Big Data Analytics

Sentiment analysis over YouTube comments

Geneviève Masioni

Abstract **2**

Method **3**

Comment retrieval 3

Comment preprocessing 3

Text cleaning 3

Comment classification 4

Sentiment analysis 4

Transformers classifiers explained 5

Training 5

Prediction 6

Comment summarization 6

Results **6**

Limits of the method **7**

Truncated comments because of tensor limitations 7

Lexicons based emotion detection 7

Other problems 7

Challenges **7**

Emoticons 7

Subjectivity and tone 7

Context, polarity and comparisons 8

Multilingual sentiment analysis 8

Conclusion **8**

Abstract

Background

Sentiment analysis (or opinion mining) is a natural language processing technique usually performed on textual data to determine whether a piece of data is positive, negative, or neutral. Traditionally, it allows businesses to monitor sentiment in customer feedback and therefore tailor their products or services to match their customers' needs. This technique is here applied to YouTube comments to help online content creators monitor their audience's opinion without reading all the comments under a video. This is particularly useful for those whose content generates thousands to millions of reactions.

Aims

This project has multiple goals that work towards allowing the user to (1) get an overview of the audience's opinion on a piece of content, (2) roughly know what is discussed in the comment section, (3) spot intense negative emotions and take actions (report or ban toxic viewers).

Method

The key steps of this project are :

- *Data collection* using the Youtube API. Creation of JSON files to design a NoSQL database using Elasticsearch. Each comment is a document and one JSON file is created per video (a reference to the video is stored in the document).
- *Data preprocessing* : text cleaning and comment classification (sentiment, emotion, toxicity and topic) using Tensorflow/ PyTorch.
- *Data analysis* : analysis of the classified comments to answer 3 key questions ;
    - What are the overall sentiment and emotions ?
    - What are people talking about ?
    - Are they any inappropriate comments ? Who is publishing those comments ?
- *Data summarisation* in a Kibana dashboard.

Results

Dashboard sentiment, topics and emotions portrayed under a video's comment section.

Tools

- Youtube API : data collection (video comments).
- NoSQL with Elasticsearch : for its tokenization feature that allows us to search through text.
- Python Transformers package (based on Tensorflow and PyTorch) for text classification using different models (sentiment, emotion, toxicity and topic classification).
- Kibana : data visualisation (final dashboard).
- Jupyter notebook : source code and synthetic explanation of our process.

Method

1. Comment retrieval

We collected publicly available video comments using the Youtube API. A JSON file was created for each video. The files were used to create a NoSQL database in which aach comment represents a document.

Comments were extracted on a multilingual selection of videos. Those videos were selected from channels whose content generates hundreds to thousands of reacomments.

Channel	Language	Size (subscribers)	Category	About
Damon Dominique	English	410 000	Entertainment	Lifestyle
The coding train	English	1,35 Millions	Education	Computer Science tutorials
Answer in Progress	English	754 000	Education	Computer Science experiments
Ben Névert	French	465 000	Education	Public debate on societal issues
Léna Mahfouf	French	2,14 Millions	Entertainment	Lifestyle
Martina D'Antiochia	Spanish	4,07 Millions	Entertainment	Lifestyle
TEDx Talks	English	33,5 Millions	Education	Conferences

Table 1 Selected channels

2. Comment preprocessing

Pre-processing ensures all comments are similarly recognized and extracted in the sentiment analysis process.

Text cleaning

We removed irrelevant information such as links (e.g. spams or self-promotion) and special characters (e.g. “\n”).

It’s important to emphasis that we did not apply the following pre-processing techniques :

- Remove punctuations (e.g. "?, !, ;")
- Remove emojis and hashtags

That is because punctuation, emojis and hashtags are a vital part of social media sentiment analysis because they carry emotion. So we ensured that emojis are not left out of data processing because that could lead to false positives or negatives. Removing those key elements affects the performance of the emotion classifier, as proven by our tests :

Sample	Class	Score
how much for the maple syrup? $20.99? That's ricidulous!!! 😡	**anger**	**0.53**
	joy	0.32
how much for the maple syrup? $20.99? That's ricidulous!!!	anger	0.50
	joy	0.38
how much for the maple syrup $20.99 That's ricidulous	joy	0.51
	anger	**0.43**

Table 2 Impact of punctuation and emojis on the emotion prediction accuracy

3. Comment classification

Sentiment analysis

We would like the content creator to be able to not only know the general sentiment in the comment section but also the emotions that are conveyed, what the audience is broadly talking about and if they are inappropriate comments that require them to take disciplinary actions.

To determine these features, we use an automatic classification method based on machine learning. We rely on pre-trained machine learning models from the Transformers package based on Tensorflow or PyTorch.

Model	Task	Classes	Accuracy (test set)
DistilBERT base uncased finetuned SST-2	Sentiment analysis	{ positive, negative }	91.3 %

Distilbert base uncased emotion	Emotion classification	{ sadness, joy, love, anger, fear, surprise }	93.8 %
Detoxify	Toxicity classification	{ toxic, severe_toxic, obscene, threat, insult, identity_hate }	98 %
	Topic classification	{ }	

Table 3 Text classification models used

Transformers classifiers explained

The transformer classifier is fed a text and returns an integer that indicates the comment's sentiment (or emotion, toxicity and topic).

Training

The models are pre-trained on an English corpus. They learn to associate a text input with a tag (e.g. *positive*, *negative*) using a vector of features. Classic feature extraction approaches are based on bag of words[1], bag of n-grams[2] or word embeddings/ word vectors with their frequency.

Our models use the DistilBERT[3] training method for generate smaller (40 %), faster and cheaper general-purpose language representation models (compared to classic BERT[4] based models).

Those models were fine-tuned on different datasets : the Stanford Sentiment Treebank v2 (SST2)[5] task from the GLUE Dataset[6] for sentiment analysis, the Twitter Sentiment Analysis dataset[7] for emotion classification and Wikipedia comments[8] for toxicity classification.

[1] Jason Brownlee, A gentle introduction to the Bag-of-Words model (2017), https://machinelearningmastery.com/gentle-introduction-bag-words-model/

[2] Machine learning Glossary, Bag-of-n-grams, https://machinelearning.wtf/terms/bag-of-n-grams/

[3] Sanh et al., DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter (2019), https://arxiv.org/abs/1910.01108

[4] Devlin et al., BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2018.

[5] Standford NLP, Stanford Sentiment Treebank v2 (SST2), https://www.kaggle.com/atulanandjha/stanford-sentiment-treebank-v2-sst2

[6] The General Language Understanding Evaluation, https://gluebenchmark.com/tasks

[7] HuggingFace, Twitter Sentiment Analysis dataset, https://huggingface.co/datasets/viewer/?dataset=emotion

[8] Kaggle, Toxic comment classification challenge, https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/data

Prediction

An unseen text input is transformed into a vector of features which is then used by the model to classify the comment. Traditional classification algorithms are statistical models such as Naïve Bayes, Logistic Regression, Support Vector Machines, or Neural Networks.

4. Comment summarization

The sentiment analysis results are condensed using different charts to easily visualize the data and help the content creator answer the following questions :

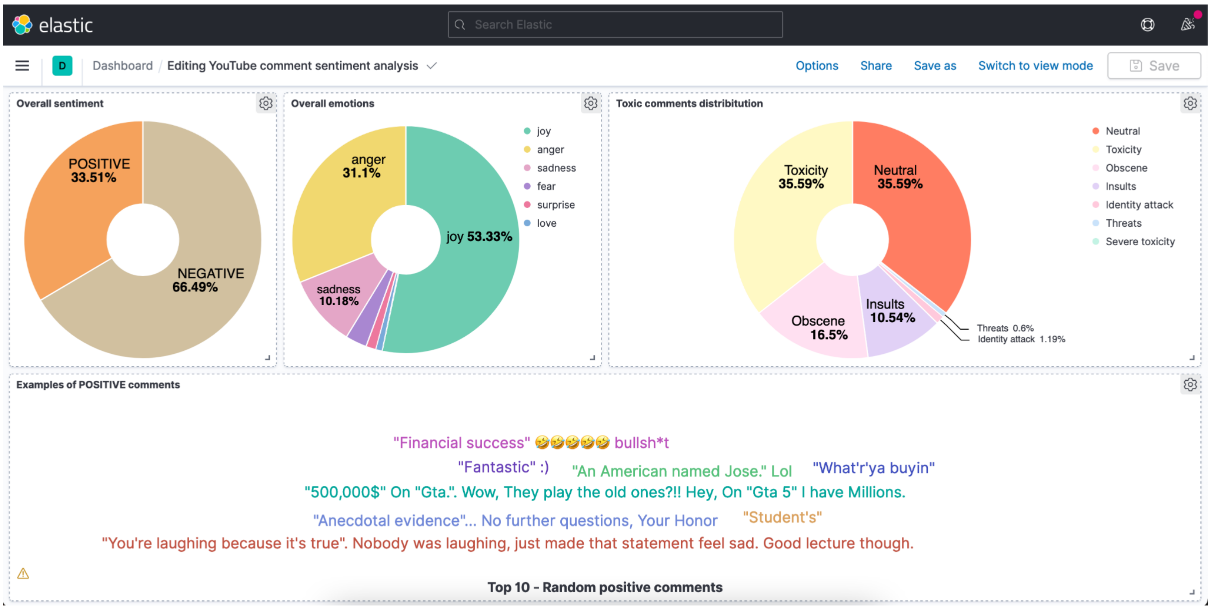
- What are the overall sentiment and emotions ?
- What are people talking about ?
- Are they any inappropriate comments ? Who is publishing those comments ?

To allow that understanding, we generated the following charts :

- Bag of words : a word cloud of the positive, neutral and negative comments.
- Average length of comments : in number of words. Is the audience very verbose ?
- Topics : bar chart counting the number of comments discussing a given topic. In the best case scenario they represent objective aspects of a piece of content a creator can work on improving.
- Overall sentiment : pie chart showing the distribution of positive, negative and neutral comments.
- Overall emotion : pie chart showing the distribution of emotions in the comments.
- Sentiment/ emotion examples : a set of comments that best represent each sentiment or emotion.
- Haters : list of users responsible for posting inappropriate comments.

Results

Using Kibana, a dashboard was created to summarize the audience’s opinion on a video.



"Financial success" 🤣🤣🤣🤣🤣 bullsh*t

"Fantastic" :)

"An American named Jose." Lol

"What'r'ya buyin"

"500,000$" On "Gta.". Wow, They play the old ones?!! Hey, On "Gta 5" I have Millions.

"Anecdotal evidence"... No further questions, Your Honor

"Student's"

"You're laughing because it's true". Nobody was laughing, just made that statement feel sad. Good lecture though.

Limits of the method

Truncated comments because of tensor limitations

To be properly classified, the comments had to be truncated at 510 characters to match the size of the sentiment analysis tensor (512). To avoid this loss of information, Longformers[9] could be used to process longer forms of text with BERT-like models.

Lexicons based emotion detection

When they don’t use machine learning algorithms, emotion detection systems use lexicons (i.e., lists of words and the emotions they convey). The main limit of lexicons is that emotions are expressed in several ways. A system that only relies on words’ connotations will fail to distinguish truly negative (e.g., *your video is so bad or your nasal voice is killing me*) and falsely negative comments (*e.g., your videos are always badass, you are killing it !*).

Other problems

Other classic problems in opinion mining involve object identification (what is the comment talking about ?), feature extraction, grouping synonyms, opinion orientation classification, selection of opinion oriented sentences (comparative and regular sentences ; facts or opinion), writing styles, strength/ intensity of opinions, misleading opinions due to spam, sentences with mixed views.

Challenges

Emoticons

A welcome extension would be to process Western (e.g. :D) and Eastern (e.g. ¯ \ (ツ) / ¯) emoticons to make them classifiable using a comprehensive list of emojis[10] and their unicode characters. Meaning, creating and using a dictionary to map emoticons to emojis so that they can be processed by our emotion classifier.

Subjectivity and tone

Sentiment analysis is highly subjective since it’s estimated that people only agree around 60-65% of the time when determining the sentiment of a given text. We try to work around that by applying the same classification criteria to all of our data, which helps improve accuracy.

[9] Victor Karlsson, Longformer — The Long-Document Transformer (2020), Medium, https://medium.com/dair-ai/longformer-what-bert-should-have-been-78f4cd595be9
[10] Unicode, Comprehensive list of emojis, https://unicode.org/emoji/charts/full-emoji-list.html
Wikipedia, List of emoticons, https://en.wikipedia.org/wiki/List_of_emoticons

On the other hand, the tone can completely change the overall sentiment of a comment. When it comes to irony or sarcasm, people use positive words to express a negative sentiment and vice versa. Our model fails to notice this subtlety.

Context, polarity and comparisons

Sentiment analysis without context is difficult because our model cannot take context into account if it is not mentioned explicitly. One challenge that arises from context is polarity : it is difficult to classify a comment that says “Absolutely nothing” without knowing if they are talking about what they liked or disliked in a video.

Furthermore, to successfully identify the overall sentiment, comparisons must also be taken into account but some comparisons need contextual cues to be classified correctly.

Multilingual sentiment analysis

As mentioned in the Comment retrieval section, our channel and video selection is multilingual : English, French, Spanish. That required thorough preprocessing to complete that task successfully.

Our approach is to use a language classifier to automatically detect a language, then use a targeted sentiment analysis model to classify our comments. At this stage of the project, only English contents are classified.

Conclusion

Sentiment analysis is traditionally used to help businesses know what makes their customers happy or frustrated, so that they can tailor their offer to meet their customers’ needs.

We applied that technique to YouTube videos to help content creators monitor their audience’s opinion. They benefit from sorting data at scale in a real-time manner in order to:

- Improve the quality of their content based of objectives feedbacks,
- Adjust their content strategy to meet their audience’s needs and wishes,
- Help de-escalate a heated debate by taking disciplinary actions before they spiral out of control (penalize those who don’t respect their channel guidelines by reporting and banning them).

The result of our project is a dashboard that gives an overview of the audience’s sentiment and emotions as well as the main topics they discuss. A key step in the future is to turn this tool into a browser extension or web application.