend ti...	1.0	I thought wonderful way spend time hot summer ...
3	Basically there's a family where a little boy ...	0.0	Basically family little boy Jake think zombie ...
4	Petter Mattei's "Love in the Time of Money" is...	1.0	Petter Mattei Love Time Money visually stunnin...

Stemming : Used to cut down the common prefixes and suffixes used in words.

Lemmatization : Grouping together the different inflected forms of a word so they can be analysed as a single item. Basically, it links words with similar meaning to one word

**We removed unnecessary data using nltk:*

Stopwords : Commonly used word (such as “the”, “a”, “an”, “in”)

Punctuation : Removing all the punctuation marks such as “ , . etc

Null values : Removed all the rows with null values.

Count Vectorization: Counting the number of occurrences each words appears in the dataset.

Tf-idf (frequency–inverse document frequency): Numerical statistic that is intended to reflect how important a word is to a document in a collection

Deleting the unnecessary Columns and labelling sentiment
Negative→ 1
Positive→ 0

In [157]:
```
df = df.drop(columns = ['Unnamed: 0'])
df['sentiment'] = df['sentiment'].where(df['sentiment'] == 'negative').replace('negative',0)
df['sentiment'].fillna('None', inplace=True)
df['sentiment'] = df['sentiment'].replace('None',1)
```

Removing Punctuation , stopwords and lemmatization of the reviews.

In [163]:
```
def clean_text_sentence(txt):
  no_punct = "".join([c for c in txt if c not in string.punctuation])
  tokens = re.split('\W+',txt)
  txt = " ".join([wn.lemmatize(word) for word in tokens if word not in stop])
  return txt
```

In [164]:
```
df['clean_review_sentence'] = df['review'].apply(lambda x: clean_text_sentence(x))
```

Translated and preprocessed testing dataset

	Text	sentiment
0	An attempt to make a film on this issue should...	1
1	They have a weirdness	1
2	John Abraham looks good and believable even in...	1
3	Friends may find the end of the story a tragic...	0
4	The story cleverly lends its appeal to the moo...	1
5	The film's background score and song-music are...	1
6	In this way, how the dance takes him to other ...	1
7	There is nothing in this film to entice or ent...	0
8	This is the basis on which he said that Rahul ...	0
9	But perhaps it did not have any effect in savi...	0

Code for Google Translation API:

In [30]:
```
from googletrans import Translator


translator=Translator()
```

In [31]:
```
translations = {}

for column in df_copy.columns:
    # Unique elements of the column
    unique_elements = df_copy[column].unique()
    for element in unique_elements:
        # Adding all the translations to a dictionary (translations)
        translations[element] = translator.translate(element).text
translations
```

```
{'इसी मुद्दे पर फ़िल्म बनाने का प्रयास सराहा जाना चाहिए': 'An attempt to make a film on this issue should be appreciated',
 'उनकी भाव-भंगिमाओं में अनोखापन है': 'They have a weirdness',
 'जॉन अब्राहम एक्शन दृश्यों में यों भी अच्छे और विश्वसनीय लगते हैं।': 'John Abraham looks good and believable even in action sequences.',
 'दोस्तों को नायक की मौत पर कहानी का अंत एक त्रासद अंत लग सकता है': 'Friends may find the end of the story a tragic end on the death of the protagonist',
 'कहानी बड़ी होशियारी से अपनी अपील को बच्चों के मिजाज से उड़ान भरते हुए युवा व परिपक्\u200dव सोच-अप्रोच का विस्तार प्रदान करती है': 'The story cleverly lends its appeal to the mood of the young and mature, taking flight from the mood of the children.',
 'फिल्म का बैकग्राउंड स्कोर और गीत-संगीत उल्लेखनीय है। ': "The film's background score and song-music are notable.",
 'एेसे में डांस उसको कैसे दूसरे शहरों में लेकर जाता है यह बखूबी दिखाया गया है': 'In this way, how the dance takes him to other cities is shown very well',
 ' दर्शकों को रिझाने या बहलाने के लिए इस फिल्म में कुछ भी नहीं है': 'There is nothing in this film to entice or entice the audience',
 'यह उन्\u200dहोंने जिस आधार पर कहा उसी वंशवाद के विरोध का ढोल राहुल गांधी पीट रहे हैं।': 'This is the basis on which he said that Rahul Gandhi is beating the drum of opposition to dynasty.',
 'लेकिन शायद इसका भी कोई असर फिल्म को संजोने में नहीं हो पाया': 'But perhaps it did not have any effect in saving the film.',
 '‘बाहुबली 2’ भव्\u200dयता और विशालता में पहली से ज\u200dयादा बड़ी और चमकदार हो गई हैं': "'Bahubali 2' has become bigger and brighter than ever
```

```
In [149]: df_hindi.info
```

```
Out[149]: <bound method DataFrame.info of      sentiment                                clean_review_sentence
0            1                attempt make film issue appreciated
1            1                                          weirdness
2            1  john abraham look good believable even action ...
3            0  friend may find end story tragic end death pro...
4            1  story cleverly lends appeal mood young mature ...
..         ...                                                ...
793          0        naked truth misuse strong column democracy
794          0                                  portrayal false
795          0              could connect audience hasina parkar
796          0  hand film based oriya background current spice...
797          0                  lack theme scene get detail depth

[797 rows x 2 columns]>
```

```
In [148]: df.info
```

```
Out[148]: <bound method DataFrame.info of                                                review  sentiment  length
0      One of the other reviewers has mentioned that ...        1.0    1761
1      A wonderful little production. <br /><br />The...        1.0     998
2      I thought this was a wonderful way to spend ti...        1.0     926
3      Basically there's a family where a little boy ...        0.0     748
4      Petter Mattei's "Love in the Time of Money" is...        1.0    1317
...                                                  ...        ...     ...
70995  o . k , so the acting is not up to par , but i...        0.0      74
70996                      but what about the script ?          0.0      27
70997              well , that's not up to par either .         0.0      36
70998  emmerich and screenwriting pal dean devlin see...        0.0     130
70999  the dialogue is banal ( broderick looks at a l...        0.0     202

                                clean_review_sentence
0      One reviewer mentioned watching 1 Oz episode h...
1      A wonderful little production   The filming te...
2      I thought wonderful way spend time hot summer ...
3      Basically family little boy Jake think zombie ...
4      Petter Mattei Love Time Money visually stunnin...
...                                                  ...
70995                          k acting par never movie
70996                                            script
70997                                      well par either
70998  emmerich screenwriting pal dean devlin seem ca...
70999  dialogue banal oderick look lot fish utters lo...
```

```
In [147]: df_hindi.describe()
```

Out[147]:

	sentiment
count	797.000000
mean	0.526976
std	0.499585
min	0.000000
25%	0.000000
50%	1.000000
75%	1.000000
max	1.000000

```
In [20]: df.describe()
```

Out[20]:

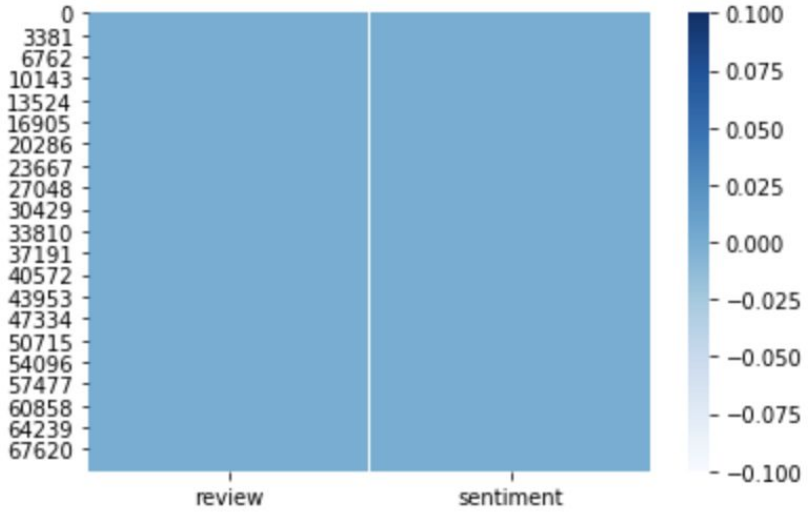
	sentiment	length
count	71000.000000	71000.000000
mean	0.570775	808.187901
std	0.494969	955.503133
min	0.000000	1.000000
25%	0.000000	125.000000
50%	1.000000	592.000000
75%	1.000000	1083.000000
max	1.000000	12988.000000

Data Visualization:

1. **Explore Dataset**

```
In [15]: sns.heatmap(df.isnull(), cbar = True, cmap ='Blues')
```

```
Out[15]: <matplotlib.axes._subplots.AxesSubplot at 0x7fe3bddc9e10>
```

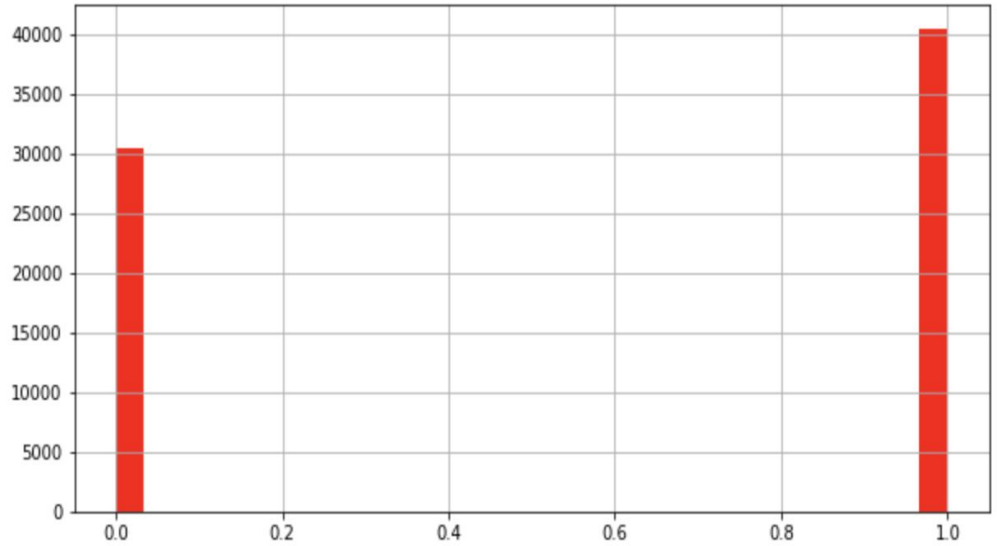


Heatmap showing that our **Training data does NOT have any Null values** In reviews as well as sentiment.

BINS-->Number of histogram bins to be used. A histogram displays numerical data by grouping data into "bins" of equal width. Each bin is plotted as a bar whose height corresponds to how many data points are in that bin. Bins are also sometimes called "intervals", "classes", or "buckets". figsize(horizontal, vertical)

In [16]:
```
df['sentiment'].hist(bins=30, figsize = (10,5), color='r')
```

Out[16]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe3dc7be890>
```



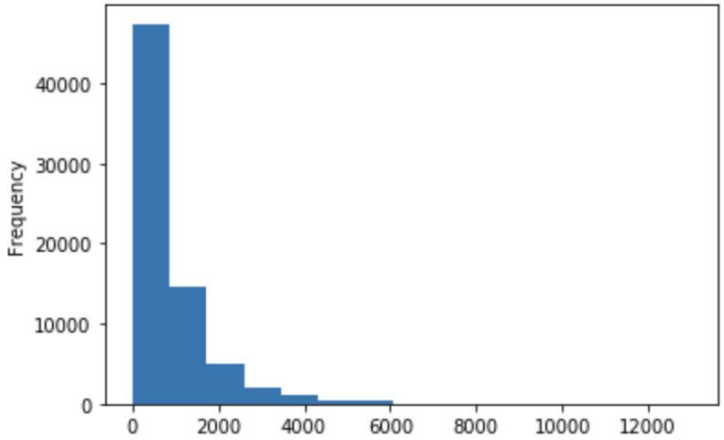
2.

Histogram showing the number of reviews that have 0(negative) and 1(positive) labels.

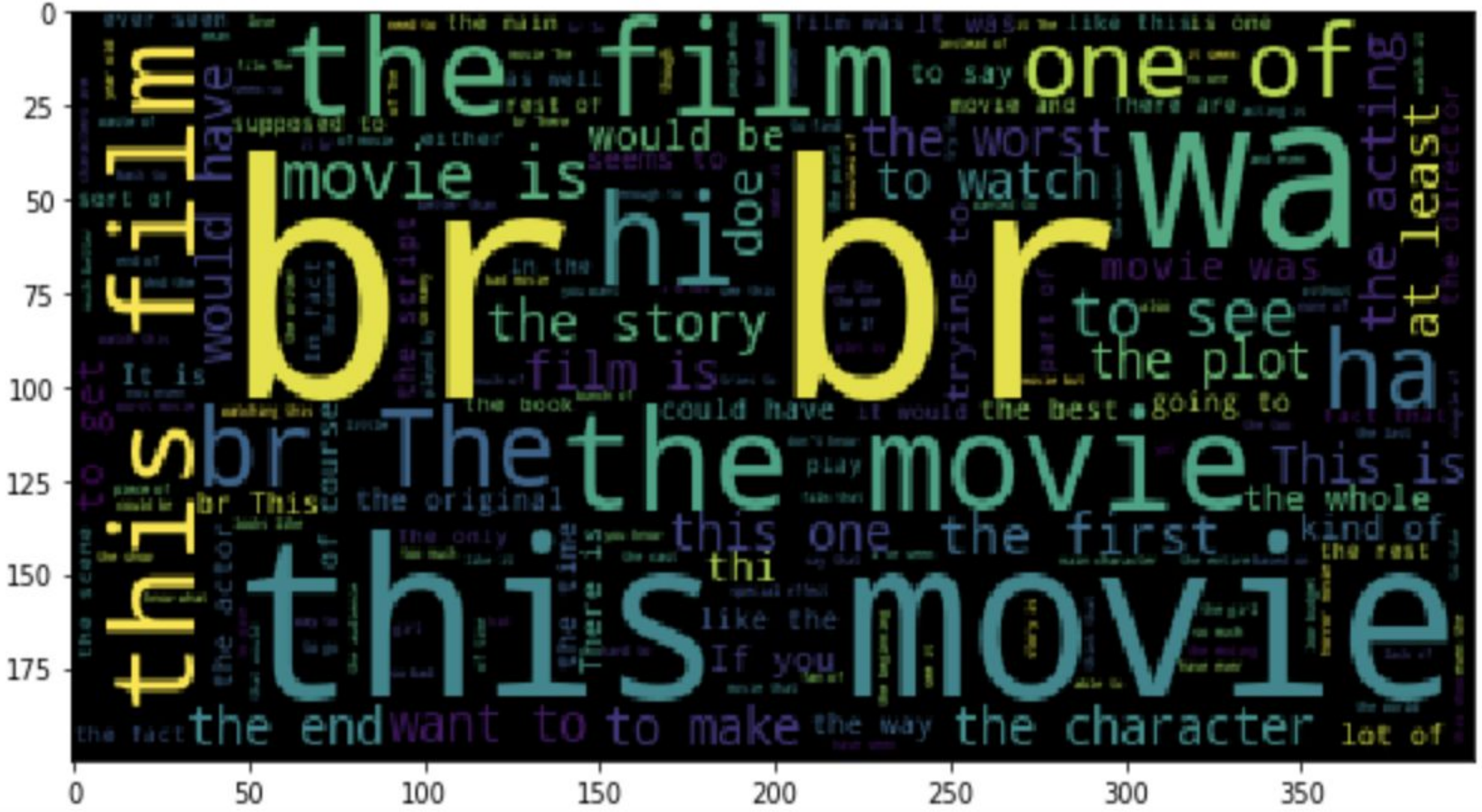
3.

Calculating the length of each review
And filling it in another column subsequently.
Then plotting a histogram with number of reviews on the y-axis and length of reviews on x-axis.

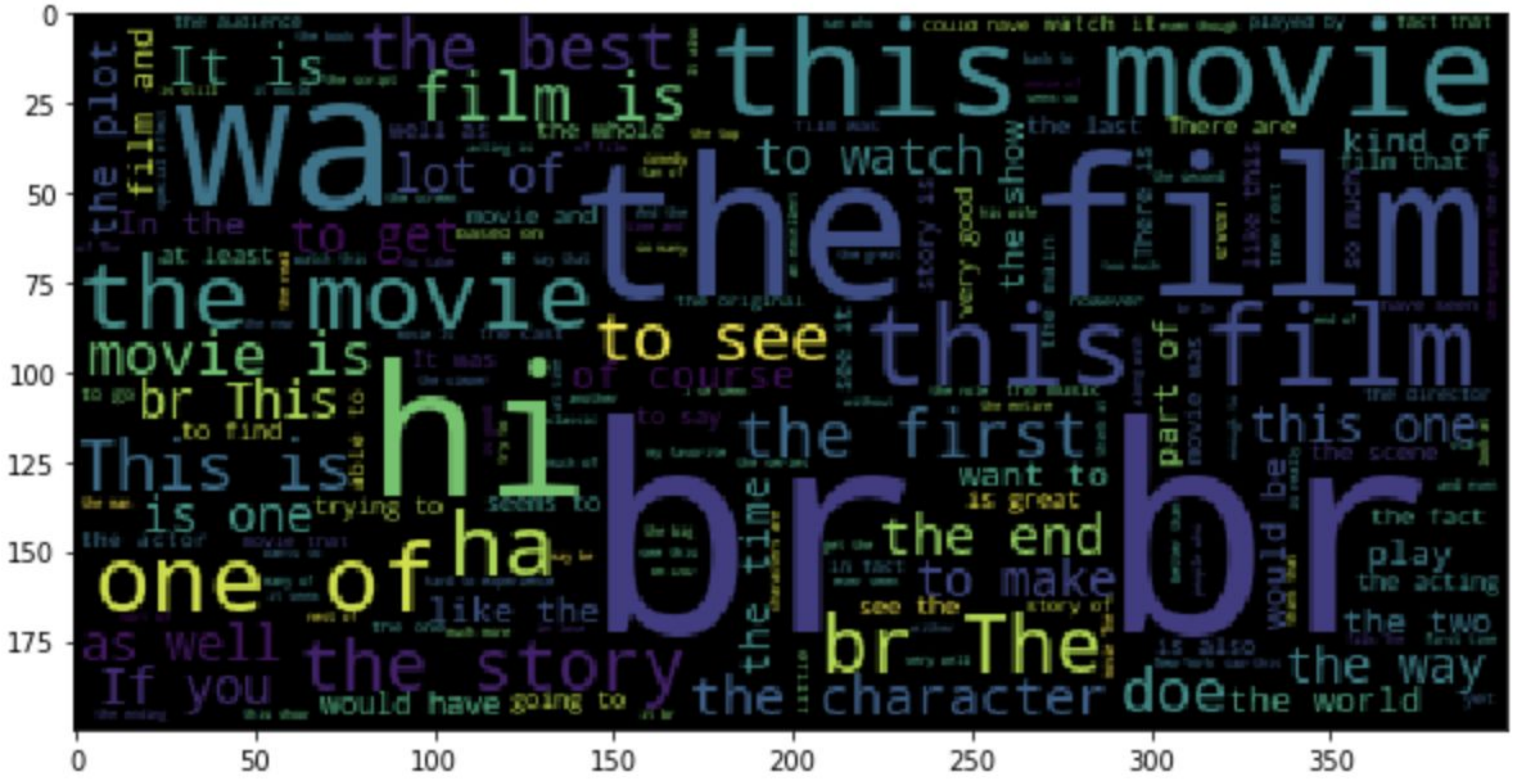
```
In [18]: df['length'] = df['review'].apply(lambda x: len(x))

In [19]: df['length'].plot(bins=15, kind='hist')

Out[19]: <matplotlib.axes._subplots.AxesSubplot at 0x7fe3dd0c3d90>
```



WordCloud: Negative:



Positive:



this movie
wa
the best
It is
film is
the film
to watch
lot of
the plot
film and
the movie
to see
this film
movie is
hi
br This
of course
the first
this one
This is
br br
is one
ha
want to
the end
one of
to make
like the
as well
the story
br The
If you
the character
doe
the way
the world
the time
would be
the two
the acting
play
the fact
kind of
the last
There are
the show
very good
part of
to get
In the
at least
to find
trying to
is great
would have
going to
0
25
50
75
100
125
150
175
0
50
100
150
200
250
300
350

Vectorization and Feature Transformation:

Count Vectorizer

In [38]:
```
vectorizer = CountVectorizer(analyzer = clean_text_sentence)
#vectorizer.get_feature_name
X_train_feat = vectorizer.fit_transform(X_train).toarray()
X_test_feat = vectorizer.transform(X_test).toarray()
print(X_train_feat)
print(X_test_feat)
```

```
[[177   0   1 ...   0   0   0]
 [ 93   0   0 ...   0   0   0]
 [ 95   0   0 ...   0   0   0]
 ...
 [  3   0   0 ...   0   0   0]
 [ 14   0   0 ...   0   0   0]
 [ 18   0   0 ...   0   0   0]]
[[ 4  0  0 ...  0  0  0]
 [ 0  0  0 ...  0  0  0]
 [ 7  0  0 ...  0  0  0]
 ...
 [ 4  0  0 ...  0  0  0]
 [10  0  0 ...  0  0  0]
 [ 5  0  0 ...  0  0  0]]
```

In [39]:
```
X_train_feat.shape, X_test_feat.shape
```

TF-IDF

In [91]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf_vect = TfidfVectorizer(analyzer = clean_text_sentence)
X_train_feat_tfidf = tfidf_vect.fit_transform(X_train).toarray()
X_test_feat_tfidf = tfidf_vect.transform(X_test).toarray()
print(X_train_feat_tfidf)
print(X_test_feat_tfidf)
```

```
[[0.59972839 0.         0.00837001 ... 0.         0.         0.        ]
 [0.52265305 0.         0.         ... 0.         0.         0.        ]
 [0.58933807 0.         0.         ... 0.         0.         0.        ]
 ...
 [0.51468295 0.         0.         ... 0.         0.         0.        ]
 [0.52119914 0.         0.         ... 0.         0.         0.        ]
 [0.52637164 0.         0.         ... 0.         0.         0.        ]]
[[0.36503074 0.         0.         ... 0.         0.         0.        ]
 [0.         0.         0.         ... 0.         0.         0.        ]
 [0.43541643 0.         0.         ... 0.         0.         0.        ]
 ...
 [0.37022407 0.         0.         ... 0.         0.         0.        ]
 [0.49274301 0.         0.         ... 0.         0.         0.        ]
 [0.44999666 0.         0.         ... 0.         0.         0.        ]]
```

In [92]:
```
X_train_feat_tfidf.shape, X_test_feat_tfidf.shape
```

Out[92]: ((71000, 129), (797, 129))

Accuracy and Result Evaluation:

1. Count Vectorization

Algorithm	Accuracy
Random Forest Classifier	53.19
Random Forest Regressor	54.32
Multinomial Naive Bayes	51.19
Gaussian Naive Bayes	47.30
Logistic Regression	52.57
KNeighbor Classifier	54.32

2. **TF-IDF**

Algorithm	Accuracy
Random Forest Classifier	51.81
Random Forest Regressor	51.81
Multinomial Naive Bayes	52.69
Gaussian Naive Bayes	48.43
Logistic Regression	52.19
KNeighbor Classifier	50.69

Code of the maximum accuracy Algorithm: Random Forest

In [53]:
```
regressor_rfr = RandomForestRegressor(n_estimators = 500, random_state = 0)
regressor_rfr
```

Out[53]:
```
RandomForestRegressor(bootstrap=True, ccp_alpha=0.0, criterion='mse',
                      max_depth=None, max_features='auto', max_leaf_nodes=None,
                      max_samples=None, min_impurity_decrease=0.0,
                      min_impurity_split=None, min_samples_leaf=1,
                      min_samples_split=2, min_weight_fraction_leaf=0.0,
                      n_estimators=500, n_jobs=None, oob_score=False,
                      random_state=0, verbose=0, warm_start=False)
```

In [54]:
```
regressor_rfr.fit(X_train_feat, y_train)
```

Out[54]:
```
RandomForestRegressor(bootstrap=True, ccp_alpha=0.0, criterion='mse',
                      max_depth=None, max_features='auto', max_leaf_nodes=None,
                      max_samples=None, min_impurity_decrease=0.0,
                      min_impurity_split=None, min_samples_leaf=1,
                      min_samples_split=2, min_weight_fraction_leaf=0.0,
                      n_estimators=500, n_jobs=None, oob_score=False,
                      random_state=0, verbose=0, warm_start=False)
```

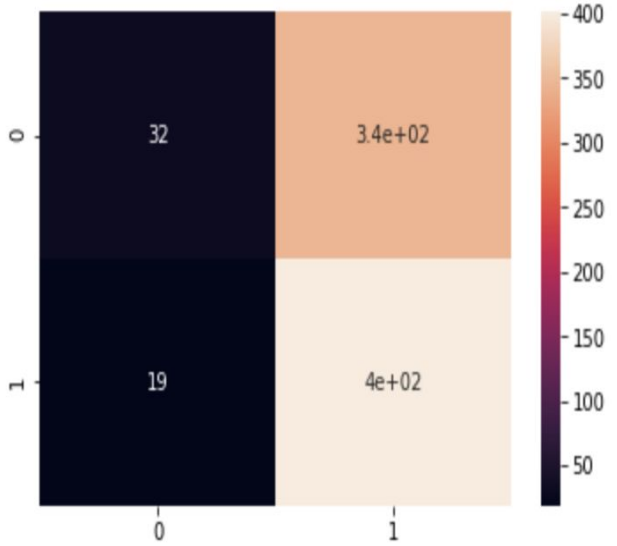
In [55]:
```
predictions_rfr = regressor_rfr.predict(X_test_feat)
predictions_rfr
```

```
0.584     , 0.56      , 0.678     , 0.444     , 0.686     ,
0.554     , 0.682     , 0.552     , 0.57200606, 0.57488889,
0.688     , 0.694     , 0.638     , 0.58275   , 0.56      ,
0.594     , 0.57266667, 0.638     , 0.53266667, 0.634     ,
0.706     , 0.516     , 0.62      , 0.698     , 0.626     ,
0.53      , 0.57      , 0.5983    , 0.624     , 0.664     ,
0.676     , 0.642     , 0.644     , 0.626     , 0.756     ,
0.572     , 0.622     , 0.542     , 0.37666667, 0.734     ,
0.636     , 0.602     , 0.6156    , 0.642     , 0.6       ,
0.578     , 0.6       , 0.724     , 0.5745    , 0.6815    ,
0.5488    , 0.652     , 0.538     , 0.604     , 0.726     ,
0.584     , 0.616     , 0.45486667, 0.702     , 0.596     ,
0.684     , 0.736     , 0.462     , 0.542     , 0.626     ,
0.72      , 0.74      , 0.49175   , 0.592     , 0.574     ,
0.626     , 0.658     , 0.553     , 0.67      , 0.688     ,
0.653     , 0.615     , 0.578     , 0.676     , 0.52      ,
```

Result Evaluation:

In [56]:
```
cm = confusion_matrix(y_test, predictions_rfr.round())
sns.heatmap(cm, annot=True)
```

Out[56]: <matplotlib.axes._subplots.AxesSubplot at 0x7fe378adaa50>



In [60]:
```
print(classification_report(y_test,predictions_rfr.round()))
```

	precision	recall	f1-score	support
0	0.63	0.08	0.15	377
1	0.54	0.95	0.69	420
accuracy			0.54	797
macro avg	0.58	0.52	0.42	797
weighted avg	0.58	0.54	0.43	797

In [57]:
```
accuracy_score(y_test, predictions_rfr.round())
```

Out[57]: 0.5432873274780426

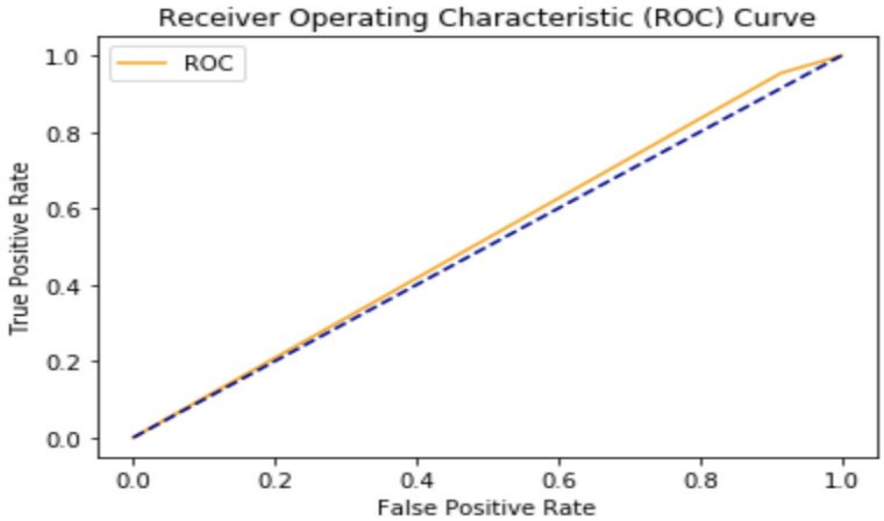
In [64]:
```
accuracy_score(y_test, predictions_rfr.round())
```

Out[64]:
```
0.5432873274780426
```

In [65]:
```
roc_auc_score(y_test, predictions_rfr.round())
```

Out[65]:
```
0.5198212706833396
```

In [66]:
```
fper, tper, _ = roc_curve(y_test, predictions_rfr.round())
plot_roc_curve(fper, tper)
```



SCOPE OF THE PROJECT:

1. Improving Accuracy by more preprocessing of Dataset. Also by implementing our Features like N-gram for better results.

2. Deployment of our ML model

3. Training our model in such a way so that it can understand emojis and detect sarcasm.

4. Divide our sentiment into more classes like (neutral , somewhat positive, somewhat negative)

References

[1]https://www.oreilly.com/library/view/applied-text-analysis/9781491963036/ch04.html
[2]https://towardsdatascience.com/text-classification-using-k-nearest-neighbors-46fa8a77acc5
[3]https://www.oreilly.com/library/view/applied-text-analysis/9781491963036/ch04.html
[4]https://www.youtube.com/watch?v=ZeQgNhc0vag

[5]https://github.com/shubham721/Sentiment-Analysis-On-Hindi-Reviews

[6]https://medium.com/analytics-vidhya/neural-machine-translation-for-hindi-english-sequence-to-sequence-learning-1298655e334a

[7]https://towardsdatascience.com/google-translate-api-for-python-723093c2144b


THANK
YOU