Forst: A Challenge to the NTCIR-16 QA Lab-PoliInfo-3 Task

Naoki Igarashi
Yokohama National University
Japan
igarashi-naoki-xy@ynu.jp

Daiki Iwayama
Yokohama National University
Japan
iwayama-daiki-kc@ynu.jp

Hideyuki Shibuki
Besna Institute Inc.
Japan
shib@besna.institute

Tatsunori Mori
Yokohama National University
Japan
tmori@ynu.ac.jp

ABSTRACT

In this paper, we describe the development of a system for QA Alignment and a system for Fact Verification. We submitted 11 results for the QA Alignment, 6 results including 4 late submissions for the Fact Verification. As a result, an F-measure of .7753 for the QA Alignment and an F-measure of .8563 for the Fact Verification were obtained.

KEYWORDS

QA Alignment, Fact Verification

TEAM NAME

Forst

SUBTASKS

QA Alignment(Japanese), Fact Verification(Japanese)

1 INTRODUCTION

We tackled the QA Alignment and Fact Verification subtasks in the NTCIR-16 QA Lab-PoliInfo-3 task [2]. In this paper, we describe the development of a system for QA Alignment and a system for Fact Verification. Section 2 describes the QA Alignment system and results. Sections 3 describes the Fact Verification systems and results. Finally, Section 4 provides some concluding remarks.

2 QA ALIGNMENT

2.1 Approach

In the QA Alignment task, we need to determine the question of the questioner and its corresponding answer from the minutes, sentence by sentence. However, one question passage of a questioner may contain more than one question, and one answer passage of a respondent may also contain more than one answer. Therefore, our system performs this task in two stages, the "segmentation" stage and the "matching" stage as shown in Figure 1.

In the segmentation stage, the statements of the questioner and the answerer, which contain multiple questions and answers, are divided into individual questions and answers. The segmentation is done by a rule-based approach using cue expressions such as "伺います" to find the division points between questions and between answers.

In the matching stage, we used similarity to map the individual questions and answers. The similarity is based on the number of overlaps in the original form of the words and the word embeddings.

2.2 Related Work

Kimura et al. [1] state that summarizing the minutes is to summarize the relationship between questions and answers. Since the minutes are in the form of a batch question and a batch answer, they proposed to divide the statements by using expressions such as "伺います" and "まず" as clues. In this system, we divide the statements by using expressions.

In matching, we use word2vec by Thomas et al.[6] word2vec is a method to convert words into numerical vectors. By vectorizing, the similarity between words can be calculated. In this system, important words are identified by tf-idf, and the word vectors of those words are used to create vectors that represent the sentence after segmentation. Another way to represent sentence features is Doc2Vec by Le et al.[5]. Le et al. proposed a method to generate fixed-length feature vectors from variable-length text such as sentences, paragraphs, and documents. The generated feature vectors can be used to find the similarity between different documents. In this system, multiple variable-length texts are obtained by segmentation, and the similarity of these texts is calculated for mapping, but Le's method is not used to generate feature vectors.

2.3 Method

2.3.1 Preparation. Since the minutes contain data such as moderator's Statement except for questions and answers, we divide it into a dictionary format for each value of "QuestionerID" and "QorA".

2.3.2 Segmentation of Question. Segmentation is done by giving a common SegmentID to the sentences that compose a single questions. The following process is performed for each "QuestionerID".

(1) Set the current "SegmentID" to 1. Repeat the processes of (2),(3) for all sentences.
(2) When a sentence contains a clue expression (Q) shown in Table 1.
  (a) When SegmentID does not change in the previous sentence, and it is not the first sentence. Set the "SegmentID" of sentence to current "SegmentID" and add 1 to the value of "SegmentID".
  (b) When the "SegmentID" changes in the previous sentence and contains 「^あわせて」. Set the "SegmentID" of sentence to current "SegmentID" and add 1 to the value of "SegmentID".

(c) When none of the above. Set the "SegmentID" of sentence to current "SegmentID".

(3) When a sentence does not contain a clue expression. Set the "SegmentID" of sentence to current "SegmentID"

Table 1: Clue expressions(Q) represented by regular expressions

伺い [^、]*ます。?	お?尋ね [^、]*します
お答えください。	(見解|所見|答弁) を求め [^、]*ます
お?聞かせて?	(いかがで|どうで)(しょうか|すか)。
質問に移ります。	どう認識して (い) ますか
再質問 (いた) します	

2.3.3 Segmentation of Answer. Segmentation is done by giving a common "SegmentID" as the process of the question. The following process is performed for each "QuestionerID".

(1) Set the current "SegmentID" to 1. Repeat the processes of (2),(3),(4),(5) for all sentences.
(2) When the speaker changes.
  (a) When a sentence contains a clue expression (Z) shown in Table 2. Set the "SegmentID" of sentence to 0.
  (b) When none of the above. Add 1 to the value of "SegmentID". Set the "SegmentID" of sentence to current "SegmentID".
(3) When a sentence contains a clue expression (A) shown in Table 3. Add 1 to the value of "SegmentID". Set the "SegmentID" of sentence to current "SegmentID".
(4) When a sentence contains a clue expression (L) shown in Table 4. Set the "SegmentID" of sentence to 0.
(5) When none of the above. Set the "SegmentID" of sentence to current "SegmentID" shown in Figure 5.

Table 2: Clue expressions(Z) represented by regex

(代表|一般) 質問にお答え?	(つ|点) のご?質問
点についてお答え	

Table 3: Clue expressions(A) represented by regex

^最初に	お答えを?(いたし|し) ます
^次に	尋ね (が|で)(すが|あり|ござ)
^次いで	^終わりに
^まず	に?ついてで?(す|あります|ございます)(が|けれど)?。
^初めに	ご?質問 (で|が)(ござい|あり) ま (す|し)

Table 4: Clue expressions(L) represented by regex

他の質問に (ついて|つきまして) は

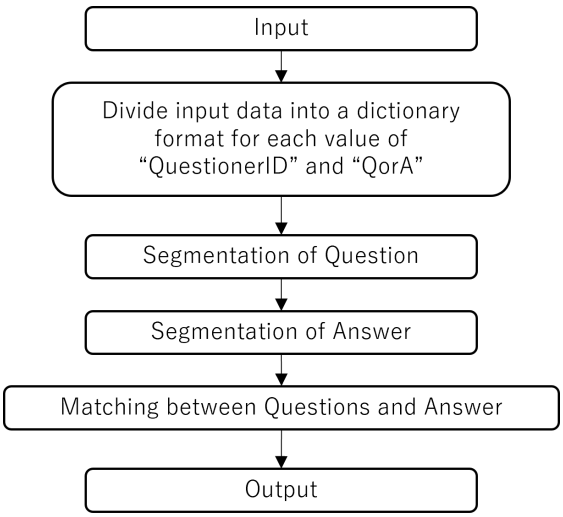

Figure 1: QA Alignment Pipeline

2.3.4 Matching between questions and answers. In this stage, the corresponding pairs of segmented questions and answers are determined based on their similarity. To calculate the similarity, we used the number of overlapping base form of word and the word embeddings. For details, see Section 2.3.5. The following process is performed for each "QuestionerID".

(1) Caluculate similarity between all questions and answers for which pairs have not been found.
(2) The pair with the highest similarity is determined to be the corresponding pair and given the same "QAID".
(3) Do (1),(2) until all pairs are decided by either question or answer.

2.3.5 Calculation of the similarity.

(A) base form of word We do morphological analysis of the input data to obtain the base form of the included words. The top n words in the number of occurrences are used as stop words. We use MeCab as a morphological analyzer and mecab-ipadic-NEologd as a dictionary. The base form of the words in each question and answer was obtained in the same way, and the stop words are removed. The number of overlapping words in each question-answer pair is used as the similarity.

(B) Word embeddings We create a vector that represents the characteristics of each question and answer. We calculate the tf-idf value of all words in each question/answer. We get the word embeddings of the top 30 words in the tf-idf value. To obtain the word vectors, we use word2vec, which has been trained on Japanese Wikipedia. The tf-idf weighted average of the obtained word vectors is used as the feature vector for each question and answer. The cosine similarity of the obtained feature vectors is used as the similarity of the pair.

$$feature\_vector = \frac{\sum_{n=1}^{30} tfidf_n \cdot \boldsymbol{W}_n}{\sum_{n=1}^{30} tfidf_n}$$

(C) A+B Rank the similarity of the pairs obtained by A and B. The rank is used as score. We regard the pair with the lowest total score of A and B as the highest similarity.

2.4 Result

Forst (ID197) ⋯ using the method A to determine the similarity. stop word number is 300.

Forst (ID261) ⋯ using the method C to determine the similarity. Stop word number is 200. Use the top 30 words of the tf-idf value.

Forst (ID262) ⋯ using the method C to determine the similarity. Stop word number is 100. Use the top 30 words of the tf-idf value.

Table 5: The results of QA Alignment

ID	F-measure	Precision	Recall
197	0.7746	0.7854	0.7716
261	0.7703	0.7615	0.7837
262	0.7699	0.7594	0.7852

2.5 Discussion

In the segmentation stage, segmentation was done using cue expressions. In terms of the number of segments, the segmentation was relatively correct, especially for answers.On the other hand, the accuracy of the segmentation of questions varied greatly depending on the characteristics of the speaker's way of speaking. In the figures2.3 the horizontal axis corresponds to the number of segmentations obtained by this method and the vertical axis corresponds to the correct number of segmentations. The straight line in the figure corresponds to y=x, and the further away from this line, the more different the number of segmentations. In the future, we would like to conduct segmentation that considers not only cue expressions but also topic transitions.

In the matching stage, when the main topic is common in a group of questions by the same questioner, the main topic is often common in the corresponding answers. In this case, there were cases where the wrong pair of questions and answers were selected. The use of subtopics in determining the degree of similarity may help to identify minor differences. As a pairing decision method, we used the greedy method, which selects the pair with the greatest similarity at the time. In the future, we would like to use a method that considers the optimization of the entire pairs.

3 FACT VERIFICATION

3.1 Approach

The Fact Verification task requires determining whether the content of a summary sentence given as evaluation data is present in the minutes. We examine the number of matching nouns between sentences to determine whether they corresponded. We used a four-step rule-based approach, including sentence-to-sentence comparison, to determine the range of correspondences. The first step is to narrow the minutes by date information. The second step is to select the candidate sentences by speaker information from the narrowed minutes. The third step is to find one representative sentence, which is most likely to match the summary sentence being evaluated, by word similarity. The fourth step is to determine the precise matching range of the minutes by using the representative sentence and cue expression specific to the minutes.

In the comparison of summary sentences with the minutes, we introduce a fixed length sliding window for the minutes. According to the preliminary experiment, where the window size is varied from one to five sentence length, we obtained the best correspondence results when the window size is two sentences length. Therefore, we adopt the setting as the baseline.

The results obtained with the baseline show, that some of the summary sentences that were incorrectly judges to be matched with parts of the minutes that have different topics or wrong pairs

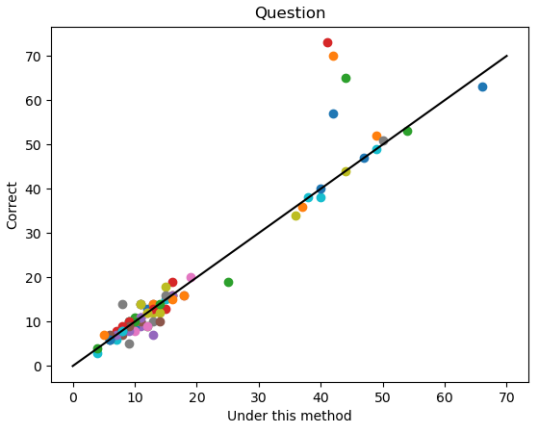

Figure 2: Compare Number of Segment to Correct

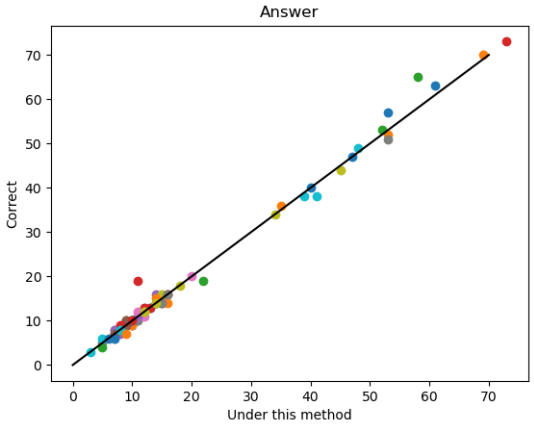

Figure 3: Compare Number of Segment to Correct

of questions and answers. From the observation, by using "RelatedUtteranceSummary", which is a summary of the statements of the questions (or answers) corresponding to the summary sentences, the relationship between the questions and the answers is taken into account.

Although, the best results were obtained when the window size of two sentence length was adopted, there were also results for three to five sentences that were judges to be true or false, unlike the results for two sentences. However, there were some case where the more than two sentence length of window size is suitable. In addition, the formal run method does not take into account the importance of the words, so if there are many words that are used generically in the minutes, they may be matched to the wrong range. For those reasons, in late submission, we tried a method to determine the similarity between sentences by cosine similarity of the tf-idf vector, and to consider the influence of context of two or more surrounding sentences by introducing a weighted function to smooth the similarity according to surroundings sentences.

3.2 Related Work

Kimura et al.[1] pointed out that it is useful to use cue expressions specific to the division of utterance and important parts of the minutes. For example, the sentence-initial expression "次に" is used to indicate a transition in topic, and the sentence-final expression "伺います" is observed when the speaker asks a question. Kazuki et al.[7] and Jiawei et al.[8], who participated in the Segmentation Task of QA Lab-PoliInfo[3], actually used this cue expressions for utterance segmentation and showed that the results were as good as those obtained by machine learning. We also use this cue expressions.

In addition, Kazuki et al.[7], in their study of mapping between the minutes and summary sentences, calculates the cosine similarity of the tf-idf vector between the pre-segmented ranges and summary sentences, and output the utterance range with the maximum similarity as the corresponding range of the summary sentence. However, this task needs to determine whether the corresponding range is truly the basis of the summary sentence, or not. The word similarity between multiple sentences and a single sentence becomes lower when the number of words differs greatly. In other words, there is not likely to be a large difference in the similarity values between the summary sentences that should be matched and those that are not. We consider that the similarity between the range of utterances and the summary sentences is not appropriate as a criterion to determine whether the range of correspondence is truly the basis for the summary sentences. Therefore, we used the similarity between a sentence in the minutes and the summary sentence as a criterion for correspondence.

As for the correspondence between questions and answers, Kimura et al.[1] found that the "batch question batch answer" method is used. In other words, a group of questions and the corresponding group of answers are listed in succession alternately in the minutes. If the content of the "RelatedUtteranceSummary" is included in the range of answers (questions) adjacent to the question (or answer) associated with the "Utterance", it is clear that the mapping is in the right place.

3.3 Method

Figure 4 show the overview of the proposed method.

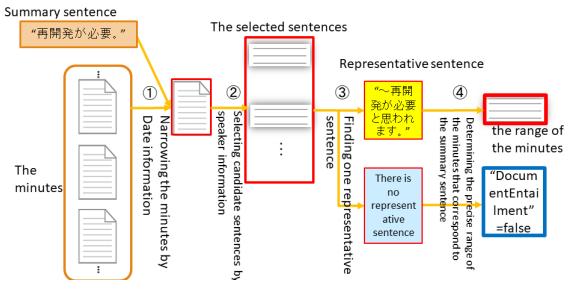

Figure 4: Overview of the proposed method

3.3.1 Preparation. We perform morphological analysis using MeCab[4] on the text and summary sentences of the minutes to extract nouns, adjectives, and verbs. In some cases, the number of words is extremely small in the morphological analysis of summary sentences (e.g., "新たな支援は考えていない。", "知事の所見は。"). When there are less than four words or two nouns in a summary sentence, we also perform morphological analysis on the "ContextWord", which show the main topic of the summary sentence, in order to obtain some supplement information for the summary sentence.

"UtteranceSummary", which is the content of the summary sentence, may consist of multiple sentences. In order to judge the similarity for each sentence of the summary, the summary is divided into sentences in advance.

The speaker information in the evaluation data may be specified by the position name instead of the speaker's name. One position name may be shared by different persons depending on the date of the meeting. In order to select the candidate sentences using speaker information, it is necessary to map speaker's name to their position. In the minutes, speaker's name is followed by "君". Therefore, we can find the utterance of the proceedings narrowing the minutes by date information and looking for the utterance that includes the position and "君" in the minutes (For example, if the evaluation data is "Utterance" : "知事" , "Date" : "23-6-17", we can find "Line" : 76, "Utterance" : "知事石原慎太郎君" in the minutes). By extracting the speaker's name between the position and "君", we can map the position to the speaker's name. The preparation stage creates a dictionary containing the date information of the evaluation data, the position , and the speaker's name.

3.3.2 Narrowing the minutes by date information. The "Date" in the evaluation data and the "Year", "Month", and "Day" in the meeting minutes data are used to narrow the minutes by excluding statements which are irrelevant to the summary sentence in terms of time constraint.

3.3.3 Selecting candidate sentences by speaker information. From the meeting minutes narrowed by the stage in Section 3.3.2, we select candidate sentences of the minutes that may correspond to the

summary sentence using the information of "Speaker" in the both of the evaluation data and the meeting minutes data. As described in Section 3.3.1, "Speaker" in the evaluation data may contain only names of positions without speaker's names. For this reason, we refer to the dictionary generated in preparation stage to convert the position in the evaluation data into the speaker's names.

3.3.4 Finding one representative sentence. For each candidate utterances in the minutes that is selectes in the stage of Section 3.3.3, the similarity to the summary sentence is calclated in turn. We used the the words in sentences to calclate the similarity in four viewpoint.

(1) Number of shared nouns between sentences
The number of nouns that are shared between the two adjacent selected sentences and the summary sentence is counted. If the number is more than half of the nouns in the summary sentence, the selected sentences are considered as one of candidates of representative sentences. The candidate with the largest number become the representative sentences. If there are multiple candidates with the largest number, the candidate that shared more proper nouns with the summary data is chosen as representative sentences. In representative sentences, one sentence that shares more nouns with the summary sentence is considered a representative sentence. If there is no candidate sentences matches, the summary sentence is assumed to be an imaginary sentence, which does not have corresponding sentence in the minutes. In this case, "DocumentEntailment" is set to false.

(2) Relationship between questions and answers
The output data obtained using the method using the method described in Section 3.3.4(1) is used as input data for a new evaluation, and the "RelatedUtteranceSummary" included in the evaluation data is used to check whether the matching range is valid or not. If the "UtteranceType" of the evaluation data is "question", scan the utterance after the matching range in the minutes. And if it is "answer", scan the utterance before the matching range in the minutes. In this case, the scanning sentence and the summary sentence are judged whether or not the representative sentence can be obtained by matching the words as in Section 3.3.4(1). If a representative sentence is obtained, it is judged that the correspondence has been made to the correct place.
If no representative sentence is obtained, it is judged that the correspondence has been made to the wrong part or to an imaginary summary sentence, and "DocumentEntailment" is set to false.

(3) Correspondence using the tf-idf vectors
We use the cosine similarity of the tf-idf vectors of nouns, adjectives and verbs in the selected sentences and the summary sentence to find the representative sentence.the sentence with the largest similarity is considered a representative sentence. If the similarity of the representative sentence does not exceed the threshold value, the summary sentence is assumed to be a fictitious sentence with no corresponding sentence, and "DocumentEntailment" is set to False.

(4) Smoothing similarity by taking account of similarity of surrounding sentences using a window function
The cosine similarity in (3) is smoothed by,taking account of the surrounding sentences. The Hamming window function is used for the smoothing. The Hamming window function for the l-th sentence is expressed by the following equation.

$$h_l(i) = 0.54 - 0.46 \cos 2\pi \frac{i-l}{W} (|i-l| < \frac{W}{2}) \quad (1)$$

Since the weighting of the Hamming window function becomes smaller from the center of the window to the outside, we can express the difference of similarity between the surrounding sentences while emphasizing the selected sentences. By taking account of the similarity of all of i-th sentences i in the specified window of the width W, the final similarity for the l-th sentence is the sum of the product of the cosine similarity $sim(i)$ with the summary sentence and the Hamming window function $h_l(i)$,as the following formula.

$$sum\_sim(l) = \sum_{i=l-\frac{W}{2}}^{l+\frac{W}{2}} h_l(i) \cdot sim(i) \quad (2)$$

In our experiment, the window width W is set to be 5. The representative sentence is obtained by using the smoothed similarity as described in (3)

3.3.5 Determining the precise range of the minutes that correspond to the summary sentence by using cue expressions specific to the minutes. The representative sentence obtained in Section 3.3.4, can be regarded as the core part of the minute that corresponds to the summary sentences. The range of the minute that corresponds to the summary sentence may be wider than the representative sentence itself. In order to determine the range, we use "Opening cue expressions" and "Ending cue expressions", which may appear at the beginning of one statement and the end of it, respectively.
From the representative sentence, the sentences are examined backward if one of "Opening cue expressions" is included. The first sentence found is regarded as the beginning of the minutes that corresponds to the summary sentence. As the same way, "Ending cue expressions" are used for finding the end of the minutes that corresponds to the summary sentence. Table6 shows the clue expressions represented by regular expressions. Figure5 shows an example of how to extract the correspondence range when "地域防災計画の修正にどう取り組むか" is given as the summary sentence and "都は、首都直下地震への備えを固めるため、地域防災計画の修正にそのように取り組むのか、見解を伺います。" as the representative sentence.

3.4 Result

Table7 shows the results of our method.
Two results were submitted as formal run, and the method described in Section 3.3.4 (1) was used for submission ID 257, and the method described in Section 3.3.4 (2) was used for submission ID 292. We also submitted four results as late submission. Post ID 338 and Post ID 340 used the method described in Section 3.3.4 (3), with similarity thresholds of 0.50 and 0.40, respectively. Post ID

Table 6: the clue expressions represented by regular expressions

Opening cue expressions	^まず|^最初に|^初めに|^次に|^次いで| ^最後に|^終わりに| ^[一二三四五六七八九十]+点目| ^[^、]+についてで (す|あります|ございます)(が|けれど)| ^終わり (ま|で) す。|^以上で| ^ありがとうございま| 他の質問に (ついて|つきまして) は
Endng cue expressions	’伺い [^、]*ます。|お尋ね [^、]*します|お答えください。| (見解|所見|答弁|対策) を求め [^、]*ます。| (いかがで|どうで)(しょうか|すか)。| .+質問を (終わります|終了します)

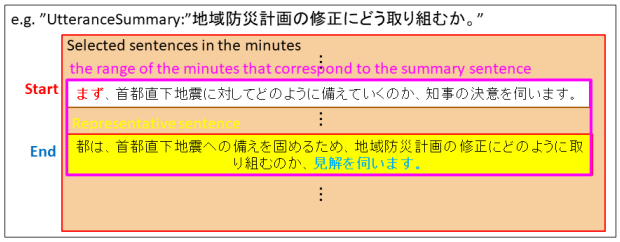

Figure 5: an example of how to extract the correspondence range

340 and Post ID 341 use the method described in Section 3.3.4 (4), and the similarity threshold is set to 0.50 and 0.40, respectively.

Table 7: The results of Fact Verification

ID:method	F-measure	Precision	Recall
257:Shared noun	0.8040	0.8113	0.8110
292:Shared noun + QandA	0.8389	0.8466	0.8451
338:tf-idf similarity(th=0.50)	0.6857	0.6864	0.6925
339:tf-idf similarity + smoothing(th=0.50)	0.7980	0.7989	0.8065
340:tf-idf similarity(th=0.40)	0.7970	0.7964	0.8058
341:tf-idf similarity + smoothing(th=0.40)	0.8563	0.8591	0.8642

3.5 Discussion

As can be seen from the results of post ID 257, relatively good results were obtained even the number of shared nouns is only taken account of. This is because many of the nouns used in the minutes are law names or event names, for which there are no alternative expressions. Therefore, the identical expressions often appear in both the summary and the minutes, and finding the representative sentence by the number of shared nouns works well.

The results of the method considering the relationship between questions and answers slightly increased by about 0.034 from the method without considering the relationship. The data for which the correspondence was improved by considering the relationship between questions and answers is assumed to be because relatively many sentences in the meeting minutes were judged as candidate sentences for correspondence due to the small amount of information in the summary sentences in the first place.

In addition, the results of late submission show that it is better to consider the surrounding sentences rather than the similarity between the summary sentence and a single sentence in the meeting minutes. This is because most of the summaries used in this task were summaries of a relatively small range of sentences (2 to 5 sentences) in the minutes, so the difference in similarity was more obvious than the similarity of a single sentence.

In this method, we did not consider the semantic similarity of words, so it would be difficult to find the corresponding part for secondary information, which are not summarized sentences like sentences in newspaper articles because of mismatch of surface expressions for same objects and events. Therefore, it is necessary to consider the semantic similarity of words in our future work.

4 CONCLUSION

In this paper, we described the development of a system for QA Alignment and a system for Fact Verification.As a result, an F-measure of 0.7753 for the QA Alignment and an F-measure of 0.8563 for the Fact Verification were obtained.

REFERENCES

[1] Yasutomo Kimura, Satoshi Sekine, and Kentaro Inui. 2018. [Towards Summarizing Local Council Proceedings]tihougikaikaigiroku no youyaku ni mukete (in Japanese). *Proceedings of the Twenty-fourth Annual Meeting of the Association for Natural Language Processing (NLP2018)* (2018), 596–599.

[2] Yasutomo Kimura, Hideyuki Shibuki, Hokuto Ototake, Yuzu Uchida, Keiichi Takamaru, Madoka Ishioroshi, Masaharu Yoshioka, Tomoyoshi Akiba, Yasuhiro Ogawa, Minoru Sasaki, Kenichi Yokote, Kazuma Kadowaki, Tatsunori Mori, Kenji Araki, Teruko Mitamura, and Satoshi Sekine. 2022. Overview of the NTCIR-16 QA Lab-PoliInfo-3 Task. *Proceedings of The 16th NTCIR Conference* (6 2022).

[3] Yasutomo Kimura, Hideyuki Shibuki, Hokuyo Ototake, Yuzu Uchida, Keiichi Takamaru, Kotaro Sakamoto, Madoka Ishioroshi, Teruko Mitamura, Noriko Kando, Tatsunori Mori, Harumichi Yuasa, Satoshi Sekine, and Kentaro Inui. 2019. Overview of the NTCIR-14 QA Lab-PoliInfo task. *Proceedings of the 14th NTCIR Conference on Evaluation of Information Access Technologies* (2019), 121–140.

[4] Taku Kudo. 2007. MeCab : Yet Another Part-of-Speech and Morphological Analyzer. (2007). https://taku910.github.io/mecab/

[5] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*. PMLR, 1188–1196.

[6] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems* 26 (2013).

[7] Kazuki Terazawa, Daiki Shirato, Tomoyodhi Akiba, and Shigeru Masuyama. 2019. AKBL at NTCIR-14 QA Lab-PoliInfo Task. *Proceedings of the 14th NTCIR Conference on Evaluation of Information Access Technologies* (2019), 190–197.

[8] Jiawei Yong, Shintaro Kawamura, Katsumi Kanasaki, Shoichi Naitoh, and Kiyohiko Shinomiya. 2019. RICT at the NTCIR-14 QALab-PoliInfo Task. *Proceedings of the 14th NTCIR Conference on Evaluation of Information Access Technologies* (2019), 141–158.