Spam Detection in YouTube Comments

Aleksandar Nikolić

Software Engineering and Information Technologies

Faculty of Technical Sciences, University of Novi Sad

Motivation

YouTube is currently the biggest video sharing service with millions of registered users. As such, it has a community of people who interact with each other on daily basics through comment sections of the videos. Some people use popular videos to advertise their YouTube channels and business in the comments. This behavior makes it difficult for other people to communicate and talk about topics relative to the video. Spam filters can help detect these types of comments.

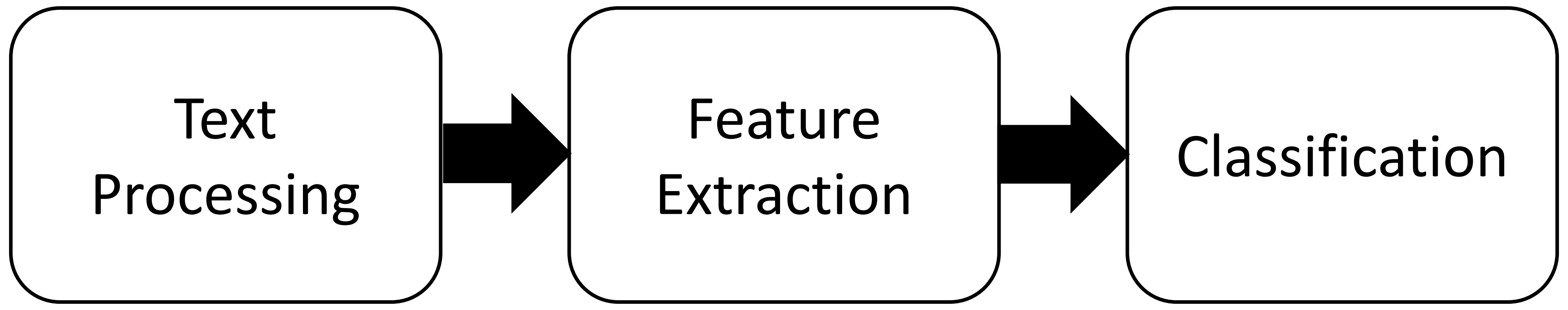
Dataset

The collection is composed of one CSV file per dataset, where each line has the following attributes:

- COMMENT_ID
- AUTHOR
- DATE
- CONTENT
- TAG

Only CONTENT and TAG were used in this project.
There are 1956 comments in total across all files.

Pipeline



Text Processing → Feature Extraction → Classification

Methodology

Text Processing

Comments were first converted to lowercase, all links in comments were substituted with a unique key word and common English words (“a”, “the”) were removed.

Feature Extraction

Combination of three types of n-grams based on words was used: 1-gram (bag of words), 2-gram and 3-gram. Features that were present in less than 1% of comments were removed. This resulted in 347 features in total.

Classification

For classification, Support Vector Machine (SVM) and Multilayer Perceptron (MLP) implementation from Scikit-learn Python library was used and results were compared. SVM kernel that was most successful was linear kernel. Logistic activation function for MLP was used with one hidden layer that consisted of 15 neurons.

Validation and Results

Cross-validation was used. 70% of data was used for training and 30% was used for testing. Training and testing data was randomly selected from the dataset. After trying multiple combination of n-grams, best results were achieved by combining three n-gram types. To reduce data noise and make a more general model, features that have a low frequency were removed.

SVM Classification Report and Confusion matrix

	Precision	Recall	f1-score	support
Non-spam	0.93	0.98	0.96	292
Spam	0.98	0.93	0.96	295
avg / total	0.96	0.96	0.96	587

$$\begin{bmatrix} 287 & 5 \\ 20 & 275 \end{bmatrix}$$

MLP Classification Report and Confusion matrix

	Precision	Recall	f1-score	support
Non-spam	0.92	0.96	0.94	292
Spam	0.95	0.92	0.94	295
avg / total	0.94	0.94	0.94	587

$$\begin{bmatrix} 279 & 13 \\ 23 & 272 \end{bmatrix}$$

Both SVM and MLP were very successful. SVM had a 96% success rate, while MLP had 94%. Training was done on Intel(R) Core(TM) i5-4670K CPU @ 3.40GHz.
Training time for SVM was 0.044 seconds, while time for MLP was 1.630 seconds, which is a lot longer. This leads to a conclusion that SVM is more suitable for solving this problem.