Topic threading for structuring a large-scale news video archive

Ichiro Ide[1,2], Hiroshi Mo[2], Norio Katayama[2], and Shin'ichi Satoh[2]

[1] Graduate School of Information Science, Nagoya University
1 Furo-cho, Chikusa-ku, Nagoya-shi, Aichi 464–8601, Japan
ide@is.nagoya-u.ac.jp
[2] National Institute of Informatics
Research Organization of Information and Systems
2–1–2 Hitotsubashi, Chiyoda-ku, Tokyo 101–8430, Japan
{mo,katayama,satoh}@nii.ac.jp

Abstract. We are building a broadcast news video archive where topics of interest can be retrieved and tracked easily. This paper introduces a structuring method applied to the accumulated news videos. First they are segmented into topic units and then *threaded* according to their mutual relations. A user interface for topic thread-based news video retrieval is also introduced. Since the topic thread structure is formed so that it has fewer number of emerging links from each topic than a simple link structure of related topics, it should lessen the tedious selection during a tracking process by a user. Although evaluation of the effect of threading and user study on the interface is yet to be done, we have found the interface informative to understand the details of a topic of interest.

1 Introduction

Broadcast video, especially news video contains a broad range of human activities which could be considered as a valuable cultural and social heritage. We are building a broadcast news video archive where topics of interest can be retrieved and tracked easily. The archive is supported by an automatic archiving system, a back-end contents analysis process, and a front-end user interface. In this paper, we will mainly focus on introducing the back-end contents analysis process, where the accumulated news videos are segmented into topic units and then *threaded* according to their mutual relations, and the front-end user interface.

The automatic archiving system captures and records broadcast news video streams including closed-caption texts (transcripts of audio), while the meta data are stored in a relational database. Currently, we have approximately 495 hours (312 GB of MPEG-1 and 1.89 TB of MPEG-2 format videos, and 23.0 MB of closed-caption texts) in the archive, obtained from a specific Japanese daily news program since March 16, 2001 (1,034 days in total). Each night, after the day's program is added to the database, the back-end contents analysis process will run. The process will be finished by the next morning so that a user can browse through the archive that reflects the topics added the previous night.

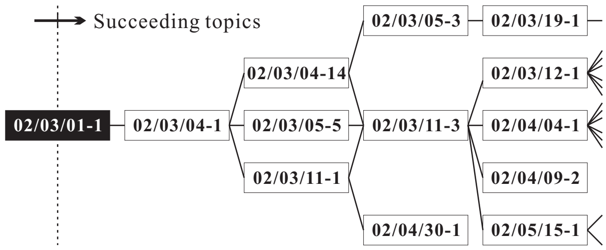

Fig. 1. Part of a topic thread structure extracted from the archive. Topics are labeled in the following format "Year/Month/Day-Topic#".

Topic thread structure in a news video archive A news video archive may seem merely an accumulation of video files recorded every day. Majority of previous works on news video structure analysis concentrated on segmenting a video stream into semantic units such as topics for retrieval (The most recent one: Yang *et al.* 2003). However, such retrieval is efficient only while the size of the archive remains relatively small. Once the archive grows larger, even the selection among the retrieved units becomes tedious for a user. Although several groups are dealing with news video archives of a comparable size with ours (Merlino *et al.* 1997; Christel *et al.* 1999), they do not look into the semantic relations between chains of topics. Works in Web structure mining is somewhat related to our work, but the existence of the chronological restriction makes our target substantially different.

We consider that linking semantically related topics in chronological order (*threading*) should be a solution to overcome this problem by providing a user with a list of topic threads instead of a whole list of individual topics. Once the topic thread structure of the entire archive has been revealed, it will no longer be a mere accumulation of video files, but a video archive where the threads complexly interweave related topics. Figure 1 shows an example of a topic thread structure starting from a topic of interest, which was actually obtained from the archive by the method proposed in this paper. As seen in the example, topic threads merge and diverge along time reflecting the transition of a story in the real world. Compare Fig. 1 with Fig. 2 which shows a simple link structure of related topics sorted chronologically. When providing a topic tracking interface by showing topics linked from a topic of interest, the fewer the number of links from each node exist, the less tedious the selection should be for a user.

In the following Sections, the topic segmentation and threading methods are introduced, followed by the introduction of a thread-based browsing interface.

2 Topic structuring

2.1 Topic segmentation

A news topic is a semantic segment within a news video which contains a report on a specific incident. Compared with topic segmentation in general docu-

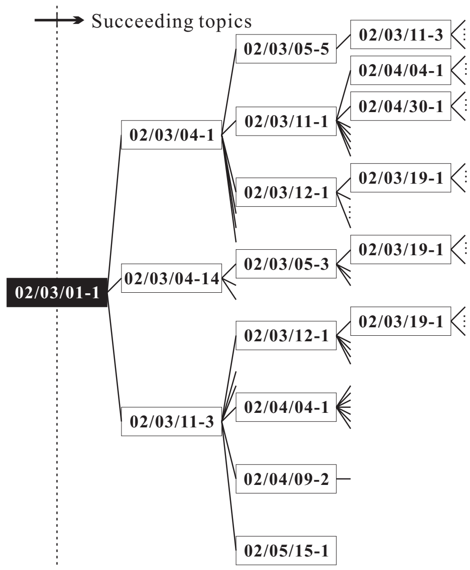

Fig. 2. Example of a simple topic link structure without threading.

ments, topic boundaries in a news video should be relatively clear, since a news video is naturally a combination of originally individual topics. Following this assumption, we will detect a topic boundary by finding a point between sentences where the keyword distributions within preceding and succeeding windows are distinctly different. The ideal window sizes are those when they are exactly the same with the actual topic lengths on both sides. Since the actual topic length is unknown beforehand, we will set elastic windows at each point between sentences to evaluate the discontinuity in various window sizes.

Procedure The following steps were taken to detect a topic boundary from a closed-caption text broadcast simultaneously with the video. The closed-caption currently used is basically a transcript of the audio speech, though occasionally it is overridden by a source script or omitted when a superimposed caption is inserted in the video.

1. Apply morphological analysis to each sentence of a closed-caption text to extract compound nouns. A Japanese morphological analysis software, JUMAN (Kyoto Univ. 1999) was employed. Compound nouns were extracted since combination of adjacent nouns was considered as more distinctive to represent a topic, rather than a group of individual nouns.
2. Apply semantic analysis to the compound nouns to generate a keyword frequency vector for each semantic class (general, personal, locational / organizational, or temporal) per sentence ($\boldsymbol{k_g}, \boldsymbol{k_p}, \boldsymbol{k_l}, \boldsymbol{k_t}$), which has frequencies as values. A suffix-based method (Ide *et al.* 2003) was employed for the analy-

sis, which classifies compound nouns both with and without proper nouns, according to suffix dictionaries for each semantic class.

3. At each boundary point between sentences i and $i+1$, set a window size w, and measure the difference of keyword distributions between w preceding and succeeding sentences. The difference (or rather resemblance) is defined as follows, where $i = w, w+1, ..., i_{max} - w$ when i_{max} is the number of sentences in a daily closed-caption text, and $S = \{g, p, l, t\}$.

$$R_{S,w}(i) = \frac{\sum_{m=i-w+1}^{i} \boldsymbol{k_S}(m) \cdot \sum_{n=i+1}^{i+w} \boldsymbol{k_S}(n)}{\left|\sum_{m=i-w+1}^{i} \boldsymbol{k_S}(m)\right| \left|\sum_{n=i+1}^{i+w} \boldsymbol{k_S}(n)\right|} \tag{1}$$

(2)

   We set $w = 1, 2, ..., 10$ in the following experiment.

4. The maximum of $R_{S,w}(i)$ among all w is chosen at each boundary as follows.

$$R_S(i) = \max_w R_{S,w}(i) \tag{3}$$

   In preliminary observations, although most boundaries were correctly detected regardless of w, there was a large number of over-segmentation. We considered that taking the maximum should mutually compensate for over-segmentations at various window sizes, due to the following tendencies.
   - Small w: Causes numerous over-segmentations, but has the advantage of showing significantly high resemblance within a short topic.
   - Large w: Does not always show high similarity within a short topic, but shows relatively high resemblance within a long one.

5. Resemblances evaluated in separate semantic attributes are combined as a weighted sum as follows.

$$R(i) = \sum_{S=\{g,p,l,t\}} a_S R_S(i) \tag{4}$$

   Different weights are assigned to each semantic class under the assumption that certain attributes should be more important than others when considering topic segmentation especially in news texts.
   Multiple linear regression analysis was applied to manually segmented training data (consists of 39 daily closed-caption texts, with 384 manually given topic boundaries) to determine the weights. The following weights were obtained as a result.

$$(a_g, a_p, a_l, a_t) = (0.23, 0.21, 0.48, 0.08) \tag{5}$$

   The weights show that temporal nouns (*e.g.* today, last month) are not distinctive in the sense of representing a topic, where the other three, especially locational / organizational nouns act as distinctive keywords.
   Finally, if $R(i)$ does not exceed a certain threshold θ_{seg}, the point is judged as a topic boundary.

6. To concatenate over-segmented topics, create a keyword vector $\boldsymbol{K_S}$ for each topic, and re-evaluate the resemblances between adjoining stories i and $j(= i+1)$ by the following function.

$$R(i,j) = \sum\nolimits_{S=\{g,p,l,t\}} a_S \frac{\boldsymbol{K_S}(i) \cdot \boldsymbol{K_S}(j)}{|\boldsymbol{K_S}(i)|\,|\boldsymbol{K_S}(j)|} \tag{6}$$

As for a_S, the same weights as in Equation 5 were used.
If $R(i,j)$ does not exceed a certain threshold θ_{cat}, the topics are concatenated. This process continues until no more concatenation occurs.

Experiment and evaluation The procedure was applied first to the training data used in Step 5. to define the thresholds ($\theta_{seg} = 0.28, \theta_{cat} = 0.08$), and later to the entire archive ranging from March 16, 2001 to April 9, 2004 (1,034 days with 132,581 sentences in total). The whole process takes approximately 5 seconds per day on a Sun Blade-1000 workstation with dual UltraSPARC-III 750MHz CPUs and 2GB of main memory. As a result, 13,211 topics with more than two sentences were extracted. Topics with only one sentence (25,403 topics) were excluded since they tend to be noisy fragments resulting from over-segmentation.

Table 1. Evaluation of topic extraction.

Condition	Both ends strict	One end strict / loose	Both ends loose
Recall	30.0%	34.6%	95.4%
Precision	28.5%	32.8%	90.5%

Evaluation was performed by applying the same procedure to manually segmented test data (consists of 14 daily closed-caption texts with 130 topics, set aside from the training data), which resulted as shown in Tab. 1. Boundaries were counted as correct if they matched exactly with the manually determined ones in ‘strict’ condition, and allowing ± 1 sentences in ‘loose’ condition. The ‘loose’ condition is acceptable for our goal since sentences at true boundaries tend to be short and relatively less informative regarding the main news content.

2.2 Topic threading

A topic thread structure starting from a topic of interest is formed so that related topics are linked in chronological order. The difference between simply expanding related topics node by node as in Fig. 2 is that the proposed method replaces a link to a subordinate node if possible. The structure will therefore be rather *flat*; few branches at each node, and a long sequence of related topics instead. By this method, a topic that may eventually be reached in the tracking process will be pushed down so that a user needs not select among dozens of topics that he/she would never even need to see.

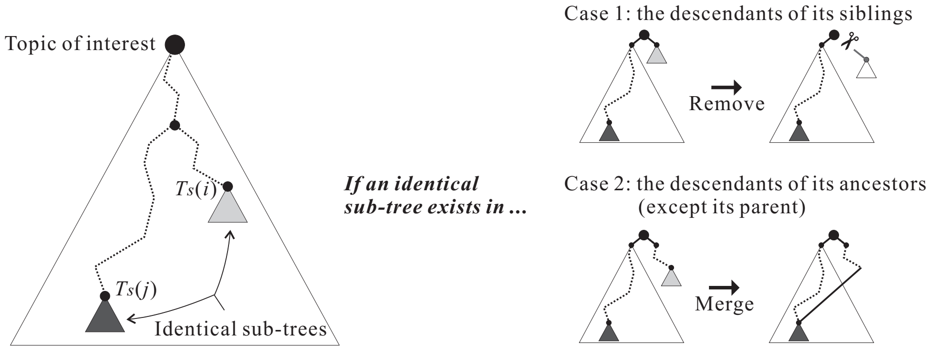

Fig. 3. Topic threading scheme.

Procedure The thread structure is formed by the following algorithm.

1. Expand a topic link tree starting from the topic of interest so that it satisfies the following conditions.
   (a) Children are topics related to a parent, under the condition that their time stamps always succeed their parent's chronologically.
   (b) Siblings are sorted so that their time stamps always succeed their left-siblings' chronologically.

   The resemblance between topics are evaluated by Equation 6. When $R(i, j)$ exceeds a threshold θ_{trk}, the topics are considered as related. This procedure forms a simple topic link tree such as the structure in Fig. 2.

   Since evaluating numerous resemblances between various topics consumes too much time for real time processing in the user interface, resemblances between all possible topic pairs are evaluated beforehand. Currently, it takes roughly 1,400 seconds to add one new topic (comparing one topic against approximately 12,000 topics), which will keep on increasing as the archive grows larger.
2. For each sub-tree $T_s(i)$, if an identical sub-tree $T_s(j)$ exists on the left-side, perform either of the following operations.
   (a) Remove $T_s(i)$ if $T_s(j)$ is a descendant of $T_s(i)$'s sibling.
   (b) Else, merge $T_s(i)$ with $T_s(j)$ if $T_s(j)$ is a descendant of $T_s(i)$'s ancestor except its parent.

   The sub-tree is removed in (a) instead of merging, to avoid creating a short-cut link without specific meaning. The removal and merger scheme is shown in Fig. 3. As a result of this operation, the thread structure will form a chronologically-ordered directed graph.

Note that this is in the case of forming a succeeding thread structure. A chronologically opposite algorithm is applied when forming a preceding thread structure.

To reduce computation time, the following conditions are applied in practice.

1. Pruning: Perform Step 2. whenever an identical story is found during the expansion of the tree in Step 1.
2. Approximation: Interrupt the expansion of the tree at a certain depth N_{trk}.

Although Condition 2. approximates the result, this will not affect much when referring to the direct children of the root (topic of interest) if N_{trk} is set to an appropriate value (We found $N_{trk} = 3 \sim 5$ as sufficient in most cases).

3 Topic thread-based video retrieval

We built a topic retrieval interface, namely the "Topic Browser", so that a user can browse through the entire news video archive by tracking up and down the topic threads. The interface consists of a "Topic Finder" and a "Topic Tracker", which can be switched by tabs.

The "Topic Finder" The "Topic Finder" is the portal to the tracking process; it retrieves a list of topics that contain the query term (Figure 4). A topic is represented by its meta data (date, time, topic ID), a thumbnail image (the first frame of the associated video), and an excerpt of the associated closed-caption text. The user can select the initial topic for tracking among the listed topics by actually viewing the video and associated close-caption text displayed on the right side of the interface.

The "Topic Tracker" Once the user selects an initial topic, he/she will choose the "Topic Tracker" tab. To provide a list of topic threads starting from the initial topic, it performs the threading process as described in Sect. 2.2 on the fly (Figure 5). A topic thread is represented by the first topic in it, and key phrases that represent it so that the user can distinguish the difference with other threads. The first topics are selected to represent the threads since they were evaluated that they do not resemble each other, thus are considered as nodes where the topics diverge. The key phrases are noun sequences selected exclusively from the representative topic so that they do not overlap with those in other threads. The interface allows the user to set θ_{trk}, N_{trk} to adjust the number of threads to be displayed and the computation time.

The user will keep on selecting a topic over and over until he/she understands the details of the story during the tracking process, or finally finds a certain topic. The tracking direction is switchable so that it could go back and forth in time.

While "Topic Tracking" is in general a part of the "Topic Detection and Tracking (TDT) task" (Wayne 2000) in the TREC Conference, their definition of *tracking* and *detection* is somewhat static compared to what we are trying to realize in this interface. The point is that the proposed topic threading method extracts various paths that gradually track topics of interest that a user may follow requires our *tracking* to be more dynamic.

Fig. 4. The “Topic Finder” interface. Result of a query “Bin Laden”.

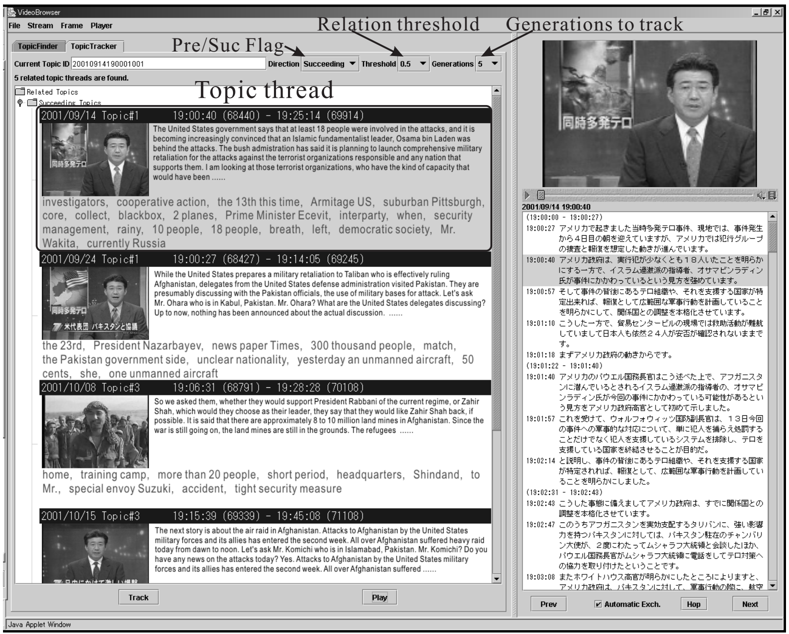

Fig. 5. The “Topic Tracker” interface.

4 Conclusions

We have proposed a method to reveal a topic-thread structure within a large-scale news video archive. The thread structure was applied to a video retrieval interface so that a user can track topics of interest along time. Since even the most relevant topic-based video retrieval method (Smeaton *et al.* 2003) considers the topic structure as simple links of related topics, the proposed approach is unique. Although precise evaluations and user studies are yet to be done, we have found the interface informative to understand the details of a topic of interest.

We will further aim at integrating image-based relations employing such methods as described in (Yamagishi *et al.* 2003) to link video segments that are related by semantics that could not be obtained from text. Precision of the tracking might be improved by refining the relation evaluation scheme by comparing a topic to a group of topics in a thread. A user study will also be performed to improve the retrieval interface after introducing relevance feedback in the tracking process, refining the keyword/thumbnail selection scheme, and so on. Evaluation of the method to the TDT (Wayne 2000) corpus is an important issue, though the system will have to be adapted to non-Japanese transcripts.

Acknowledgement

Part of the work presented in this paper was supported by the Grants-in-Aid for Scientific Researches (15017285, 15700116) from the Japanese Society for the Promotion of Science, and the Japanese Ministry of Education, Culture, Sports, Science and Technology.

References

Christel, M. G., Hauptmann, A. G., Ng, T. D.: Collages as dynamic summaries for news video. Proc. 10th ACM Intl. Conf. on Multimedia (2002) 561–569

Ide, I., Hamada, R., Sakai, S., Tanaka, H.: Compilation of dictionaries for semantic attribute analysis of television news captions. Systems and Computers in Japan **34(12)** (2003) 32–44

Kyoto University: Japanese morphological analysis system JUMAN version 3.61.

Merlino, A., Morey, D., Maybury, M.: Broadcast news navigation using story segmentation. Proc. 5th ACM Intl. Conf. on Multimedia (1997) 381–391

Smeaton, A. F., Lee, H., O'Conner, N. E., Marlaw, S., Murphy, N.: TV news story segmentation, personalization and recommendation. AAAI Press Technical Reports, **SS-03-08** (2003) 60–65

Wayne, C. L.: Multilingual topic detection and tracking: Successful research enabled by corpora and evaluation. Proc. 2nd Intl. Conf. on Language Resources and Evaluation **3** (2000) 1487–1493

Yamagishi, F., Satoh, S., Hamada, T., Sakauchi, M.: Identical video segment detection for large-scale broadcast video archives. Proc. 3rd Intl. Workshop on Content-based Multimedia Indexing (2003) 135–142

Yang, H., Chaisorn, L., Zhao, Y., Neo, S., Chua, T.: VideoQA: Question and answering on news video. Proc. 11th ACM Intl. Conf. on Multimedia (2003) 632-641