Accurate and Efficient Extraction of Hierarchical Structure of Web Communities for Web Video Retrieval

Ryosuke Harakawa (student member)†, Takahiro Ogawa (member)†, Miki Haseyama (member)†

Abstract This paper presents an accurate and efficient method for extracting hierarchical structure of Web communities, i.e., Web video sets with similar topics for Web video retrieval. First, efficient canonical correlation analysis (CCA), named sub-sampled CCA, is derived to obtain link relationships that represent similarities between latent features of Web videos. Moreover, the obtained link relationships enable application of an algorithm based on recursive modularity optimization to extract hierarchical structure of Web communities. Different from previously reported methods, our method can extract the hierarchical structure for the whole target dataset since the algorithm enables recursive reduction of its processing targets. This means it becomes unnecessary to perform screening of Web videos, and we can avoid performance degradation caused by discarding relevant Web videos in the screening, which occurred in previously reported methods. Consequently, our method enables extraction of the hierarchical structure with high accuracy as well as low computational cost.

Key words: hierarchical structure, Web community, Web video, video retrieval.

1. Introduction

With the proliferation of digital contents including Web videos∗1), more and more users are retrieving Web videos to access their desired contents. Usually, to access them, users retrieve Web videos by inputting queries that identify the desired contents into a retrieval system. Then it returns retrieval results to users. However, if suitable queries cannot be input, the retrieval results may include irrelevant contents to the desired ones. Consequently, it becomes difficult for users to access the desired contents from the retrieval results2).

To overcome this difficulty, clustering-based retrieval methods3)-12) have been proposed. Since users can easily grasp the overview of retrieval results via the clustering results, easy access to the desired contents becomes feasible. In particular, hierarchical clustering-based methods7)-12) are useful for grasping the overview of retrieval results since relationships between the obtained clusters become clear. Some methods use visual features and enable retrieval of videos based on hierarchical exhibition of videos7)-9). Another method for retrieving Web videos based on exhibition of hierarchical topic structure obtained via textual analysis and WordNet13) was proposed10). Also, to realize accurate Web video retrieval, we have proposed methods that extract hierarchical structure of Web communities by using not only a single modality but also multi-modalities, i.e., visual, audio and textual features11)12). In this paper, we define Web communities as Web video sets with similar topics. Moreover, the hierarchical structure is defined as the property of Web communities being divided into their sub-communities. However, one of our previously proposed method11) has a limitation in computational cost. The other proposed method12) solves this problem by introducing a community graph that can handle many Web videos efficiently, but this method needs to perform screening of Web videos for reducing computational cost. Thus, there is a problem that this method cannot realize both high accuracy of retrieval results and low computational cost simultaneously.

In this paper, we present a new method to extract hierarchical structure of Web communities for Web video retrieval, which can solve the above problem. The proposed method enables extraction of the hierarchical structure with high accuracy as well as low computational cost by introducing a graph analysis algorithm based on recursive modularity optimization14). First, as motivated by our previous work12), we efficiently calculate canonical correlation analysis (CCA)15)-based link relationships that represent similarities between latent features of Web videos. Moreover, the obtained link relationships enable application of the graph analysis algorithm based on recursive modularity optimization14), which makes screening of Web videos unnecessary and
enables accurate and efficient extraction of the hierarchical structure of Web communities. By exhibiting the obtained hierarchical structure to users, retrieval of the desired Web videos becomes feasible.

Finally, we explain our contribution of this paper. To the best of our knowledge, there has been no work that enables accurate and efficient hierarchical clustering via multimodal features for Web video retrieval. Here, to realize effective retrieval of Web videos that exist in quantity, (1) efficient calculation of multimodal features and (2) fast and accurate hierarchical clustering are necessary. To realize (1), our previous method\textsuperscript{12)}, which enables efficient calculation of multimodal features and definition of a graph, is useful. To realize (2), it is undesirable to repeatedly calculate global features of the graph whose computational cost is high since screening of Web videos becomes necessary and may cause the performance degradation as with our previous method\textsuperscript{12}). Rather, fast and accurate hierarchical clustering based only on local structure of the graph is necessary. To realize this, the method\textsuperscript{14)} is optimal. However, since the method\textsuperscript{14)} has been proposed not for Web video retrieval but for general graph analysis, the method\textsuperscript{14)} cannot be directly introduced into Web video retrieval. Hence, our contributions are twofold. First, we enable application of the method\textsuperscript{14)} to Web video retrieval by newly defining a graph as motivated by the method\textsuperscript{12}). Note that we need to define an undirected graph to adopt the method\textsuperscript{14)} meanwhile our previous work\textsuperscript{12)} defines a directed graph. Thus, in this paper, we define an undirected graph that can preserve information of edge directions (see Eq. (9)). Second, we show the effectiveness of introducing the method\textsuperscript{14)} into Web video retrieval by the experiments. In particular, although the method\textsuperscript{14)} has not been proposed for Web video retrieval originally, we show that introduction of the method\textsuperscript{14)} successfully enables accurate and efficient extraction of the hierarchical structure for Web video retrieval.

This paper is organized as follows. An outline of our method is shown in Sec. 2. In Sec. 3, we show a method to efficiently derive CCA-based link relationships between Web videos. We explain a method to accurately and efficiently extract the hierarchical structure of Web communities for Web video retrieval in Sec. 4. In Sec. 5, we present results of experiments on actual Web videos to verify the accuracy and computational cost of our method. Concluding remarks are given in Sec. 6.

2. Outline of the Proposed Method

Our method consists of the following two phases.

Phase I: Derivation of CCA-based Link Relationships between Web Videos
We calculate CCA\textsuperscript{15)}-based link relationships via visual, audio and textual features, which represent similarities between latent features of Web videos. Here, efficient CCA, named sub-sampled CCA, is derived to obtain the canonical correlation for large training pairs with low computational cost according to our previous work\textsuperscript{12)}.

Phase II: Accurate and Efficient Extraction of Hierarchical Structure of Web Communities
By constructing a graph whose nodes are Web videos based on the obtained link relationships, we enable application of the algorithm based on recursive modularity optimization\textsuperscript{14)} to extract hierarchical structure of Web communities. Here, screening of Web videos for reducing computational cost becomes unnecessary since the graph analysis algorithm enables recursive reduction of the target nodes. Thus, we can avoid the performance degradation caused by discarding relevant Web videos in the screening, which occurred in our previous work\textsuperscript{12)}.

Finally, users can retrieve the desired Web videos by selecting Web communities associated with the desired contents according to the hierarchical structure. The following sections show these details.

3. Phase I: Derivation of CCA-based Link Relationships between Web Videos

In Sec. 3.1, we define Web video features for deriving the link relationships. A scheme for obtaining the link relationships is shown in Sec. 3.2.

3.1 Definition of Web Video Features for Deriving Link Relationships
In this paper, we adopt visual, audio and textual features of Web videos according to our previous work\textsuperscript{12)}. First, we apply a shot segmentation method\textsuperscript{16)} to each Web video $f_i$ ($i = 1, 2, \ldots, N$; $N$ being the number of Web videos) and obtain shots $s_i^{q_i}$ ($q_i = 1, 2, \ldots, M_i$; $M_i$ being the number of shots within $f_i$) from each Web video. Then, from each shot $s_i^{q_i}$, we calculate a visual feature vector $v_i^{q_i}$, an audio feature vector $a_i^{q_i}$ and a textual feature vector $t_i^{q_i}$ as follows.

**Visual Feature Vector** (*p* dimensions): The HSV color histogram with $p$ bins is calculated every $P_{f}$ frames of Web video $f_i$, and its vector is obtained.\textsuperscript{**} Then the vector median\textsuperscript{17)} is calculated

\textsuperscript{**} In the experiments, we set $P_f$ as the vectors were obtained once a second.
for the frames in each shot $s^i_k$, and the calculated vector is defined as the visual feature vector $v^q_i$ ($=[v^q_i(1), v^q_i(2), \ldots, v^q_i(p)]^T$). Thus, we obtain $M_i$ visual feature vectors $v^q_i$ ($q_i = 1, 2, \ldots, M_i$) from each Web video $f_i$.

**Audio Feature Vector (22 dimensions):** Each shot $s^i_k$ in Web video $f_i$ is divided into some clips***, and we classify each clip into four audio classes, i.e., silence, speech, music and noise, based on a previously reported method. Then the audio class that is most included in the clips within each shot $s^i_k$ is selected. For all clips classified into the selected audio class in each shot $s^i_k$, we calculate the averages and the standard deviations of the following 11 features:

- volume, zero-crossing rate, pitch, frequency centroid, frequency bandwidth, subband energy ratio (0-630 Hz, 630-1720 Hz, 1720-4400 Hz and 4400-11025 Hz), non-silence ratio and zero-ratio.

For each shot $s^i_k$, the audio feature vector $a^q_i$ ($=[a^q_i(1), a^q_i(2), \ldots, a^q_i(22)]^T$) is obtained by aligning the above obtained features. Thereby, $M_i$ audio feature vectors $a^q_i$ ($q_i = 1, 2, \ldots, M_i$) are calculated from each Web video $f_i$.

**Textual Feature Vector ($U$ dimensions):** We obtain the keywords that appear in text attached to Web videos $f_i$ ($i = 1, 2, \ldots, N$) such as title and description. Moreover, we apply TF-IDF to the keywords and obtain the weights of them. By aligning the obtained weights for each Web video, we obtain the vector $\eta_i$ ($=[\eta_i(1), \eta_i(2), \ldots, \eta_i(K)]^T$). Furthermore, by applying random projection to $\eta_i$ ($i = 1, 2, \ldots, N$), their dimensions are reduced to $U(<K)$ by the following procedures. First, a $K \times U$ matrix $R = (r_{ij})$ is calculated as follows:

$$r_{ij} = \begin{cases} \sqrt{3} \text{ with probability of } \frac{1}{6} \\ 0 \text{ with probability of } \frac{1}{3} \\ -\sqrt{3} \text{ with probability of } \frac{1}{6} \end{cases} \quad (1)$$

Then a new $U$-dimensional vector $t_i$ is calculated as follows:

$$t_i = R^T\eta_i, \quad i = 1, 2, \ldots, N. \quad (2)$$

Thus, $M_i$ textual feature vectors $t^q_i$ ($= t_i$) ($q_i = 1, 2, \ldots, M_i$) are obtained for each shot $s^i_k$ in a Web video $f_i$.

In this way, the visual feature vector $v^q_i$, audio feature vector $a^q_i$ and textual feature vector $t^q_i$ are calculated from each shot $s^i_k$ in the Web video $f_i$.

### 3.2 Derivation of CCA-based Link Relationships

To obtain link relationships between Web videos, we calculate similarities between Web videos that enable comparison between different modalities via efficient CCA, named sub-sampled CCA, according to the reference. Here, similarity calculation with low computational cost can be realized since we introduce a clustering-based sub-sampling scheme of training pairs for CCA. Specifically, we first perform k-means clustering to $v^q_i$, $a^q_i$ and $t^q_i$ ($i = 1, 2, \ldots, N, q_i = 1, 2, \ldots, M_i$) for each modality. Then we obtain the cluster centers $v^{cent}_j$, $a^{cent}_j$ and $t^{cent}_j$ ($j = 1, 2, \ldots, N_{clus}; N_{clus}$ being the number of cluster centers) corresponding to visual, audio and textual feature vectors, respectively. Furthermore, from $v^q_i$, $a^q_i$ and $t^q_i$ ($i = 1, 2, \ldots, N, q_i = 1, 2, \ldots, M_i$), we select vectors with the shortest Euclidean distance to the obtained cluster centers $v^{cent}_j$, $a^{cent}_j$ and $t^{cent}_j$ ($j = 1, 2, \ldots, N_{clus}$) for each modality. In this paper, the index of the visual feature vectors, which are selected from $v^q_i$ ($i = 1, 2, \ldots, N, q_i = 1, 2, \ldots, M_i$) that is most similar to the cluster centers $v^{cent}_j$ ($j = 1, 2, \ldots, N_{clus}$), is denoted by $I_v$, and $|I_v|$ ($= N_{clus}$) denotes the number of elements. In the same manner, we respectively define $I_a$ and $I_t$ for the audio and textual feature vectors, where $|I_a| = |I_t| = N_{clus}$. Then we denote the index set of vectors that become calculation targets for CCA by $I$ ($= I_v \cup I_a \cup I_t$), and the number of elements is denoted by $N_{sub}$ ($= |I|$). Here, the selected vectors are denoted by $x^v_i$, $x^a_i$ and $x^t_i$ ($i' = 1, 2, \ldots, N_{sub}$).

Next, we apply CCA to these selected vectors and similarities between Web videos that enable comparison between different modalities. First, three kinds of matrices $X_v$, $X_a$ and $X_t$ are calculated by using $x^v_i$, $x^a_i$ and $x^t_i$ as follows:

$$X_v = [x^v_1 - \bar{x}_v, x^v_2 - \bar{x}_v, \ldots, x^v_{N_{sub}} - \bar{x}_v]^T, \quad (3)$$

$$X_a = [x^a_1 - \bar{x}_a, x^a_2 - \bar{x}_a, \ldots, x^a_{N_{sub}} - \bar{x}_a]^T, \quad (4)$$

$$X_t = [x^t_1 - \bar{x}_t, x^t_2 - \bar{x}_t, \ldots, x^t_{N_{sub}} - \bar{x}_t]^T, \quad (5)$$

where $\bar{x}_v$, $\bar{x}_a$ and $\bar{x}_t$ are the average vectors of $x^v_i$, $x^a_i$ and $x^t_i$, respectively. By using $X_k$ ($k \in \{v, a, t\}$), we calculate linear transformations that maximize correlations among visual, audio and textual features. Specifically, we calculate the matrices that project $X_k$ into the latent space of $X_l$ ($l \in \{v, a, t \mid k \neq l\}$). As a result

*** For more details in the division into clips, refer to the paper.

**For a detailed algorithm for k-means clustering for CCA, refer to the paper.**
of CCA, we obtain coefficient matrices $\mathbf{W}_k$ and correlation matrices $\mathbf{A}_k,l$ in which each diagonal element is the canonical correlation coefficient\textsuperscript{5,6}. Next, by using $\mathbf{W}_k$, $\mathbf{A}_k,l$ and $k^q_l$ ($\in \{v^q_i - \bar{x}^q_i, a^q_i - \bar{x}^q_i, t^q_i - \bar{x}^q_i\}$), $\zeta^l_q$, i.e., the projection results of $k^q_l$ into the latent space of $l^q_i$ ($\in \{v^q_i - \bar{x}^q_i, a^q_i - \bar{x}^q_i, t^q_i - \bar{x}^q_i\}$), are calculated as follows:

$$
\zeta^l_q = \begin{cases} 
\mathbf{W}_k^T k^q_l & \text{if } k = l \\
\mathbf{A}_k,l \mathbf{W}_k^T k^q_l & \text{otherwise.}
\end{cases}
$$

(6)

Hence, from $v^q_i$, $a^q_i$ and $t^q_i$, we obtain the vectors $\zeta^l_q$, $\zeta^q_i$, and $\zeta^l_q$ that can be compared between different kinds of features. In the experiments shown later, visual, audio and textual feature vectors are projected into the latent space of textual features by setting $l = t$ in Eq. (6). Note that the projection results of all feature vectors $v^q_i$, $a^q_i$, and $t^q_i$ ($i = 1, 2, \cdots, N, q_i = 1, 2, \cdots, M_i$) can be obtained by applying CCA to a small number of vectors $x^q_i$, $x^q_i$ and $x^q_i$ ($i' = 1, 2, \cdots, N_{sub}$) selected via k-means clustering.

Thus, we enable efficient calculation of CCA to obtain latent features of Web videos.

Next, we compute similarities $s_{ij}$ between Web videos $f_i$ and $f_j$ via the obtained latent features as follows:

$$
s_{ij} = \max_{q_i, q_j} \frac{(\zeta^l_{q_i})^T \zeta^l_{q_j}}{\|\zeta^l_{q_i}\| \|\zeta^l_{q_j}\|}.
$$

(7)

$$
\xi^l_{q_i} = [(\zeta^l_{q_i})^T, (\zeta^q_{q_i})^T, (\zeta^l_{q_i})^T]^T.
$$

(8)

Moreover, we build link relationships between Web videos based on the obtained similarities and the metadata of Web videos, namely “related videos”\textsuperscript{5,6}. In particular, we consider that a Web video $f_i$ links to a Web video $f_j$ if “related videos” of $f_i$ include $f_j$. Although the only use of the low-level features extracted from Web videos may cause the semantic gap\textsuperscript{23}, i.e., the difference between the low-level features and high-level interpretation of humans, the metadata “related videos” is useful for obtaining Web videos that are similar to each other\textsuperscript{23}. Therefore, we introduce the metadata into our method. Then we construct a graph $G = (V, E)$ whose nodes and edges are respectively Web videos $f_i$ ($i = 1, 2, \cdots, N$) and weighted links. The edge weight $e_{ij}$ between Web videos $f_i$ and $f_j$ is defined as follows:

$$
e_{ij} = \begin{cases} 
2s_{ij} & \text{if } f_i \text{ links to } f_j \text{ and } f_j \text{ links to } f_i \\
s_{ij} & \text{if } f_i \text{ links to } f_j \text{ exclusive or } f_j \text{ links to } f_i \\
0 & \text{otherwise.}
\end{cases}
$$

(9)

If $f_i$ and $f_j$ do not link to each other, we do not build the edge between $f_i$ and $f_j$. Note that we need to define an undirected graph to adopt the method\textsuperscript{4} in the next section meanwhile our previous work\textsuperscript{20} defines a directed graph. Thus, by Eq. (9), we define an undirected graph that can preserve information of edge directions. By constructing a graph $G$, we enable application of a method\textsuperscript{4} to extract the hierarchical structure for Web video retrieval as shown later. The contribution of this paper includes this.

4. Phase II: Accurate and Efficient Extraction of Hierarchical Structure of Web Communities

In Sec. 4.1, we present a method for accurately and efficiently extracting the hierarchical structure. A Web video retrieval method that uses the hierarchical structure is presented in Sec. 4.2.

4.1 Extraction of Hierarchical Structure of Web Communities

After deriving the link relationships between Web videos, i.e., the graph $G = (V, E)$, we extract hierarchical structure of Web communities, i.e., Web video sets with similar topics. Specifically, by using $G$, the hierarchical structure of Web communities is extracted via an algorithm based on recursive modularity optimization\textsuperscript{14}. The algorithm\textsuperscript{14} is a very fast method for graph analysis by which 118 million nodes can be processed in 152 minutes. In addition, our proposed similarities can be easily introduced into this method. Therefore, since this method is optimal for extracting the hierarchical structure efficiently and accurately, we adopt this method. Extraction of the hierarchical structure consists of the following two phases.

In the first phase, each node $f_i$ ($i = 1, 2, \cdots, N$) is assigned to each different Web community. For each node $f_i$, the gain of modularity $Q$ is evaluated when a node $f_i$ is set to a Web community containing a neighbourhood node $f_j$. Then $f_i$ is re-assigned to a Web community for which the positive gain is maximum. Note that modularity $Q$ is an evaluation measure for the division results of the communities in the graph, which are defined by the following equation\textsuperscript{41}:

$$
Q = \frac{1}{2m} \sum_{i=1}^{N} \sum_{j=1}^{N} (e_{ij} - \frac{k_i k_j}{2m}) \delta_{ij},
$$

(10)

where $2m = \sum_{i=1}^{N} \sum_{j=1}^{N} e_{ij}$, $k_i = \sum_{j=1}^{N} e_{ij}$ and $\delta_{ij}$ is 1 if $f_i$ and $f_j$ belong to the same Web community and 0 otherwise. Thus, the higher the modularity is, the better division results of the communities are. This
process is applied to all nodes iteratively and sequentially until no more improvement of the modularity can be obtained.

In the second phase, we construct a new graph whose nodes are the Web communities obtained in the first phase. Here, the edge weight between the two new nodes is the sum of the edge weight of the original graph found during the first phase. Also, each new node has a self-loop that is derived from the weighted edges of the corresponding original nodes obtained in the first phase. In this paper, we call a pair of the first and second phases “a pass” and this iteration number is denoted by $q (= 1, 2, \ldots, Q_h; Q_h$ being the number of all passes). Moreover, the passes, i.e., the first phase to find the Web communities from the new graph and the second phase to construct the newer graph, are iterated until no more improvement of the positive gain of modularity can be obtained. Then we denote the obtained Web communities by $Com^q_{nq} (n_q = 1, 2, \ldots, T_q; T_q$ being the number of Web communities where pass is $q$). Consequently, since we can attain Web communities with different levels of resolution according to the number of passes, it becomes feasible to extract the hierarchical structure of Web communities.

Finally, we note the following for the above recursive processes. According to the increase of $q$, the number of Web videos within the new nodes becomes larger and the number of new nodes decreases in the second phase. Thereby, unlike our previously reported method\textsuperscript{12} that needs to perform screening of Web videos for handling many Web videos, our method can extract the hierarchical structure for the whole target Web videos by the above efficient recursive phases. Therefore, our method can avoid the performance degradation caused by discarding relevant Web videos in the screening unlike our previous method\textsuperscript{12}. For detailed procedures of extraction of the hierarchical structure, refer to Algorithm 1.

### 4.2 Web Video Retrieval Using Hierarchical Structure of Web Communities

In the proposed method, it becomes feasible to perform Web video retrieval by using the obtained hierarchical structure of Web communities. First, we obtain the hierarchically extracted Web communities $Com^q_{nq} (n_q = 1, 2, \ldots, T_q, q = 1, 2, \ldots, Q_h)$. Then we rank each Web video $f_i (i \in \{1, 2, \ldots, N\})$ that belongs to the Web community $Com^q_{nq}$ in the descending order of the following measure $R^q_{nq} (i)$:

$$R^q_{nq} (i) = \sum_{j=1}^{N} e_{ij} \delta_{ij},$$

where $\delta_{ij}$ is 1 if $f_i$ and $f_j$ belong to the same Web community and 0 otherwise. Thus, Web videos in each Web community are ranked on the basis of the number of weighted links in the graph of each Web community, i.e., the degree centrality of the graph. Moreover, we exhibit the Web communities in the order of $Com^2_{nq}, Com^1_{nq}r_{h}, \ldots, Com^1_{nq}$, that is, from larger Web communities to smaller Web communities. We notice that the larger Web communities include Web videos with various topics and the smaller Web communities contain Web videos with similar topics. Then users select the Web communities associated with the desired contents according to the exhibited hierarchical structure and retrieve the desired Web videos based on the ranking $R^q_{nq} (i)$ of each Web video. Hence, our method enables users to easily grasp an overview of many Web videos via the hierarchical structure of Web communities. As a consequence, the users can retrieve the desired Web videos even if varied topics are contained in the retrieval results since they cannot input suitable queries that identify the desired contents.
5. Experimental Results

In this section, we present experimental results for actual Web videos to verify the accuracy and computational cost of our method.

5.1 Datasets

By using YouTube Data API (v3)*7, we built datasets as follows. First, we gave a keyword as a query and obtained the top 50 Web videos. Then we repeatedly obtained 10 Web videos contained in links, i.e., metadata “related videos” of the selected Web videos, until about 3000 Web videos were collected. We collected only Web videos with lengths of less than 1800 seconds and ones to which Freebase topics were attached. Note that Freebase is a knowledge database maintained by a community supported by Google, and YouTube videos are associated with their relevant topics, e.g., “Samsung Galaxy S5” and “Google Nexus”. The detailed conditions of each dataset are shown in Table 1. Here, we note that these datasets were collected in the different time and conditions from the datasets shown in our previous paper12). Therefore, even if a query used to collect the dataset was the same, Web videos contained in each dataset were different from them used in the experiments shown in a paper12).

5.2 Evaluations

In Fig. 1, we show the hierarchical structure of Web communities obtained by our method. From this figure, we can see that the hierarchical structure enables users to easily grasp an overview of many Web videos since Web videos containing varied topics are clustered into ones with similar topics.

Next, we quantitatively evaluate the accuracy of our method. Note that we denote the proposed method by (Ours) in this evaluation. We compared (Ours) with the following methods.

(C1): This is a method to extract the hierarchical structure of Web communities in the same manner as our method, but this method does not use similarities between Web videos. Specifically, we defined edge weights $e_{ij}$ between Web videos $f_i$ and $f_j$ as follows:

$$e_{ij} = \begin{cases} 1 & \text{if } f_i \text{ links to } f_j \text{ or } f_j \text{ links to } f_i \\ 0 & \text{otherwise} \end{cases}$$

$i = 1, 2, \cdots, N, j = 1, 2, \cdots, N$.

(C2-1): This is a method based on the conventional method11) to extract the hierarchical structure of Web communities. Although the method11) does not use modularity originally, we introduced modularity into this method for comparison in this experiment. This method performs screening of Web videos to select about 1000 representative Web videos in a dataset.

(C2-2): This is a same method as (C2-1), but this method performs screening of Web videos to select about 2000 representative Web videos in a dataset.

(C3-1): This is a conventional method12) to efficiently extract the hierarchical structure of Web communities. This method performs calculation of sub-sampled CCA-based link relationships between Web videos as in our method, but screening of Web videos is performed to select about 1000 representative Web videos in a dataset.

(C3-2): This is a same method as (C3-1), but this method performs screening of Web videos to select about 2000 representative Web videos in a dataset.

(C4): This is a conventional method5) that extracts Web communities via Web video features and metadata “related videos”; however, this method does not extract the hierarchical structure.

(C5): This is a method based on a work5) with a Web community extraction scheme by affinity propagation25). This method uses only Web video features without the use of metadata “related videos”, and the hierarchical structure is not explored. Note that we do not use the original similarities in a paper25) but use our proposed similarities in this experiment.

---

*7 https://developers.google.com/youtube/v3/
Table 2 shows the number of Web videos after screening of Web videos by (C2-1), (C2-2), (C3-1) and (C3-2).

For the evaluation, we used the following F-measure and average precision (AP@k):

\[
\text{F-measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}},
\]

\[
\text{AP@k} = \frac{1}{k} \sum_{i=1}^{k} x_i \text{prec}_i,
\]

where \( k \) is the number of Web videos provided as the retrieval results, \( R_k \) is the number of “relevant Web videos” within \( k \) Web videos of the retrieval results, \( x_i \) is 1 if the \( i \)-th retrieved Web videos are “relevant Web videos” and 0 otherwise, and \( \text{prec}_i \) is the precision when \( i \) Web videos are retrieved. Here, Freebase topics related to each Web community were selected by human assessors, and then we defined each ground truth for the evaluation. Note that a few ground truths were defined on the basis of titles of Web videos since the number of their Freebase topics were too small to perform the evaluation. Table 3 shows the defined ground truths.

Since it was difficult to evaluate retrieval results when all Web communities were selected, we performed evaluation as follows. First, we selected the three largest Web communities at the highest hierarchy. Then we retrieved Web videos by using Web communities in the lowest hierarchy corresponding to the selected ones to
Table 3 Ground truths for the quantitative evaluation. Web videos with the following Freebase topics or titles were defined as “relevant Web videos”.

<table>
<thead>
<tr>
<th>Target</th>
<th>Frebase topics related to “Apple iPhone” and “Samsung Galaxy S5”.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target 1</td>
<td>Frebase topics related to “Samsung Galaxy Note”.</td>
</tr>
<tr>
<td>Target 2</td>
<td>Frebase topics related to “Universe”.</td>
</tr>
<tr>
<td>Target 3</td>
<td>Frebase topics related to “products of Ferrari”.</td>
</tr>
<tr>
<td>Target 4</td>
<td>Frebase topics related to “Google Nexus” and “Products of Motorola, Inc.”.</td>
</tr>
<tr>
<td>Target 5</td>
<td>Frebase topics related to “Japanese”.</td>
</tr>
<tr>
<td>Target 6</td>
<td>Frebase topics related to “Portugal”.</td>
</tr>
<tr>
<td>Target 7</td>
<td>Frebase topics related to “Automobile Mercedes-Benz”.</td>
</tr>
</tbody>
</table>

perform evaluation. Tables 4 and 5 show F-measures and average precision when Web videos were retrieved. When comparing our method with (C1), we can see the effectiveness of our CCA-based link relationships via visual, audio, and textual features. Also, since our method does not need to perform screening of Web videos by our efficient recursive processing unlike (C2-1), (C2-2), (C3-1) and (C3-2), evaluation results obtained by our method can outperform those obtained by these comparative methods. When comparing our method with (C4) and (C5), we can confirm that the use of the hierarchical structure enables accurate retrieval.

Next, we show computational times for obtaining the hierarchical structure of Web communities in Table 6. From this table, the computational efficiency of our method can be confirmed. In particular, it is significant to realize the results although our proposed method does not perform screening of Web videos unlike (C2-1), (C2-2), (C3-1) and (C3-2).

Consequently, we can show the effectiveness of introducing the method\textsuperscript{14} into Web video retrieval. In particular, although the method\textsuperscript{14} has not been proposed for Web video retrieval originally, we can confirm that introduction of the method\textsuperscript{14} successfully enables accurate and efficient extraction of the hierarchical structure for Web video retrieval.

6. Conclusions

In this paper, we have proposed an accurate and efficient method to extract hierarchical structure of Web communities for Web video retrieval. First, we derived CCA-based link relationships that represent similarities between latent features of Web videos by using visual, audio and textual features. Here, similarity calculation with low computational cost can be realized on the basis of efficient CCA, named sub-sampled CCA. Moreover, we enabled application of the algorithm based on recursive modularity optimization by constructing a graph whose nodes are Web videos via the obtained link relationships. Then it became feasible to extract the hierarchical structure of Web communities. Here, unlike the previously reported methods that needed to perform screening of Web videos, it can be realized for the whole target Web videos since its algorithm enabled recursive reduction of the target nodes. Experimental results have verified that our method enabled extraction of the hierarchical structure with high accuracy as well as low computational cost.

Acknowledgement

This work was partly supported by Grant-in-Aid for Scientific Research (B) 25280306, Japan Society for the Promotion of Science (JSPS), and Grant-in-Aid for Scientific Research on Innovative Areas 24120002 from the MEXT.

References

### Table 4: Quantitative evaluation for Web video retrieval by F-measures.

(A) Average values of F-measures.

<table>
<thead>
<tr>
<th>Dataset 1</th>
<th>(Ours)</th>
<th>(C1)</th>
<th>(C2-1)</th>
<th>(C2-2)</th>
<th>(C3-1)</th>
<th>(C3-2)</th>
<th>(C4)</th>
<th>(C5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target 1</td>
<td>0.506</td>
<td>0.543</td>
<td>0.175</td>
<td>0.277</td>
<td>0.277</td>
<td>0.352</td>
<td>0.345</td>
<td>0.037</td>
</tr>
<tr>
<td>Target 2</td>
<td>0.274</td>
<td>0.526</td>
<td>0.186</td>
<td>0.249</td>
<td>0.173</td>
<td>0.263</td>
<td>0.257</td>
<td>0.054</td>
</tr>
<tr>
<td>Target 3</td>
<td>0.447</td>
<td>0.355</td>
<td>0.303</td>
<td>0.264</td>
<td>0.313</td>
<td>0.359</td>
<td>0.302</td>
<td>0.062</td>
</tr>
<tr>
<td>Target 4</td>
<td>0.574</td>
<td>0.0679</td>
<td>0.108</td>
<td>0.0927</td>
<td>0.294</td>
<td>0.128</td>
<td>0.056</td>
<td>0.064</td>
</tr>
<tr>
<td>Target 5</td>
<td>0.417</td>
<td>0.510</td>
<td>0.224</td>
<td>0.180</td>
<td>0.229</td>
<td>0.215</td>
<td>0.429</td>
<td></td>
</tr>
<tr>
<td>Target 6</td>
<td>0.544</td>
<td>0.284</td>
<td>0.136</td>
<td>0.107</td>
<td>0.0670</td>
<td>0.0931</td>
<td>0.0734</td>
<td>0.143</td>
</tr>
</tbody>
</table>

(B) Maximum values of F-measures.

<table>
<thead>
<tr>
<th>Dataset 1</th>
<th>(Ours)</th>
<th>(C1)</th>
<th>(C2-1)</th>
<th>(C2-2)</th>
<th>(C3-1)</th>
<th>(C3-2)</th>
<th>(C4)</th>
<th>(C5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target 1</td>
<td>0.384</td>
<td>0.330</td>
<td>0.192</td>
<td>0.300</td>
<td>0.169</td>
<td>0.287</td>
<td>0.132</td>
<td>0.0737</td>
</tr>
<tr>
<td>Target 2</td>
<td>0.317</td>
<td>0.230</td>
<td>0.210</td>
<td>0.237</td>
<td>0.180</td>
<td>0.216</td>
<td>0.0873</td>
<td>0.0634</td>
</tr>
<tr>
<td>Target 3</td>
<td>0.471</td>
<td>0.460</td>
<td>0.251</td>
<td>0.502</td>
<td>0.229</td>
<td>0.265</td>
<td>0.170</td>
<td>0.0590</td>
</tr>
<tr>
<td>Target 4</td>
<td>0.429</td>
<td>0.053</td>
<td>0.582</td>
<td>0.242</td>
<td>0.277</td>
<td>0.190</td>
<td>0.0880</td>
<td>0.0947</td>
</tr>
<tr>
<td>Target 5</td>
<td>0.517</td>
<td>0.180</td>
<td>0.155</td>
<td>0.248</td>
<td>0.334</td>
<td>0.295</td>
<td>0.0381</td>
<td>0.0502</td>
</tr>
</tbody>
</table>

### Technical Details


Table 5  Quantitative evaluation for Web video retrieval by average precision, i.e., AP@k. We set k to the number of Web videos within Web communities used for retrieval.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Target 1</th>
<th>Target 2</th>
<th>Target 3</th>
<th>Target 4</th>
<th>Target 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours</td>
<td>0.947</td>
<td>0.857</td>
<td>0.368</td>
<td>0.357</td>
<td>0.488</td>
</tr>
<tr>
<td>(C1)</td>
<td>0.372</td>
<td>0.368</td>
<td>0.126</td>
<td>0.126</td>
<td>0.368</td>
</tr>
<tr>
<td>(C2-1)</td>
<td>0.947</td>
<td>0.857</td>
<td>0.368</td>
<td>0.357</td>
<td>0.488</td>
</tr>
<tr>
<td>(C2-2)</td>
<td>0.947</td>
<td>0.857</td>
<td>0.368</td>
<td>0.357</td>
<td>0.488</td>
</tr>
<tr>
<td>(C3-1)</td>
<td>0.947</td>
<td>0.857</td>
<td>0.368</td>
<td>0.357</td>
<td>0.488</td>
</tr>
<tr>
<td>(C3-2)</td>
<td>0.947</td>
<td>0.857</td>
<td>0.368</td>
<td>0.357</td>
<td>0.488</td>
</tr>
<tr>
<td>(C4)</td>
<td>0.947</td>
<td>0.857</td>
<td>0.368</td>
<td>0.357</td>
<td>0.488</td>
</tr>
<tr>
<td>(C5)</td>
<td>0.947</td>
<td>0.857</td>
<td>0.368</td>
<td>0.357</td>
<td>0.488</td>
</tr>
</tbody>
</table>

Table 6  Computational time for obtaining the hierarchical structure of Web communities. We used a computer with Intel® CPU 3.07GHz and 16GB RAM.

(A) Extraction of Web video features. (The image size was 320 × 180 pixels and the frame rate was 24 fps. The audio signal was sampled at 22.05 kHz.)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Visual feature</th>
<th>Audio feature</th>
<th>Textual feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Ours)</td>
<td>(C1-1)</td>
<td>(C2-1)</td>
<td>(C3-1)</td>
</tr>
</tbody>
</table>

(B) Derivation of CCA-based link relationships between Web videos.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>(Ours), (C3-1) and (C2-2) include k-means clustering, but (C2-1), (C2-2), (C4) and (C5) do not.</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Ours)</td>
<td>(C1-1)</td>
</tr>
<tr>
<td>961.8 (sec)</td>
<td>1849.6 (sec)</td>
</tr>
</tbody>
</table>

(C) Preceding of Web videos for extracting hierarchical structure of Web communities.

Note that (C2-1), (C2-2), (C3-1) and (C3-2) need to perform this processing.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>(Ours)</th>
<th>(C1-1)</th>
<th>(C2-1)</th>
<th>(C3-1)</th>
<th>(C3-2)</th>
<th>(C4)</th>
<th>(C5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1442.4 (sec)</td>
<td>1745.0 (sec)</td>
<td>1745.0 (sec)</td>
<td>1442.2 (sec)</td>
<td>1442.2 (sec)</td>
<td>3090.1 (sec)</td>
<td>1442.2 (sec)</td>
<td></td>
</tr>
</tbody>
</table>

(D) Extraction of the hierarchical structure of Web communities. As for (C4) and (C5), computation time for obtaining not the hierarchical structure but Web communities is shown since the methods do not explore the hierarchical structure.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>(Ours)</th>
<th>(C1-1)</th>
<th>(C2-1)</th>
<th>(C3-1)</th>
<th>(C3-2)</th>
<th>(C4)</th>
<th>(C5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0 (sec)</td>
<td>0.9 (sec)</td>
<td>103.3 (sec)</td>
<td>5266.2 (sec)</td>
<td>5.7 (sec)</td>
<td>22.6 (sec)</td>
<td>13.1 (sec)</td>
<td>185.7 (sec)</td>
</tr>
</tbody>
</table>

(E) Total time for obtaining the hierarchical structure except for extraction of Web video features, i.e., the sum of (B), (C) and (D).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>(Ours)</th>
<th>(C1-1)</th>
<th>(C2-1)</th>
<th>(C3-1)</th>
<th>(C3-2)</th>
<th>(C4)</th>
<th>(C5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>963.8 (sec)</td>
<td>0.9 (sec)</td>
<td>1957.3 (sec)</td>
<td>7120.2 (sec)</td>
<td>7672.1 (sec)</td>
<td>7689.2 (sec)</td>
<td>2426.5 (sec)</td>
<td>1147.3 (sec)</td>
</tr>
</tbody>
</table>


Ryosuke Harakawa received his B.S. and M.S. degrees in Electronics and Information Engineering from Hokkaido University, Japan in 2013 and 2015, respectively. He is currently pursuing a Ph.D. degree at the Graduate School of Information Science and Technology, Hokkaido University. His research interests include audiovisual processing and Web mining. He is a student member of the IEEE, IEICE, and Institute of Image Information and Television Engineers (ITE).

Takahiro Ogawa received his B.S., M.S. and Ph.D. degrees in Electronics and Information Engineering from Hokkaido University, Japan in 2003, 2005 and 2007, respectively. He is currently an assistant professor in the Graduate School of Information Science and Technology, Hokkaido University. His research interests are multimedia signal processing and its applications. He has been an Associate Editor of ITE Transactions on Media Technology and Applications. He is a member of the IEEE, EURASIP, IEICE, and Institute of Image Information and Television Engineers (ITE).

Miki Haseyama received her B.S., M.S. and Ph.D. degrees in Electronics from Hokkaido University, Japan in 1986, 1988 and 1993, respectively. She joined the Graduate School of Information Science and Technology, Hokkaido University as an associate professor in 1994. She was a visiting associate professor of Washington University, USA from 1995 to 1996. She is currently a professor in the Graduate School of Information Science and Technology, Hokkaido University. Her research interests include image and video processing and its development into semantic analysis. She has been a Vice-President of the Institute of Image Information and Television Engineers, Japan (ITE), an Editor-in-Chief of ITE Transactions on Media Technology and Applications, a Director, International Coordination and Publicity of The Institute of Electronics, Information and Communication Engineers (IEICE). She is a member of the IEEE, IEICE, Institute of Image Information and Television Engineers (ITE) and Acoustical Society of Japan (ASJ).