



# YOUTUBE SPAM DETECTION

Artificial Intelligence for Cybersecurity

Antonio Osele

# TABLE OF CONTENTS

---

## Goal of the project

Detecting spam messages from YouTube comments

01



## Data cleaning

Import, clean and preprocess the data

02



## Data analysis

Analyze the data to understand it better

03



04

## Classification

Use different algorithms to classify the data



05

## Validation

Evaluate the results with the Stratified K-Folds cross-validator



06

## Conclusions

Project outcome and possible improvements



# YOUTUBE SPAM

---

YouTube comments are known for having lots of spam, ranging from self advertisement or irrelevant messages to straight up phishing and scam attempts. The goal of the project is to train a model able to detect such comments.



# ABOUT THE DATASET

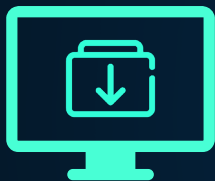
---

The dataset<sup>[1]</sup> contained 1956 instances of real comments extracted from five of the most viewed videos on YouTube. Each instance was labeled as spam or ham. Other attributes are: comment ID, author, date.

<sup>[1]</sup> <https://archive.ics.uci.edu/ml/datasets/YouTube+Spam+Collection>

# DATA CLEANING

---



## IMPORT

Import and concatenate  
the datasets



## CLEAN

Remove unnecessary  
features



## PREPROCESS

Add more useful  
features

# DATA ANALYSIS

---



**HISTOGRAMS**



**HEATMAP**

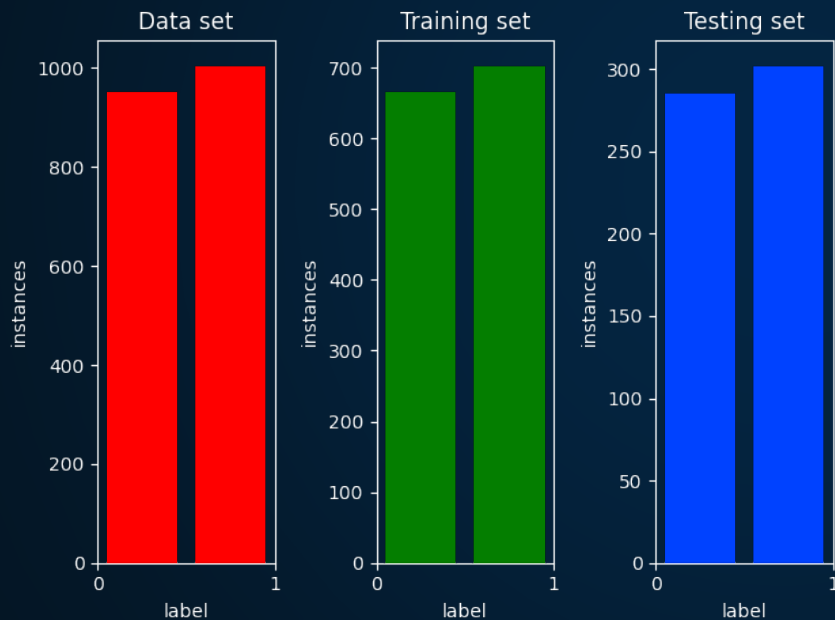


**WORD CLOUD**



# DATASET DISTRIBUTION

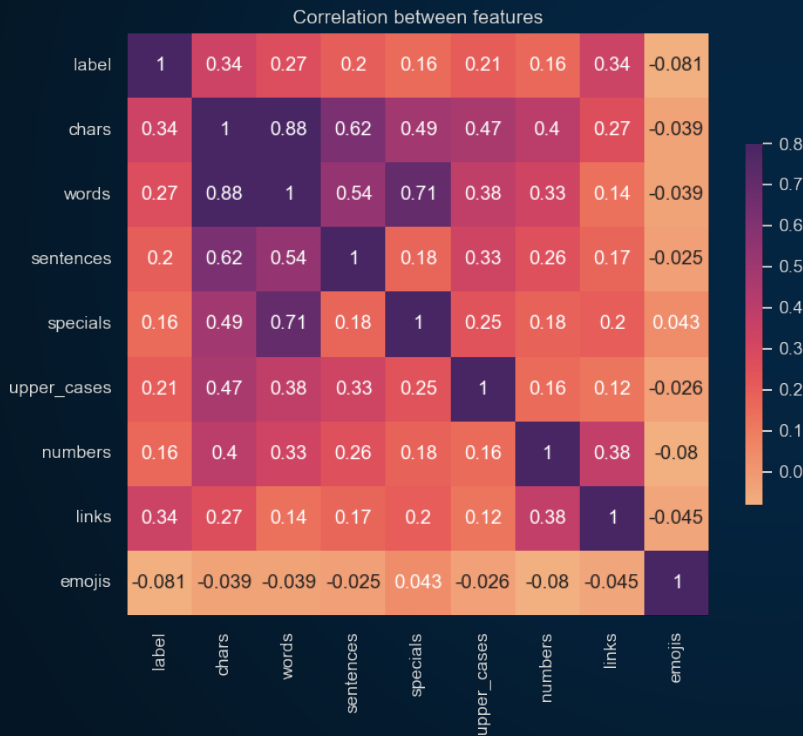
---



## Balanced data

The raw dataset was already balanced. After the 70/30 split of training and testing data, the ratio between spam and ham is unchanged.

# FEATURE CORRELATION

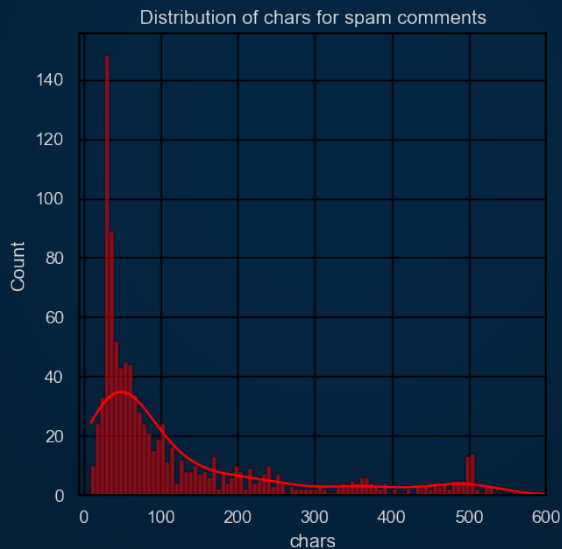
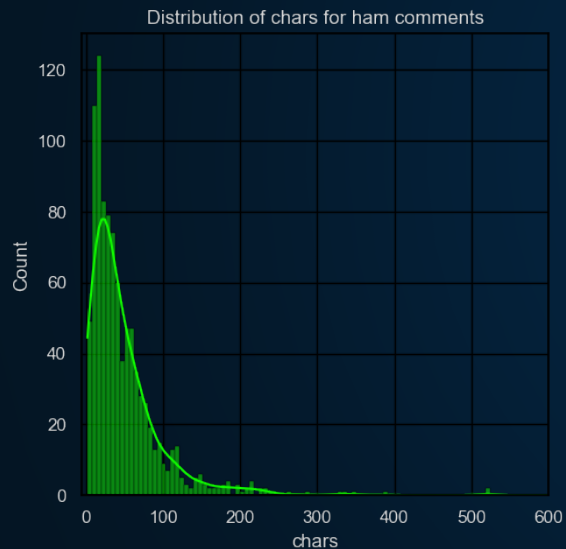


## Links and emojis

Character count and links are more prevalent in spam, whereas emojis are slightly less present.



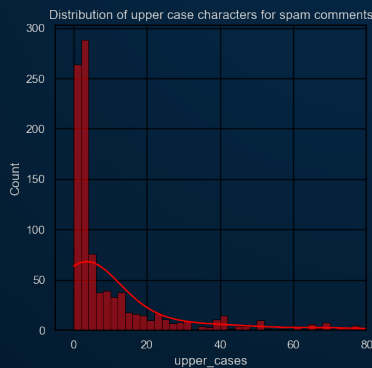
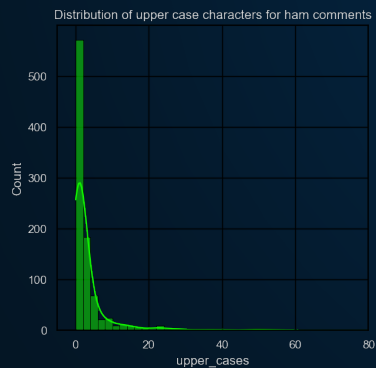
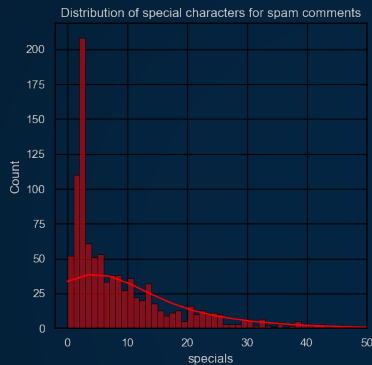
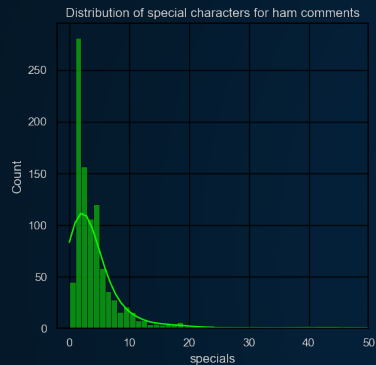
# FEATURE DISTRIBUTION



## Long comments

**Ham comments are on average 200 characters or less. Spam comments instead tend to be longer, with a secondary peak at around 500 characters.**

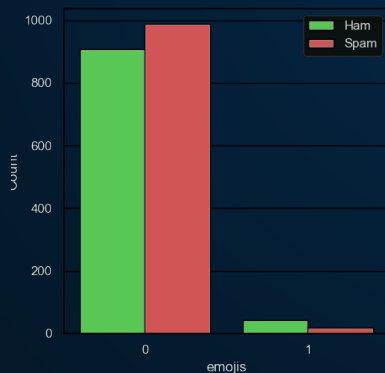
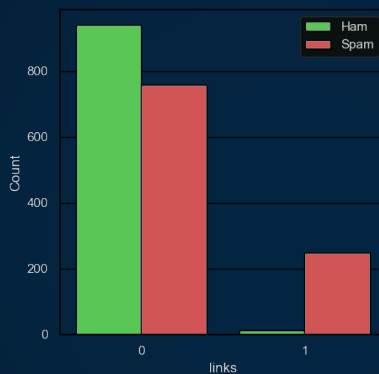
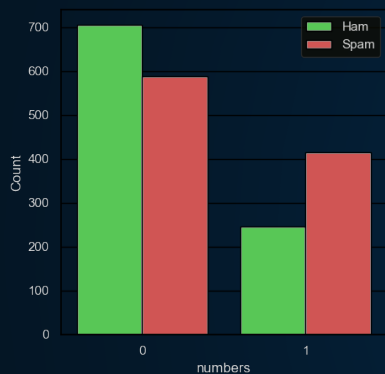
# FEATURE DISTRIBUTION



## Other characters

Something similar can be observed with the distribution of special and upper case characters, being more spread out in spam comments than in ham.

# FEATURE DISTRIBUTION

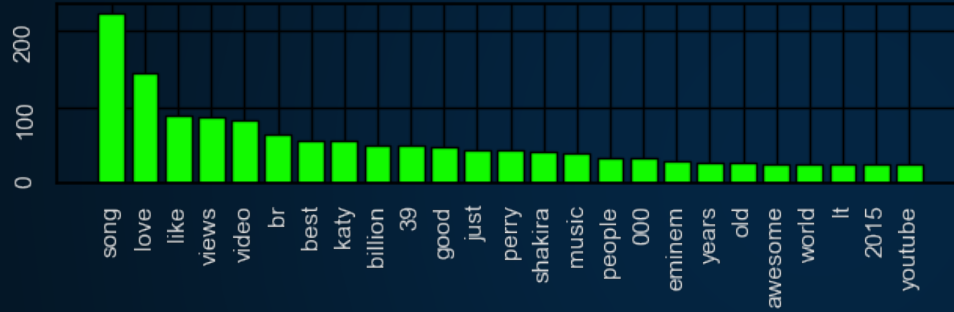


## Feature presence

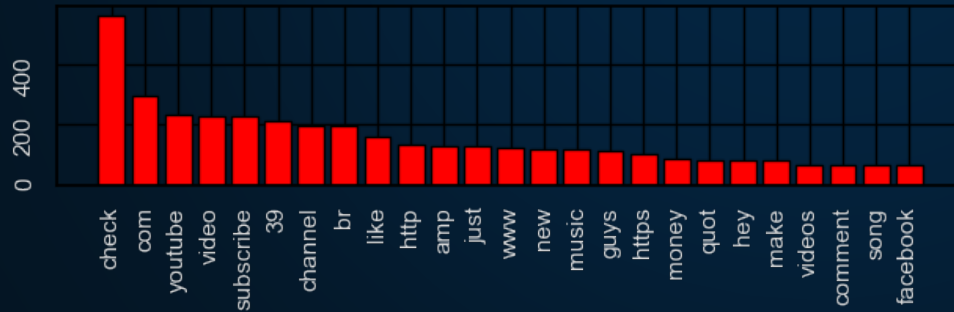
Here we can see how spam are more likely to contain numbers and links. At the same time they don't have as much emojis.

# WORD FREQUENCY

Frequency of words in ham comments



Frequency of words in spam comments



## Common words

It's easy to see that ham comments engage normally with the video while spam comment are mostly self advertisement or phishing.

## A word cloud to show in a different way the most used words in spam comments

## A word cloud to show in a different way the most used words in spam comments

The background is a dark navy blue. It features several decorative elements: a large, light blue gear outline in the bottom left corner; a smaller, solid light blue gear with a circular outline above the central text box; a solid light blue gear with a circular outline in the bottom right corner; and various horizontal lines in white and light blue at the top and right edges.

## VECTORIZATION

Count Vectorizer was used to tokenize the text of the comments, remove accents, punctuation and stop words.

# CLASSIFIERS

---



## K-NEIGHBORS

- + simple, fast
- sensitive to outliers



## GAUSSIAN WITH RBF

- + versatile (different kernels)
- inefficient if high features



## SVM WITH SGD

- + fast, unbiased by outliers
- sensitive to feature scaling



## SVC

- + effective in high dimensions
- sensitive to hyperparameters

# CLASSIFIERS

---



## MULTINOMIAL NB

- + fast, unbiased by outliers
- assumes all features have the same relevance



## COMPLEMENT NB

- + same as MNB but faster on text classification tasks



## DECISION TREE

- + easy to explain and visualize
- slow, prone to overfitting



## RANDOM FOREST

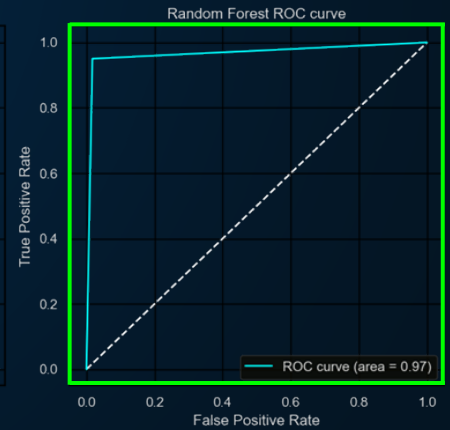
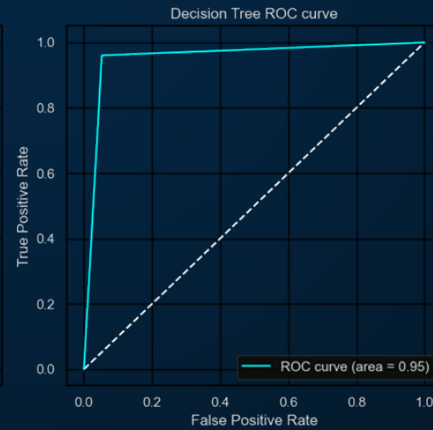
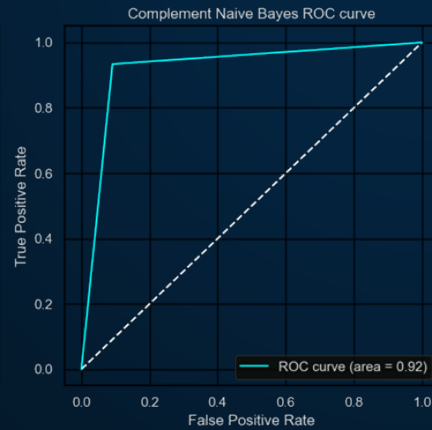
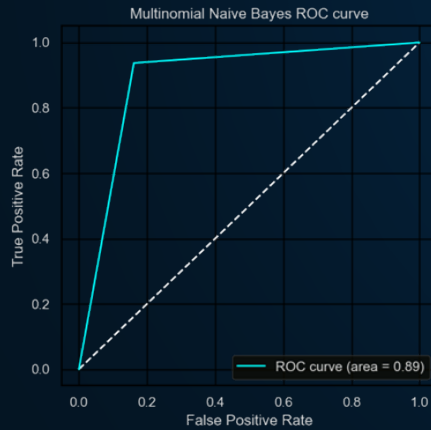
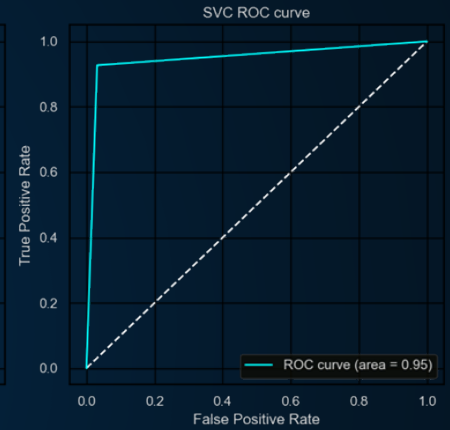
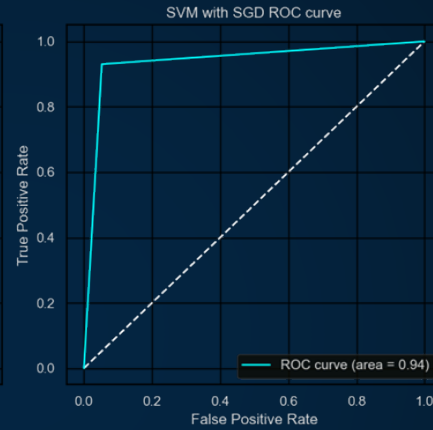
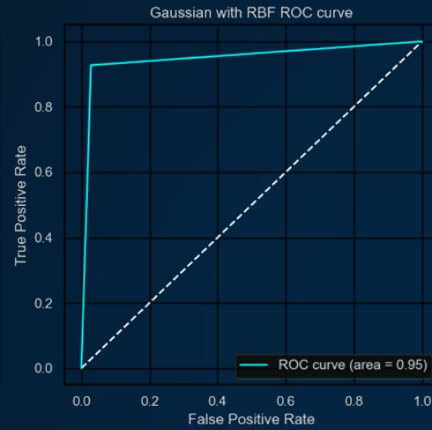
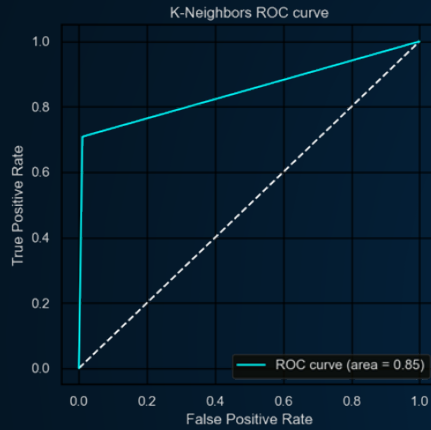
- + very accurate
- hard to interpret, prone to overfitting



# METRICS

CLASSIFIER	CONFUSION MATRIX	ACCURACY	PRECISION	RECALL	F1
K-NEIGHBORS	<div><div>[2823]</div><div>[ 88214]</div></div>	0.844	0.986	0.708	0.824
GAUSSIAN WITH RBF	<div><div>[277 8]</div><div>[ 22 280]</div></div>	0.948	0.972	0.927	0.949
SVM WITH SGD	<div><div>[270 15]</div><div>[ 21 281]</div></div>	0.938	0.949	0.930	0.939
SVC	<div><div>[276 9]</div><div>[ 22 280]</div></div>	0.947	0.968	0.927	0.947
MULTINOMIAL NAIVE BAYES	<div><div>[239 46]</div><div>[ 19 283]</div></div>	0.889	0.860	0.937	0.896
COMPLEMENT NAIVE BAYES	<div><div>[259 26]</div><div>[ 20 282]</div></div>	0.921	0.915	0.933	0.924
DECISION TREE	<div><div>[270 15]</div><div>[ 12 290]</div></div>	0.954	0.950	0.960	0.955
RANDOM FOREST	<div><div>[280 5]</div><div>[ 15 287]</div></div>	0.965	0.982	0.950	0.966

# ROC CURVES



# STRATIFIED K-FOLD CROSS-VALIDATOR

CLASSIFIER	CONFUSION MATRIX	AVG ACCURACY	AVG PRECISION	AVG RECALL	AVG F1
K-NEIGHBORS	<div><div>92823</div><div>308697</div></div>	0.831	0.860	0.835	0.828
GAUSSIAN WITH RBF	<div><div>91041</div><div>77928</div></div>	0.940	0.942	0.940	0.940
SVM WITH SGD	<div><div>88962</div><div>78927</div></div>	0.931	0.941	0.930	0.924
SVC	<div><div>88467</div><div>90915</div></div>	0.920	0.922	0.920	0.920
MULTINOMIAL NAIVE BAYES	<div><div>816135</div><div>97908</div></div>	0.881	0.884	0.881	0.881
COMPLEMENT NAIVE BAYES	<div><div>836115</div><div>100905</div></div>	0.890	0.892	0.890	0.890
DECISION TREE	<div><div>89061</div><div>72933</div></div>	0.932	0.934	0.926	0.929
RANDOM FOREST	<div><div>89160</div><div>67938</div></div>	0.935	0.938	0.936	0.930

# CONCLUSIONS

---

## RESULTS

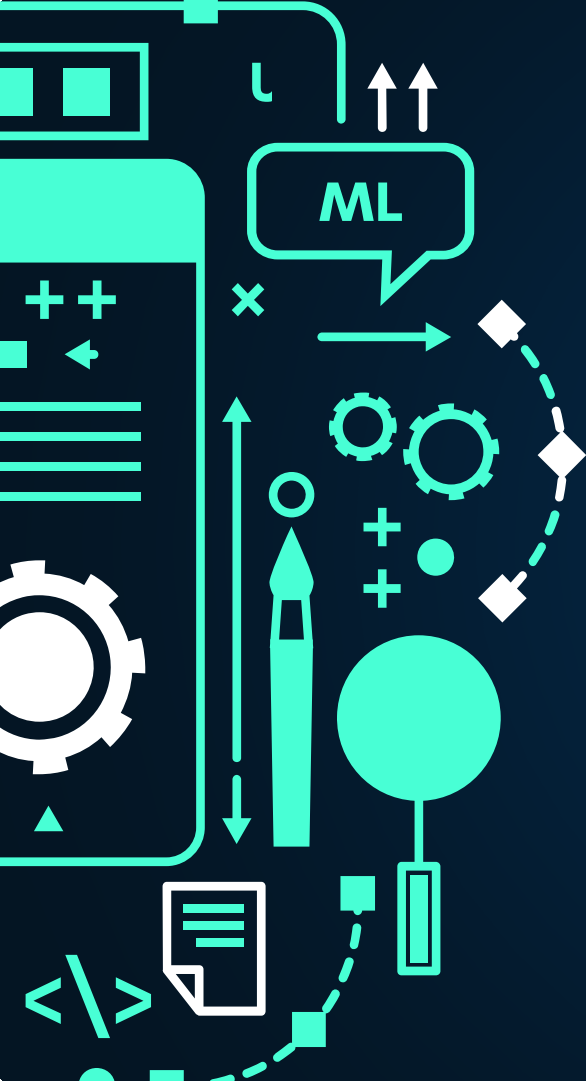
Even if the dataset wasn't very big it achieved acceptable results and the model is able to correctly classify most comments. Overall the initial goals of the project were reached.



## POSSIBLE IMPROVEMENTS

- Collect more data to expand the data set
- Optimize with Grid/Random search
- Discriminate benign links
- Test with comments from other videos





# THANKS!

Antonio Osele  
647926  
antonio.osele@studenti.unipi.it