NTCIR 17 Conference: Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies, December 12-15, 2023, Tokyo, Japan
DOI: https://doi.org/10.20736/0002001304

AKBL at the NTCIR-17 QA Lab-PoliInfo-4 Task

Tasuku Shin
Toyohashi University of Technology
Japan
shin.tasuku.gm@tut.jp

Haruki Ishikawa
Toyohashi University of Technology
Japan
ishikawa.haruki.yu@tut.jp

Yuki Gato
Toyohashi University of Technology
Japan
gato.yuki.am@tut.jp

Eiji Kuramoto
Toyohashi University of Technology
Japan
kuramoto.eiji.vp@tut.jp

Tomoyoshi Akiba
Toyohashi University of Technology
Japan
akiba@cs.tut.ac.jp

ABSTRACT

AKBL team participated in the Question Answering-2, Answer Verification, Stance Classification-2, and Minutes-to-Budget Linking subtasks. For the Question Answering-2 subtask, our system extracts relevant transcripts from question metadata and summarizes them using a T5 model pre-trained in Japanese. For the Answer Verification subtask, our method first generates the pseudo-fake data automatically by round-trip translation, and then fine-tunes the pre-trained BERT with the training data and pseudo-fake data. For the Stance Classification-2 subtask, our best system is a binary classifier using RoBERTa. For the Minutes-to-Budget Linking subtask, it was realized using a ranking method based on Okapi BM25.

KEYWORDS

round-trip translation, Okapi BM25, pre-trained language model, fake data generation

TEAM NAME

AKBL

SUBTASKS

Question Answering-2 (Japanese)
Answer Verification (Japanese)
Stance Classification-2 (Japanese)
Minutes-to-Budget Linking (Japanese)

1 QUESTION ANSWERING-2

1.1 Method

The flow of the system created for the QA-2 task is shown in Figure 1. The following is a rough flow of the system.

- **step 1:** Extract the part of the minutes of the Tokyo Metropolitan Assembly that is relevant to a given question.
- **step 2:** Create a single query that separates the question and the minutes by tokens.
- **step 3:** Put the query into the summarization model and generate summary sentences.

The detailed description of each step is provided as follows.

In step1, from the dataset distributed at NTCIR-17 QA Lab-PoliInfo-4, we identified its relevant part (the excerpt) by using the date, the questioner, and the respondent of the given question as clues. In step 2, the question and the excerpt are combined to make a single query. Figure 2 shows the format used to make the query string. By using the format, we hope that it would help the summarization model understand the intent of the question and make a summary according to it. For step 3, we employed pre-trained language model, T5, trained with Japanese Web text. It was fine-tuned by using the training data provided by the PoliInfo-4 task organizers, where the input and output were the aforementioned query and the golden summary, respectively.

1.2 Result

Table 1 shows the results of the experiment conducted in this study.

Table 1: Results of Question Answering-2 subtask

	F-score
DryRun	0.0407
FormalRun	0.1162
Baseline.	0.2736

1.3 Discussion

The dry run produced a very low score, but the subsequent formal run was able to improve the score somewhat. However, it is clear that this result is far from the baseline score, so various issues remain to be addressed, such as examining the method for extracting multiple minutes and the format for inputting the data into the summarization model.

2 ANSWER VERIFICATION

2.1 Overview

For the Answer Verification task, we created a fact classifier by fine-tuning the pre-trained BERT with textual entailment evaluation data. The training data for the textual entailment recognition was PoliInfo-4 dry-run training data and additional pseudo-fake data mechanically created using round-trip translation based on the PoliInfo-4 dry-run training data.

2.2 Method

To automatically generate a pseudo-fake claim $\hat{\mathbf{x}}$ from a given true claim $\mathbf{x}$, two methods are proposed in [1]: *Direct Manipulation* and *Round-Trip Manipulation*. In both methods, basic operations are applied to a given text (either $\mathbf{x}$ for *Direct Manipulation* or the translation $\mathbf{y}$ of $\mathbf{x}$ in another language for *Round-Trip Manipulation*) to

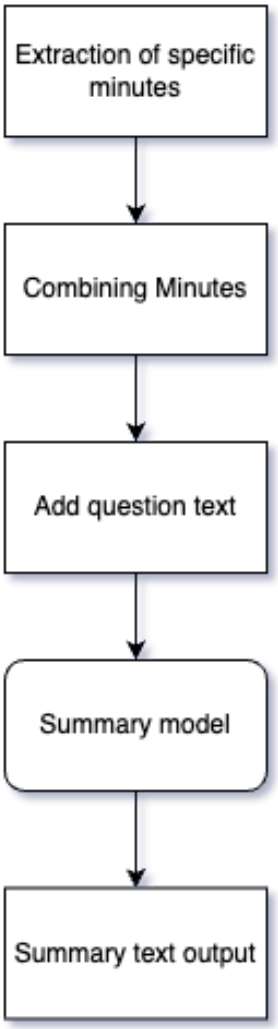

Figure 1: Flowchart of this system

question: Actual question text.
context:Combined Minutes

Figure 2: Specific examples of query minutes

change the polarity. For the Answer Verification task, we used the *Round-Trip Manipulation* method.

2.2.1 Basic Operations for Changing Polarity. We introduced three basic text operations to change the polarity of a given claim as follows.

- Insertion and deletion of negation (NEG)
  "A loves B" → "A does not love B"
- Conversion to antonyms (ANT)
  "A loves B" → "A hates B"
- Subject-Object Exchange (SOE)
  "A loves B" → "B loves A"

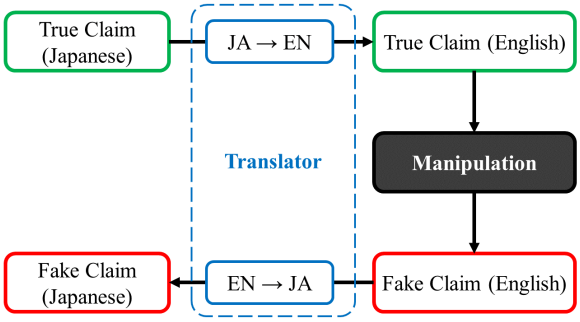

Figure 3: The schematic diagram of Round-Trip Manipulation

2.2.2 Round-Trip Manipulation. Instead of directly manipulating sentences, we employ round-trip translation for manipulation. We call this method Round-Trip Manipulation (hereinafter referred to as RTM). The process flow of RTM is shown in Figure 3.

The RTM first translates a claim into another language by employing an existing MT system. In this task, we chose to translate the target claims written in Japanese into English. Then, one or some basic operations are applied to the translated English sentence. Finally, the revised English sentence is translated back into Japanese.

The reason why we employ machine translation is that, even though the operations on text might introduce some artifacts, we hope the back-translation can alleviate them to produce natural-looking sentences.

2.3 Experiment

The 730 instances (415 true and 315 fake) of the original training data of the Answer Verification task were used for training our baseline model. For generating pseudo-fake claims by round-trip translation, we selected 415 true instances from them. In addition, we selected 298 true instances from the training data of the PoliInfo-3 Fact Verification task whose "UtteranceType" is Answer and "DocumentEntailment" is True. The reserved 53 true claims, which were not selected for the test data creation, were also added to them. A total of 766 true instances were used as the seed of the pseudo-fake claim generation methods.

For the RTM, fake claims were created by all three basic operations, the insertion and deletion of negation (NEG), Conversion to antonyms (ANT), and Subject-Object Exchange (SOE).

In NEG, our system adds or removes "not" from verbs and auxiliary verbs found in the part-of-speech analysis. In ANT, our system finds adverbs and adjectives and replaces them with antonyms using WordNet. In SOE, our system finds the subject and object of a sentence and swaps them.

We also generated pseudo-true claims by using simple Round-Trip Translation (hereinafter referred to as RTT), in order to mitigate the bias introduced by extending only the fake claims. We implemented them by using the natural language processing tool

NLTK[1]. For our machine translation in two directions, we used the DeepL API[2]. An example of the data generated by the above methods is attached in table 2.

Table 2: Example of RTT & RTM

Method		Example of generated sentence	True/Fake
ORG		国は有効な対策打ち出していない。 *The State has not put forward effective measures.*	True
RTT		政府は有効な対策を打ち出せていない。 *The State has not put forward effective measures.*	True
RTM	NEG	政府は効果的な対策を打ち出して いません。 *The State has put forward effective measures.*	Fake
	SOE	その 対策 は、効果的な 政府 には至っていない。 *The measures has not put forward effective State.*	Fake
	ANT	政府は 効果のない 対策は打ち出していない。 *The State has not put forward ineffective measures.*	Fake

In the formal run, 785 fake instances from GDADC[3] were distributed. Among these instances, we selected 50 for training purposes. The reason for this will be explained subsequently.

The number of data used for training is shown in Table 3.

Table 3: Statics of answer verification training data

Dataset		Q&A pairs	facts	fakes
PoliInfo-4	ORG	730	415	315
	RTT	415	415	0
	NEG	333	0	333
	SOE	265	0	265
	ANT	312	0	312
PoliInfo-3	ORG	351	351	0
	RTT	351	351	0
	NEG	286	0	286
GDADC(selected)		785 (50)	0	785 (50)

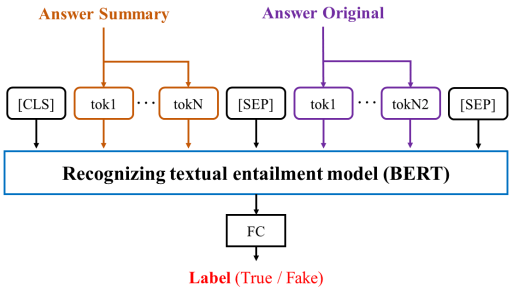

Figure 4: The schematic diagram of fact classifier

For the pre-trained language model for textual entailment, we used BERT-base model published by Inui laboratory at Tohoku University[4] using Hugging Face. Figure 4 shows the schematic diagram of the fact classifier.

[1]https://www.nltk.org
[2]https://www.deepl.com/pro-api?cta=header-pro-api
[3]https://sites.google.com/view/poliinfo4/game
[4]https://huggingface.co/cl-tohoku

2.4 Result

The evaluation results with the formal run data are shown in Table 4.

The baseline model is the pre-trained BERT fine-tuned using PoliInfo-4 dry-run training data, and the evaluation result of its output is ID 68.

ID 70 represents the evaluation result of the output from a model fine-tuned using a combination of PoliInfo-3 ORG, PoliInfo-4 ORG, RTT, NEG, SOE, and ANT to refine the baseline model.

Using the model from ID 70 to classify GDADC data resulted in 50 instances being falsely classified as true when they were actually fake. ID 138 represents the evaluation result of the output from a model fine-tuned using the training data of ID 70 along with these 50 instances.

ID 161 represents the evaluation result of the output from a model fine-tuned using the training data of ID 138 along with PoliInfo-3 RTT and NEG.

Table 4: The results of Answer Verification task

ID	accuracy	F-measure
68	0.7900	0.8073
70	**0.8800**	**0.8966**
138	0.8400	0.8621
161	0.7400	0.7451

2.5 Discussion

Fine-tuning the baseline model with pseudo-fake data generated mechanically by RTM resulted in a higher accuracy than that of the baseline model. In addition, an accuracy varied depending on the type of pseudo-fake data used for fine-tuning. We speculate that aligning the number of true instances with the number of fake instances in the overall training data improves the score.

Using fake data from GDADC for fine-tuning did not yield an improvement in accuracy. This can be attributed to two primary factors. The first factor is the mismatch between the formal-run test data and GDADC. While the formal-run test data was mechanically generated, the fake data from GDADC was human-created. The second factor is the bias towards fake predictions caused by training with a higher proportion of fake data.

3 STANCE CLASSIFICATION-2

3.1 Overview

Stance Classification-2 subtask aims to classify whether a politician's utterance on a particular bill is a "agree" or "disagree" stance. [2] We experimented with several pre-trained language models and developed a binary classifier using RoBERTa as the best model.

3.2 Method

We developed a binary classifier using RoBERTa. The columns "utterance" and "target" from the Stance Classification-2 dataset are used as input to the classifier. The input text is provided as:

`[CLS] target [SEP] utterance [SEP]`

If the input length exceeds 512 tokens, the input text is truncated from the end.

3.3 Experiment

We investigated BERT and RoBERTa using Stance Classification-2 DryRun dataset. The dataset is split into training and validation at a ratio of 8 to 2. We used the BERT-base model published by the Inui Laboratory at Tohoku University [5] and the RoBERTa-base model published by the rinna, Inc [6]. We used AdamW with weight decay, and trained the both models for 6 epochs. The batch size is set to 16.

3.4 Result

The results of the experiment are shown in Table 5. The ID columns in the table are on the PoliInfo-4 leaderboard. From the table 5, RoBERTa has higher accuracy than BERT. Based on the experimental results, we used RoBERTa in FormalRun. The results of our best system in DryRun and FormalRun are shown in Table 6.

Table 5: Results of Stance Classification-2 DryRun

	accuracy	ID
BERT	0.9082	18
RoBERTa	0.9271	19

Table 6: Results of Stance Classification-2

	accuracy	ID
DryRun	0.9271	19
FormalRun	0.9308	211

3.5 Discussion

To improve the performance of our system, we are currently investigating the use of additional information as a input.

4 MINUTES-TO-BUDGET LINKING

4.1 Overview

Minutes-to-Budget Linking task aims to link budget discussion statements in Congress with the relevant budget tables. In this task, it was realized using a ranking method based on Okapi BM25. Experiments on the selection of correct answers were also conducted, evaluated, and discussed.

4.2 Method

Scraping In this task, the council meeting minutes and the budget table are given in HTML-formatted data.

Since it is more convenient for the budget discussion remarks and the budget table to be in JSON format data, scraping is performed. For the scraping, we use BeautifulSoup, a Python library. The statements in the minutes of the council meetings in which the budget is being discussed are marked with a sentenceID, so only these are extracted by scraping. For the budget table, the part enclosed by table tags is retrieved as the budget table data.

[5]https://huggingface.co/cl-tohoku/bert-base-japanese-whole-word-masking
[6]https://huggingface.co/rinna/japanese-roberta-base

At the same time, morphological analysis using SudachiPy is also performed. The tokenizer is C and the type of dictionary used is full. The extracted parts of speech are limited to common nouns and proper nouns, and the obtained data are converted to JSON format. Table7 below shows the utterances before and after the conversion.

Table 7: Morphological analysis Before and After

Original	本市はこれまで、将来にわたって効率的かつ安定的に持続可能な行政運営をなし得る財政構造とするため、収支改善に向けた歳入確保や歳出削減に取り組んでおります。 The City has been working to secure revenue and reduce expenditures in order to improve its financial structure to ensure efficient, stable, and sustainable administrative operations in the future.
After scraping	市, 将来, 効率的, 安定的, 持続可能, 行政, 運営, 財政, 構造, ため, 収支, 改善, 歳入, 確保, 歳出削減 city, future, efficient, stable, sustainable, administration, management, fiscal, structure, because, balance, improvement, revenue, secure, expenditure, reduction

Ranking To achieve the linkage between the utterances and the budget table, a ranking method is used. Each utterance is a query and the budget table is a corpus group, which is ranked using Okapi BM25. For each query, the top k corpora with the highest Okapi BM25 values are output as the correct answers. The method of selecting correct answers with high evaluation is discussed in the experiment described below.

4.3 Experiment

Regarding how many of the top ranked corpora should be considered as outputs, we investigated the evaluation of two methods, one that uses the top k cases as output and the other that uses the top n% of cases as output. The results are shown in the table8.

Table 8: F scores when top k case and top N %

top k case	F score	top n %	F score
k = 4	0.2577	n = 3	0.2348
		n = 5	0.2554
		n = 10	0.2555
		n = 15	0.2534
		n = 20	0.2289

For the types of parts of speech to be extracted for morphological analysis, we investigated the evaluation of two methods, one that extracts only common nouns and proper nouns, and the other that extracts all parts of speech. The results are shown in table9.

Table 9: Types of parts of speech extracted by morphological analysis and F scores

part of speech	F score
Common and proper nouns	0.2555
All parts of speech	0.0253

4.4 Result

From the table8, the highest result was obtained when the top four responses were output as correct answers. However, when the top 5-15% are output as correct answers, the evaluation is almost the same. From the table9, it can be seen that the evaluation is much lower when all parts of speech are extracted compared to when only common nouns and proper nouns are extracted. This suggests that parts of speech such as particles and auxiliary verbs are noisy and have a negative impact on accuracy.

4.5 Discussion

Based on the results of the experiment, the correct response rate of the method based on ranking using BM25 is approximately 25%, and we expect no further dramatic improvement. In this task, Otaru city council meeting minutes and budget tables for the years H28 and R3 are given as input, but the HTML tag writing style is not standardized by year. Therefore, with the current simple scraping method, there is a possibility that important information may be missing from year to year. It will be necessary to consider a more flexible scraping method.

REFERENCES

[1] Haruki Ishikawa, Ryuto Kobayashi, Yuki Gato, and Tomoyosi Akiba. 2023. Pseudo Fake Data Generation using Round-Trip Translation for Fact Verification on Political Dabates. *Proceedings of the 10th International Conference on Advanced Informatics: Concepts, Theory and Applications.*

[2] Yasuhiro Ogawa, Yasutomo Kimura, Hideyuki Shibuki, Hokuto Ototake, Yuzu Uchida, Keiichi Takamaru, Kazuma Kadowaki, Tomoyoshi Akiba, Minoru Sasaki, Akio Kobayashi, Masaharu Yoshioka, Tatsunori Mori, Kenji Araki, Satoshi Sekine, and Teruko Mitamura. 2023. Overview of the NTCIR-17 QA Lab-PoliInfo-4 Task. *Proceedings of The 17th NTCIR Conference.* https://doi.org/10.20736/0002001326