Detecting child unsafe videos using transcripts

Shritishma Reddy & Pratik Kamble

April, 16th 2019

1 Abstract

In this project, we propose an end-to-end pipeline for the detection of child unsafe videos using video transcripts. Given the spread of digital media and increase in media consumption by kids (ages 3-12) over the past decade, there is a great need for the filtering of unsafe content found within reach of said age group. While traditional parental systems do exist, this would be an online scalable system, that would only benefit from the increasing video consumption.

2 Introduction

This pipeline has three phases:

- Data acquisition: Collection of a holistic video-transcript dataset representative of children viewing habits and trends.
- Unsafe pattern detection: Analyze speech patterns and clustering hotspot zones to form a timeline representation.
- Video Classification: Classification of videos into unsafe types, if unsafe, viz. religious, abusive, drug use, etc.

2.1 Data Acquisition

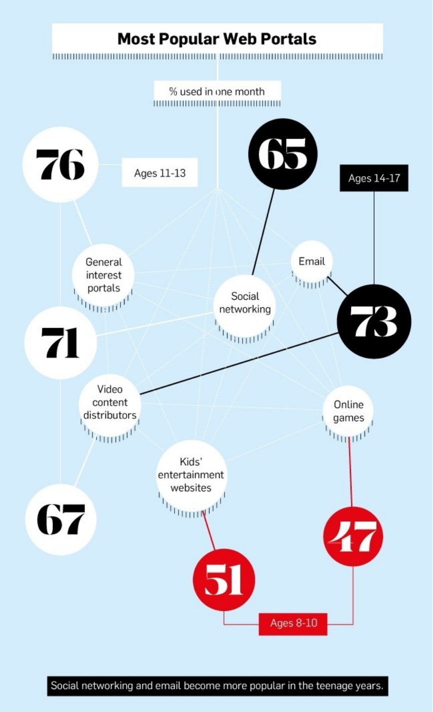
The data acquisition process depends a lot on the type of genre distribution. A study [**?**] has found that children consume just over three hours of media, as of 2015; the numbers having since then gone up further.

The time spent with on screen media dramatically increases from the toddler to preschool to school-age years. Children under two have a screen time average of 53 minutes per day. This increases to almost two and a half hours per day among two to four year old and almost three hours for kids in the five to eight year old range. By age eight, 96% of children have watched TV, 90% have used a computer, 81% have played console video games, and 60% have played games or used apps on a portable device. Thus by order of data-source, we should have a dataset representative mainly of TV show content, followed by YouTube and lastly, game video data.

Luckily, all video footage from various sources can be obtained from YouTube, along with transcript data. TV show snippets, actual online vlog and other consumable media, as well as video game footage can be found on the streaming platform.

2.2 Unsafe Pattern Detection

Once we have the annotated data, we have to run through it and classify according to unsafe-ness. Once criterion that is of importance is categories that may influence minds may be quite different from ones for adolescents or adults.



Social networking and email become more popular in the teenage years.

Figure 1: Major content sources and distribution

As seen from the figure , the main categories of media streaming are entertainment websites, omline games, email, social networking and general interest portals. Breaking down the type of content found on these sites, the main type of genres are as follows:

- Religion-based hate speech: This occurs in chatrooms and online forums, both high target kid influence
- Color-based hate speech: Depending on race and ethnicity, such sort of speech may be hidden in meaning and harder to detect.
- Foul Language: Kids usually pick up on foul language and the age of such exposure is lower this decade than it ever was.
- Other offensive terminology: Dealing with kids in the lower age bracket, systems need not worry too much about false positives, because of the ease of influence of media on early youth.

2.3 Video Classification

Once the data collection for the transcripts is done, a simple classical model is used in cascade, to first analyze and mark scenes and zones in the time frame with dangerous content.

This meta information is then passed to a zonal mapping pipeline, that, depending on how many zones of what clusters there are, estimates a score for each category mentioned prior. Finally, depending on the thresholding for each genre distribution across ages, a weighted average will judge the videos overall scores.

3 Annotation and labeling

The first step in the pipeline is the data collection and annotation. It is important to keep in mind the divisons and genre distribution amidst children and have a dataset representative of that. We are collecting 5 videos in each positive category, approximately of 5 minutes each, unless the genre specifies (eg, vlogs are usually longer). The videos should be at such an age genre balance that younger children usually are pressured into by peers, and boundary videos are easy gateways to explicit content.

This gives us a total of 4 categories of positive videos, for each age group bracket within kids, targetting most common videos in the given set.

$$N = n_{videos} \times n_{categories} \times n_{agegroups} \times m \times \delta \times \alpha$$

Here n_{videos} is 5 per category, $n_{agegroups}$ is 3, m being the average length per video of selection. Now the average rate of speech is δ. This may vary across various demographics, but we choose the global average, 110 words per minute. Let's assume, since we are targetting positive samples to seed the dataset, there is a 20% conversion rate, α. This means that 1 minute in a positive truth, 5 minute video contains positive samples of the content we're trying to flag. This is a fair assumption, and goes in line with average statistics [**?**].

3.1 Raw Transcript Collection

After aggregating the various sources of media and content, our dataset consisted of 60 video subtitle files (*.srt). We have used a multi-threaded procedure to get transcripts for our videos. This saved a lot of time in video collection. These files were formatted using the srt standard and had to be converted into a 'DataFrame' for annotation. After cleaning invalid records and repeat rows, we were left with around 7000 observations.

3.2 SRT Parsing and Dataset Aggregation

Once we had a collection of formatted transcripts, they had to be aggregated and labeled. Each record of the dataset is of the form:

```
 Vedio_Id Genre       start     end       text
0  3tgZ...      0       0.00   3.71  mr. mine you say babe ...
1  3tgZ...      0       3.72  22.79  mr. mine babe what’s d...
2  3tgZ...      0      22.79  22.80  of a dynamic punch mr. line
3  3tgZ...      0      22.80  28.40  of a dynamic punch you’ll...
4  3tgZ...      0      28.40  28.41  you’ll get stronger your
```

Every sample has a start and end marker, the genre for the video, and other meta information (as seen above). The timestamps and genre will help correlate the overall accuracy of the system with certain trends and patterns across genres. Additionally, some inference can be drawn as to the average "unsafe" duration of a video and the subsequent variation with genres.

Now, this annotated data is passed onto the model. The preliminary embedding and tagging of every sentence of a video is handled by the same. The results from said model are then passed onto an aggregator; it takes into account sentence based transcript data, augments it with additional meta information about the video and then makes a prediction regarding the safety of the video in terms of child consumption.

3.2.1 Youtube Dataset Statistics:

- 77% of our dataset is "Clean".
- 12% of our dataset is "Abusive"
- 11% of our dataset is "Hate"
- Average running time of videos in our dataset is 4.5 mins.

4 Model - Sentence labeling

The model choice we ultimately went with was a bi-LSTM model with a simple embedding layer (from Keras). We used sequence model architectures i.e., BLSTM with character grams as inputs for training. BLSTM and LSTM models usually consists of three sequential layers:

- Embedding Layer
- Bidirectional LSTM
- Fully Connected Layer

To compare the benefits of this model against other classical/hybrid approaches, we trained a model using classical approaches.[**?**]. Across the board, the accuracy of the chosen model was better than all other models considered (*a comparitive analysis has been provided in the* ***results*** *section.*) A small improvement to the pipeline could be considered at this stage, viz. an ensemble approach with a mix of this bi-lstm model and a more classical approach.

4.1 Structure

- To deal with complex structured conversation data, we plan to use word level trigrams to represent each conversation.
- Sequence of trigrams sent through an embedding layer which takes one-hot vector of each character trigram in the sentence and learns the lower dimension representation for it.

4.2 Functionality & Reasoning

- Bi-Directional LSTMs have the ability to capture much richer sequential patterns from both directions of a sequence.
- Each character trigram from the previous embedding layer is fed as a sequence of trigrams, to the BLSTM layer.
- The output of the BLSTM layer is given as input to a Fully Connected (FC) layer which models the interactions between these features.
- The final softmax node in the FC layer outputs the probability of the sentence belonging to the inappropriate class.

The initial training and validation was done on the Davidson dataset. This model gave excellent results, resulting in 99% training accuracy and 89% test accuracy on the baseline dataset. Results on the test dataset and comparison across all models is presented in the *results* section.

5 Zonal clustering and overall classification

All the transcript data has been labeled as follows:

- $\begin{bmatrix}1 & 0 & 0\end{bmatrix}$ Abusive
- $\begin{bmatrix}0 & 1 & 0\end{bmatrix}$ Hate
- $\begin{bmatrix}0 & 0 & 1\end{bmatrix}$ Clean

This data is then mapped into a timeline heat map to indicate stronger zones of the video that may be unsafe. As part of the further work, this is the step wherein more meta information, age group, region, etc would be added so as to provide a better guess as part of the second phase (*something like an additional FC layer, with the transcript heatmap as a feature subset*).

Now, as part of the first post processing step, we take the labeled video timeline and calculate the percentage of the video that's ”unsafe”. This is then cross-referenced with the genre based averages and an appropriate genre is determined. Once this part has been done, each genre has a statical threshold for it's unsafeness (again, precalculated). If the video then falls beyond such a threshold then, it is classified as unsafe. Additionally, it is run across other averages as well, and a score is given depending on the sensitivity of the system (*higher sensitivity would lead to more false positives*).

These are then the final labels that are taken into account. As part of the results section, we have compared the end-to-end accuracy with ground truth obtained from manual viewing and judgment.

6 Results

6.1 Data preparation & model training

6.1.1 Data preparation

We have used Davidson dataset and 20% of data corresponding to each label of our YouTube dataset for training. The Davidson dataset is highly biased towards abusive data and contains very less hate and clean data. In order to make our training dataset bias free, we use about 4500 clean, 2000 hate and 1500 abusive sentences. Since,clean sentences are not as structured as hate and abusive sentences, we use higher proportion of clean data as compared to the other two classes. The Davidson dataset primarily consists of tweets. Hence, urls, hashtags, punctuation, user mentions etc are removed initially. All the sentences are then lemmatized and stemmed. Stopwords are also removed from all the sentences. Since, we are using a tfidf based tokeniser, removing stop words improves the rank of relevant words in the tokeniser. We use this clean data to train our tokeniser. From the above preprocessed dataset, we take all the words and train a glove model and get glove embeddings for the words in our training data.

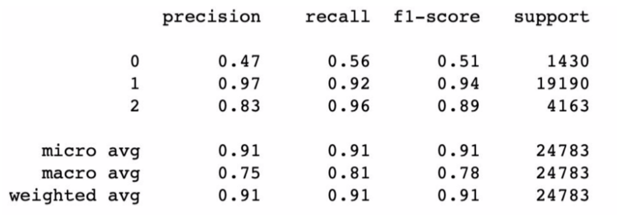
6.1.2 Model Training

Classical Approach:
This model has been proposed by ?Automated Hate Speech Detection and the Problem of Offensive Language." ICWSM.
Features

- We lowercased each sentence and stemmed it using the Porter stemmer, then create bigram, unigram, and trigram features, each weighted by its TF-IDF.
- To capture information about the syntactic structure we use NLTK (Bird, Loper, and Klein 2009) to construct Penn Part-of-Speech (POS) tag unigrams, bigrams, and trigrams
- To capture the quality of each tweet we use modified Flesch-Kincaid Grade Level and Flesch Reading Ease scores, where the number of sentences is fixed at one.
- We also use a sentiment lexicon designed for social media to assign sentiment scores to each tweet (Hutto and Gilbert 2014).
- We also include binary and count indicators for features for the number of characters, words, and syllables in each sentence.

Model

- We first use a logistic regression with L1 regularization to reduce the dimensionality of the data.
- We tested each model using 5-fold cross validation, holding out 10% of the sample for evaluation to help prevent overfitting.
- After using a grid-search to iterate over the models and parameters we find that the Logistic Regression and Linear SVM tended to perform significantly better than other models.
- We decided to use a logistic regression with L2 regularization for the final model.
- We trained the final model using the Davidson dataset and used it to predict the label for each sentence of Youtube Transcript Dataset.
- We use a one-versus-rest framework where a separate classifier is trained for each class and the class label with the highest predicted probability across all classifiers is assigned to each tweet.
- All modeling was performing using scikit-learn.



```
              precision    recall  f1-score   support

           0       0.47      0.56      0.51      1430
           1       0.97      0.92      0.94     19190
           2       0.83      0.96      0.89      4163

   micro avg       0.91      0.91      0.91     24783
   macro avg       0.75      0.81      0.78     24783
weighted avg       0.91      0.91      0.91     24783
```

Figure 2: Classical training

The accuracies of the classical model when tested on our dataset was around 20%. The classical features fail to capture contextual data and sentence structure efficiently.

Bi-LSTM Model:

Model

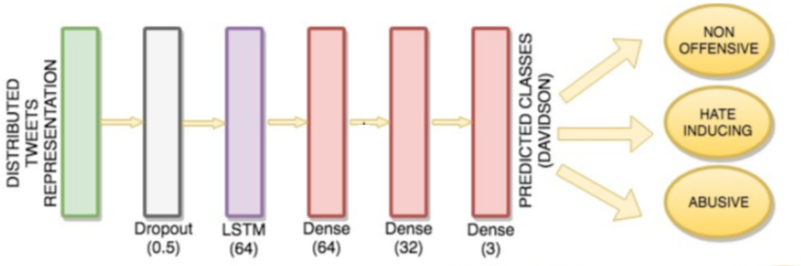
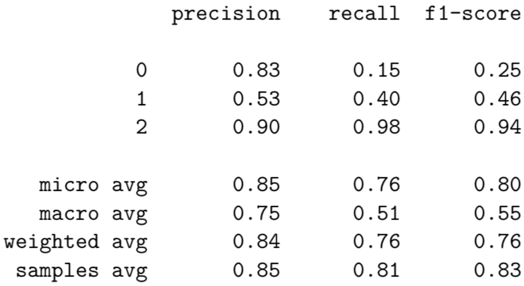


Figure 3: Architecture

- We use an initial Embedding Layer to which we give our trained glove embeddings as weights.
- Bi-Directional LSTMs have the ability to capture much richer sequential patterns from both directions of a sequence.
- The output of the BiLSTM layer is given as input to a Fully Connected (FC) layer which models the interactions between these features.
- Then 2 Dense layers with appropriate drop out ratios are added sequentially.
- The final soft-max node in the FC layer outputs the probability of the sentence belonging to the inappropriate class

The accuracies of the bilstm model when tested on our dataset was around 84%.



```
              precision    recall  f1-score

           0       0.83      0.15      0.25
           1       0.53      0.40      0.46
           2       0.90      0.98      0.94

   micro avg       0.85      0.76      0.80
   macro avg       0.75      0.51      0.55
weighted avg       0.84      0.76      0.76
 samples avg       0.85      0.81      0.83
```

Figure 4: Accuracies

6.2 Final Choice & Testing

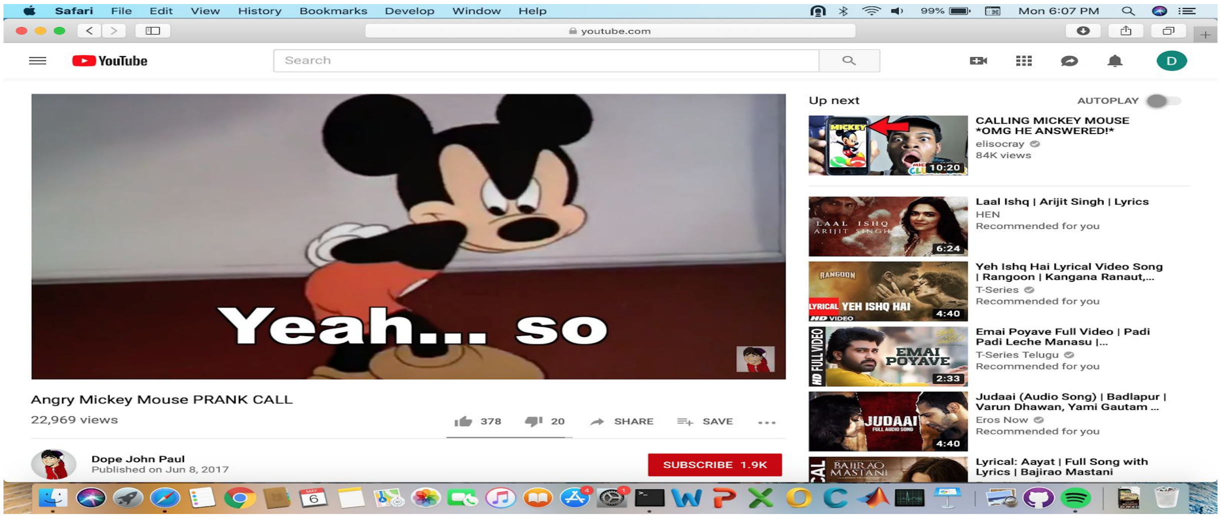
Since, the bilstm model performs better than the classical model, we proceed using the bilstm model for our final script.

6.3 Final Pipeline

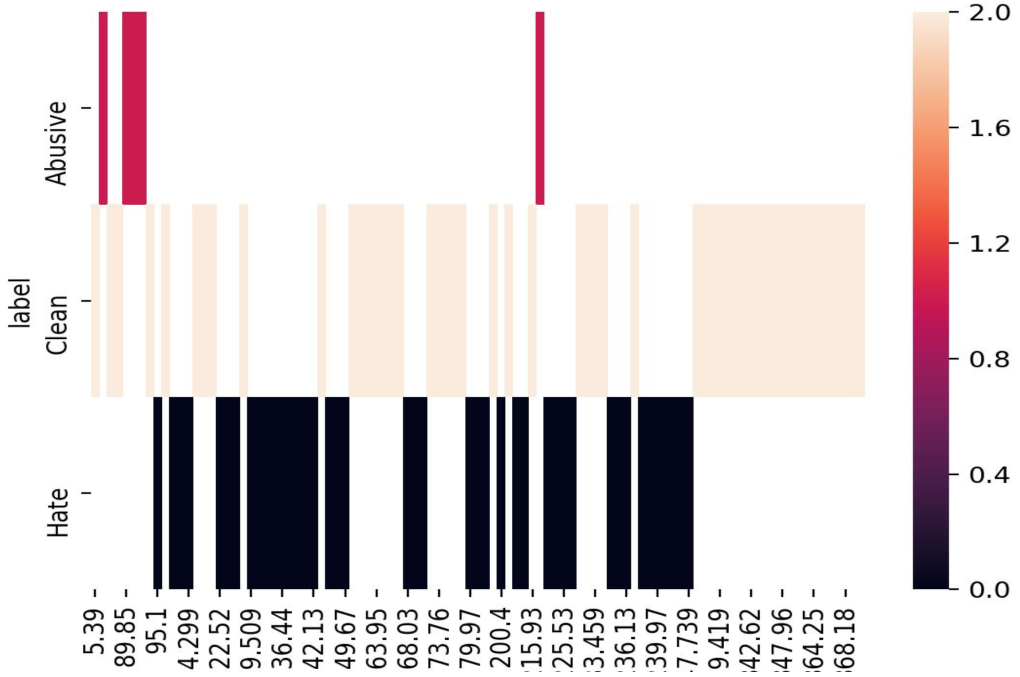
The final script that we have written takes the youtube link as input and shows a histogram of distribution of abusive, hate and clean labels for all the transcripts extracted. It also outputs a final heat map showing distribution of abusive, hate and clean subtitles vs time.

6.3.1 Results

Video:


Safari File Edit View History Bookmarks Develop Window Help
99% Mon 6:07 PM
youtube.com
YouTube
Search
Yeah... so
Angry Mickey Mouse PRANK CALL
22,969 views
378
20
SHARE
SAVE
Dope John Paul
Published on Jun 8, 2017
SUBSCRIBE 1.9K
Up next
AUTOPLAY
CALLING MICKEY MOUSE *OMG HE ANSWERED!*
elisocray
84K views
10:20
Laal Ishq | Arijit Singh | Lyrics
HEN
Recommended for you
6:24
Yeh Ishq Hai Lyrical Video Song | Rangoon | Kangana Ranaut,...
T-Series
Recommended for you
4:40
Emai Poyave Full Video | Padi Padi Leche Manasu |...
T-Series Telugu
Recommended for you
2:33
Judaai (Audio Song) | Badlapur | Varun Dhawan, Yami Gautam ...
Eros Now
Recommended for you
4:40
Lyrical: Aayat | Full Song with Lyrics | Bajirao Mastani

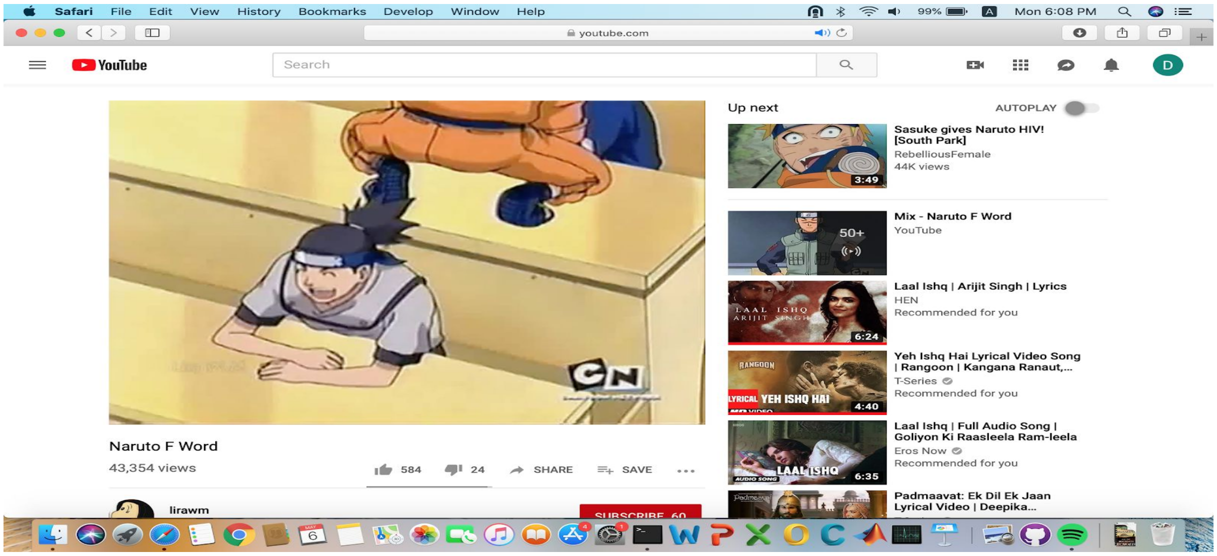
Transcript Sentences Classification:



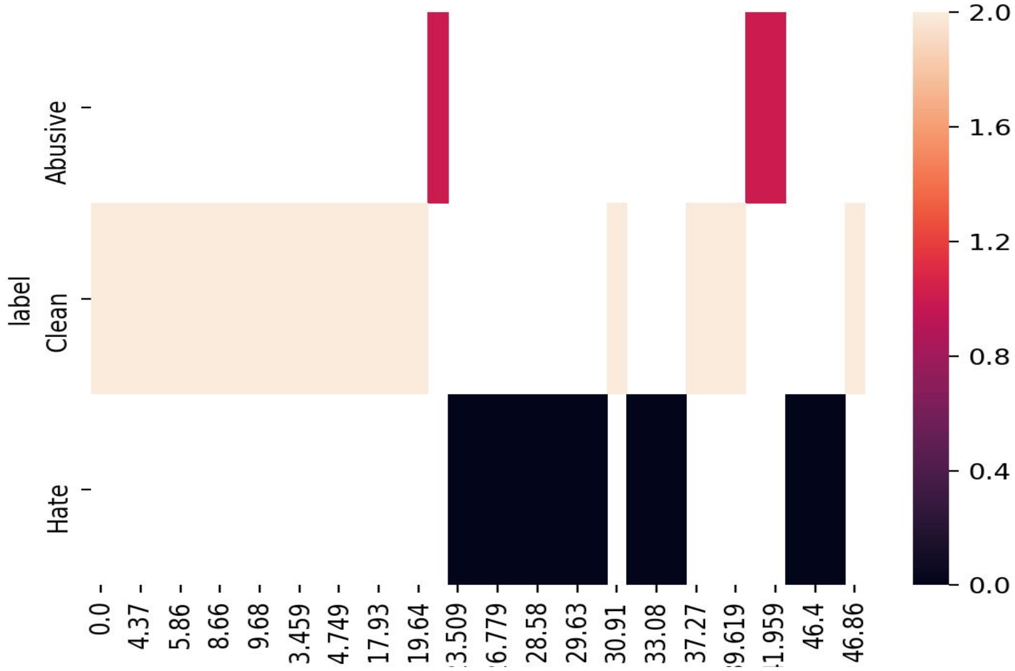
```
Clean       52
Hate        42
Abusive      5
Name: label, dtype: int64
```

Histogram of labels :

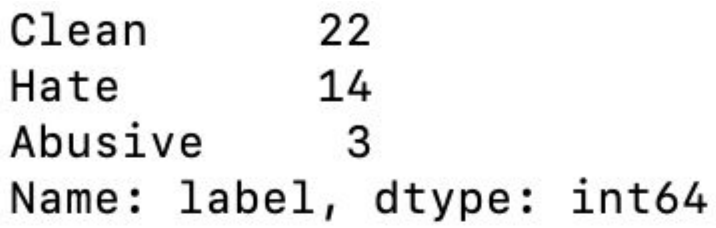
Video:


Safari File Edit View History Bookmarks Develop Window Help
99%
Mon 6:08 PM
youtube.com
YouTube
Search
D
Up next
AUTOPLAY
Sasuke gives Naruto HIV! [South Park]
RebelliousFemale
44K views
3:49
Mix - Naruto F Word
YouTube
50+
Laal Ishq | Arijit Singh | Lyrics
HEN
Recommended for you
6:24
Yeh Ishq Hai Lyrical Video Song | Rangoon | Kangana Ranaut,...
T-Series
Recommended for you
4:40
Laal Ishq | Full Audio Song | Goliyon Ki Raasleela Ram-leela
Eros Now
Recommended for you
6:35
Padmaavat: Ek Dil Ek Jaan Lyrical Video | Deepika...
Naruto F Word
43,354 views
584
24
SHARE
SAVE
lirawm
SUBSCRIBE 60

Transcript Sentences Classification:

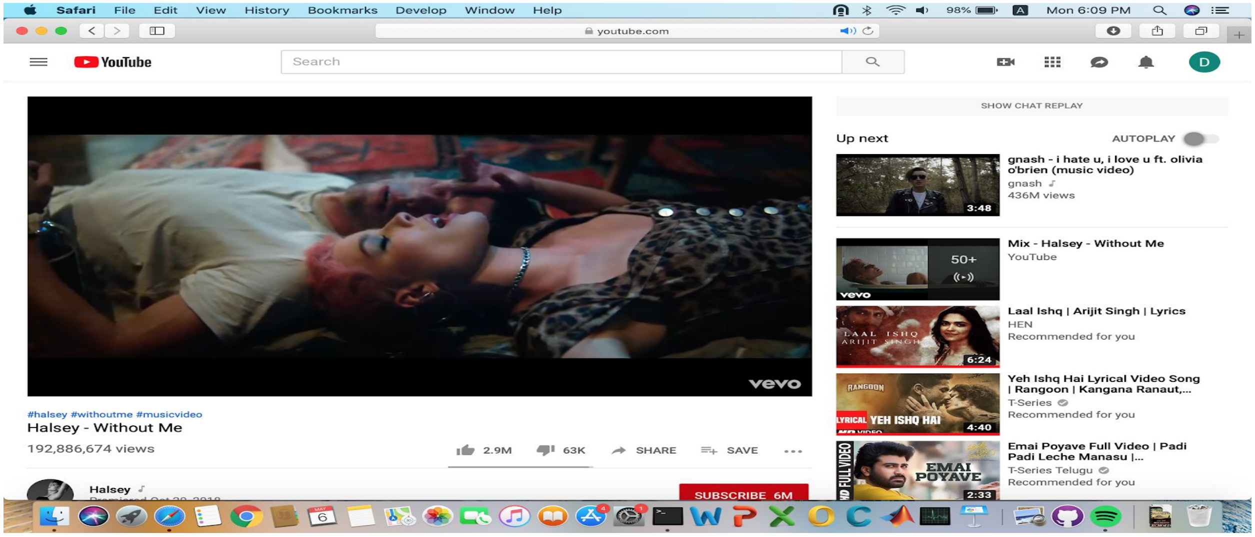


Histogram of labels:



```
Clean       22
Hate        14
Abusive      3
Name: label, dtype: int64
```

Video:



gnash - i hate u, i love u ft. olivia o'brien (music video)
gnash
436M views

Mix - Halsey - Without Me
YouTube

Laal Ishq | Arijit Singh | Lyrics
HEN
Recommended for you

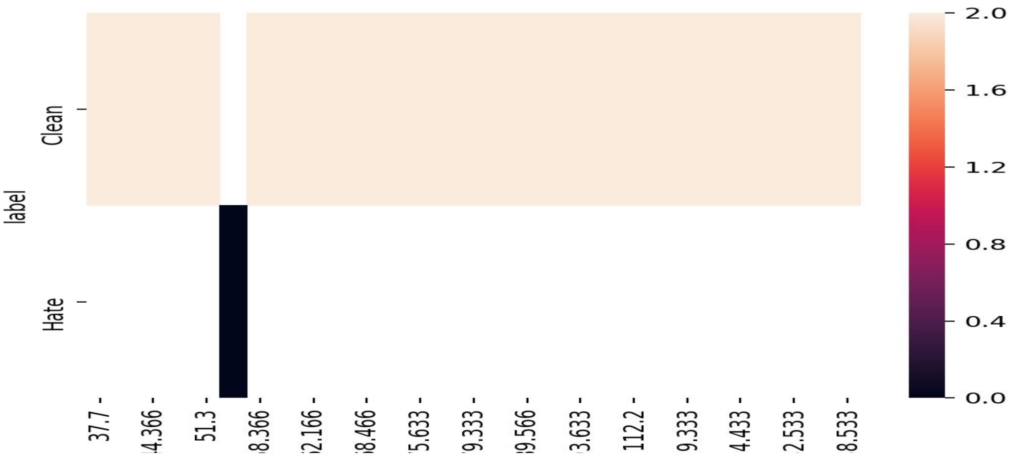
Yeh Ishq Hai Lyrical Video Song | Rangoon | Kangana Ranaut,...
T-Series
Recommended for you

Emai Poyave Full Video | Padi Padi Leche Manasu |...
T-Series Telugu
Recommended for you

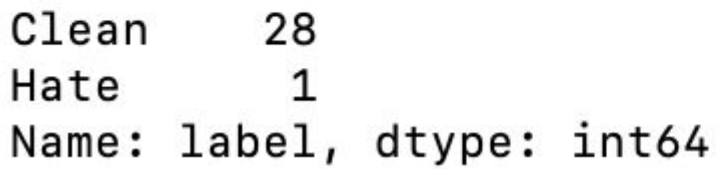
#halsey #withoutme #musicvideo
Halsey - Without Me
192,886,674 views

Halsey

Transcript Sentences Classification:



Histogram of labels:



```
Clean     28
Hate       1
Name: label, dtype: int64
```

7 Conclusion

Thus, with the classification methods of the bi-lstm model with a combination of the classical method gives us the best performance.We successfully analysed youtube videos from their transcripts and produced histograms and heat maps reflecting the transcript profanity.

7.1 Further Work

Future work includes expanding the dataset. BERT Embeddings could be used in the model. Video frames can be collected and multi-modal analysis can be done to make our predictions more reliable. [1].Metadata from Youtube can be collected and used for analysis too.

References

[1] Kaushal, R., Saha, S., Bajaj, P., Kumaraguru, P.: KidsTube: detection, characterization and analysis of child unsafe content promoters on YouTube. In: 2016 14th Annual Conference on Privacy, Security and Trust (PST), pp. 157?164. IEEE (2016)