# Automatic Question-Answer Alignment for Japanese Diverse Local Assembly Minutes

Yoshiki Higashiyama and Tomoyosi Akiba



September 29, 2024

# Automatic Question-Answer Alignment for Japanese Diverse Local Assembly Minutes

Yoshiki Higashiyama
*Toyohashi University of Technology*
Aichi, Japan
higashiyama.yoshiki.ta@tut.jp

Tomoyosi Akiba
*Toyohashi University of Technology*
Aichi, Japan
akiba@cs.tut.ac.jp

***Abstract*—In the question-and-answer sessions in Japanese local assembly minutes, various topics about local administration are discussed, from which residents can learn about the administrative policy. However, in the discussion in some councils, a single council member asks several questions in a batch, then a prefectural governor and persons in charge answer their corresponding questions one by one. This kind of argument structure is difficult for residents to read since the text of the question and that of the answer are separated from each other. This study aims to improve the readability of the minutes and make local politics more easily accessible to residents. In order to achieve that goal, this work proposes to transform them into "one question, one answer" format through two stage processes, text segmentation and Question and Answer alignment. We employ several robust segmentation methods for the segmentation so that the proposed method can be applied to various discussion styles of Japanese local assembly minutes. Our experimental evaluation showed that the proposed methods performed well on not only the specific minutes but also those other than the minutes used for extracting the training data of our methods.**

***Index Terms*—local assembly minutes, alignment, segmentation**

## I. Introduction

In the National Diet of Japan and local council meetings, various topics are discussed through question-and-answer sessions. What is said is recorded in the minutes, which can be viewed by anyone. Through these publicly available minutes, residents can grasp the content of the agenda and learn about measures and proposals. However, in some councils, a single council member asks several questions in a batch, then a prefectural governor and persons in charge answer their corresponding questions one by one. Figure 1 shows a typical example of such argument structure often observed in Japanese assembly minutes. We call this type of argument structure "collective questioning and collective answering".

The Japanese assembly minutes recorded in the "collective questioning and collective answering" format are difficult for residents to read because the text of the question and the text of the answer are separated from each other. In this work, we propose a method for transforming local assembly minutes in "collective questioning and collective answering" format into a "one question, one answer" format as shown in Figure 2 by dividing questions and answers into topic units and mapping questions and answers on the same topic. A similar problem has been set as an evaluation task in the QA Lab-PoliInfo task of the 2021-2022 evaluation-based workshop NTCIR16 as the QA Alignment task [1]. However, the QA Alignment task targets only the Tokyo Metropolitan Assembly Minutes, and it is not clear whether the methods developed for the task would be applicable to various local assembly minutes other than Tokyo. Therefore, this work proposes a method that is robust to various types of minutes.

Chapter 2 reviews related work on topic segmentation ans alignment of questions and answers. In Chapter 3, we describe our approach, segmentation and alignment. Chapter 4 presents experiments and results of segmentation and alignment. Chapter 5 summarizes key findings.

This study was supported by JSPS Grant-in-Aid for Scientific Research 23K11118.

## II. Related Work

### A. Topic Segmentation

Galley et al. [2] uses information about the occurrence of identical words in a sentence (lexical chains) to compute cosine similarity and detect regions of decreasing similarity, thereby segmenting the text. Lukasik et al. [3] proposed three Bidirectional Encoder Representations from Transformers(BERT)-based architectures: the first is to fine-tune a pre-trained BERT using a sequence of tokens in the text that comes before and after the position where the text is to be split and split the text with that model; the second is transforming each text into a sequence of tokens in BERT and passes it to Bidirectional Long Short-Term Memory (Bi-LSTM) for segmentation; the third is transforming each text into a sequence of tokens in BERT and passes it to BERT for segmentation, as in the second. Solbiati et al. [4] use pre-trained BERT to transform sentences into special tokens and then segment the text by computing cosine similarity. Ohsugi et al. [5] use regular expressions to capture the typical keyphrases that appear in the sentences before and after the split, and split the text accordingly if one of then is found by the regular expressions.

This study constructed a binary classifier that takes two sentences from a question or an answer and determines if there is a split between them. We employed pre-trained BERT for that classitication.

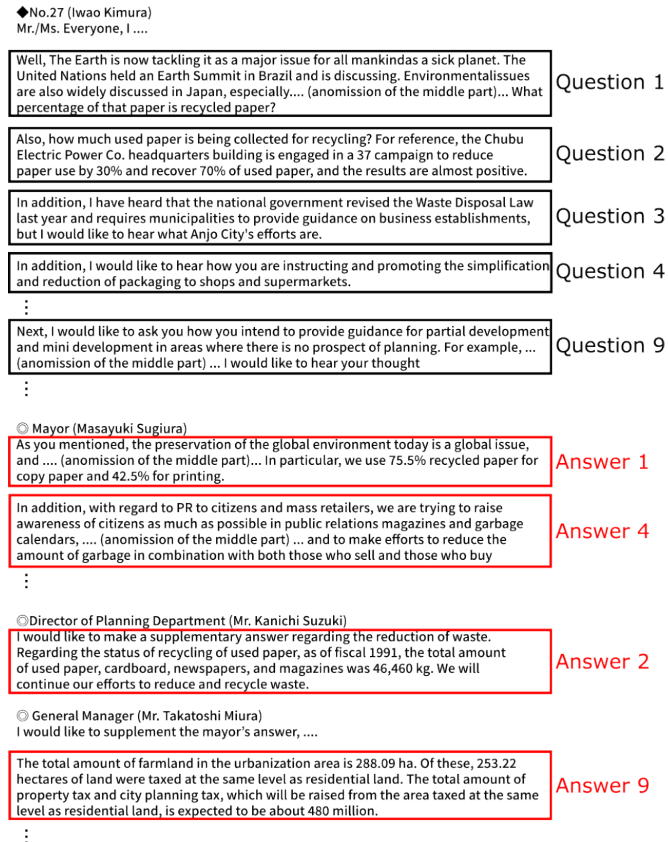


Fig. 1. typical example of the argument structure often observed in Japanese assembly minutes
The same numbers (Question1 and Answer1) are matched to each other.

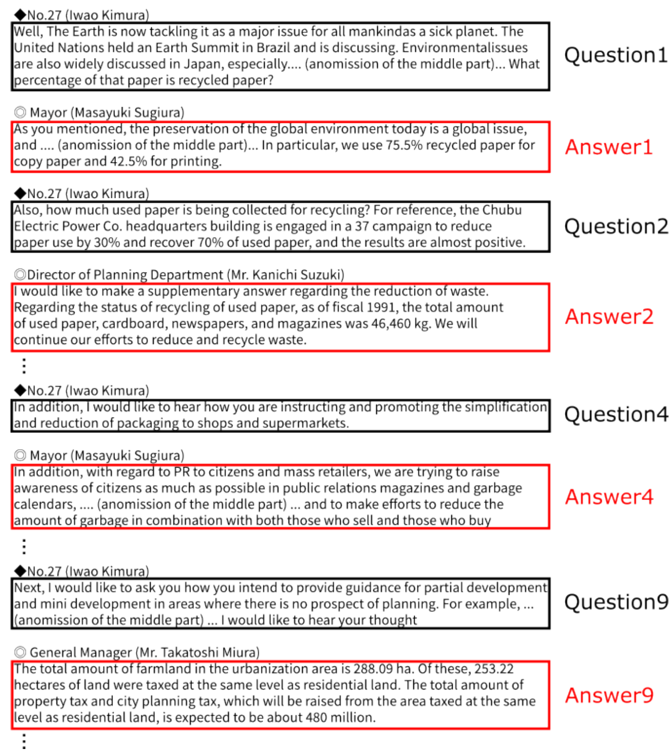


Fig. 2. "one question, one answer" format

### B. Alignment of Questions and Answers

Ohsugi et al. [5] transformed the text into an Okapi BM25 [7] word weight vector. Then, it solves a perfect matching problem that maximises the cosine similarity between the question and answer vectors. The problem was computed and aligned by applying the Hungarian algorithm [8]. Yamato et al. [9] transformed text into vectors using Sentence-BERT. Vectorisation is performed using a model fine-tuned with gold data from the QA alignment task. Alignment was then performed by calculating the cosine similarity between the question and answer vectors. This study uses the method of Ohsugi et al [5].

## III. Approach

The proposed method consists of two stages: segmentation (Section III-A) and alignment (Section III-B), as shown in Figure 1. Segmentation splits the text, and alignment maps the segmented questions to the answers. Previous studies [5] have used regular expressions for segmentation. This regular expression cannot be adapted to diverse Japanese local assembly minutes due to its specificity to the Tokyo Metropolitan Assembly minutes. Manipulating a new regular expression specitic to each of the minutes is expensive. We explored segmentation methods not just for one but robust for various local governments.

### A. Segmentation

Segmentation is performed on Japanese assembly minutes, to find topic breaks in the questions and answers and to split the text. Unsupervised and supervised methods are used.

*1) Unsupervised segmentation:* We employed two methods: one that uses lexical chaining (referred to as LCseg) [2] and one that uses pre-trained sentenceBERT (referred to as unsupervised BERT) [4].

A lexical chain is constructed to consist of all repetitions ranging from the first to the last appearance of the word in the document. The lexical chain score is computed by generating this lexical chain from the documents. The $score(R_i)$ of a chain $R_i$ of a word $t_i$ is expressed as follows, where $freq(t_i)$ is combining frequency of $t_i$, $L_i$ is the length of the chain, $L$ is the length of the whole document.

$$score(R_i) = freq(t_i) \cdot \log \frac{L}{L_i}$$

$$cosine_{LCseg}(A, B) = \frac{\sum_i w_{i,A} \cdot w_{i,B}}{\sqrt{\sum_i w_{i,A}^2 \sum_i w_{i,B}^2}}$$

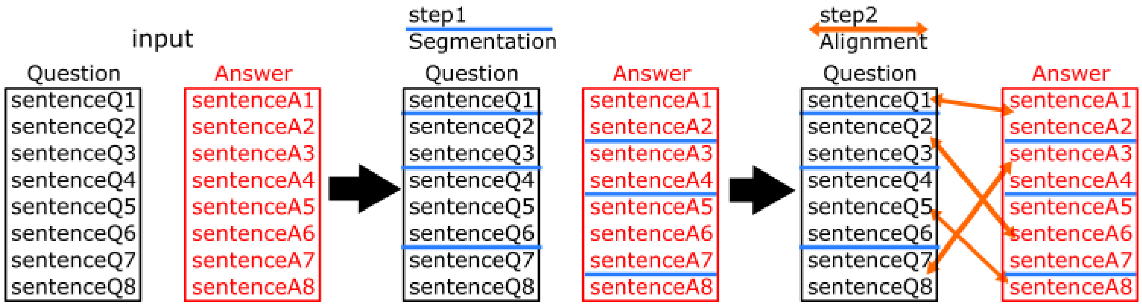


Fig. 3. Flow of the proposed approach

*where*

$$w_{i,\Gamma} = \begin{cases} score(R_i) & if\ R_i\ overlaps \quad \Gamma \in \{L, R\} \\ 0 & otherwise \end{cases}$$

LCseg computes lexical cohesion scores ($cosine_{LCseg(A,B)}$) using lexical chain scores $score(R_i)$ that overlap with the two analysis windows (A and B) of adjacent fixed window width k. The sentence is split if its score is less than the threshold.

On the other hand, unsupervised BERT methods firstly convert a text of a segment candidate into a vector by using sentenceBERT [6] by taking a maximum pooling of the hidden vectors of its last layer. The cosine similarity between two consecutive segment candidates is calculated to determine if the text should be split at that position by comparing with a pre-defined threshold. We employed three pre-trained sentenceBERT from (1) 'stsb-xlm-r-multilingual' [1] and (2) 'paraphrase-multilingual-mpnet-base-v2' [2], which is Japanese sentenceBERT, (3) 'sentence-bert-base-ja-mean-tokens' [3].

*2) Supervised segmentation:* This approach consists of a binary classifier that takes two consecutive sentences and classify them as 1 if there is a boundary between them and 0 if not.

The training dataset used was those used in the QA Alignment task [1]. The training data used in this task is from the Tokyo Metropolitan Assembly minutes, for which segmentation is provided, and covers six years from 2011 to 2016. In addition, data from the minutes of the Tokyo Metropolitan Assembly automatically segmented using regular expressions are also used as training data. This data is takes from the minutes from 2004 to 2011 and 2017 to 2019. The regular expression used is the one used in the previous study [5] and is shown in the table I. The validation data used in the same task [1] with the gold labels (correct labels for segmentation) are used as validation data. This is the data for the year 2020. In addition, we added to the validation data the minutes from the 2021 to 2023, which were automatically split using the regular expression of table I.Details of the dataset are given in TableII.

[1]https://huggingface.co/sentence-transformers/stsb-xlm-r-multilingual
[2]https://huggingface.co/sentence-transformers/paraphrase-multilingual-mpnet-base-v2
[3]https://huggingface.co/sonoisa/sentence-bert-base-ja-mean-tokens

TABLE I
REGULAR EXPRESSIONS FOR SEGMENTATION IN PREVIOUS STUDIES

<table>
<tr><th>Pattern</th><th>Regular expressions</th></tr>
<tr><td>Question</td><td>(お)?(伺い｜尋ね)(を)?(いた)?し?(させていただき)?(ます｜たい) ｜ (見解｜答弁｜所見｜課題｜認識｜考え｜説明) をお?(求め｜伺い｜聞かせ｜尋ね) ｜ (お)?(答え｜聞かせ)(て｜を)?ください｜ありがとうございました｜いかがですか｜どうですか｜ではありませんか｜るものです｜ (どのように｜どう)(考えて｜認識して｜取り組む) ｜のですか｜のでしょうか</td></tr>
<tr><td>Answer</td><td>(お)?答え (を)?(いた)?(｜し｜申し上げ) ます｜初めに、｜次 (いで｜に｜は)、｜まず、｜他方で、｜最後に、｜続きまして、｜について (です｜であります｜でございます) ｜の (お話｜お尋ね)(がございました｜でございます) ｜ (の｜に関する)(ご)?質問で (ございま)?す｜ (質問｜指摘｜言及｜お尋ね) が?ございました｜ (質問｜指摘) を?いただきました</td></tr>
</table>

TABLE II
TRAINING DATASET

| data | gold segmentation | period | number of segment |
|---|---|---|---|
| training data | yes | 2011-2016 | 9,492 |
| | no | 2004-2011, 2017-2019 | 14,708 |
| validation data | yes | 2020 | 2,109 |
| | no | 2021-2023 | 3,922 |

We trained two separate models; one for segmenting questions and the other for answers. We employed a pre-trained BERT and fine-tuned it by using our training data. As a pre-trained Japanese BERT model, we used 'bert-based-japanese-whole-word-masking'[1] published by the Natural Language Processing Laboratory, Tohoku University. This model is a BERT model pre-trained on the Japanese Wikipedia. They are also trained with whole-word-masking enabled, which masks consecutive tokens corresponding to a single word at a time, for Masked Language Modelling (MLM) to predict masked words. Two fine-tuning models were prepared, one for the question text and one for the answer text.

### B. Alignment

The alignment process maps segmented questions and answers one to one. First, the text of each segment is transformed to a vector. In the proposed approach, the text is transformed

[1]https://huggingface.co/tohoku-nlp/bert-base-japanese-whole-word-masking

into a word weight vector of Okapi BM25 [7], following previous research [5].

The BM25 assigns a weight to a word in a document by combining the term frequency (TF), the inverse document frequency (IDF) of the word, and the total number of words in the document (DL). The score for word $w$ in document $D$ is expressed as

$$score(D, w) = IDF(w)\frac{f(w, D)(k_i + 1)}{f(w, D) + k_i(1 - b + b \cdot \frac{|D|}{avgdl})}$$

The inverse document frequency (IDF) of a word is the inverse of the number of documents in which a word appears. The document set here is either the segments for questions or that for answers. Let N be the number of all documents and $n(q_i)$ be the number of documents containing $q_i$, IDF is expressed as

$$IDF(q_i) = \log \frac{N}{n(q_i)}$$

The cosine similarity between the Okapi BM25 vectors of questions and answers is calculated to find the mapping between segmented questions and answers.

Due to the nature of the 'one question, one answer' format, question-answer pairs are to be one-to-one. It can be viewed as a perfect matching problem, in which the sum of the cosine similarity between vectors is to be maximized. This problem can be solved by applying the Hungarian algorithm [8].

## IV. EXPERIMENT

The performance of the conventional and proposed approaches is compared using novel assembly minutes. For our evaluation, we selected one session of a plenary meeting of a Japanese city, Anjo, whose argument structure follows "collective questioning and collective answering" format. We manually annotated its assembly minutes with topic segmentations both on questions and answers and aligned them with each other so that question and answer segments with a same topic are connected. Questions and answers by Iwao Kimura from the second regular meeting of the Anjo City Council in 1992, No. 2, were covered. We also evaluated the performance of the compared methods on the Question and Answer aligned assembly minutes of Tokyo metropolitan provided from the PoliInfo-3 QA Alignment task.

### *A. Segmentation evaluation experiments*

*1) Method:* The evaluation metrics used for the segmentation are Pk [10], WindowDiff [11] and F-measure.

Pk evaluates whether two texts separated by a distance k belong in the same segments or not correctly. The smaller the value the better, as it represents the probability of error. Pk has the problem that FalsePositive is less penalized. An evaluation metrics called WindowDiff (WD), which solves this problem, is also used. WindowDiff calculates how many boundaries are contained between two texts separated by a distance of k and gives a score for the number of discrepancies.

TABLE III
SEGMENTATION EVALUATION USING THE TOKYO METROPOLITAN ASSEMBLY MINUTES

| Question | | | |
|---|---|---|---|
| Method | Pk ↓ | WD ↓ | F1 ↑ |
| Regular expressions | 0.06 | 0.07 | 0.94 |
| LCseg | 0.55 | 0.55 | 0.03 |
| Unsupervised BERT(1) | 0.56 | 0.56 | 0.09 |
| Unsupervised BERT(2) | 0.52 | 0.52 | 0.13 |
| Unsupervised BERT(3) | 0.50 | 0.50 | 0.18 |
| Supervised BERT | 0.33 | 0.34 | 0.56 |
| Answer | | | |
| Method | Precision | Recall | F1 |
| Regular expressions | 0.05 | 0.08 | 0.91 |
| LCseg | 0.32 | 0.35 | 0.44 |
| Unsupervised BERT(1) | 0.32 | 0.35 | 0.45 |
| Unsupervised BERT(2) | 0.32 | 0.35 | 0.45 |
| Unsupervised BERT(3) | 0.32 | 0.29 | 0.45 |
| Supervised BERT | 0.07 | 0.09 | 0.90 |

TABLE IV
SEGMENTATION EVALUATION USING THE ANJO CITY ASSEMBLY MINUTES

| Question | | | |
|---|---|---|---|
| Method | Pk ↓ | WD ↓ | F1 ↑ |
| Regular expressions | 0.18 | 0.22 | 0.55 |
| LCseg | 0.39 | 0.41 | 0.20 |
| Unsupervised BERT(1) | 0.41 | 0.41 | 0.09 |
| Unsupervised BERT(2) | 0.34 | 0.35 | 0.19 |
| Unsupervised BERT(3) | 0.43 | 0.45 | 0.16 |
| Supervised BERT | 0.16 | 0.20 | 0.69 |
| Answer | | | |
| Method | Pk ↓ | WD ↓ | F1 ↑ |
| Regular expressions | 0.36 | 0.37 | 0.38 |
| LCseg | 0.26 | 0.27 | 0.57 |
| Unsupervised BERT(1) | 0.29 | 0.29 | 0.49 |
| Unsupervised BERT(2) | 0.30 | 0.33 | 0.56 |
| Unsupervised BERT(3) | 0.28 | 0.30 | 0.56 |
| Supervised BERT | 0.31 | 0.39 | 0.48 |

We also employed F-measure as another evaluation metric. The F-measure is calculated by creating a confusion matrix predicted boundaries of the proposed approach and correct boundaries. From this matrix, precision, recall, and their harmonic mean, F-measure are computed.

*2) Result:* First, the results of the segmentation evaluation on the Tokyo Metropolitan Assembly minutes are shown in tableIII. The regular expressions performed well because it is optimised for the Tokyo Metropolitan Assembly minutes. In addition, Supervised BERT performed well because it is trained by the Tokyo Metropolitan Assembly minutes.

The results on the Anjo City Council minutes, are shown in TableIV. Looking at the result on the question sentences, the regular expression still performed moderately well. We think that is because the expressions in the question used for the progress in Anjo City is similar to those for Tokyo Metropolitan. However, looking at that on the answer sentences, the regular expression performed poorly. We think that is because the expressions in the answer for Anjo is quite different from Tokyo. On the other hand, all the proposed methods performed better on the answer for Anjo City. Among them, SupervisedBERT outperformed the regular expression both on the question and the answer, even though it is trained by using the data taken from the Tokyo Metropolitan Assembly Minutes. Those results show the robustness of the proposed segmentation methods.

### B. Alignment evaluation experiments

We also evaluated the QA alignment performance based on the results of the segmentations obtained by the compared methods. We employed Ohsugi's QA alignment method [5] for all compared segmentation methods.

*1) Method:* We employed F-measure of a given QA alignment result used in the Polinfo3 [1] as our evaluation metric. The calculation process of the metric is as follows. First of all, for each aligned question and answer segment $(Q_i, A_i)$, all pair of sentences $S_i = \{(q, a) : q \in Q_i, a \in A_i\}$ are extracted, then all the pairs are merged into a set $S = \bigcup_i S_i$. The extraction process is performed both on system's output and the reference QA alignment, so we have two sets of sentence pairs $S_{system}$ and $S_{reference}$. By comparing $S_{system}$ and $S_{reference}$, precision and recall are calculated. The final evaluation metric F-measure is the harmonic mean of them. Please refer to [1] for more details.

*2) Result:* The results of the automatic alignment between questions and answers are shown in the table V. From the table V, for the Tokyo Metropolitan Assembly, the value of the F-measure exceeds 0.8 when using regular expressions. However, for the Anjo City Assembly minutes, the use of regular expressions did not perform well. On the other hand, the proposed supervised segmentation method successfully improved the F-measure by 0.17, the recall by 0.09 and the precision by 0.3 compared to the regular expression.

The proposed Supervised BERT was still performed well on the Anjo City Assembly minutes, where regular expressions were not well adapted. However, even using the Supervised BERT, its recall is still low. The reason for the low recall is that the proposed alignment method assumes one-to-one pairs of questions and answers. The test data of the Anjo City show that there are more cases where two answers correspond to one question than in the Tokyo Metropolitan Assembly. The row indicated by "correct segment" in the table V is the result of the alignment by using the correctly segmented text. It shows the recall is still low. Therefore, we need to improve the alignment method so that it allows one-to-many alignments.

## V. CONCLUSIONS

In this study, we proposed new segmentation methods robust for diverse Assembly minutes. Among them, supervised BERT was implemented as a binary classifier that determines whether a segment boundary exists between two consecutive sentences. We have demonstrated the performance of our methods across novel assembly minutes. The proposed method (supervised BERT) improved the performance of automatic QA mapping for novel assembly minutes for which the method used in previous work did not work well.

TABLE V
ALIGNMENT EVALUATION USING THE ASSEMBLY MINUTES

| Tokyo Metropolitan Assembly minutes | | | |
|---|---|---|---|
| Method | Precision | Recall | F1 |
| Regular expressions | 0.84 | 0.87 | 0.86 |
| LCseg | 0.02 | 0.33 | 0.03 |
| Unsupervised BERT(1) | 0.04 | 0.28 | 0.07 |
| Unsupervised BERT(2) | 0.05 | 0.24 | 0.08 |
| Unsupervised BERT(3) | 0.06 | 0.25 | 0.09 |
| Supervised BERT | 0.42 | 0.54 | 0.47 |
| **Anjo City Assembly minutes** | | | |
| Method | Precision | Recall | F1 |
| Regular expressions | 0.30 | 0.32 | 0.31 |
| LCseg | 0.14 | 0.31 | 0.19 |
| Unsupervised BERT(1) | 0.27 | 0.43 | 0.33 |
| Unsupervised BERT(2) | 0.23 | 0.30 | 0.27 |
| Unsupervised BERT(3) | 0.22 | 0.33 | 0.26 |
| Supervised BERT | 0.60 | 0.41 | 0.48 |
| Correct segment | 0.89 | 0.55 | 0.68 |

## REFERENCES


[1] Yasutimo Kimura, Y. Uchida, M. Yoshioka et al.: Overview of the NTCIR-16 QA Lab-PoliInfo-3 Task, Proceedings of the 16th NTCIR Conference on Evaluation of Information Access Technologies(2022).

[2] M. Galley, K. R. McKeown, E. Fosler-Lussier, and H. Jing: Discourse Segmentation of Multi-Party Conversation, Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics(2003).

[3] M. Lukasik and B. Dadachev and G. Simões and K. Papineni: Text Segmentation by Cross Segment Attention, CoRR(2020).

[4] A. Solbiati, K. Heffernan, G. Damaskinos, S. Poddar, S. Modi and J. Calì: Unsupervised Topic Segmentation of Meetings with BERT Embeddings, CoRR(2021).

[5] Ryoto Ohsugi, Teruya Kawai, Yuki Gato, Tomoyosi Akiba and Shigeru Masuyama: AKBL at the NTCIR-16 QA Lab-PoliInfo-3 Task, Proceedings of the 16th NTCIR Conference on Evaluation of Information Access Technologies(2022).

[6] Nils Reimers and Iryna Gurevych: Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, CoRR(2019)

[7] Robertson, Stephen and Walker, S. and Jones, S. and Hancock-Beaulieu, M. M. and Gatford, M.: Okapi at TREC-3, Overview of the Third Text REtrieval Conference (TREC-3), pp.109–126(1995).

[8] H. W. Kuhn and B. Yaw: The hungarian method for the assignment problem, Naval Res. Logist. Quart(1995).

[9] H. Yamato, T. Fukunaga, M. Okada, N. Mori: Omuokdlb at the NTCIR-17 QA Lab-PoliInfo-4 Task, NII Institutional Repository(2023).

[10] Beeferman, Doug and Berger, A. and Lafferty, J.: Statistical Models for Text Segmentation, Machine Learning, Vol.34, No1-3, pp.177-210(1999)

[11] Pevzner, Lev and Hearst, Marti A.: A Critique and Improvement of an Evaluation Metric for Text Segmentation, Computational Linguistics, Vol.28, No.1, pp.19-35(2002)