Analysis of Popularity Pattern of User Generated Contents and its Application to Content-aware Networking

Tatsuya Tanaka*, Shingo Ata†, and Masayuki Murata*
* Graduate School of Information Science and Technology, Osaka University Osaka, Japan,
Email: {t-tanaka, murata}@ist.osaka-u.ac.jp
† Graduate School of Engineering, Osaka City University Osaka, Japan, Email: ata@info.eng.osaka-cu.ac.jp

***Abstract*—In recent years, social multimedia sharing services such as YouTube, which share User Generated Content (UGC) have become much attracted. An efficient control of UGC is one of important roles to achieve, e.g., an optimized placement of advertisements for end users, or content-aware caching control for improving the utilization of network resources. For this reason, it is effective to forecast the future popularity of the content as early as possible, so that we can take a proactive action to highly popular contents. In this paper, we propose a method to classify the popularity of UGCs in real time using K-means clustering, and analyze tendencies led by popularity patterns. We then propose a method to identify UGCs which are expected to be popular in future, by taking both the initial part of popularity patterns and actual counts of content retrieves into consideration. Our experimental results show that the accuracy of identification of popular UGCs can be increased around 10% by considering the initial part of popularity patterns.**

I. Introduction

Recently, User Generated Contents (UGCs) are becoming popular, which is initiated by social video sharing services such as YouTube [1] and Instagram [2]. Share and delivery of UGCs require additional Quality of Service (QoS) constraints compared to data transfers, especially when delay and jitter are serious and sensitive against Quality of Experience (QoE) in video delivery.

Content caching is a promising approach to achieve an efficient use of network resources. Many service providers actually utilize a scheme of content cache to improve the end users' QoE.

The use of content cache is more important and generic in Information Centric Networking (ICN) [3], which is being attracted recently as a future Internet architecture. One of major features of ICN is in-network caching, where ICN routers have a storage (called *content store*) to store received packets (typically called *chunks* in ICN), and stored chunks are re-used for future requests of contents. A difference from Content Distribution Networks (CDN) is that a content store acts by packet-basis, while a caching mechanism of CDN is content-basis. Also, in-network caching is a built-in capability of ICN, so that the behavior of content delivery is transparent to the users whatever a part of a content is cached or not.

In both cases (CDN or ICN), a strategy of caching contents is a key for the overall performance of content delivery. Since the resource of storage for caching is limited, a cache replacement algorithm is needed to update the cache storage. Conventionally, Least Recently Used (LRU) is widely used for the replacement of contents in cache. LRU works fine when content request are uniformly distributed. However, it sometimes degrades the overall performance when the distribution is heavily biased. For example, the Zipf distribution of requests causes an increase of unused caches in future and overall inefficiency. In particular, the most portion of content cache becomes instantly popular in short period, and would not be requested much after losing its popularity.

The main reason is that LRU only focuses on the history of access frequencies, and has an implicit assumption that the characteristic of access frequencies is stable from the past to the future. However, especially for UGCs, access frequencies of contents heavily depend on their popularity, which may vary significantly in very short term. Therefore, a strategy of content placement should consider not only a history of access frequencies in the past but also a lifetime of the content in the future. Such tendency would appear clearer in UGCs because the total number of UGCs extremely higher than professionally generated contents.

Keeping those background in mind, we consider that a forecast of future popularity of contents is quite important for making a caching strategy of UGCs in both CDN and ICN. For example, in order to suppress the peak load of the video distribution server, it is effective to perform a proactive caching, which actively caches popular contents in advance. For the effective proactive caching, the accurate prediction of the popularity of contents is important [4].

In addition, social UGC sharing services have an advertisement framework which offers a user an advertisement closely related to the playing content. Optimization of advertisement may also need a prediction of content popularity in future.

In this paper, we aim to forecast a future popularity of

a content based on the measurement of its access frequencies. First we collect time-series view counts of YouTube videos and analyze the variation of popularity (we call *popularity pattern* in this paper) by using a clustering technique. Previously there is a literature which collects popularity patterns of video contents [5], by the unit of 1 day, while we collect time-series data in a miniaturized time granularity of 1 hour unit to reveal the tendency of popularity pattern.

Based on the findings in the analysis of popularity pattern, we propose a method to predict a future popularity of a content from the measurement of the variation access frequencies around initial phase (i.e., first 3 hours from the time when the content is initially published). We use the Naive Bayes classifier, to decide whether the content will become highly popular or not in next 7 days. Through simulation results, we show that our prediction method can improve around 10% of accuracy to identify highly popular contents in future.

This paper is organized as follows. In Section II, we introduce related works and describe the novelty of this paper. In Section III, we describe the method of measurement and analysis of popularity pattern for video contents in YouTube. Section IV describes a method to identify highly popular contents in future using Naive Bayes classifier, and shows the simulation-driven evaluations. Finally, we conclude this work with future research topics in Section VI.

II. Related Works

Forecasting the dynamic of UGC popularity is more difficult than VoD (Video-on-Demand), due to the incalculable number of videos, the diversity of content, and popularity dynamics. Moreover, it is known that the view counts of each video differ greatly. Therefore, the viewing trend of UGC is discussed in many literatures.

In [6], Figueiredo et al. investigated the popularity dynamics of videos, which are categorized in following three types, i.e., videos in popular ranking, videos which had been deleted by the infringement of copyright, and videos which had been selected by inputing random words in the search engine of YouTube. In [7], Borghol et al. showed that the popularity for the content, which is randomly selected from YouTube, may vary in the unit of one week. In [8], Gursun et al. analyzed the access pattern of YouTube, and showed that the daily access pattern of most contents are classified into two types: a frequent access and a sporadic access. They also proposed a method of forecasting future view counts by using Principal Component Analysis (PCA). PCA is a multivariate technique that analyzes a data table in which observations are described by several inter-correlated quantitative dependent variables. In [9], Szabo et al. paid their attention to a linear correlation between early view count and view count at thirty days later from initial publish in logarithmic graph, and described that future view count can be predicted by coordinating the parameters of a liner model.

In [10], Pinto et al. focus on the problem that multiple videos having the same tendency in terms of cumulative view counts in the past may have significantly different tendencies in future. For the problem, they proposed a method to predict cumulative view count at arbitrary day by using linear regression. In [11], Tirad et al. grouped contents according to locality and predicted the short-term access pattern in a multi-model using an autoregressive model.

In [5], Kitade et al. proposed a classifying method with k-means clustering which is often used as non-hierarchical cluster analysis to extract content of which a lot of audience are expected in the future by using the pattern of early popularity dynamics.

As already mentioned, in regard to network control, content-aware caching control and load balancing of the popular content, especially for UGCs, may require fine grained (in the unit of one hour) identification of popular contents. However, to the best of our knowledge, above literatures only considered in the unit of 1 day or longer.

III. Collection and Analysis of view count in YouTube

A. Data Collection Method

We collect the view counts of recently uploaded YouTube videos by using YouTube Data API version 3.0 [12] to analyze the trend of the popularity pattern. Specifically, we collect everyday the names of videos newly uploaded, and continuously obtain their hourly view counts until one week from the initial upload. For this purpose we develop a program which periodically runs to get daily view counts of all published videos.

The period of data collection is from Oct. 14, 2015 to Dec. 5, 2015, and we removed the videos which are failed to collect its view count. We also removed videos that were removed or disappeared before 30 days from the initial upload. Finally, we use total 87,830 videos which meet the following conditions as the dataset.

- Cumulative view count for 8 days after the initial upload is more than 10.
- Duration when a daily view count is less than 5 is 6 days or less.
- All hourly view counts are positive (in some videos, hourly view counts sometimes become negative due to an adjustment to the accounting policy of YouTube [13]).

B. Observations of YouTube Access Patterns

In this section, we analyze various characteristics of YouTube dataset. Fig. 1(a) is the CCDF (Complementary Cumulative Distribution Function) of view count at s hour(s) after the initial upload ($s = 1, 2, 3, 6$), Fig. 1(b) is the CCDF of view count at s day(s) after the initial upload ($s = 1, 2, 3, 7, 14$).

Both graphs are double logarithmic plot and depict the curve close to linear at the foot. These figures show that a few of videos get large view counts extremely. Moreover, from Fig. 1(a), we observe that view count at 1 hour from the initial upload is the largest. The reason is considered that the information of the video is marked as "Newly uploaded" on the social service, and many people try to view by checking the list of newly uploaded videos. As time progress, the list is replaced by newer contents and the content would be dropped from the list. From Fig. 1(b), we observe that view count decreases significantly as the increase the number of days after the initial upload. The tendency is clearer for the videos with higher view counts.

Fig. 1(c) is the CCDF of cumulative view counts at s day(s) after the initial upload ($s = 1, 2, 3, 7, 14$). This figure also depicts the curve close to linear at the foot. We also observe that a few videos get large view counts extremely.

We next investigate viewing trend by the time-of-day of initial upload (0 ~3, 4~7, 8~11, 12~15, 16~19, 20~23 in UTC). Fig. 2(a), 2(b), and 2(c) are the CCDF of view count at 1 hour, 1 day, and 1 week after the initial upload, respectively. From these figures, we can observe that the difference of the time-of-day has a significant impact to the difference of view count at 1 hour after the initial upload, and the impact becomes decreased as the time progress. Such tendency is clearly caused by the number of people actively using the Internet. By 1-hour measurement, the number of people may vary significantly by the time-of-day, however, the variation would be rounded in aggregation in both day and week.

C. Analyzing Popularity Pattern by Clustering

In this section, we analyze the trend of popularity pattern in YouTube with k-means clustering which is often used as non-hierarchical clustering for hourly view count in first n hour(s) from initial upload.

In [5], Kitade et al. analyze a pattern of daily view count with k-means clustering. On the contrary, in this paper, we collect hourly view counts and reveal trend of pattern on a finer granularity (i.e., 1 hour). To be concrete, we normalize each hourly view count by the maximum value of hourly view count. Then we get a n-dimensional vector having values of $0 \leq s \leq 1$ in each element. We classify videos with k-means clustering by using these vectors, then we classify videos into clusters in which frequency patterns of videos are similar.

Fig. 3 is the result of k-means clustering with 5 clusters by using the pattern of hourly view for first 24 hours (i.e., $n = 24$). The numbers in parentheses in the legend is the number of videos that are classified to the corresponding cluster.

Fig. 3(a) plots the average of the normalized views of each cluster. From this figure, Cluster 1, which has the largest number of videos, the normalized view count is relatively high at the initial phase (e.g., just after the initial upload), but the view count becomes lowest at 5 hours and later. The result implies that videos in Cluster 1 are viewed only the phase when they are listed in "Newly uploaded". It seems that they are not much attracted and few people recommend to others. On the other hand, in Cluster 5, the normalized view count is not so high just after the initial upload, but it reaches the highest value after 16 hours or later from the initial upload.

Fig. 3(b) shows the average of hourly view counts of each cluster. Cluster 5 keeps the higher average than others. Also, there is a periodic repeats by 24 hours in all clusters (especially remarkable in Cluster 5). Initially, the variation of access frequencies is caused by mainly the system (e.g., "Newly Uploaded" list), but after 24 hours, the spread of interest for the video by other services or word of mouth, which depend on human life cycle.

Fig. 3(c) is the average of daily view counts until 30 days from the initial upload of each cluster. Cluster 5 also keeps higher average than others. Fig. 3(d) is the CCDF of view count after 7 days from the initial upload. Cluster 5 tends to have larger view counts than others. From these observations, it is clear that videos having the large normalized view counts in first 24 hours also have a tendency to earn large view counts for future.

From above results, we consider that there are some typical popularity patterns; (1) it has a large number of view counts at early phase but the number is decreased sharply as the time progress, (2) it keeps a certain view counts continuously to achieve a stable popularity.

IV. Identification of Popular Content using Naive Bayes Classifier

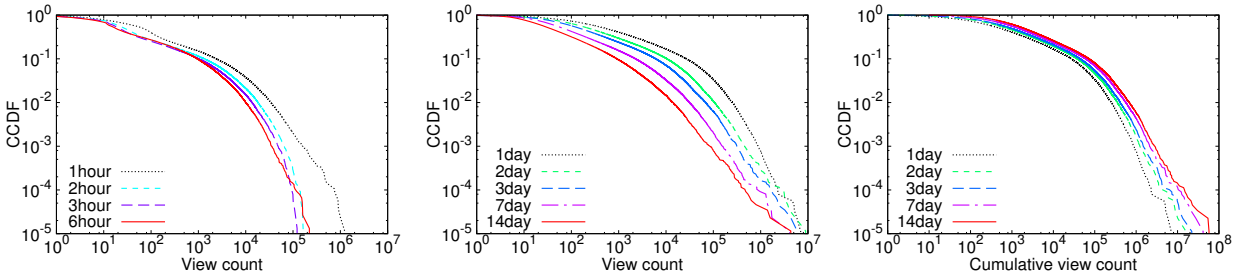
In this section, we propose a method which identifies popular content by using a supervised machine learning (ML). We use a series of view count patters for first Y hours from the initial upload as a data of ML, and then apply a Naive Bayes classifier for identifying whether the content will be popular in next d days.

A. Outline of Naive Bayes Classifier

A Naive Bayes classifier is a kind of supervised learning based on applying Bayes' theorem. From the learning data, when input features $F_1, \cdots, F_n$ are given, it calculates a probability of each data to be assigned to a category C. Based on this probability, a classification category is determined for the test data. The classifier is represented by

$$\text{classify}(f_1, \cdots, f_n) = \arg\max_c p(C = c) \prod_{i=1}^{n} p(F_i = f_i | C = c).$$

For some types of probability models, a Naive Bayes classifiers can be trained very efficiently in a supervised learning setting. Despite their naive design and simple assumptions, a Naive Bayes classifiers have worked quite well in many complex real-world situations. Thus, we apply a Naive Bayes classifier for identifying popular contents and confirm the its efficacy.

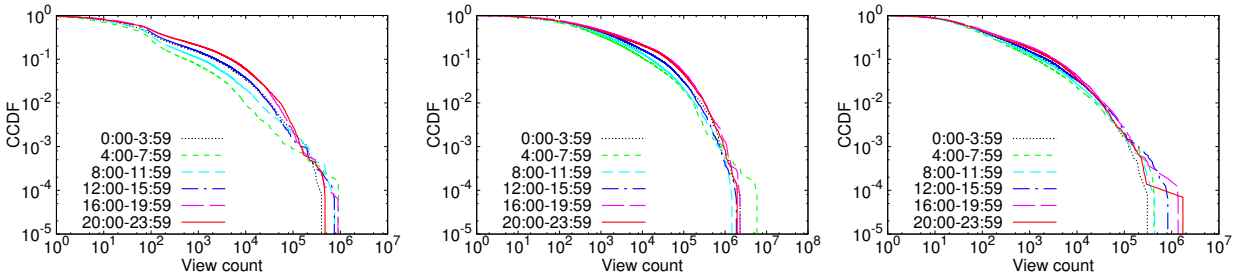


(a) CCDF of view count at s hour(s) after initial upload ($s = 1, 2, 3, 6$)

(b) CCDF of view count at s day(s) after initial upload ($s = 1, 2, 3, 7, 14$)

(c) CCDF of cumulative view count at s day(s) after initial upload ($s = 1, 2, 3, 7, 14$)

Fig. 1. Characteristics of View Counts

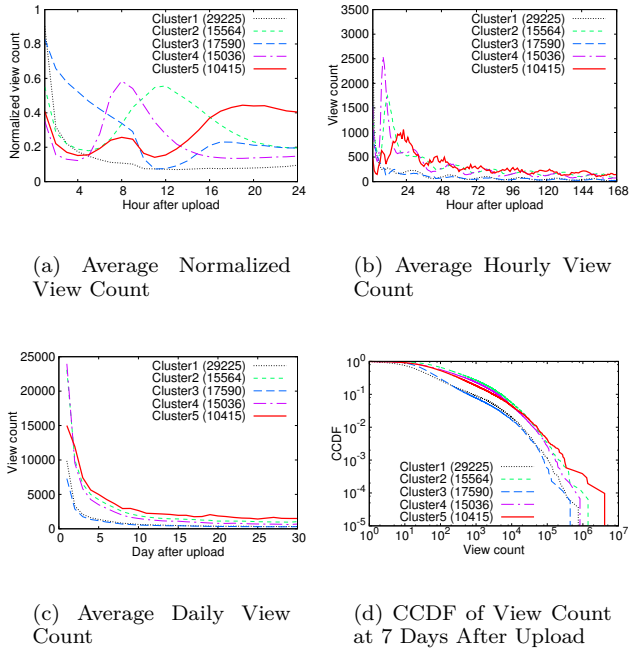


(a) CCDF of view count at 1 hour after initial upload

(b) CCDF of view count at 1 day after initial upload

(c) CCDF of view count at 7 days after initial upload

Fig. 2. Difference of Characteristics by Time-of-Day



(a) Average Normalized View Count

(b) Average Hourly View Count

(c) Average Daily View Count

(d) CCDF of View Count at 7 Days After Upload

Fig. 3. Results of k-means Clustering (24 Hours Dataset)

B. Identification Method of Popular Contents

Here we suppose H as the time to identification, Y (hours) as the window size of measurement to obtain a test data of identification, and d (days) as the target time to identify the popularity, i.e., this method identifies the popularity of d days after the time of identification H. To prepare the training data, we use popularity patterns of video contents which have been uploaded $Y + d$ before the identification time H. For the test data, we use videos which have been uploaded Y before the identification time H.

Following is a process to retrieve popularity pattern of video contents through YouTube Data API.

1) Obtain the list of newly uploaded videos for every minute.
2) Get hourly view counts of videos which have been uploaded within last Y period.
3) Get daily view counts of videos which have been uploaded between $Y - d$ and d before the measurement time.
4) Prepare the training data of Naive Bayes classifier from the measurement data obtained by Step. 2.
5) Through machine learning, identify the popularity for video contents which have been uploaded after Y from the initial upload.

C. Identification Procedure

In the evaluation, we focus on two types of contents : stably popular contents and highly popular contents.

We first define *stably popular content* as the one which satisfies the following condition.

- The coefficient of variation of daily view counts in the first d days is lower s% of all videos in the training data.($s = 1, 5, 10$)

The coefficient of variation is defined as the ratio of the standard deviation to the average. That is, we suppose that the daily view count at i days after the prediction time is denoted by x_i, so the coefficient of variation (CV) is given by

$$CV = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \frac{1}{n}\sum_{i=1}^{n} x_i)^2}}{\frac{1}{n}\sum_{i=1}^{n} x_i}. \quad (1)$$

The smaller the value of CV is, the more stable view counts and popularity of the video are.

Next, a *highly popular content* is defined which satisfies following two conditions.

- Definition 1 : Daily view counts in d days are top 1% of all videos in the training data.
- Definition 2 : Cumulative view counts in d days are top 1% of all videos in the training data.

$F_1, \cdots, F_n$ in Section IV-A are normalized variables obtained by dividing hourly view counts by the maximum hourly view count in first Y hours and rounded off to the first decimal place.

In the case of identification of *highly popular content*, the number of digits of max hourly view count in first Y hours from the initial upload is prepared. As an input to these prepared variables, we apply the Naive Bayes classifier.

V. Evaluation Results

A. Evaluation conditions

In this paper, we use the half selected contents at random from 87,830 videos as the training data, and identify both *stably popular* and *highly popular* contents for the rest.

Videos of which coefficient of variation in the forecast period is small are predicted from the initial views data from upload by using the Naive Bayes Classifier.

The input is normalized view counts to be used in the prediction. We compare this prediction with the case that we select the same number of videos of which coefficient of variation is small in the order as the Naive Bayes classifier selected.

For comparison purpose, we also evaluate the result of identification of highly popular content by using the Naive Bayes classifier with View Count based Selection (VCS). VCS is to select the same number of videos of which cumulative view counts in first Y from the initial upload is large in the order as the Naive Bayes classifier selected.

For performance metric, we use an *identification accuracy*, which is the total number of video contents that are correctly identified as *stably* or *highly popular contents* divided by the total number of all videos.

TABLE I
Identification Accuracy of *Stably Popular Contents*
($Y = 72, d = 7$)

Definition	NBC	Selection based on Initial CV
Lower 1%	0.077	0.056
Lower 5%	0.142	0.135
Lower 10%	0.211	0.199

TABLE II
Identification Accuracy of *Stably Popular Contents*
($Y = 168, d = 7$)

Definition	NBC	Selection based on Initial CV
Lower 1%	0.157	0.001
Lower 5%	0.165	0.050
Lower 10%	0.248	0.042

B. Identification of Stably Popular Contents

Table I shows the identification accuracy of *stably popular contents* by using view counts of every 6 hours ($Y = 72$, $d = 7$). Table II shows the identification accuracy by using daily view counts ($Y = 168$, $d = 7$). NBC in Table I and Table II is an abbreviation for a Naive Bayes classifier.

As shown in these tables, we can observe that the identification accuracy of the proposed method (labeled by NBC) is higher than the method based on initial coefficient of variation (labeled by VCS). In particular, when the window of initial phase (Y) is large, the identification accuracy of the Naive Bayes classifier is much higher. This is because that there are many videos which have volatile popularity at initial phase but become stable rapidly after the initial phase.

C. Identification of Highly Popular Contents

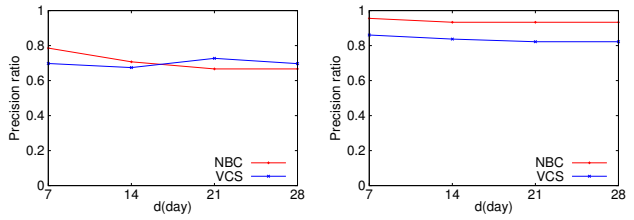
In this section, we show the result of the prediction of highly popular contents. Table III shows the identification accuracy of *highly popular contents* by using hourly view counts ($Y = 3$, $d = 7, 14$).

In all cases in Table III, the precision ratio of the naive Bayes classifier is higher than that of VCS. Therefore, it is clear that when we use view counts for the first 3 hours, the identification which takes the popularity pattern into account has a higher accuracy.

Next, we show the variation of popularity in regard to the hourly or daily view counts, and cumulative view counts. Fig. 4 shows the results by changing d. Fig. 4(a) shows the transition of precision ratio of Definition 1 when we fix $Y = 3$ and change the value of d. Fig. 4(b) shows that of Definition 2. In the case of Definition 1, the accuracy of the Naive Bayes classifier tends to decrease with the increase of d. In the case of Definition 2, the accuracy of the Naive Bayes classifier is maintained at high level.

TABLE III
the identification accuracy of *Highly Popular Contents*
($Y = 3$, $d = 7, 14$)

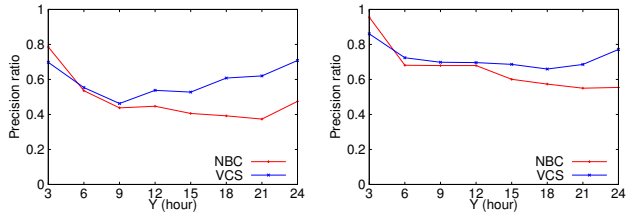
	Identification Accuracy of Top 1% Videos			
Target day	Day 8	2~8 days	Day 15	2~15 days
NBC	0.785	0.956	0.707	0.933
VCS	0.697	0.860	0.674	0.837



(a) Prediction of daily view count after d days

(b) Prediction of cumulative view count until after d days

Fig. 4. Transition of precision ratio of when we fix $Y = 3$ and change the value of d



(a) Prediction of daily view count after 7 days

(b) Prediction of cumulative view count until after 7 days

Fig. 5. Transition of precision ratio when we fix $d = 7$ and change the value of Y

Next, Fig. 5(a) shows that the transition of precision ratio of Definition 1 when we fix $d = 7$ and change the value of Y. Fig. 5(b) shows that of Definition 2. In the Naive Bayes classifier, when the numbers of input are too many, the popularity evolution pattern is too diverse and precision ratio decreases. Thus, we use view counts of every $Y/3$ hours and set the number of inputs as 3. When we increase Y, it can be seen that precision ratio of the prediction using the absolute value of initial view counts becomes higher.

From the above, when we predict future popularity using view counts of initial 3 hours from upload, the precision ratio of the Naive Bayes classifier is higher than that of VCS. This is because there are many videos that is popular just after upload but become unpopular a few days later. Therefore, we presume that the Naive Bayes classifier is able to capture the evolution of content popularity, which finally provides high precision in out results.

VI. Conclusion

In this paper, we firstly collected the time-series data of view counts and analyzed the viewing trend of YouTube. The result shows that a small number of videos has extremely large view counts. Moreover, we analyzed the popularity evolution pattern just after upload by clustering the time-series data of hourly view counts with k-means clustering. In the result, it became clear that there are the popularity change patterns which have the large absolute value of view count at early stage from upload but fail to maintain view count for long period of time and have stable normalized view count pattern so highly popularity is maintained over future. Furthermore, we applied the Naive Bayes classifier to identification both *stably popular* and *highly popular* contents. In the result, we revealed that the identification of the Naive Bayes classifier that takes the change pattern of view count into account grows in performance. For our future works, this prediction approach will be further evaluated in the control of content caching and advertisement targeting.

Acknowledgment

We would like to express our gratitude to Dr. Noriaki Kamiyama of NTT Network Technology Laboratories for helpful comments to this work. We would also like to express our gratitude to Dr. Suyong Eum of Osaka University for helpful comments to our writing and English. This research and development work was supported by the MIC/SCOPE #165007007.

References

[1] "YouTube." https://www.youtube.com/.
[2] "Instagram." https://www.instagram.com/.
[3] M. Zhang, H. Luo, and H. Zhang, "A survey of caching mechanisms in information-centric networking," *IEEE Communications Surveys & Tutorials*, vol. 17, pp. 1473–1499, July 2015.
[4] N. Kamiyama, R. Kawahara, T. Mori, and H. Hasegawa, "Multicast Pre-distribution VoD System," *IEICE transactions on communications*, vol. E96-B, pp. 1459–1471, June 2013.
[5] Y. Kitade, "Analyzing popularity dynamics of YouTube content and its application to content cache design," Master's thesis, Graduate School of Information Science and Technology, Osaka University, Feb. 2015.
[6] F. Figueiredo, F. Benevenuto, and J. 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*, pp. 745–754, Feb. 2011.
[7] Y. Borghol, S. Mitra, S. Ardon, N. Carlsson, D. Eager, and A. Mahanti, "Characterizing and modelling popularity of user-generated videos," *Performance Evaluation*, vol. 68, pp. 1037–1055, Nov. 2011.
[8] G. Gürsun, M. Crovella, and I. Matta, "Describing and forecasting video access patterns," in *Proceedings of IEEE INFOCOM*, pp. 16–20, Apr. 2011.
[9] G. Szabo and B. A. Huberman, "Predicting the popularity of online content," *Communications of the ACM*, vol. 53, pp. 80–88, Aug. 2010.
[10] H. Pinto, J. M. Almeida, and M. A. 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*, pp. 365–374, Feb. 2013.
[11] J. M. Tirado, D. Higuero, F. Isaila, and J. Carretero, "Multi-model prediction for enhancing content locality in elastic server infrastructures," in *Proceedings of the eighteenth International Conference on High Performance Computing*, pp. 1–9, Dec. 2011.
[12] "YouTube Data API." https://developers.google.com/youtube/v3/.
[13] "YouTube Help." https://support.google.com/youtube/answer/2991785/.