



OVERVIEW



- Opinion Miner is the process of using natural language processing, text analysis, and statistics to analyze customer sentiment.
- The best businesses understand the sentiment of their customers—what people are saying, how they're saying it, and what they mean.
- Customer's opinion can be found in tweets, comments, reviews, or other places where people mention your brand. Opinion Miner is the domain of understanding these emotions with software, and it's a must-understand for developers and business leaders in a modern workplace. As with many other fields, advances in deep learning have brought opinion miner into the foreground of cutting-edge algorithms.
- Today we use natural language processing, statistics, and text analysis to extract, and identify the opinion of words into positive, negative, or neutral categories.



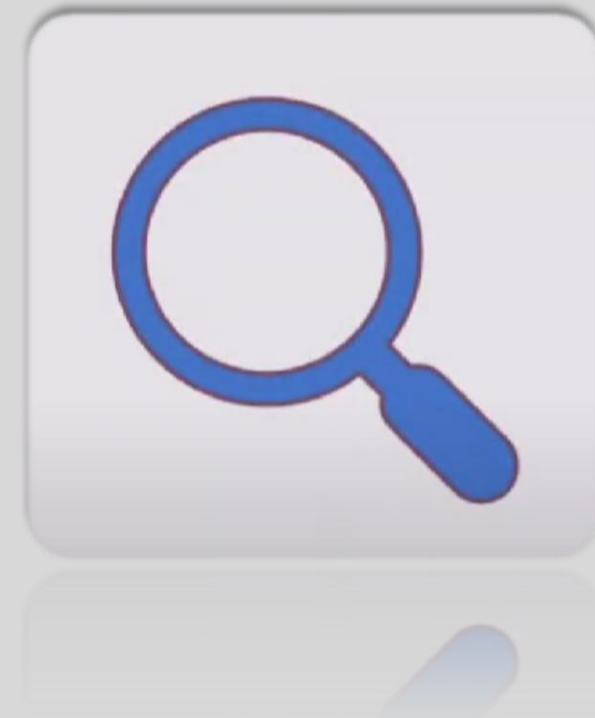
WHAT OUR PROJECT AIMS TO ACHIEVE?

- With the advancement of technology, we can now use high build libraries to derive meaningful insights from social media.
- What could be better than to use the technology to analyze people's response to certain products in order to get the overall acceptance of products in the market.
- The best way to achieve this is through comments of youtube.

ENTITIES OF THE PROJECT

Entities of the project are the comments and transcripts of the following videos. Further for a better understanding of the project we include the number of likes, comments, time of the comment and user-link.

- Twice music video
<https://www.youtube.com/watch?v=i0p1bmr0EmE>
- Iphone 14 pro review
<https://www.youtube.com/watch?v=SdLShOCvVeM>
- Black adam movie trailer
<https://www.youtube.com/watch?v=X0tOpBuYasI>
- NBA Game highlights
<https://www.youtube.com/watch?v=k9aVDS8WOG0&t=2s>



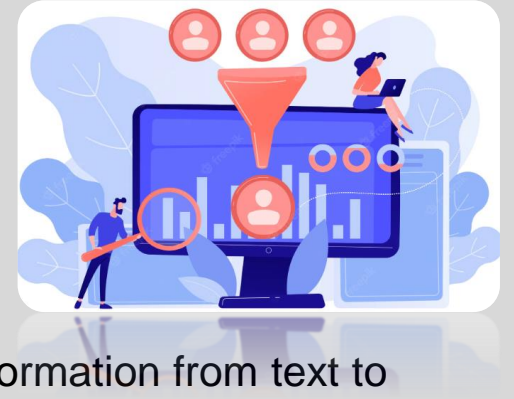


STEPS INVOLVED IN PROJECT





DATA EXTRACTION



- Sentiment analysis is a natural language processing task that involves extracting information from text to determine whether it is positive, negative, or neutral.
- In the context of sentiment analysis, data extraction typically involves identifying and extracting the words or phrases in the text that are most relevant for determining the sentiment of the text. This can involve a variety of techniques, such as part-of-speech tagging, named entity recognition, and keyword extraction.
- The extracted data is then used to calculate various metrics, such as the overall sentiment of the text, the strength of the sentiment, and the subjectivity of the text.
- These metrics can be used to classify the text as positive, negative, or neutral, or to identify the sentiment of individual sentences or phrases within the text

```
In [*]: # For data extraction from youtube ie getting comments from youtube
```

```
from youtube_comment_scraper_python import *  
import pandas as pd
```

```
link = input("Input links:")  
saved = input("Output name:")  
youtube.open(link)
```

```
response = youtube.video_comments()  
all_data = []  
for i in range(0, 5): # It will scroll 10 times  
    response = youtube.video_comments()  
    data = response['body']  
    all_data.extend(data)  
df = pd.DataFrame(data)  
df.to_csv(saved)
```

Starting DataKund...

Input links:https://www.youtube.com/watch?v=k6rc4CS1TTA

Output name:argentinavscroatia.csv



Lens 2 months ago

I honestly think the Dynamic Island is just another one of the small marketing schemes that apple does that works so well. Even though there is still a notch, this new feature made it look new for most people.

👍 27K 🗨️ Reply

▼ 487 replies



Lars Hofman 1 day ago

Many reviewers bash the always on display for being too bright and mistaken it for still being on. I switched from android to the iPhone 14 Pro space black and in combination with the Omber leather case and the standard background with the always on display, the phone looks really gorgeous on my desk. Give it a chance to get used to!

👍 2 🗨️ Reply



Buhdee TV 1 day ago

I'm interested about the E-Sim since it's a new feature on the iPhone I'm curious how will it affect when you travel to another country. Since it's only U.S does that mean your carrier must include whichever country you're going to without being charged?

👍 5 🗨️ Reply

▼ 2 replies



Samuel Eggleton 9 days ago

Never forget the LG V10. Honestly my favorite smartphone I've ever owned.

👍 4 🗨️ Reply



Kuba 11 days ago

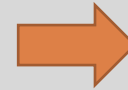
I don't understand why they removed noise cancellation since iPhone 13.... It was such a cool feature. Every new phone instead of iPhones has that.


```
In [89]: #Libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import os

# Import functions for data preprocessing & data preparation
from sklearn.preprocessing import LabelEncoder
from sklearn.utils import resample
from sklearn.feature_extraction.text import CountVectorizer
from nltk.sentiment.vader import SentimentIntensityAnalyzer

from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
from nltk.stem import PorterStemmer, LancasterStemmer
from nltk.stem.snowball import SnowballStemmer
from nltk.corpus import stopwords
from nltk.corpus import wordnet
import string
from string import punctuation
import nltk
import re
```

READING THE DATA FROM CSV FILE



IMPORTING ALL THE LIBRARIES

```
In [90]: data = pd.read_csv('comments.csv')
data.columns
data1=data.drop(['Unnamed: 0', 'Likes', 'Time', 'user', 'UserLink'],axis=1)
data1
```

Out[90]:

	Comment
0	Love how Dr. Fate's design looks and how cool ...
1	I can't get over how good everything looks. Dr...
2	Really hoping that this can save DC's movie un...
3	U cant deny how good this looks.Now if they ca...
4	From this trailer, I have a feeling that this ...
...	...
275	I want to see this. It may be one of his most ...
276	wow thats very amazing. I can't wait to see.
277	Doctor Fate is why i'm watching
278	This looks fire. DC looks like they stepping t...
279	Shazam : "I don't want fight you Black Adam."B...

280 rows × 1 columns





DATA CLEANING AND SENTIMENT ANALYSIS


```
In [91]: nltk.download('vader_lexicon')
sentiments = SentimentIntensityAnalyzer()
data1["Positive"] = [sentiments.polarity_scores(i)["pos"] for i in data1["Comment"]]
data1["Negative"] = [sentiments.polarity_scores(i)["neg"] for i in data1["Comment"]]
data1["Neutral"] = [sentiments.polarity_scores(i)["neu"] for i in data1["Comment"]]
data1["Compound"] = [sentiments.polarity_scores(i)["compound"] for i in data1["Comment"]]
score = data1["Compound"].values
sentiment = []
for i in score:
    if i >= 0.05 :
        sentiment.append('Positive')
    elif i <= -0.05 :
        sentiment.append('Negative')
    else:
        sentiment.append('Neutral')
data1["Sentiment"] = sentiment
data1.head()
```

```
[nltk_data] Downloading package vader_lexicon to
[nltk_data] C:\Users\Asus\AppData\Roaming\nltk_data...
[nltk_data] Package vader_lexicon is already up-to-date!
```

Out[91]:

	Comment	Positive	Negative	Neutral	Compound	Sentiment
0	Love how Dr. Fate's design looks and how cool ...	0.384	0.000	0.616	0.8910	Positive
1	I can't get over how good everything looks. Dr...	0.153	0.000	0.847	0.6801	Positive
2	Really hoping that this can save DC's movie un...	0.375	0.000	0.625	0.9216	Positive
3	U cant deny how good this looks.Now if they ca...	0.302	0.049	0.649	0.9262	Positive
4	From this trailer, I have a feeling that this ...	0.131	0.000	0.869	0.4416	Positive

ELIMINATING ALL THE STOP WORDS



VADER is able to detect the polarity of sentiment (how positive or negative) of a given body of text when the data being analysed is unlabelled.

```
In [54]: # For removing stopwords ie most used words like in ,an ,the, on etc
nltk.download('stopwords')
stop_words = stopwords.words('english')
porter_stemmer = PorterStemmer()
lancaster_stemmer = LancasterStemmer()
snowball_stemmer = SnowballStemmer(language="english")
lizr = WordNetLemmatizer()
```

```
[nltk_data] Downloading package stopwords to
[nltk_data] C:\Users\Asus\AppData\Roaming\nltk_data...
[nltk_data] Package stopwords is already up-to-date!
```

```
In [55]: # For text processing like removing special characters ,removing new line characters,converting from uppercase to lower case
def text_processing(text):
    # convert text into lowercase
    text = text.lower()

    # remove new line characters in text
    text = re.sub(r'\n',' ', text)

    # remove punctuations from text
    text = re.sub('[%s]' % re.escape(punctuation), "", text)

    # remove references and hashtags from text
    text = re.sub("^a-zA-Z0-9$","", text)

    # remove multiple spaces from text
    text = re.sub(r'\s+', ' ', text, flags=re.I)

    # remove special characters from text
    text = re.sub(r'\W', ' ', text)

    text = ' '.join([word for word in word_tokenize(text) if word not in stop_words])

    # stemming using porter stemmer from nltk package - msh a7sn 7aga - momken: lancaster, snowball
    # text=' '.join([porter_stemmer.stem(word) for word in word_tokenize(text)])
    # text=' '.join([lancaster_stemmer.stem(word) for word in word_tokenize(text)])
    # text=' '.join([snowball_stemmer.stem(word) for word in word_tokenize(text)])

    # lemmatizer using WordNetLemmatizer from nltk package
    text=' '.join([lizr.lemmatize(word) for word in word_tokenize(text)])

    return text
```



```
In [97]: le = LabelEncoder()  
data_copy['Sentiment'] = le.fit_transform(data_copy['Sentiment'])
```

```
In [98]: processed_data = {  
    'Sentence':data_copy.Comment,  
    'Sentiment':data_copy['Sentiment']  
}  
  
processed_data = pd.DataFrame(processed_data)  
processed_data.head()
```

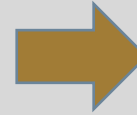
```
Out[98]:
```

	Sentence	Sentiment
0	love dr fate design look cool scene look power...	2
1	get good everything look dr fate magic cyclone...	2
2	really hoping save dc movie universe looking n...	2
3	u cant deny good looksnow follow rest movie go...	2
4	trailer feeling movie going one movie would ne...	2

```
In [99]: processed_data['Sentiment'].value_counts()
```

```
Out[99]: 2    205  
        1     39  
        0     36  
        Name: Sentiment, dtype: int64
```

DATA SAMPLING AND PREPARING FINAL DATA SET



ENCODING

```
In [101]: df_neutral = processed_data[(processed_data['Sentiment']==1)]  
df_negative = processed_data[(processed_data['Sentiment']==0)]  
df_positive = processed_data[(processed_data['Sentiment']==2)]  
  
# upsample minority classes  
df_negative_upsampled = resample(df_negative,  
                                replace=True,  
                                n_samples= 205,  
                                random_state=42)  
  
df_neutral_upsampled = resample(df_neutral,  
                                replace=True,  
                                n_samples= 205,  
                                random_state=42)  
  
# df_positive_upsampled = resample(df_positive,  
#                                 replace=True,  
#                                 n_samples= 58,  
#                                 random_state=42)  
  
# Concatenate the upsampled dataframes with the neutral dataframe  
final_data = pd.concat([df_negative_upsampled,df_neutral_upsampled,df_positive])
```

```
In [102]: final_data['Sentiment'].value_counts()
```

```
Out[102]: 0    205  
        1    205  
        2    205  
        Name: Sentiment, dtype: int64
```

```
In [103]: corpus = []  
for sentence in final_data['Sentence']:  
    corpus.append(sentence)  
corpus[0:5]
```



ALGORITHM USED



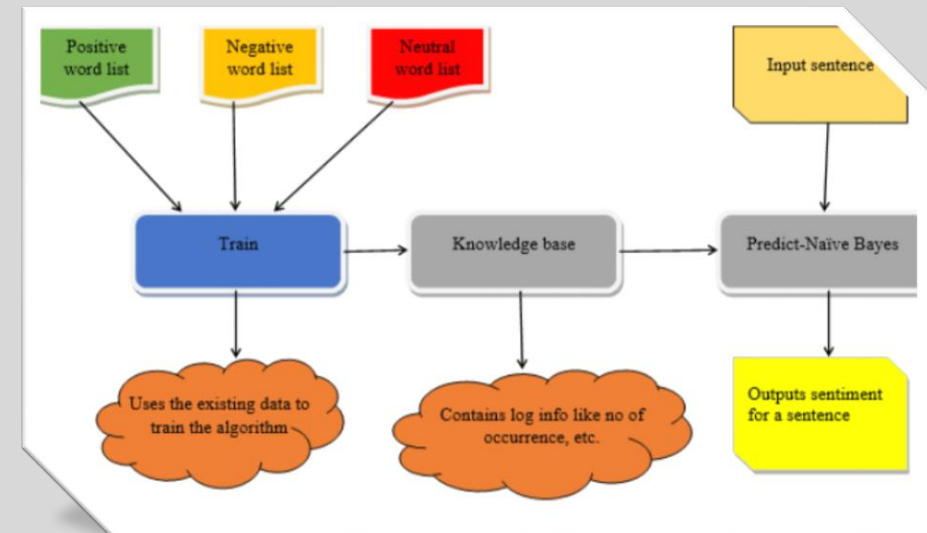
NAÏVE BAYES



ALGORITHM

WHY NAÏVE BAYES??

We have applied Naïve Bayes algorithm because One of the advantages of using a Naive Bayes classifier for sentiment analysis is that it is relatively simple and easy to implement, yet it can still achieve good performance on many sentiment analysis tasks. Additionally, the model can handle large amounts of data and can be trained efficiently, making it well-suited for use in production environment



```
In [104]: from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=1500)
X = cv.fit_transform(corpus).toarray()
y = final_data.iloc[:, -1].values
```

```
In [105]: from sklearn.naive_bayes import GaussianNB
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=0)
classifier = GaussianNB()
classifier.fit(X_train, y_train)
```

```
Out[105]: ▾ GaussianNB
GaussianNB()
```

```
In [106]: from sklearn.metrics import confusion_matrix, accuracy_score
y_pred = classifier.predict(X_test)
cm = confusion_matrix(y_test, y_pred)
cm
```

```
Out[106]: array([[58,  0,  0],
                 [ 0, 70,  0],
                 [11,  1, 45]], dtype=int64)
```

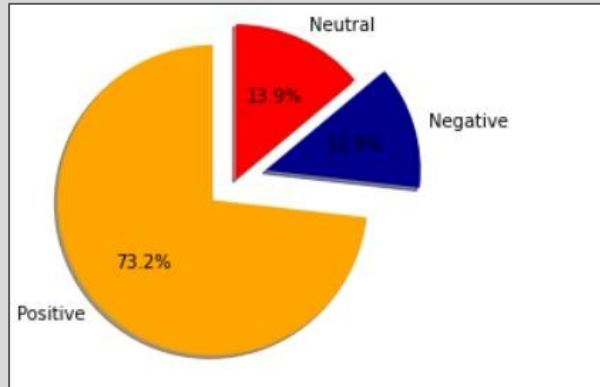
```
In [107]: nb_score = accuracy_score(y_test, y_pred)
print('Accuracy', nb_score*100)
```

Accuracy 93.51351351351352

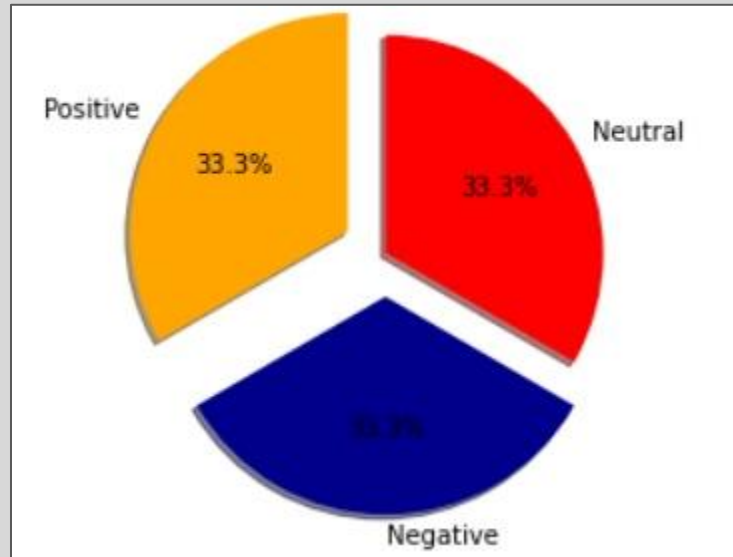


RESULT

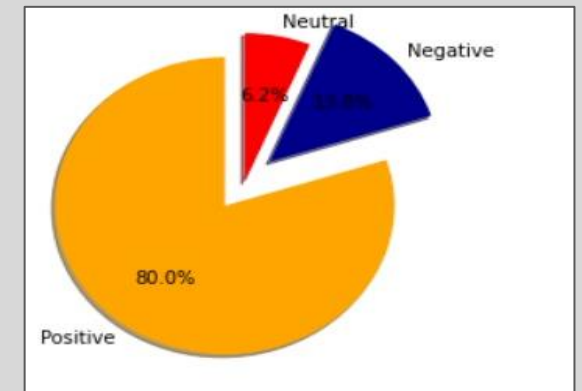
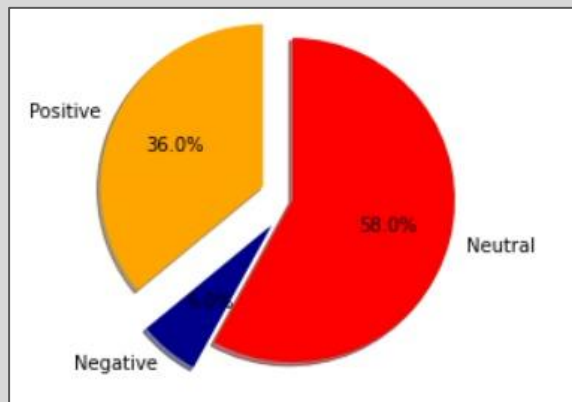
Pie chart depicting the number of comments of positive ,negative and neutral sentiment in each dataset



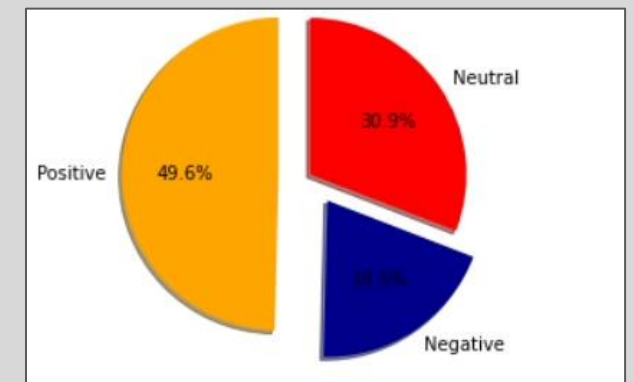
iPhone 14 Pro Review



Tvice Music Video



Movie Trailer



NBA Game Highlights



RESULT



```
In [129]: nb_score = accuracy_score(y_test, y_pred)
          print('Accuracy',nb_score*100)
```

Accuracy 94.87179487179486

IPHONE REVIEW

```
nb_score = accuracy_score(y_test, y_pred)
print('Accuracy',nb_score*100)|
```

Accuracy 85.0

NBA GAME HIGHLIGHTS

TWICE MUSIC

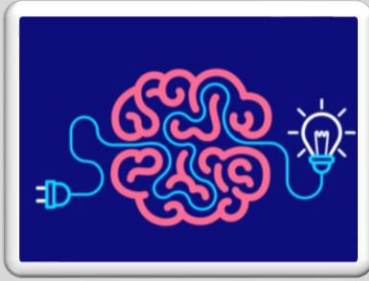
```
In [234]: nb_score = accuracy_score(y_test, y_pred)
          print('Accuracy',nb_score*100)
```

Accuracy 90.0

MOVIE TRAILER

```
In [107]: nb_score = accuracy_score(y_test, y_pred)
          print('Accuracy',nb_score*100)
```

Accuracy 93.51351351351352



CONCLUSION



- Sentiment analysis has the potential to provide valuable insights into the attitudes and opinions of YouTube users towards specific videos, channels, and brands.
- This information can be useful for content creators, advertisers, and other stakeholders to better understand and engage with their audience.
- Additionally, the use of sentiment analysis on YouTube comments can help improve the overall user experience on the platform by automatically identifying and flagging inappropriate or offensive comments.
- It can also aid in the enforcement of community guidelines and policies, ensuring that the platform remains a safe and welcoming space for all users.
- Overall, sentiment analysis on YouTube comments is a promising and rapidly-developing technology that has the potential to provide valuable insights and improve the user experience on the platform.

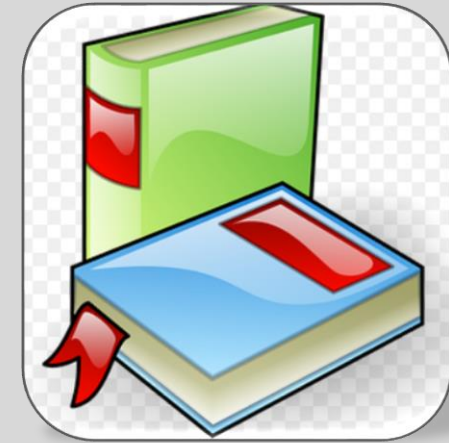
FUTURE SCOPE



- The use of sentiment analysis on YouTube comments has the potential to provide valuable insights into the opinions and attitudes of users towards the content on the platform.
- This information can be useful for content creators, advertisers, and other stakeholders to better understand and engage with their audience.
- In the future, sentiment analysis of YouTube comments may become even more advanced and sophisticated, with the ability to handle a wider range of languages, dialects, and cultural differences.
- It may also become more adept at handling sarcasm and other forms of irony, as well as the inherent subjectivity of sentiment.
- Additionally, the integration of sentiment analysis with other tools, such as analytics and engagement metrics, may provide even greater insights into the attitudes and behaviors of YouTube users



REFERENCES



- <https://monkeylearn.com/sentiment-analysis/>
- <https://devopedia.org/sentiment-analysis>
- <https://monkeylearn.com/blog/opinion-mining/>
- <https://www.saspublishers.com/media/articles/SJET31B71-75.pdf>
- [https://www.google.co.in/books/edition/Opinion Mining and Sentiment Analysis/XQswsqLLKrEC?hl=en&gbpv=1&dq=history+of+sentiment+analysis&printsec=frontcover](https://www.google.co.in/books/edition/Opinion_Mining_and_Sentiment_Analysis/XQswsqLLKrEC?hl=en&gbpv=1&dq=history+of+sentiment+analysis&printsec=frontcover)
- <https://www.kaggle.com/code/arunkumar1809/youtube-comments-sentiment-analyser>
- <https://www.nltk.org/>
- “New Sentiment analysis” article by Antony Samuels of University of Southern California. Article on “Predicting like-ratio on YouTube videos using sentiment analysis on comments” in 2018 by Martin Hyberg and Isaacs.
- Sentiment Analysis on Youtube Comments to Predict sentiments on Youtube Video Like Proportions ISAC LORENTZ & GURJIWAN SINGH.