IIAI Open Conference Publication Series
IIAI Letters on Informatics and Interdisciplinary Research
Vol.006, LIIR458
DOI: https://doi.org/10.52731/liir.v006.458

A Method for Converting Lecture Videos into Microcontents and Visualizing Them

Masako Furukawa *, Yoshitomo Yaginuma †

Abstract

With the spread of online courses, a large amount of lecture videos has become available. In order to make more advanced use of existing lecture videos, it is necessary to convert the videos into microcontents and visualize them in a way that allows users to easily find the parts they need. For this purpose, in this paper, we propose a method for converting lecture videos into microcontents. The method vectorizes transcripts of videos using Doc2Vec and then divides videos based on the distance between these vectors. In addition, we compared visualization methods using principal component analysis, multidimensional scaling, and t-SNE, and found that t-SNE is suitable for the visualization.

Keywords: Microcontents, Lecture video, Doc2Vec, Visualization

1 Introduction

In the field of education, online courses such as MOOCs have become widespread, making it common to have learning opportunities over the Internet. With the spread of online courses, a large amount of lecture videos has become available. Many of these digital learning materials are available free of charge, and it is expected that they will be put to effective use. Regarding the recommendation of such educational contents, for example, Yang et al proposed a Knowledge Entity-Aware Model using Deep Learning [1]. Khan et al proposed session-based course recommendation frameworks using Deep Learning [2]. Wang et al proposed a page jump recommendation model for e-book system [3]. Zhu et al proposed instructional video clip recommendation system for MOOC forum questions [4].

However, in order to make more advanced use of existing lecture videos, it is necessary to convert the videos into microcontents and show them in a way that allows users to easily find the parts they need. Microcontents are small semantic units of learning materials, and by combining microcontents, learners can realize learning according to their interests, objectives, and abilities [5]. Visualization system and recommendation system and so on have been proposed to utilize such microcontents [6-9]. Converting learning materials into microcontents can also be used to analyze learning behavior that reflects the semantic contents.

On the other hand, there are many existing learning materials that have been developed without considering conversion to microcontents. When trying to convert existing lecture videos into microcontents, for example, it is necessary to divide a one-hour lecture video into several

* National Institute of Informatics, Tokyo, Japan
† The Open University of Japan, Chiba, Japan

minutes of contents. As a basis for automating such a process, this paper proposes a method to convert lecture videos into microcontents and visualize them. The method vectorizes transcripts of videos using Doc2Vec and performs dividing of videos based on the distance between these vectors. In addition, to realize visualization that enables us to find what we need in microcontents, we compared visualization methods using principal component analysis, multidimensional scaling, and t-SNE.

In the following, in section 2, we first describe the microcontents search interface, which is based on the microcontents conversion and visualization methods described in this paper. In section 3, we propose a method to convert lecture videos into microcontents. In section 4, we discuss visualization methods for microcontents.

2 Interface for Microcontents Search

Figure 1 shows a prototype of the interface for searching microcontents using the visualization method described in section 4. In the upper left corner, there is a text field for entering search keywords, where "natural language processing" is entered as the search keyword. Below that, icons and transcripts of microcontents containing the search keywords are shown. The images in in the middle of each video are used as icons. The original transcripts are in Japanese, but the English translations are shown here.

The right side of the interface is the area that uses the visualization results described in section 4, and the area can be zoomed in and out, and moved up, down, left, and right, just as in general map applications. The area displayed in Figure 1 corresponds to the upper right area of Figure 5, and icons of microcontents containing search keywords are displayed. This interface allows the user to see the similarity between microcontents, and to find the microcontents they need by paying attention to the similarity. By clicking on the displayed icons, the corresponding microcontents can be shown.

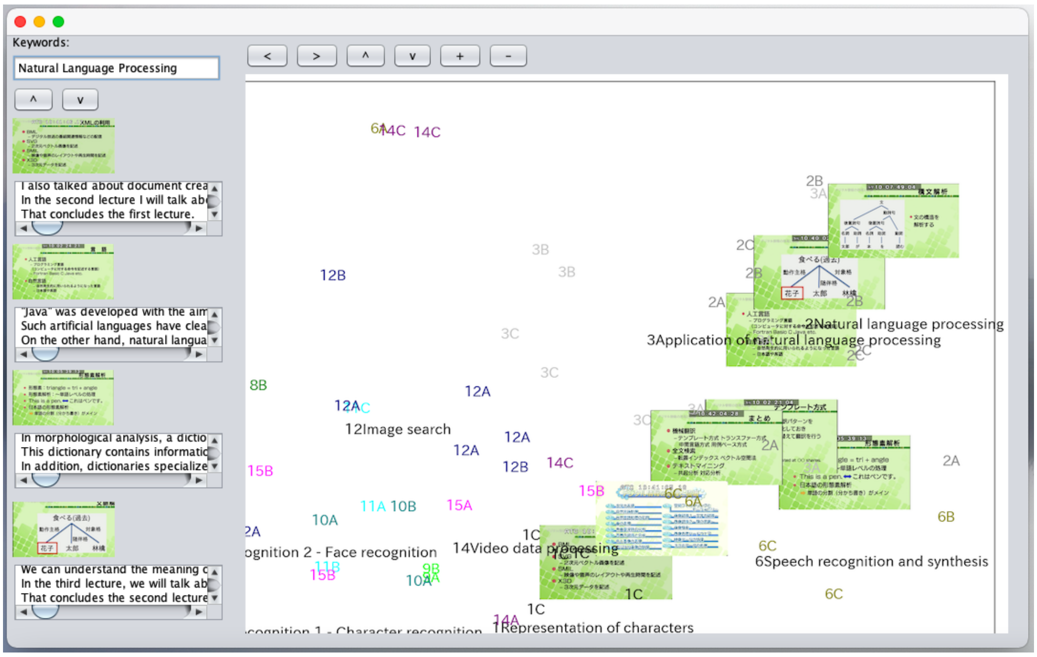

Figure 1: Interface for microcontents search

3 Converting Lecture Videos into Microcontents

As a method for converting lecture videos into microcontents, we propose a method which vectorizes transcripts of videos using Doc2Vec and then divide videos based on the distance between these vectors.

The concrete procedure for video dividing is as follows: First, Assuming the number of sentences in transcripts is w, the texts in the range of w/16 before and w/16 after the x-th sentence are vectorized using Doc2Vec. Since the target teaching materials are expressed in Japanese, we used the Doc2Vec model trained on Japanese Wikipedia [10] for vectorization. Doc2vec models trained in other languages are also publicly available, so it is also possible to target other languages.

Then, the difference between the document vectors is calculated as follows:

Difference between the document vectors = 1-(v1 ・ v2)/|v1||v2|.

Here, the vector in the range of w/16 before the x-th sentence is v1 and the vector in the range of w/16 after the x-th sentence is v2. If the angle between v1 and v2 is θ, then this becomes (1-cosθ).

An evaluation experiment of the proposed method was conducted using 15 45-minute videos. For each video, a transcript file with one sentence per line was prepared and used for the analysis. Each video contains between 185 and 264 sentences. In each of these videos, one to two intermission telops, that show the title of the contents, are included. Since these intermissions are desired to be detected, they are used as the dividing points that must be detected.

Figure 2 shows the difference between the document vectors for one of the videos. The horizontal axis represents the sentence number, and the vertical axis represents the difference between the document vectors. The green vertical lines are the dividing points where the intermission telops are shown in the video. The red vertical dotted lines are the division points detected by the proposed method. The numbers on the upper side of the lines indicate the order of detection.

Looking at the points detected, the division point 7 corresponding to the intermission is correctly detected as the break. The part up to this point corresponds to the part that talks on CCDs. In this range, three division points (2, 8, and 4) are detected. The division point 2 corresponds to the point where the discussion moves from a general explanation of what semiconductors are to an explanation of n-type and p-type semiconductors. The division point 8 corresponds to the point where the discussion moves to the explanation of the case where n-type and p-type semiconductors are combined. The division point 4 corresponds to the point that moves to the explanation of CCDs.

The division point 3 corresponding to the intermission is also correctly detected as the break. The area between division point 7 and division point 3 corresponds to the part that talks about displays. In this range, two division points (1 and 6) are detected. The division point 1 corresponds to the point where the discussion moves from the explanation of cathode-ray tubes, etc. to liquid crystal displays (LCDs). The division point 6 corresponds to the point where the topic moves to the explanation of polarization used in liquid crystal displays. The part after the division point 3 describes printers, and division point 5 roughly corresponds to the point where the explanation moves from inkjet printers to laser printers.

From these results, we can see that the proposed method is able to detect semantic division points well.

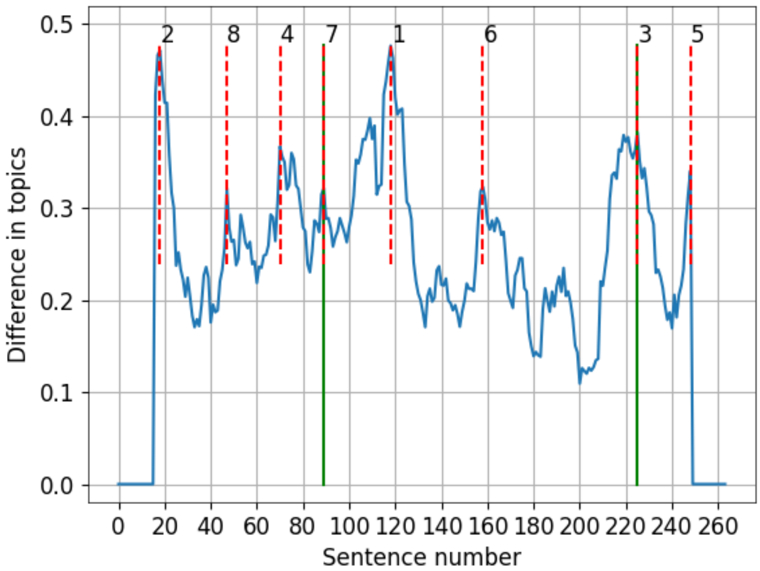

Figure 2: Differences between document vectors (Doc2Vec)

4 Visualization of Microcontents

Next, visualization of the microcontents obtained by dividing the videos is discussed. The visualization procedure involves extracting the transcripts corresponding to each microcontent and converting them into 300-dimensional vectors using Doc2Vec. Here, we performed three methods to visualize these 300-dimensional vectors: principal component analysis, multidimensional scaling, and t-SNE, and compared the results.

In the previous section, we detected eight breaks per video, which means that each video is divided into nine microcontents. In the following experiments, we visualized these 135 microcontents.

4.1 Principal Component Analysis

Figure 3 shows the results of the principal component analysis. Principal component analysis was performed on the 300-dimensional vectors obtained from 135 microcontents, and the dimension was reduced to two dimensions. The horizontal axis is the first axis obtained by the principal component analysis, and the vertical axis is the second axis obtained by the principal component analysis.

When visualizing microcontents, it is desirable to place similar microcontents as close together as possible and dissimilar microcontents as far apart as possible. To see whether this was done properly, microcontents were labeled with a combination of the video number and a letter representing the parts separated by intermissions, such as 1A. In addition, the title of each video is written in black at the center of gravity of each video.

The results show that, microcontents are well grouped together in each video. For example, 5A, 5B, and 5C, which are labeled in red in the upper center of the figure, are close to each other. However, microcontents are not necessarily displayed with uniform density, and there are areas where microcontents are densely displayed.

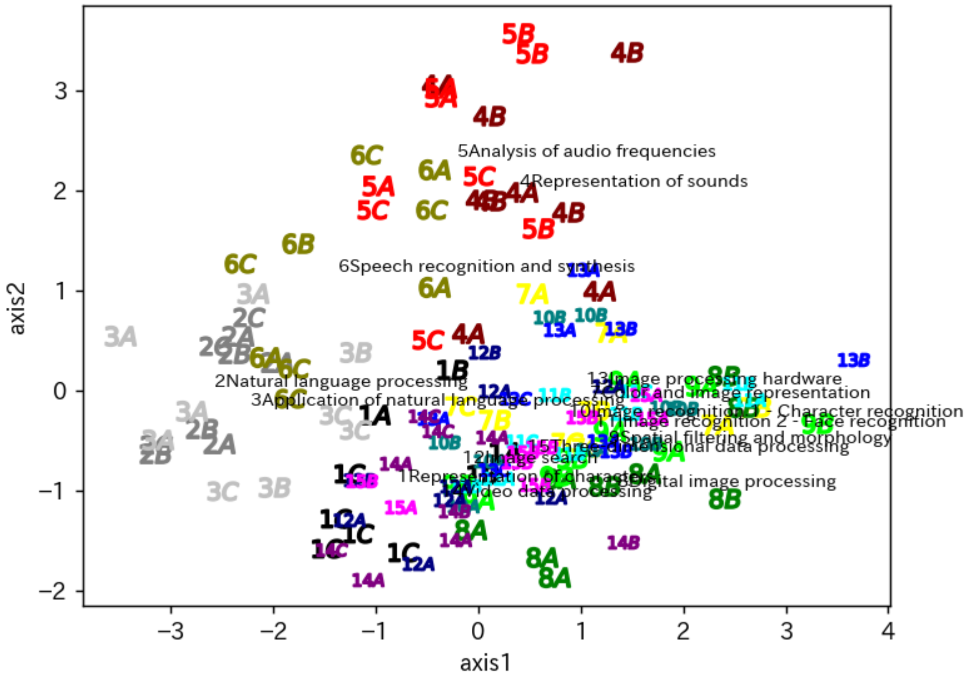

Figure 3: Visualization results (Principal component analysis)

4.2 Multidimensional Scaling

As the second method, visualization using the multidimensional scaling was performed. As the distance between vectors, we used (1 - Cosine similarity), which is often used to calculate the distance between document data, instead of the Euclidean distance. We then used the multidimensional scaling to find the two-dimensional coordinates that best reproduce the distance. The result is shown in Figure 4.

Looking at Figure 4, for example, 5A, 5B, and 5C, indicated by the red labels in the left center of the figure, are clustered close to each other. This indicates that the similar microcontents are grouped together. We also see that the labels are more evenly distributed than in the case of the principal component analysis. This is because the distance used is (1 - Cosine similarity), and the distances between labels are similar values.

4.3 t-SNE

The third method used t-SNE. t-SNE is a method for nonlinear dimensionality reduction that arranges data so that similar data are close to each other and dissimilar data are far from each other with a high probability.

Figure 5 shows the results of visualization. Similar to other methods, we can see that labels of same videos are clustered close together. However, the results of t-SNE depend on a parameter called perplexity. Figure 5 shows the results when the perplexity is 40, and it can be seen that microcontents are displayed with a more uniform density.

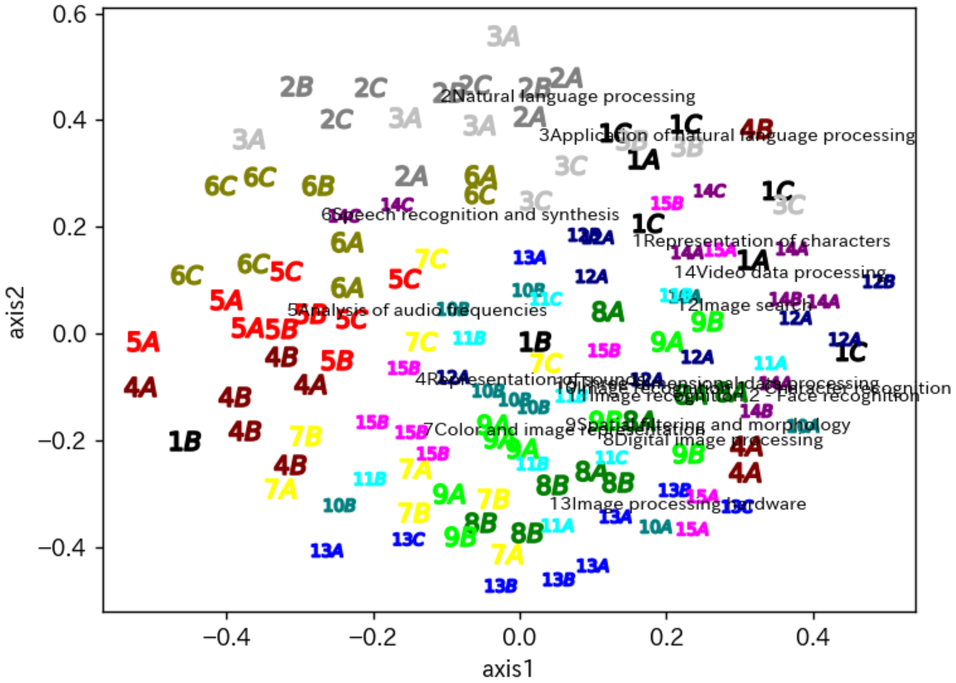

Figure 4: Visualization results (Multidimensional scaling)

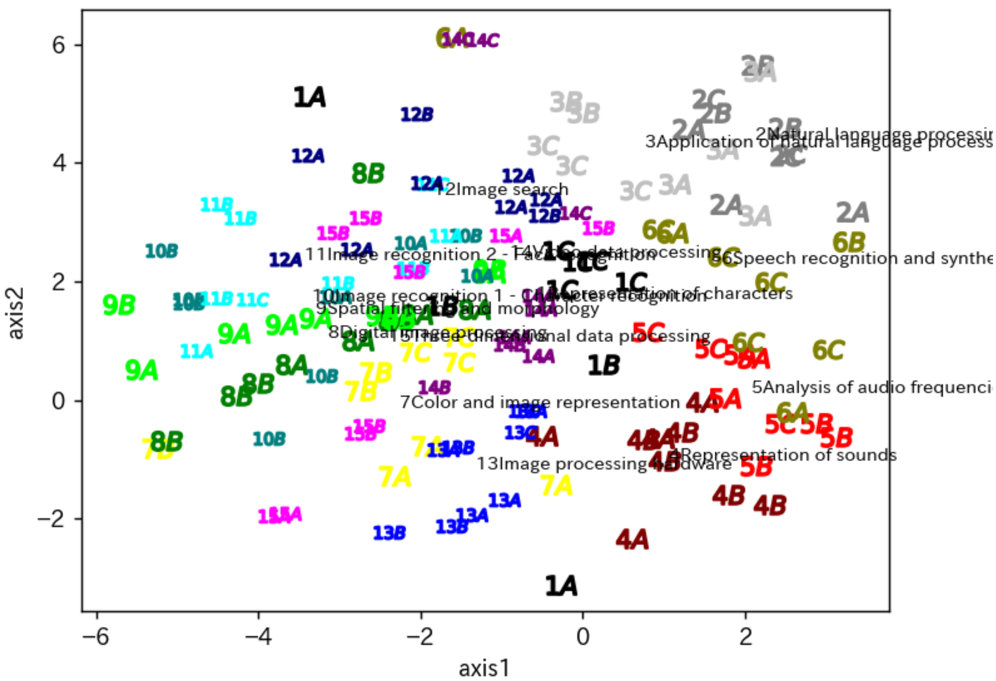

Figure 5: Visualization results (t-SNE, perplexity=40)

4.4 Comparison of Visualization Methods

Here we compare the three visualization methods mentioned so far. If the methods are good, the arrangement should be such that similar microcontents are close together and different microcontents are far apart. In this case, the value of (between-class variance / within-class variance) is expected to be larger. Figure 6 shows the values of (between-class variance / within-class variance) for each method assuming that the same label, such as 1A, is the same class. The arrangement of t-SNE changes depending on the perplexity parameter, so the results are shown for perplexities of 30, 40, and 50.

The results show that the multidimensional scaling method has a smaller value of (between-class variance / within-class variance). Principal component analysis and t-SNE have similar values, but, when the perplexity is 40, the value of (between-class variance / within-class variance) is the largest, indicating that this is the most suitable arrangement.

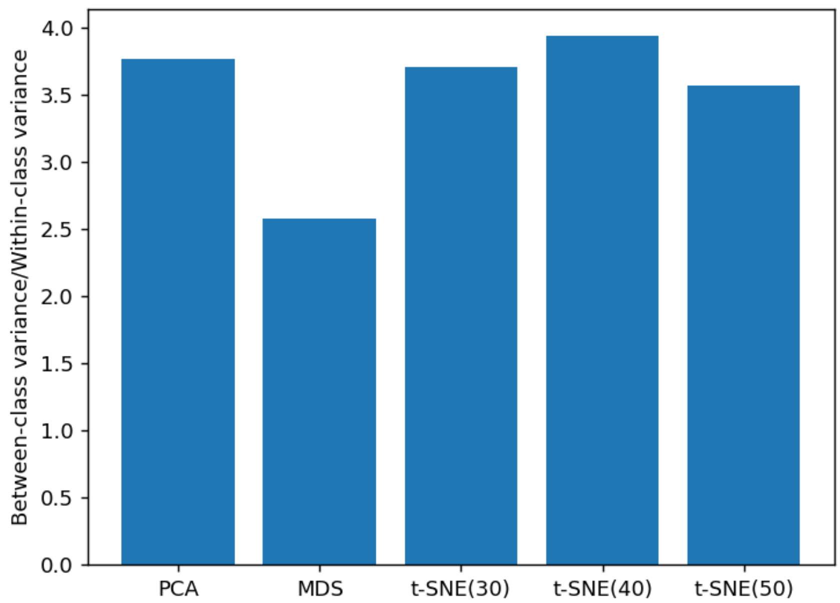

Figure 6: between-class variance / within-class variance

5 Conclusion

In this paper, we proposed a method for converting lecture videos into microcontents. The method vectorizes transcripts of videos using Doc2Vec and then divides videos based on the distance between these vectors. In addition, we compared visualization methods using principal component analysis, multidimensional scaling, and t-SNE, and found that t-SNE is suitable for the visualization. Future challenges include further improving the accuracy of video content division, evaluating the search interface, and developing functions such as advanced recommendation of microcontents.

Acknowledgement

This work was partially supported by JSPS KAKENHI Grant Numbers JP24K06300.

References

[1] T. Yang, B. Ren, B. Ma, M.A.Z. Khan, T. He, and S. Konomi, "Making Course Recommendation Explainable: A Knowledge Entity-Aware Model using Deep Learning," Proceedings of the 17th International Conference on Educational Data Mining, pp.667-672, 2024.

[2] M.A.Z. Khan, and A. Polyzou, "Session-based Course Recommendation Frameworks using Deep Learning," Proceedings of the 16th International Conference on Educational Data Mining, pp.269-277, 2023.

[3] W. Wang, E. Kumamoto, and C.Yin, "A page jump recommendation model and result interpretation based on structured annotation methods," Proceedings of the 17th International Conference on Educational Data Mining, pp.371-378, 2024.

[4] P. Zhu, C. Hauff, and J. Yang, "MOOC-Rec: Instructional Video Clip Recommendation for MOOC Forum Questions," Proceedings of the 15th International Conference on Educational Data Mining, pp.705-709, 2022.

[5] A. Leene, "Microcontent Is Everywhere," Proceedings of Microlearning Conference 2006, pp.20-40, 2006.

[6] M. Furukawa and Y. Yaginuma, "Visual Interface for Looking over Micro Contents," 10th International Congress on Advanced Applied Informatics (IIAI-AAI), pp.930-931, 2021.

[7] J. Lin, G. Sun, T. Cui et al., "From ideal to reality: segmentation, annotation, and recommendation, the vital trajectory of intelligent micro learning," World Wide Web 23, pp.1747–1767, 2020. https://doi.org/10.1007/s11280-019-00730-9

[8] M. Souza and S. Amaral, "Educational Microcontent for Mobile Learning Virtual Environments, " Creative Education, 5, pp.672-681, 2014.

[9] M. Furukawa and Y. Yaginuma, "A method for dividing lecture videos into microcontents, " IEICE Technical Report, ET2024-26, pp.1-4, 2024.10 (In Japanese)

[10] DOC2VEC model learned by Japanese WIKIPEDIA, https://yag-ays.github.io/project/pretrained_doc2vec_wikipedia/ (In Japanese)