Data Analytics of Video Popularity

Ziyu He, Haoxiang Gao
Department of Electrical Engineering
Columbia University in the City of New York
zh2255@columbia.edu, hg2412@columbia.edu

Abstract—**Predictions and analysis on popularity of user-created web content, especially video, is becoming increasingly important and valuable to gain insights in web content's dissemination in a dynamic distribution system, to benefit decision making in online marketing and designing of web content. In this paper, we aim to conduct a comprehensive data-driven study of influential factors of YouTube channels' popularity. Analysis in this paper is achieved with the following steps: (1) Collecting related information from various sources in regard to each individual YouTube channel; (2) Data preprocessing algorithms to extract useful features from unstructured raw data; (3) Training and validating machine learning models for prediction of quantified channel popularity and inference of relative importance of predictive features; (4) Developing an item based recommender based on previous analysis and its online visualization. With data of more than 10,000 YouTube channels and 80,000 YouTube videos, our analysis shows that popularity of current YouTube channels can be quantified as 3 clusters with different levels of accumulated views; frequency of publishing videos, interaction of content creator and reference of its videos on online social media are critical factors to promote popularity of a YouTube channel. In this paper, we also designed a cascaded Random Forest model that can solve the imbalanced classification problem in prediction.**

Keywords - Machine Learning; Web Content; Prediction; Social Media; Data Visualization

I. INTRODUCTION

In Web 2.0, due to their characteristics of low-cost in generation and dissemination, all types of online self-generated content are dominating our internet as a virtual public sphere and they are becoming an important part of the new form of so-called "We-Media" evolution. Take our research objective, YouTube as an example: YouTube has over a billion users and in each day enormous amount of information is created and consumed on YouTube including more than 100 hours of uploaded videos and billions of views [1]. Thus, the capability of mining useful knowledge from flooding visual content stream, understanding the distributing mechanisms of videos, identifying key factors in determining a video's potential value as well as predicting the popularity of a video are crucial problems which need to be solved to benefit decision making in online marketing, to help designing better distribution network and related services, and to guide the creation of online contents. For instance, if we are available to accurately predict view counts of a specific video publisher to exceed a significant number, product placement advertising strategies can be carried out for this publisher to increase potential revenue while fund in publisher promotion can be better spent on other less popular video publishers [2].

From the perspective of data mining, research in prediction of video popularity and analysis of potential predictive features involve answering the following questions: (1) How should we model this problem; (2) What are the features that are worth studying; (3) How to extract these features from mass multi-sourced and unstructured data stream; (4) How to validate our model and how to do further inference based on our model.

In our paper, our major research objectives are YouTube video publishers, i.e. channels. We modeled our prediction problem as a typical multi-classification problem which requires us to design a quantification metric of popularity (class labels in our case) in the first place hence machine learning classification models can be applied to predict the general range of popularity of a new YouTube channel. Also note that since we do not have supervised response in our case, generation of class labels itself can be treated as a clustering problem. Besides, our further studies (detailed description in Section IV) show that prediction of popularity classes suffers from imbalanced class fractions, hence we also need to design a robust classification model to deal with this problem.

The training data we used in our model is collected online with a pipeline involving multiple crawling threads on multiple cloud virtual machine (EC2) from different data sources. Structured training data is extracted by various data preprocessing algorithms from raw data according to the nature of different types of raw data, e.g. topic modeling algorithms to obtain refined categorical description of video content from textual data related to a YouTube channel (detailed description in Section IV).

Along with training, validation, and further refinement of our predictive model, we also aim to obtain the ranking of relative importance of features considered in our model which is important for further inference (detailed description in Section VI). At last, we also developed a YouTube channel popularity visualization tool and item-based

recommender to apply the result obtained from our previous study (detailed description in Section V).

The remainder of this paper is organized as follow: In Section II we will summarize the state of the art research in related topics; In Section III we will unfurl frameworks and details of our data collection and data analysis procedure. Section IV and Section V introduces the algorithms we applied in our study as well as the visualization tool we developed. Section VI presents the results and we will conclude our work in Section VII.

II. RELATED WORKS

To the best of our knowledge, lion's share of works in related topics have been focusing on predicting individual video instead of content generator itself, i.e. YouTube channel, which is the major objective in our project. However, it is still worth summarizing previous works in prediction of web content popularity to have a comprehensive understanding of previous models (machine learning, time series, probabilistic, etc.) adopted to quantify the problem, features that are proofed to be correlated with our target response and methodologies used for inference.

According to the summary of prediction models in a survey on predicting the popularity of web content conducted by Tatar et al. [3], previous research in this relevant topic can be roughly categorized as "Single Domain" and "Cross Domain", where "Single Domain" is defined as a study of video popularity regardless if the video has been created or shared from an external source while "Cross Domain" studies expand their scope to information across different source of websites.

Under this binary categorization, relevant studies can be further categorized according to their research objectives: predicting on popularity before the publication of a web content and after publication. While before publication prediction can be very challenging due to the fact that such studies usually only rely on quantification of video content metadata, after publication prediction is a more popular choice and research in this objective can be summarized as three major topics according to the quantification of target: (i) Study the cumulative growth of attention; (ii) Perform a temporal analysis of how content popularity evolves over time until the prediction moment. (iii) Use clustering methods to find web items with similar popularity evolution trends. Since these three categories of studies on aggregated user's attention of online videos make up most of the previous research, figuring out difference in objectives and state-of-art methodologies in these three areas is critical to the designing of our analysis.

(i) Cumulative growth: numbers of statistical learning and machine learning models are applied in this field of study. Important examples are: a statistical predicting model (mixture of two log-normal distributions) proposed by Kaltenbrunner et al. [15] to predict popularity of Slashdot stories; Lee et al. [16] proposed a survival analysis based Cox proportional-hazards regression to predict if a web content will have an increasing attention after certain period of time; Several studies involve regression analysis [17] [18] [19]; Classification models are also widely applied, [20] applied simple logistic regression model to address tweet classification, [22] used Random Forests to identify comments of online articles, [21] adopted SVM, Naïve Bayes and tree methods to predict popularity range of articles.

(ii). Temporal analysis: In early stage of research related to prediction of online video popularity, numbers of studies aimed at modeling video access pattern of users [3]. Important conclusions in these studies are: public attentions of web content such as online video's is generated in a transient and often unpredictable fashion [3] [4] [7] and pattern of users' requests for web pages can be modelled as distributed to Zipf 's law [5]. These conclusions are cornerstones which indicate that popularity of videos in different stages of its lifetime are highly correlated hence prediction of video popularity can be treated as a time series analysis problem in different lifetime stages [3] [6] [7], proceeded with numbers of studies focusing on study of early patterns of video popularity generation and found out that a video's long-term popularity is often determined, and can be predicted from its early views [8] [9] [10].

(iii). Clustering of evolution trends: Along with modelling in time series, a number of studies have investigated probabilistic characters of popularity prediction which show that popularity growth of videos over time can be represented by power-law or exponential distributions including Poisson distribution [3] [13] and numbers of representative time series evolution patterns of video popularity can be found [14].

Though appealing as they sound, early pattern recognition and times series analysis are not applicable in our case since it is generally very difficult to track or even define the early stage of content creators and videos of a channel can be spread and influenced by other communication medium in our internet as a global information ecosystem [3] [11] [12], e.g. online social media such as Twitter. In this case, our problem can be naturally categorized as a "Cross Domain" study in [3] if we consider relevant information from other data sources in our model.

Some interesting examples of "Cross Domain" have also been listed in [3]. Oghina et al. [22] incorporates textual features extracted from Twitter data and statistics from YouTube in a simple linear regression model to predict movie ratings on IMDb; Roy et al. [23] leverage real time

analysis of Twitter topics comparing to YouTube videos to detect disproportionate share of attention on Twitter compared to YouTube with a SVM classifier hence detect potential sudden burst of popularity on YouTube. These two important examples along with other similar research indicates that incorporating related information from multiple sources (e.g. Twitter) can significantly improve predictive accuracy on web-content, especially those disseminated in multiple web media (e.g. YouTube videos). This is because social media data stream can provide additional perspectives about the true popularity of videos outside of the originating web domain [3].

Since our design of analysis is to treat our problem as a static multi-class classification problem to predict on classes of popularity range and inspired by research in [22] [23] we will integrate information related to YouTube channels from social media such as Twitter, our paper can be summarized as a "Cross Domain" study which aims to predict the "after publication cumulative growth of attention" according to the categorization system proposed in [3].

III. SYSTEM OVERVIEW

1) Data and Features

Before we get into details of collecting data and training our model, we need to specify the features we need to extract from raw data for prediction and inference. The features we used can be summarized in three categories.

Direct quantitation of channel popularity:

- Numbers of views/subscribers/comments

Accumulated numbers of views, subscribers and comments. This information is directly related to popularity of a video channel, which we will need to further process to be our prediction response.

Viewers opinion:

- Ratio of likes/dislikes, comment/views, favorite /views (aggregation of videos)
- Sentiment analysis score of video comments
- Social media reference score

We do not know the relationship between channel popularity and its viewers' opinion, thus apart from the statistics directly indicating popularity of a video channel, we also need to consider those factors reflecting viewers' attitude towards videos of a YouTube channel.

Features in this category (and in later "Quantitative Description of Channel and Its Content") requires us to compute a specific metric of each individual video of a YouTube channel and aggregate them together. The method we used to treat these kind of features is as follow:

$$Q_i = \frac{\sum_{j=1}^{10} v_{ij} Q(x_{ij})}{\sum_{j=1}^{10} v_{ij}}$$

Where Q_i indicates a quantitation of the i^{th} channel and it is computed by the average of the corresponding metric $Q(x_{ij})$ of each of top 10 videos of this channel weighted by video's number of views v_{ij}.

Aggregated ratio of numbers of likes and dislikes, ratio of numbers of comments and views, ratio of numbers of favorites and views among the top 10 videos of a channel can roughly indicate if a channel are favored by its viewers and if the work of this channel has made an impact (i.e. attract viewers to discuss). The reason we used ratio here is that we want to maximally eliminate the effect of time (though these features might evolve with time) since our model is static.

YouTube API also provides us with comments of videos. By applying sentiment analysis of all the comments of top 10 videos of a YouTube channel, we can obtain a more refined quantitation of viewers' attitude.

As we have discussed before, dissemination of a channel's videos involves several other web media, and it is very necessary to look into these alternative data sources since they might provide a portion of popularity that can not be fully explained by internal information in YouTube. In our project, we used Twitter as an example. Under the assumption that additional views can be brought by reference of YouTube videos on Twitter and each Twitter referrer's relative importance can be represented by its number of followers, we will track the top 10 referrers of a channel's top 10 videos respectively, and aggregate them as a sum of weighted average (details in Section IV).

Quantitative description of channel and its content:

- Frequency of publish
- Duration
- Content category
- Topics obtained from topic modeling on textual description
- Named entity recognition score
- Characteristics of content creator: social media behavior

Inspired by studies in "Before Publication" [3], we also need to consider a quantitative description of YouTube channels and their content. The most straightforward way of quantifying the content created by a video channel should be data mining of video metadata, like [2]. However, analysis of large scale video metadata is computationally expensive and explanatory power of features extracted from video metadata with current techniques on video popularity is quite unclear. Hence in our project we will focus on category of video topics. Note that YouTube API does

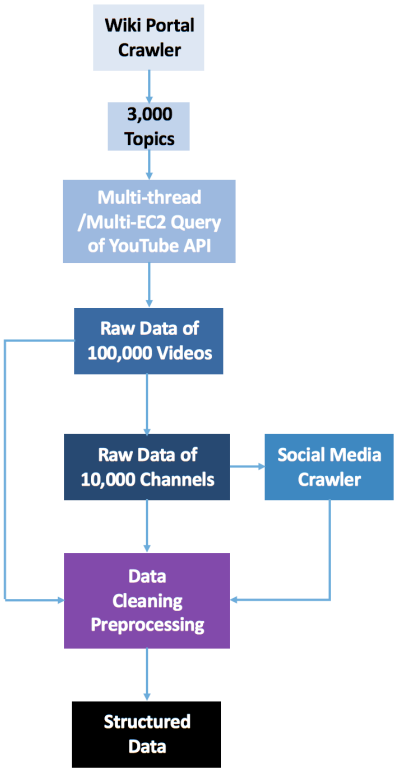
provide such information, but research which considered content category shows a low predictive performance of using this information. This can be explained by the fact that a video of a channel can usually have several overlapping content categories [3] [9] [22]. To solve this problem, we can apply topic modeling algorithms to video descriptions to obtain a more refined categorical description of video topics (i.e. multiple topics for a single video of a channel).

We have also considered the fact that occurrence of famous entities (celebrity, famous places, etc.) in video can potentially increase the popularity of a channel, hence with the textual data of video content description we can also apply a named entity recognition algorithm to capture the occurrence of named entities. The "Named entity recognition score" is the average of accumulated numbers of such occurrence in the top 10 videos of a channel weighted by each video's total numbers of views.

Moreover, we can also track YouTube channels' social media accounts and analyze the relationship between their interactive behavior and their popularity. For simplicity, we used Twitter accounts as example and use numbers of followers to estimate a channel owner's influence on social media.

2) Data Collection

Plot 3.1 shows our data collection pipeline:



Plot 3.1 Data Collection Pipeline

To start with, we used a crawler to obtain a dictionary of topics covering commonplace public interests from Wikipedia, with these topics as searching key words we collected data of around 30,000 videos from YouTube API. Note that in order to improve our data collecting efficiency we deployed several EC2 to collect data with multi-thread.

With information of 10,000 videos we can further retrieve a list of YouTube channels who published these videos as well as information of these channels. Again, since our research target is channel, we need to go back and query for data of videos published by these channels (averagely 10 videos per channel hence approximately 100,000 videos in total).

With video IDs related to each channel we can implement crawler to track these video's reference on Twitter and obtain their referrer's information (the top 10 referrers were considered). With channel IDs as keywords we can also track the information of channel owners on Twitter.

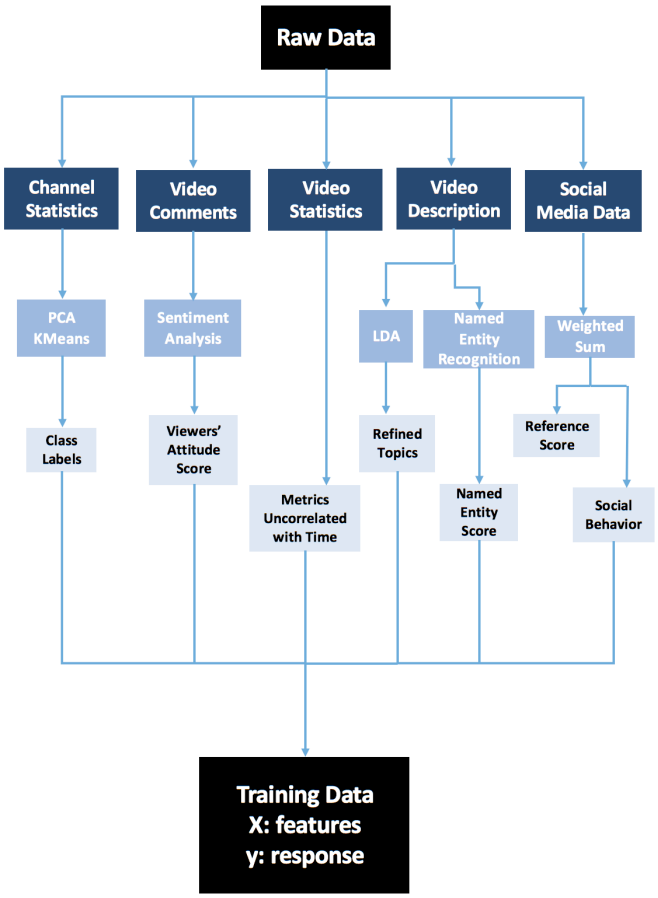
All these information including data of around 10,000 YouTube channels, 100,000 videos related to these channels and relevant information crawled from Twitter will be fed to our data cleaning and preprocessing algorithms to obtain structured data for model training.

3) Data Preprocessing and Machine Learning

Follow the feature schema we described earlier, we applied several data preprocessing algorithms to extract features we need from raw data, see plot 3.2.

Intuitively our machine learning model can predict on numbers of views to represent popularity (which will be a regression problem), however we think accurate prediction on views is not plausible or meaningful. We instead predict on relative range of popularity. Due to the fact that we lack of domain knowledge to determine such range, we have to use clustering models to discover these potential ranges.

We applied K-Means algorithm on channel statistics including accumulated numbers of views, comments, subscribers, number of videos which are directly correlated with popularity of a channel. As we obtain the result of K-Means, i.e. popularity clusters, we can check the range, mean, standard deviation of different clusters to see if our assumption on cluster number is reasonable and if channel popularity really has underlying clusters. Different cluster's range of views can be used to assign labels for each channel, hence we get our quantification of channel popularity and response for prediction. Apart from clustering, we also applied PCA on channel statistics to achieve low dimensional approximation of channels, these approximations will be used for visualization.



Plot 3.2 Data Preprocessors

Sentiment analysis algorithms are applied to compute viewer's attitude scores of channels which are computed as follow:

$$S_i^{(k)} = \frac{\sum_{j=1}^{10} v_{ij} S^{(k)}(x_{ij})}{\sum_{j=1}^{10} v_{ij}}$$

$$S^{(k)}(x_{ij}) = \frac{\sum_{t=1}^{T_{ij}} s_{ijt}^{(k)}}{T_{ij}}$$

T_{ij} = Number of comments of the jth video of the ith channel

$k \in \{\text{Positive, Negative, Compound, Neutral}\}$

Each comment of a video will be analyzed and result in "positive/negative/neutral/compound" sentiment scores $s^{(k)}_{ijt}$ and we can compute average sentiment scores of a video hence aggregate them to a weighted average as we demonstrated in Section III part 1).

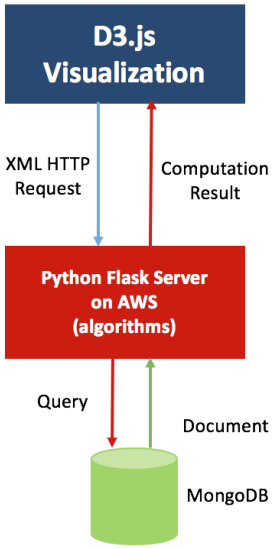
As for processing textual description of videos, we applied LDA as our topic modeling algorithm to get multiple topics of a video. The named entity algorithm is also applied to get numbers of famous people, places and organizations occurred in video description. Further aggregation to channel level is similar to viewers' attitude score.

As we discussed earlier, video reference score of a channel is computed as a weighted average of followers of top 10 referrers of this channel's top 10 videos on Twitter. Social behavior of channel owner is simply represented by number of followers on Twitter in unit of 1,000.

Each of the features we mentioned above will be integrated to a data point which represent a video channel, hence we obtain matrix X as all the training data (all the categorical features such as video topic categories are encoded as dummy variables). With the result of clustering analysis on channel popularity quantitation we can assign labels to each data point indicating which popularity class does this data point belongs to and get the response vector y. Matrix X and vector y will be our structured training data.

In our analysis, we applied Random Forests as our machine learning model since Random Forest does not require a strict probabilistic assumption on data generating process and it is a non-linear classifier which will be suitable for our case. More importantly, Random Forest can quantify the relative importance of each feature using permutation importance measure [25], which we will leverage for further inference.

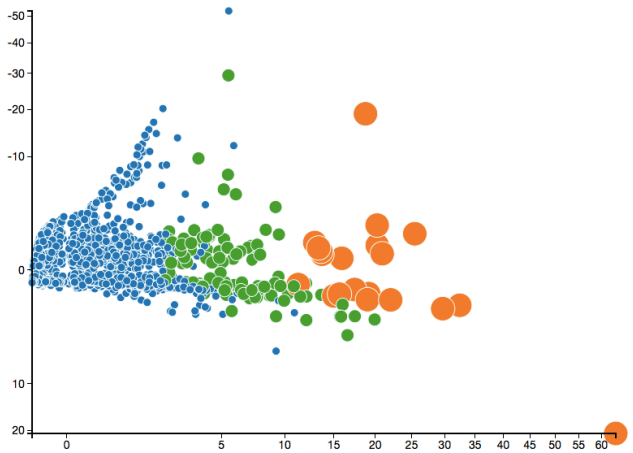
4) *Visualization and Recommendation*



Plot 3.3 Visualization and Recommendation Architecture

To visualize different channels, we built an interactive bubble chart webpage, which vividly demonstrates categorizations of video channels in the chart and key features of observations in a table. To effectively plot each video channel in two-dimension plane, we take the first principle component in PCA analysis as X-axis and the

second principal component as Y-axis as plotted in Plot 3.4. Different colors and sizes of bubbles denote the popularity of video channels. The bigger the bubble is, the more popular the video channel is.



Plot 3.4 Visualization of Video Channels

We implemented the visualization functionality using D3 Javascript Library and Boostrap at the front end and Python Flask and MongoDB at back end. D3 Javascript can immediately react to user's mouse activities, and send request to our back-end server, and the server retrieves the documents stored in MongoDB and send the JSON data back to browser at client side. Upon data being loaded, Javascript will update the tables and charts that are viewed by users.

Another key aspect of our application is an item-based recommendation system, which leverages both users' preferences and the similarity among video channels' similarities. The user is required to input their favorite five video channels in descending order. The first channel represents the video channel that the user likes most. We assigned these 5 channels different weight, say, weight = [5,4,3,2,1]. Then we calculate each video channel's distances to these 5 videos picked by user and compute the weighted average of these distances. The videos with least weighted distances are recommended to users. The main functionality of recommendation system is implemented in Python Flask backend, which responds recommended channels' JSON data to the front end's request.

IV. ALGORITHM

In this section we will present overview as well as some necessary details of algorithms and tools we proposed to use to solve the problems we've modeled so far.

1) PCA

In our visualization of direct quantitation of channel popularity, since we are considering numbers of views, comments, subscribers and comments, each channel as a data point will be in R^4 and we need a lower dimensional representation. PCA as a state-of-art lossy dimensionality reduction algorithm provides a cheap way to achieve our goal. We will find a two dimensional space spanned by two eigenvectors of empirical covariance matrix of our data with largest eigenvalues and approximate our data by projecting on this newly defined space. We first computing empirical covariance of data:

$$\Sigma = \frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)(x_i-\mu)^T$$

$$\mu = \frac{1}{N}\sum_{i=1}^{N}x_i$$

Then we will define our problem as finding the direction:

$$\max_{v} \quad v^T\Sigma v$$

$$\text{s.t.} \quad ||v|| = 1$$

In our case this is equivalent to finding the eigenvectors of covariance matrix which have largest eigenvalues and all the data points can be approximately represented as:

$$x_i' = (\xi_1^T x_i)\xi_1 + (\xi_2^T x_i)\xi_2$$

2) K-Means

With K-Means we can find underlying clusters of channel popularity. Note that K-Means is not a model selection method which means we do not initially know how many clusters are reasonable thus we need to consider several potential choice of clusters (2,3,4,5,6,7). We can use basic statistics of clustering result as validation, here is a basic pseudo code of K-Mean algorithm:

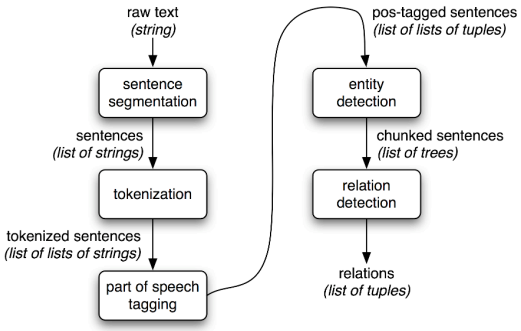
Set K as 2,3,4,5,6,7:
(1)Randomly initialize K cluster centers $\mu_1^{(0)}, ..., \mu_K^{(0)}$
(2)While $\sum_{k=1}^{K} ||\mu_k^{(t)} - \mu_k^{(t-1)}|| > \epsilon$:
Assign each data point x_i to the closest (in l_2-norm) cluster center:
$c_i^{(t+1)} = \arg\min_{k\in\{1,...,K\}} ||x_i - \mu_k^{(t)}||$
Update cluster centers as mean of data points assigned to them:
$\mu_k^{(t+1)} = \frac{\sum_{i:c_i^{(t+1)}=k} x_i}{\sum_{i=1}^{N} \mathbb{1}\{c_i^{(t+1)}=k\}}$
(3)Compute relevant statistics of each cluster

3) Sentiment Analysis

We applied NLTK VADER sentiment analysis tool [26] to process our video comments. This algorithm does not require training data and gives quantitation of positive/negative/neutral compound sentiment scores.

4) Named Entity Recognition

We also applied NLTK [27] to deal with named entity recognition problem. Framework of NLTK named entity recognition algorithm is shown below:



Plot 4.1 Named Entity Recognition Framework

Raw text of the video comments will be segmented into sentences and then sliced to words. Each sentence will be tagged with part-of-speech tags which helps identifying and validating potential named entities.

5) Topic Modeling

In topic modeling we applied Latent Dirichlet Allocation (LDA) aim to find a refined categorical topics of video descriptions. The assumption on data generation process is that we have K topics, a dictionary of V terms, D documents, the d^{th} document has N_{d} words in total and each word has an encoded value x_{di} corresponds to a specific term in dictionary. β_{k} is a V dimensional vector representing discrete distribution of topic k on V terms and θ_{d} is a K dimensional vector representing distribution of the d^{th} document on K topics:

$$c_{di} \sim \text{Discrete}(\theta_d), \quad x_{di} \sim \text{Discrete}(\beta_{c_{di}}).$$

$$\theta_d \overset{iid}{\sim} \text{Dirichlet}(\alpha), \quad \beta_k \overset{iid}{\sim} \text{Dirichlet}(\gamma)$$

c_{di} indicates the topic which the i^{th} word of the d^{th} topic belongs to which is generated by a discrete distribution of parameter θ_{d}, and value i.e. the encoded term that the i^{th} word of the d^{th} topic corresponds to is denoted by variable x_{di} and generated by a discrete distribution of $\beta_{c\text{di}}$. If we treat LDA as a Bayesian model, the model parameters θ_{d} and β_{k} have their own Dirichlet prior distribution.

From the perspective of application in our case, we need to estimate θ_{d} which shows the probability of different topics on the d^{th} document (i.e. textual description for the d^{th} video), hence we can get the top three topics with greatest probability for the video.

6) Random Forest

Random Forest is a robust, non-linear classifier and it does not require probabilistic assumptions on data generating mechanism which matches our requirement of our channel popularity prediction model. Random Forest is an ensemble tree method, the training process includes bootstrapping samples from training data and training decision tree on each of the resampled dataset with restrictions on tree depth and considered features in each node split. Prediction for new data point is achieved with majority vote (which class obtained the highest vote among all the classification trees in our model). Pseudo code of Random Forest is shown below:

Initialize:
  numbers of features m considered in each split
  depth D of tree
  number of trees T
For t = 1,...,T:
  Bootstrap sampling B_t
  Train a classification tree on B_t
    While depth < D:
      Randomly sample m features for split
      Find the best split (j, s) based on a certain loss criteria
      Split on (j, s)

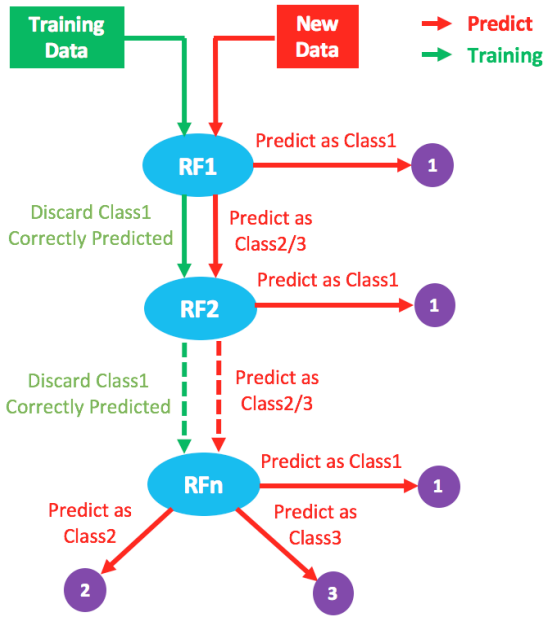
As we will see in details in Section VI, our training dataset has imbalanced classes which means one of the popularity classes has dominating portion while others only compose minority of all data. This will typically result in good prediction accuracy on the dominating class and overall. However, prediction accuracy for other classes might be worse. Following derivations support this point, suppose we have three classes and class "1" is the dominating class, our training process will keep the error rate of class "1" low to achieve a low overall error rate since its portion is significantly larger than the other two classes.

$$\begin{aligned}
Error &= \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\{f(x_i)\neq y_i\} \\
&= \frac{1}{N}\sum_{i:y_i=\text{"1"}}\mathbb{I}\{f(x_i)\neq y_i\}+\frac{1}{N}\sum_{i:y_i=\text{"2"}}\mathbb{I}\{f(x_i)\neq y_i\}+\frac{1}{N}\sum_{i:y_i=\text{"3"}}\mathbb{I}\{f(x_i)\neq y_i\} \\
&= \frac{N_1}{N}\left(\frac{N_1}{N}\sum_{i:y_i=\text{"1"}}\mathbb{I}\{f(x_i)\neq y_i\}\right)+\frac{N_2}{N}\left(\frac{N_2}{N}\sum_{i:y_i=\text{"2"}}\mathbb{I}\{f(x_i)\neq y_i\}\right)+\frac{N_3}{N}\left(\frac{N_3}{N}\sum_{i:y_i=\text{"3"}}\mathbb{I}\{f(x_i)\neq y_i\}\right) \\
&= \frac{N_1}{N}\{Class\text{"1"}\,ErrorRate\}+\frac{N_2}{N}\{Class\text{"2"}\,ErrorRate\}+\frac{N_3}{N}\{Class\text{"3"}\,ErrorRate\}
\end{aligned}$$

Similar to the trade-off in sensitivity and specificity in binary classification, with a single Random Forest classifier it is very difficult to achieve low error rate for all three classes. One potential solution is cascaded classifiers such as Viola-Jones Detector in face recognition [28]. Similar to Viola-Jones Detector, we can design a cascaded Random Forest classifier, the framework is shown in plot 4.2.

While training our cascaded Random Forest classifier, we discard correctly predicted data points with class 1 in current classifier and train the classifier in next level with the truncated training data. As for prediction, the new data point will move from the top level classifier to the bottom until it is classified as class 1, if the data point comes to the bottom it will be classified as 2 or 3. In this way, classes in training data for each classifier will be gradually balanced and even though each of the classifier will still have a relatively high error rate in predicting class 2 and 3,

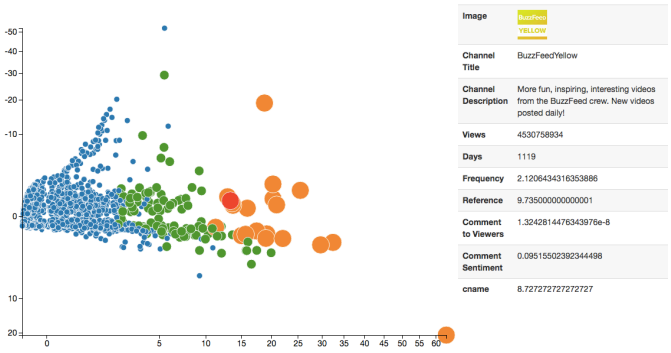
cascading them together will achieve a good overall classification accuracy for these two classes.



Plot 4.2 Cascaded Random Forest

V. SOFTWARE PACKAGE DESCRIPTION

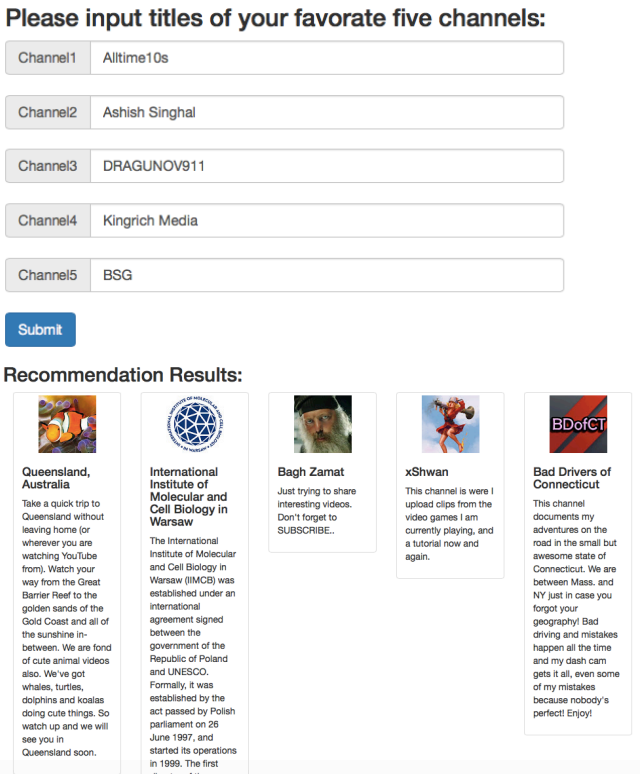
The first part of our software package is the data collection libraries and scripts, which provides an easier-to-use API suited for big data analytics applications. We also contributed an API that can crawl Wikipedia topics from Wikipedia Portal pages. The video data collection functions take topic words as input and download massive video data related to these topics as JSON files which is the friendliest to document-based database, like MongoDB and DynamoDB. To reduce the latency caused by HTTP request to YouTube API, we wrote multi-threading functions that can speed up the data collection by 10 times. The function that merge separate dataset make it easier to deploy distributed systems such as Amazon EC2.



Plot 5.1 Visualization of Video Channels

The next part of our software package is video channel data visualization built as an interactive web application, as demonstrated in Plot 5.1. Each video channel is represented as a bubble on the bubble chart and it is easy to characterize the popularity of different video channels. By clicking on the bubble, the website can dynamically show the statistics and contents of each video channel, which is highly user friendly. User can explore more features of each channel that are not available on YouTube.

The final part of our software is a recommendation system based on users' preferences and similarity of video channels' features. Such similarity is measured on features obtained from our previous analysis which means our recommender incorporates novel factors such as YouTube channels' interaction on social media, named entity recognition, etc. (see details in Section III part 1)



Plot 5.2 Video Channel Recommendation System

The user can input their 5 favorite video channels on our website as shown in Plot 5.2 and the back-end server computes the top 5 recommended videos channels.

VI. EXPERIMENT RESULTS

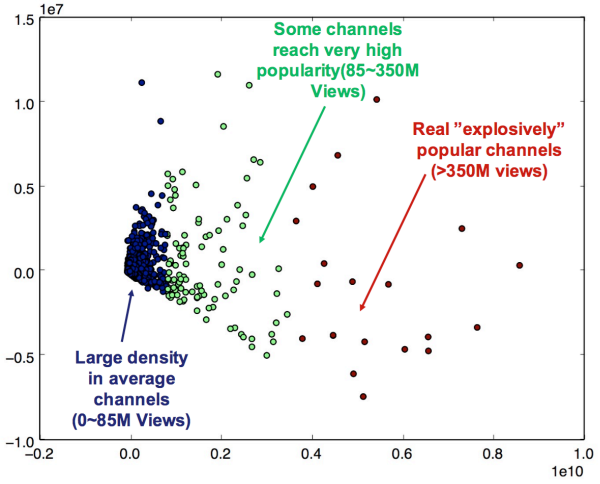
1) Clustering and PCA

When we applied K-Means clustering to direct quantitation of channel popularity, we conducted trials on different numbers of clusters (K = 2, 3, 4, 5). Statistics in accumulated views for these 5 cases is shown in Table 5.1

K	Means (Million)	Std (Million^2)	Range (Million)	Portion (%)
2	$m_1 = 3.2$ $m_2 = 362.2$	$s_1 = 13.0$ $s_2 = 164.3$	$R_1 = (0, 180.8)$ $R_2 = (180.8, 864.1)$	$N_1 = 99.4$ $N_2 = 0.6$
3	$m_1 = 2.3$ $m_2 = 155.3$ $m_2 = 533.7$	$s_1 = 7.5$ $s_2 = 70.8$ $s_3 = 136.1$	$R_1 = (0, 78.2)$ $R_2 = (79.2, 333.1)$ $R_3 = (352.0, 864.1)$	$N_1 = 98.5$ $N_2 = 1.2$ $N_3 = 0.3$
4	$m_1 = 1.7$ $m_2 = 83.2$ $m_3 = 266.2$ $m_4 = 579.4$	$s_1 = 5.1$ $s_2 = 34.0$ $s_3 = 62.5$ $s_4 = 123.9$	$R_1 = (0, 42.3)$ $R_2 = (42.7, 173.5)$ $R_3 = (179.3, 418.5)$ $R_4 = (433.2, 864.1)$	$N_1 = 97.6$ $N_2 = 1.8$ $N_3 = 0.5$ $N_4 = 0.1$
5	$m_1 = 1.0$ $m_2 = 31.8$ $m_3 = 106.0$ $m_4 = 270.7$ $m_5 = 579.5$	$s_1 = 2.3$ $s_2 = 12.5$ $s_3 = 29.8$ $s_4 = 60.83$ $s_5 = 123.9$	$R_1 = (0, 16.3)$ $R_2 = (16.3, 67.8)$ $R_3 = (69.2, 180.8)$ $R_4 = (188.5, 418.5)$ $R_5 = (433.2, 864.1)$	$N_1 = 94.8$ $N_2 = 3.6$ $N_3 = 1.1$ $N_4 = 0.4$ $N_5 = 0.1$

Table 6.1

Roughly speaking, the result of case K = 3 and case K = 4 are more reasonable. Considering the fact that portion of class "4" in case K = 4 is only 0.1% which will cause a problem in future prediction, we chose K = 3. As we can see all three clusters have different and distinct ranges which means we can assign labels as prediction response to our training data points according to their range of accumulated views.



Plot 6.1 K=3 clusters (in PCA)

We can approximate our data points with the first two principle components obtained from the result of PCA and visualize these three clusters (see Plot 6.1). As we can see, channels with relatively low popularity (in blue dots) are clustered together (in respect to distance in two dimensional space spanned by PC) and have small variance, this indicates that these are the regular channels which comprise the majority; channels with relatively high popularity (in green dots) are more scattered from majority in the space; red dots denote channels with "explosively" high popularity and are most noteworthy, their popularity outweighs all the other YouTube channels which means they can be potential targets for funders. However, these channels have a pretty high deviation hence accurate prediction can be difficult.

2) LDA

Ideally we would like to apply LDA to process video textual descriptions and get refined categorical quantification of video topics with estimated distribution of topics on dictionary as well as distribution of topics on documents (i.e. video descriptions). However, our LDA processor failed to produce reasonable topics as shown in Table 6.2.

Topics	Top 10 Terms in Topic
1	World(0.005), part(0.005), wing(0.005), music(0.004), let(0.004), m(0.004), ramp(0.004), whale(0.004), sim(0.004) , can(0.004)
2	Muslim(0.009), world(0.006), 9(0.006), antastesia(0.005), senat(0.005), en(0.004), new(0.004), album(0.004), download(0.004), itun(0.004)
3	en(0.009), ingress(0.008), un(0.006), learn(0.006), la(0.006), coraz\xf3n(0.006), madera(0.006), sign(0.005), vanguard(0.005), van(0.005)
4	d\xe2n(0.009), c\u1ee7a(0.008), 2013(0.007), nh\u1eefng(0.007), \u0111\u1ea5t(0.006), 2014(0.006), 4(0.005) , v\xe0(0.004), ng\u01b0\u1eddi(0.004), februari(0.004)
5	us(0.008), Disney(0.006), documentary(0.006), fighter(0.006), t(0.005), miley(0.005), world(0.005), jet(0.004), m(0.004), watch(0.004)
6	nbc(0.012), night(0.010), late(0.010), armi(0.006), us(0.005) , seth(0.005), tumblr(0.005), latenightseth(0.005), meyer(0.005), documentary(0.005)
7	mahi(0.027), o(0.018), ve(0.016), que(0.015), hai(0.011), eu(0.008), vey(0.007), um(0.007), suna(0.007), ho(0.007)
8	randleman(0.01), \u0b95(0.01), \u0ba4(0.009), \u0bb0(0.008), \u0baa(0.007), one(0.007), \u0bae(0.007), part(0.007), watch(0.006), v(0.006)
9	will(0.008), coryxkenshin(0.007), music(0.005) , danisnotonfir(0.005), even(0.005), far(0.005), like(0.003), look(0.003), help(0.003), thank(0.003)
10	refract(0.009), angl(0.009), inform(0.009), follow(0.006), incid(0.006), like(0.006), pleas(0.005), youtu(0.005), gmail(0.005), reflect(0.005)

Table 6.2 LDA Result

Table 6.2 shows the result of applying LDA on video descriptions when we assume there are 10 latent topics and 10 terms with the highest probability in topic distribution on dictionary are considered. As we can see, most of the keywords given by these 10 topics do not makes sense and the result is not applicable to estimate video content topics.

However, from Table 6.2 we can roughly analyze the reason why LDA failed on our data. First of all, LDA requires neat preprocessing of text data, including parsing documents to words and filtering high frequency terms. This can be difficult for our data because video description given by YouTube is filled with meaningless codes such as "\u0111" which are difficult to be identified and filtered out. This partly explains key words of topic 4. Secondly, textual video descriptions on YouTube are usually short and sometimes have very low quality, e.g. not highly correlated with the real video content.

3) Random Forest and Prediction

For single Random Forest classifier (tree number = 10) trained on features we selected, we independently collected another set of test data (preprocessed) and apply our model to predict on new data, result is summarized in Table 6.3

Overall Error Rate	Class 1 Error Rate	Class 2 Error Rate	Class 3 Error Rate
1.86% (20/1070)	1.44% (15/1038)	16% (4/25)	14.2% (1/7)

Table 6.3 Single Random Forest Test Error

The single Random Forest achieves accurate overall classification and low error rate for class 1 (the dominating class). However, the error rate for class 2 and 3 is high (see details in Section IV part 6). To solve this problem, we can apply the cascaded Random Forest classifier (with 3 cascaded models) on test data, result is shown in Table 6.4.

Model	Overall Error Rate	Class 1 Error Rate	Class 2 Error Rate	Class 3 Error Rate
Overall	1.02% (11/1070)	0.87% (9/1038)	4% (1/25)	14.2% (1/7)
Classifier #1	1.86% (20/1070)	1.44% (15/1038)	16% (4/25)	14.2% (1/7)
Classifier #2	1.40% (15/1070)	1.05% (11/1038)	12% (3/25)	14.2% (1/7)
Classifier #3	1.40% (15/1070)	1.16% (12/1038)	5% (1/25)	0% (0/7)

Table 6.4 Cascaded Random Forest Test Error

From Table 6.4 we can see that cascaded Random Forest achieves a low error rate for both dominating class and minority class (class 2 & 3). From classifier 1 to 3, the class fractions in training dataset are gradually balanced and result in decreasing error rate for minority classes. (Note that here the classification accuracy for class 3 hasn't changed, this is due to the fact that the size of our test dataset is not big enough).

4) Ranking of Features and Topics

Note that Random Forest also gives score of relative importance of features hence we can get ranking of all the data features we've studies. Results are summarized in Table 6.4.

Order	Feature	Score
1	Frequency of publishing videos	0.103
2	Reference on social media	0.119
3	Neutral/Compound comment sentiment	0.096
4	Activity of channel owner on social media	0.082
5	Rate of subscription	0.075
6	Rate of "Likes"	0.055
7	Rate of comments	0.055
8	Positive comment sentiment	0.051
9	Occurrence of named entities	0.045
10	Duration	0.026
11	Negative sentiment	0.01

Table 6.5 Feature Importance

This result confirms our previous assumption that interaction between YouTube channel and social media such as Twitter has positive correlation with a high popularity (Twitter reference score and channel's interaction on Twitter are ranked as 2nd and 4th). Furthermore, since we are predicting on a variable related with time thus it is intuitive to assume that duration (how long has a YouTube channel been existed) will be the most important feature. To our surprise, this is not the case, duration is actually ranked as the 9th important feature and the most important feature is a channel's frequency of publishing video. In this case, publishing video more frequently might potential promote a channel's popularity.

Note that we've encoded our video content categories as dummy variables, hence with feature importance we can also obtain the rank of video topics that are favored by YouTube users (see Table 6.6).

VII. CONCLUSION

1) Conclusion

In this paper, we conducted a comprehensive data-driven study on influential factors of YouTube channel popularity. From our analysis, we discovered that video popularity can

Order	Topics
1	Comedy
2	Drama
3	Horror
4	Documentary
5	Education
6	People and Blogs
7	Anime/Animation
8	Foreign
9	Nonprofits & Activism
10	Family
11	Anime/Animation
12	Sci-fi
13	Thriller

Table 6.6 Topic Ranking

be quantified as 3 clusters with distinct range in accumulated views. We also designed and implemented a cascaded Random Forest classifier to address with the imbalanced class in training data and result shows that cascaded Random Forest achieves low error rate for both dominating class and minor classes. Random Forest classifier also provides evidence to show relative importance of different features: frequency of publishing videos, interaction of content creator and reference of its videos on online social media are significant factors to promote popularity of a YouTube channel.

Apart from data analysis, we've also developed tools for data collection, data preprocessing, visualization, item-based YouTube channel recommendation.

2) Future Work

If we can somehow accurately describe content of a YouTube channel's videos in quantitative fashion, we can conduct a true "Before Publication" study. To achieve this, we can work on more robust topic modeling algorithms to process video textual information and look into data mining of video metadata.

3) Contribution

Ziyu He:

-Data Collection
-Data Preprocessing
-Machine Learning
-Analysis

Haoxiang Gao:

-Data Collection
-Visualization & Recommender
-Server

ACKNOWLEDGMENT

THE AUTHORS WOULD LIKE TO THANK PROFESSOR CHING-YUNG LIN AND ALL TAS AS WELL AS LECTURERS FOR PROVIDING SUCH AN EXCELLENT COURSE

REFERENCES

[1] YouTube Statisitics (2015): https://www.youtube.com/yt/press/en-GB/statistics.html

[2] T. Trzcinski and P. Rokita "Predicting popularity of online videos using Support Vector Regression" arXiv:1510.06223 [cs.SI]

[3] A. Tatar, M. Amorim, S. Fdida, P. Antoniadis "A survey on predicting the popularity of web content" Journal of Internet Services and Applications 2014, 5:8

[4] L. Cherkasova, M. Gupta "Analysis of enterprise media server workloads: access patterns, locality, content evolution, and rates of change". IEEE/ACM Trans Netw 12(5)7: 81–794

[5] L. Breslau, P. Cao, L. Fan, G. Phillips, S. Shenker "Web caching and Zipf-like distributions: Evidence and implications". In: INFOCOM'99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 1. IEEE, New York, NY, pp 126–134

[6] G. Szabo, B. Huberman "Predicting the popularity of online content" .CommunACM53(8)8:0–88

[7] M. Cha, H. Kwak, P. Rodriguez, Y. Ahn, S. Moon "Analyzing the video popularity characteristics of large-scale user generated content systems". IEEE/ACM Trans Netw. (TON) 17(5)1: 357–1370

[8] G. Gursun, M. Crovella, I. Matta "Describing and forecasting video access patterns". In: INFOCOM, 2011 Proceedings IEEE. IEEE, Shanghai, pp 16–20

[9] H. Pinto, J. Almeida, M. Gonçalves "Using early view patterns to predict the popularity of youtube videos". In: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining. WSDM '13. ACM, Rome, Italy, pp 365–374

[10] H. Yu, L. Xie, S. Sanner "The Lifecyle of a YouTube Video: Phases, Content and Popularity" Proceedings of the Ninth International AAAI Conference on Web and Social Media

[11] S. Roy, T. Mei, W. Zeng, S. Li "Towards cross-domain learning for social video popularity prediction". IEEE Trans Multimedia 15(1255-1267)

[12] A. Oghina, M. Breuss, M. Tsagkias, M. de Rijke "Predicting IMDb movie ratings using social media" In: Proceedings of the 34th EuropeanConferenceonAdvancesinInformationRetrieval.ECIR'12.Springer, Barcelona, Spain, pp 503–507

[13] Z. Avramova, S. Wittevrongel, H. Bruneel, D. De Vleeschauwer, "Analysis and modeling of video popularity evolution in various online video content systems: power-law versus exponential decay". In: First International Conference on Evolving Internet (INTERNET'09). IEEE, Cannes/La Bocca, pp 95–100

[14] F. Figueiredo "On the prediction of popularity of trends and hits for user generated videos". In: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining. WSDM '13. ACM, Rome, Italy, pp 741–746

[15] A. Kaltenbrunner, V. Gomez, V. Lopez, "Description and prediction of slashdot activity" In: Web Conference, 2007. LA-WEB 2007. Latin American. IEEE, Santiago, Chile, pp 57–66

[16] J. Lee, S. Moon, K. Salamatian "Modeling and predicting the popularity of online contents with Cox proportional hazard regression model". Neurocomputing 76(1)1: 34–145

[17] A. Tatar, P. Antoniadis, M. Amorim, S. Fdida "Ranking news articles based on popularity prediction". In: Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012). IEEE, Istanbul, pp 106–110

[18] A. Tatar, J. Leguay, P. Antoniadis, A. Limbourg, M. Amorim, S. Fdida, "Predicting the popularity of online articles based on user comments". In: Proceedings of the International Conference on Web Intelligence, Mining and Semantics. ACM, Sogndal, Norway

[19] S-D. Kim, S-H. Kim, H-G. Cho "Predicting the virtual temperature of web-blog articles as a measurement tool for online popularity". In: 11th International Conference on Computer and Information Technology (CIT). IEEE, Pafos, Cyprus, pp 449–454

[20] L. Hong, O. Dan, B. Davison "Predicting popular messages in Twitter". In: Proceedings of the 20th International Conference Companion on World Wide Web. ACM, Hyderabad, India, pp 57–58

[21] M. Tsagkias, W. Weerkamp, M. Rijke "Predicting the volume of comments on online news stories". In: Proceedings of the 18th ACM Conference on Information and Knowledge Management. ACM, Hong Kong, China, pp 1765–1768

[22] R. Bandari, S. Asur, B. Huberman "The pulse of news in social media: Forecasting popularity". In: ICWSM. The AAAI Press, Dublin, Ireland

[23] A. Oghina, M. Breuss, M. Tsagkias, M. Rijke "Predicting IMDb movie ratings using social media" In: Proceedings of the 34th EuropeanConferenceonAdvancesinInformationRetrieval.ECIR'12.Springer, Barcelona, Spain, pp 503–507

[24] SD. Roy, T. Mei, W. Zeng, S. Li "Towards cross-domain learning for social video popularity prediction". IEEE Trans Multimedia 15(1255-1267)

[25] B. Gregorutti, B. Michel , P. Pierre "Correlation and variable importance in random forests" arXiv:1310.5726, DOI: 10.1007/s11222-016-9646-1

[26] Hutto, C.J. & Gilbert, E.E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Eighth International Conference on Weblogs and Social Media (ICWSM-14). Ann Arbor, MI, June 2014.

[27] http://www.nltk.org/book/ch07.html

[28] P. Viola, M. Jones "Rapid Object Detection using a Boosted Cascade of Simple Features". In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001.