Judging YouTube by its Covers

Angel Iek Hou Zhang
Department of Computer Science and Engineering
University of California, San Diego

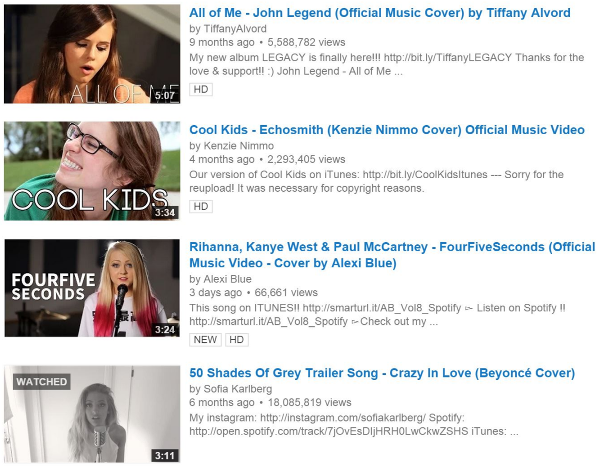
***Abstract*—In this new age of digital multimedia content and internet audiences, the entertainment industry and the path to stardom is slowly evolving. Many aspiring musicians can now achieve fame by performing the works of other famous artists, and sharing their performances on sites such as Youtube. Aspiring artists and other content generators on Youtube are always advised by marketing experts to include keywords in their video titles, descriptions and tags. In this paper, we show that this advice is sound. Using only the title of the video and the Bernoulli Naive Bayes model, we predict whether a video receives many or few views with 65.83% accuracy. We also show that the Bernoulli Naive Bayes model performs better than SVMs and k-NN on this data set of music cover videos.**

Introduction

Picture how you would react if you searched "music" on YouTube and were presented with a page of video content. Your eyes would scan the page and perhaps linger on an interesting title, a dynamic thumbnail, or an inconceivably large view count number, and possibly within the next few seconds, you will have moved your mouse to click on one of the videos. This could happen even without you having any knowledge about the quality of the content you are about to spend the next few minutes of your life watching. Considering that the decision to click on a video takes place so quickly, and almost imperceptibly, it is somewhat surprising how much information is actually presented to us in a list of YouTube videos. Figure 1 shows a sample list, in which each video is displayed on its own line with its thumbnail, duration, video title, channel title, view count, age (how long ago it was posted) and a portion of its description. Additionally, there can be information about whether or not the video is in HD, has captions, or is new.

On the other end of YouTube are the content generators. With the advent of online video platforms like YouTube, every independent musician has the entire internet as his/her potential audience. This has spawned an entire generation of YouTube stars, and resulted in an abundance of cover songs being posted on YouTube every day, by independent musicians hoping to build a fan base by performing already popular, commercially released works. For aspiring, unknown musicians, having people discover their content is directly dependent on how engaging the videos' static information is, i.e. the information accompanying the video as shown in Figure 1. This poses an interesting problem that every content generator wishes to solve - how can one craft the perfect title, thumbnail and description in order to attract attention and optimize the number of views?

This paper investigates a dataset comprising of only YouTube cover songs and aims to develop a rudimentary model that predicts the success of a music cover video based on just its title. The model chosen for this task is the bag-of-words Bernoulli Naive Bayes classifier, often used in document classification [1]. This paper shows that Bernoulli Naive Bayes outperforms support vector machines (SVMs) and k-nearest neighbors (k-NN) in the task of predicting the success of a music cover video based on just its title.



All of Me - John Legend (Official Music Cover) by Tiffany Alvord
by TiffanyAlvord
9 months ago • 5,588,782 views
My new album LEGACY is finally here!!! http://bit.ly/TiffanyLEGACY Thanks for the love & support!! :) John Legend - All of Me ...
HD

Cool Kids - Echosmith (Kenzie Nimmo Cover) Official Music Video
by Kenzie Nimmo
4 months ago • 2,293,405 views
Our version of Cool Kids on iTunes: http://bit.ly/CoolKidsItunes --- Sorry for the reupload! It was necessary for copyright reasons.
HD

Rihanna, Kanye West & Paul McCartney - FourFiveSeconds (Official Music Video - Cover by Alexi Blue)
by Alexi Blue
3 days ago • 66,661 views
This song on ITUNES!! http://smarturl.it/AB_Vol8_Spotify ▻ Listen on Spotify !! http://smarturl.it/AB_Vol8_Spotify ▻Check out my ...
NEW HD

50 Shades Of Grey Trailer Song - Crazy In Love (Beyoncé Cover)
by Sofia Karlberg
6 months ago • 18,085,819 views
My instagram: http://instagram.com/sofiakarlberg/ Spotify: http://open.spotify.com/track/7jOvEsDIjHRH0LwCkwZSHS iTunes: ...

Figure 1: List of YouTube videos and associated information

Related Work

Current literature on text mining focuses on topic/category classification [2] and predicting single consumer opinions [3] rather than predicting content popularity. On the other hand, researchers who study popular multimedia content have not fully investigated the role of text and prefer to focus on factors such as early view count, comment counts and like counts, which are not particularly useful for predicting the popularity of a completely new video [4, 5]. Text mining titles and articles has been used to assess content quality and relevance in the biomedical literature, [6] and perhaps more relevant to this study, [7] investigated the influence that titles had on the popularity of Reddit submissions. It is also noteworthy that [8] has studied what makes up quality content. However, to the best of our knowledge, there is no particular study that focuses on just how "click-inducive" a particular YouTube cover song title is and how this could contribute to the popularity growth of new musicians on YouTube.

Methodology

Data Collection

There is no publicly available dataset consisting of just YouTube cover songs, so a new dataset was constructed using the YouTube Data API. The YouTube Data API allows access to public video attributes such as video title, channel title, location, date posted, and a range of other statistics such as comment count, view count, like count and dislike count. Since the API returns a maximum of five hundred results per query, the search parameter had to be varied in order to construct a larger data set. Each query collected at most five hundred results within a five day interval and the "Published Before" and "Published After" search parameters were varied to retrieve data from between July 1 2013 to February 18 2015. All queries required the word “cover” to be in the video title, and the video category to be “Music”. There are a total of 52948 video records in the final data set, and each video record has the following attributes:

- video title
- channel title
- description
- duration
- like count
- dislike count
- view count
- comment count

For the purposes of this paper, location, date posted and video thumbnail url were not collected or analyzed, but these attributes will definitely be included in the next stage of development of the model.

Statistical Pattern

The correlation matrix for the numerical attributes of the videos in our dataset is depicted in Table I. It matches our expectation that comment count, like count and dislike count are all highly correlated with view count, since the more views a video has, the more likely it is that people have opinions about it, and conversely, if there are many opinions about a video, it must have been watched many times.

We visualize some of these strong relationships between opinion and view count by plotting them in Figure 2, 3 and 4. It is worth noting that something interesting is happening when view count reaches 100,000. Figure 3 and 2 show that comment count and like count both increase in the same manner after that point, and perhaps this is an indication of a video going “viral”. Exploring this threshold further could make for an interesting study in the future. Figure 4 shows the relationship between like count, dislike count and view count. We note that like count is almost always above dislike count, and this phenomenon might be of interest to researchers in the fields of social networks, economics, or psychology.

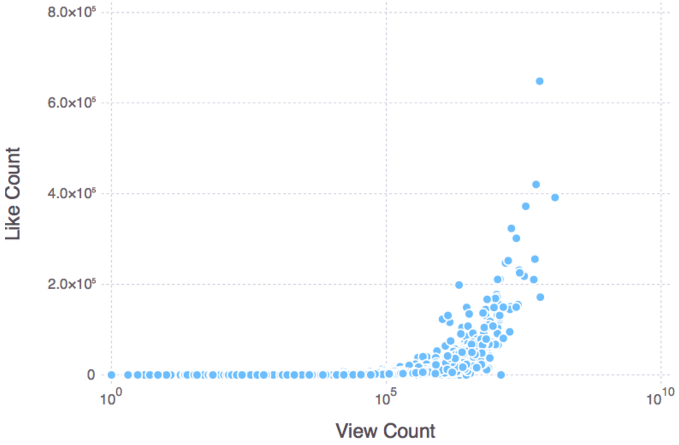


Figure 2: Like count versus view count of videos

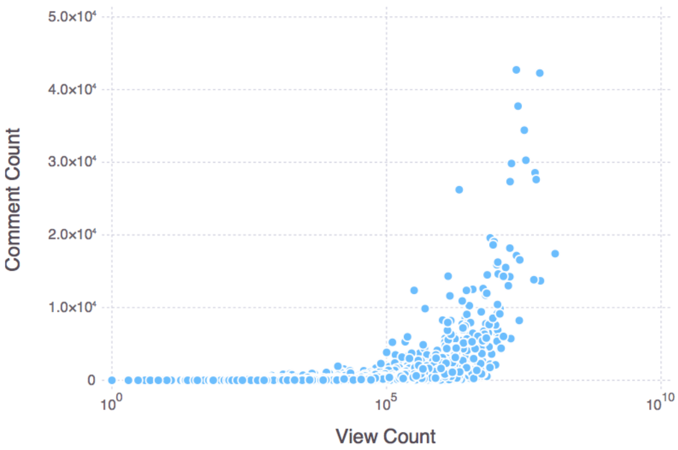


Figure 3: Comment count versus view count of videos

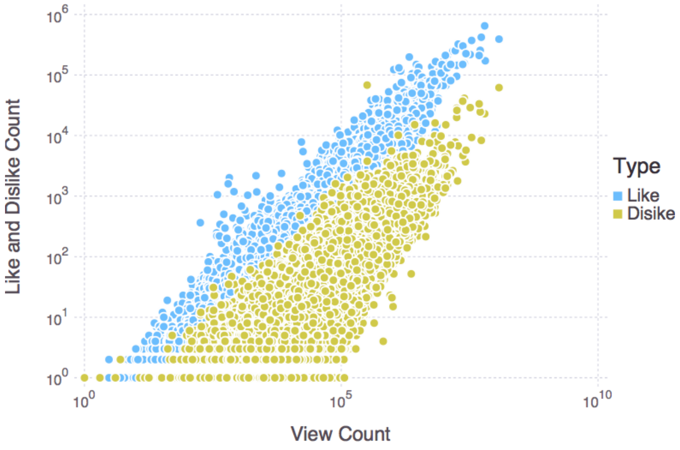


Figure 4: Like and dislike count versus view count

Table I: Correlation between numerical features

	Duration	Views	Comments	Likes	Dislikes
Duration	1.0000	-0.0017	-0.0037	-0.0049	-0.0068
Views	-0.0017	1.0000	0.7097	0.8149	0.7190
Comments	-0.0037	0.7097	1.0000	0.8702	0.6773
Likes	-0.0049	0.8149	0.8702	1.0000	0.5517
Dislikes	-0.0068	0.7190	0.6773	0.5517	1.0000

From Table I, we also observe that there is very little correlation between duration and view count. This is visualized in Figure 5, where the graph of duration is almost flat for all view counts, except for a few outliers. From the plot, we see that there does not seem to be any underlying relationship between the length of a video and its view count.

Another interesting characteristic of the dataset is observed by plotting a histogram of the view counts on a log scale. From the plot in Figure 6, we observe that for the most part, the distribution of view counts resembles a log-normal distri-

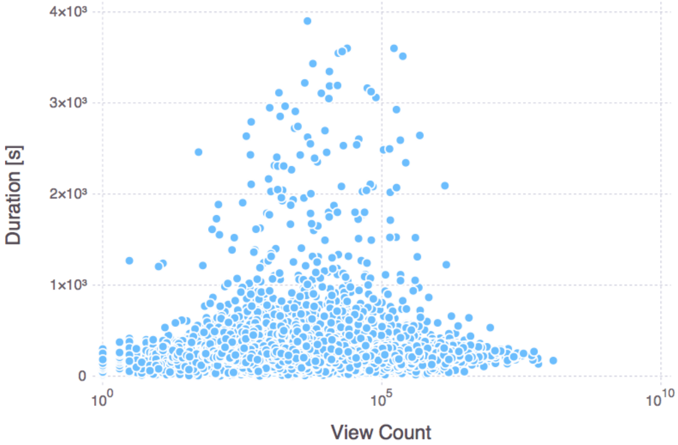


Figure 5: Effect of duration on view count

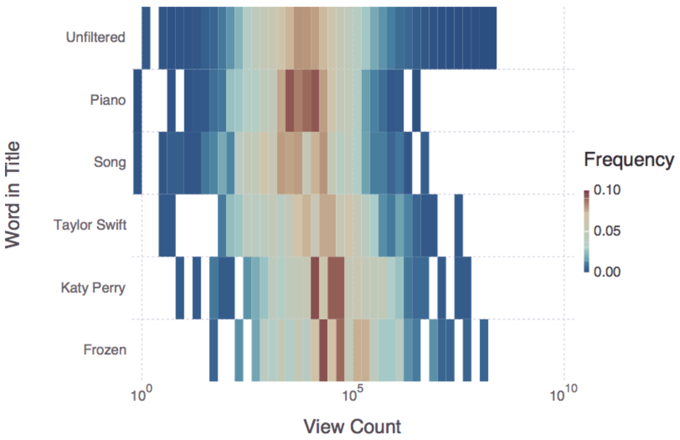


Figure 7: Comparision of view count distribution containing different key words

bution. According to [9],, this could actually be an indication of a power-law distribution, which arises from phenomena where the "rich-get-richer" principle holds. This would also match our intuition, as a video that already have a large number of views is more likely to attract new views. Perhaps people who have already watched the video share it with their friends, in which case the rate of growth of view count could even become exponential. Or perhaps new viewers are simply really curious as to why the video has so many views. In any case, this means that there might be a snowball effect after reaching a certain threshold in view counts, at which point the static information we are trying to manipulate becomes less important. Although this is not very beneficial to new musicians who need to attract those initial views, it can be comforting, and at the same time disconcerting, to know that beyond a certain point, the growth in video views is completely beyond their control. Again, further investigation into this topic is necessary before drawing any conclusions.

Figure 7 is a plot of videos that contain certain words in their title versus view count. This plot motivates the main idea behind the model - that the title of the video can indeed give some information about whether a video's view count will be high or low. Figure 7 shows that videos with neutral words such as "piano" and "song" in their title have very similar view count frequency distributions to that of the entire dataset. However, if we pick a "commercially popular" term such as "Taylor Swift", "Katy Perry" or "Frozen", there is a clear shift to the right in view count frequency.

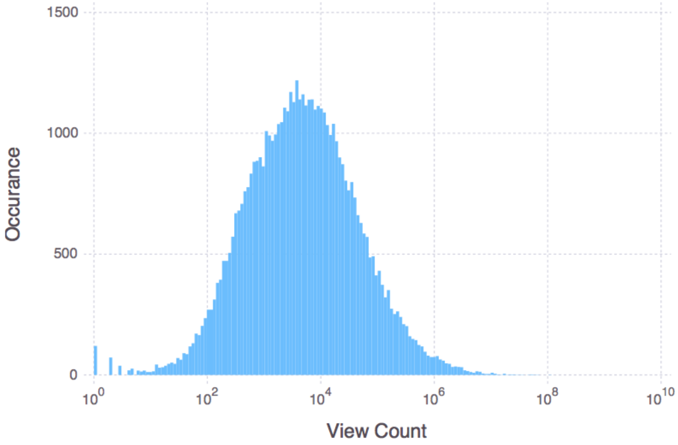


Figure 6: Distribution of view counts (without zero)

The histograms in Figure 7 are log-scale, so the results are slightly skewed because we omitted the videos with zero views, but note in Table II that even if we include the videos with zero views and examine the mean and median view counts, we still find that those values corresponding to "Taylor Swift", "Katy Perry" and "Frozen" videos are significantly higher than those corresponding to "Song", "Piano" and "Unfiltered" videos. Because some commercial songs and artists are more popular or more searched for than others, music cover videos of these songs and artists receive more views. This motivates why we proceed to model a relationship between title and view count.

Table II: View count statistics for videos containing different keywords

	Mean View Count	Median View Count
Unfiltered	72,742	4429
Piano	33,370	6492
Song	58,399	4242
Taylor Swift	201,459	12,015
Katy Perry	431,549	28,879
Frozen	1,233,651	34,638

Feature Selection

We selected view count as the measure of how successful a video is. Although it is true that like count might be more indicative of how popular a video is, and the actual quality of the content is without a doubt important, we are first and foremost interested in optimizing view count. Without viewers, there is no point in discussing how to improve content or optimize for higher like counts and lower dislike counts. Due to the underlying distribution of view counts resembling the log-normal distribution, we were careful in picking a good threshold for separating the videos into two classes - high view count and low view count. We set the threshold, θ, equal to the median view count (4429.5). Although this number does not seem very high, it is still statistically interesting and significant to be better than 50% of all other videos in terms of view count. Therefore, the label “high view count”, θ_{high}, corresponds to having a view count that is in the top 50% of all videos, and “low view count”, θ_{low}, corresponds to having a view count in the bottom 50% of all videos.

As we observed in the previous section, duration is not really correlated to view count, so we discarded it. Although comment count, like count and dislike count are all highly correlated with view count, we discard them as features too because from a content generator’s point of view, it is not so useful to know that if he/she has lots of comment counts, he/she also has a lot of view counts. Similarly, we did not use channel title as a feature, because we did not want to double count the inherent popularity associated with a channel name.

However, we should note that had we not discarded all the above features, the overall accuracy of the classifier might have improved.

The two sets of features that we ended up testing our model on are the words in the title, and the words in the description. These were represented as two separate word vectors. Although these two features could have been combined to form one giant feature vector, we kept them distinct in order to analyze the separate contributions of title and description to the view count.

Preprocessing Features

Six different vocabularies were constructed: V_1 is a unigram dictionary built from all the words that appear in the set of video titles, V_2 is a bigram dictionary built from all the two-term words that appear in the set of video titles, and V_3 is a unigram dictionary built from all the words that appear in the set of video descriptions more than five times. V_4 is a unigram dictionary of all the words that appear in the set of video titles more than once, V_5 is a unigram dictionary of all the words that appear more than three times, and V_6 is a unigram dictionary of all the words that appear more than five times in the set of video titles. The unigram dictionary consists of single words, and the bigram dictionary consists of sequences of two words, e.g. the video title “Taylor Swift likes goats” would result in $V_2 = \{\text{"Taylor"}, \text{"Swift"}, \text{"likes"}, \text{"goats"}\}$ and $V_2 = \{\text{"Taylor Swift"}, \text{"Swift likes"}, \text{"likes goats"}\}$. All the punctuation was stripped, and all words were converted to lowercase, but special unicode characters were added to the dictionary in order to account for video titles in foreign languages. In addition, in order to eliminate common words that have little to no predictive value, we removed words such as "cover", "by", "ft", "my", etc. from the vocabularies [10]. A sparse matrix was then generated for every vocabulary, such that one row in the sparse matrix represented one video’s word feature vector.

Model

We used the Bernoulli Naive Bayes bag-of-words model to predict view count. The bag-of-words model assumes that we treat each video title and description text as if they are just a string of words that are picked, with replacement, out of a bag that contains all the words in either V_1, V_2, through V_6, depending on which case we were interested in testing. This model assumes that the words are conditionally independent from each other given the view count, and makes a prediction for title j using the following rule: if $P(\theta_{\text{high}} \mid w_j) > P(\theta_{\text{low}} \mid w_j)$, where w_j is the words in title j, then predict θ_{high} (recall,θ_{high} is defined by view count > θ = 4429). If $P(\theta_{\text{low}} \mid w_j) > P\left(\theta_{\text{high}} \mid w_j\right)$, then predict that view count is low. Using Bayes rule and conditional independence of the words, we get that

$$\begin{aligned} P\left(\theta_{\text{high}} \mid w_j\right) &= \frac{P\left(\theta_{\text{high}}, w_j\right)}{P\left(w_j\right)} \\ &= \frac{P\left(w_j \mid \theta_{\text{high}}\right) \cdot P\left(\theta_{\text{high}}\right)}{P\left(w_j\right)} \\ &= \frac{\prod_{w_j} P\left(\text{word} \mid \theta_{\text{high}}\right) \cdot P\left(\theta_{\text{high}}\right)}{P\left(w_j\right)} \end{aligned}$$

Notice that the denominator is not necessary for classification purposes, since

$$P\left(\theta_{\text{high}} \mid w_j\right) = 1 - P\left(\theta_{\text{low}} \mid w_j\right)$$

, so a comparison of the numerators if sufficient. In addition, since we are using Bernoulli Naive Bayes, we only count the appearance of the word i.e. even if a word appears in a title more than once, we only account for the probability of it occuring once.

Other models we were interested in are k-NN and SVMs. SVMs are linear classifiers that have been shown to have a lot of success with text categorization and higher-dimensional, sparse data. They are also very resistant to overfitting due to the fact that they learn the hyperplane decision boundary in a higher dimensional space [11, 12]. k-NN is slightly different in flavor and less restrictive in that it does not require a linear boundary but rather, makes a prediction based on its k nearest neighbors [13]. The difficulty in using k-NN for text classification in our case, is finding a good feature space and figuring out what distance function to use.

RESULTS

To train and test the model, we split the dataset into two: 70% for training and 30% for testing. In order to tune the parameter for k-NN and how many words to discard from the vocabularies, we created a validation set by splitting the training set up further: 80% for training and 20% for validation.

We tested several different models, and found that Naive Bayes performed on V_1 yielded the best results in terms of accuracy rate, where accuracy rate is defined as $\frac{\text{\# predicted correctly}}{\text{\#predictions}}$. For the comparison of Naive Bayes, k-NN and SVMs, we used only the vocabulary V_1. We first trained using a polynomial fit SVM, which gave an accuracy rate of 0.20 on the test set. k-NN with Minkowski distance performed slightly better. Using the validation set for tuning k, we found that k = 3 gave one of the higher accuracy rates. We then retrained 3-NN on the training and validation set, and the accuracy rate on the test set was 0.58. To find the best word feature vector for Naive Bayes, we trained the model on the training set for the vocabularies

V_1, V_4, V_5 and V_6 and tested them on the validation set. We found that using V_6 (discarding all words with less than 5 occurences) gave an accuracy rate 0.631, V_5 (discarding all words with less than 3 occurrences) gave an accuracy rate of 0.639, V_4 (discarding all words with less than 1 occurence) gave an accuracy rate of 0.647 and V_1 (keeping all the words in the dictionary) gave an accuracy rate of 0.65. Since V_1 gave the best results, we retrained the model on the training set and the validation set using V_1, and found the accuracy rate on the test set to be 0.6583, with true positive rate:

$$\frac{\text{\# correct high view count predictions}}{\text{\# high view count predictions}} = 0.6394$$

and true negative rate:

$$\frac{\text{\# correct low view count predictions}}{\text{\# high low count predictions}} = 0.6770.$$

Settling on Naive Bayes as our model, we then compared the unigram and the bigram word features. We found that using V_2 resulted in an accuracy rate of 0.65, and considering how much extra information was stored, it was somewhat surprising that V_1 actually yielded a slightly higher accuracy rate.

We also compared using the title words as features in the model versus using the description words as features. V_3 yielded an accuracy rate of 0.61, which was again surprising, given how much more information is contained in descriptions.

We therefore found that Naive Bayes trained on V_1 is the best predictor for the view count label.

Evaluation and critique

To evalute the performance of our classifier, we compared it to the performance of a random classifier. Since we have an inbalanced data set where most of the videos have view counts below the mean, and only a select few have view counts above the mean, we needed to ensure that our classifier was not just guessing "low view count" most of the time. This was the reasoning behind picking θ to be the median, and defining anything above θ to be a high view count, and anything below to be a low view count. Since θ is the median, a random classifier would pick "high view count" or "low view count" with probability 0.5. Thus, an accuracy rate of 0.6583 is in fact significant, and by evaluating the true positive rate = 0.6394 and the true negative rate = 0.6770, we can safely confirm that the classifier is not just guessing "low view count" most of the time. Since the accuracy rate on the training set and the test set were similar, there was no issue of overfitting. There were also no scaling issues since the data was just a sparse matrix of ones and zeros.

Conclusion and Future Work

Achieving an accuracy rate of 0.6583 with a Bernoulli Naive Bayes classifier built on a unigram vocabulary of video title words is significantly better than the results achieved using SVMs, k-NN or random guessing. This result is significant considering the only information we used to predict a high or low view count was the video title. This verifies the intuition that including popular terms in a video title can increase the number of views.

Due to time constraints, we were not able to to use the image thumbnail and time posted data in our model. This might have resulted in a higher accuracy rate and could potentially provide more insight on how a new content generator can craft a better post.

Although we do not yet know what the impact of including images of Elsa or Katy Perry in a video thumbnail might be, we hope that our work will still be informative for aspiring YouTube stars. Simply include "Frozen" or "Katy Perry" in your title and watch the views go up![1]

References

[1] Ashraf M Kibriya, Eibe Frank, Bernhard Pfahringer, and Geoffrey Holmes. Multinomial naive bayes for text categorization revisited. In *AI 2004: Advances in Artificial Intelligence*, pages 488–499. Springer, 2005. (document)

[2] Fabrizio Sebastiani. Text categorization. In *Text Mining and its Applications to Intelligence, CRM and Knowledge Management*, pages 109–129. WIT Press, 2005. (document)

[3] Silvana Aciar. Mining context information from consumer's reviews. Technical report, Universidad Nacional de San Juan - Instituto de Informatica, 2010. (document)

[4] G. Chatzopoulou, Cheng Sheng, and Michalis Faloutsos. A first step towards understanding popularity in youtube. In *INFOCOM IEEE Conference on Computer Communications Workshops , 2010*, pages 1–6, March 2010. (document)

[5] Flavio Figueiredo, Fabrício Benevenuto, and Jussara M Almeida. The tube over time: characterizing popularity growth of youtube videos. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 745–754. ACM, 2011. (document)

[6] Dean Cheng, Craig Knox, Nelson Young, Paul Stothard, Sambasivarao Damaraju, and David S Wishart. Polysearch: a web-based text mining system for extracting relationships between human diseases, genes, mutations, drugs and metabolites. *Nucleic acids research*, 36(suppl 2):W399–W405, 2008. (document)

[7] Himabindu Lakkaraju, Julian McAuley, and Jure Leskovec. What's in a name? understanding the interplay between titles, content, and communities in social media, 2013. (document)

[8] Eric Nichols, Charles DuHadway, Hrishikesh Aradhye, and Richard F. Lyon. Automatically discovering talented musicians with acoustic analysis of youtube videos. In *Proceedings of the 2012 IEEE 12th International Conference on Data Mining*, ICDM '12, pages 559–565, Washington, DC, USA, 2012. IEEE Computer Society. (document)

[9] Meeyoung Cha, Haewoon Kwak, Pablo Rodriguez, Yong-Yeol Ahn, and Sue Moon. I tube, you tube, everybody tubes: Analyzing the world's largest user generated content video system. In *Proceedings of the 7th ACM*

[1] Disclaimer: As shown by the analysis in this paper, this only increases your chances, and is by no means a guarantee.

SIGCOMM Conference on Internet Measurement, IMC '07, pages 1–14, New York, NY, USA, 2007. ACM. (document)

[10] Swati Joshi and Dharmendra Sharma. Analysis of impact of stop words on domain specific document set. *International Journal Of Recent Advances in Engineering & Technology*, 2014. (document)

[11] Thorsten Joachims. *Text categorization with support vector machines: Learning with many relevant features*. Springer, 1998. (document)

[12] Susan Dumais et al. Using svms for text categorization. *IEEE Intelligent Systems*, 13(4):21–23, 1998. (document)

[13] Fabrice Colas and Pavel Brazdil. Comparison of svm and some older classification algorithms in text classification tasks. In Max Bramer, editor, *Artificial Intelligence in Theory and Practice*, volume 217 of *IFIP International Federation for Information Processing*, pages 169–178. Springer US, 2006. (document)