DOI: https://doi.org/10.20736/0002001289

fuys Team at the NTCIR-17 QA Lab-PoliInfo-4 Task

Daigo Nishihara
Fukuoka University
Japan
td222007@cis.fukuoka-u.ac.jp

Hokuto Ototake
Fukuoka University
Japan
ototake@fukuoka-u.ac.jp

Kenji Yoshimura
Fukuoka University
Japan
yosimura@fukuoka-u.ac.jp

ABSTRACT

This paper reports on the fuys team's NTCIR-17 QA Lab-PoliInfo-4 Minutes-to-Budget Linking (MBLink) results. We thought that related tables could be found by focusing on the cells of the table. Learning inferences were made by combining the text of <p> tag with an ID and the text of table cell. The two were encoded and combined to perform a binary classification. We considered a table relevant if there was at least one related word in the table's cells. We also tried this when the text of a table cell was joined column by column and combined with the text of a <p> tag with an ID. The best accuracy was obtained when the text in table cells was joined column by column.

KEYWORDS

BERT, binary classification, TFIDF

TEAM NAME

fuys

SUBTASKS

Minutes-to-Budget Linking (MBLink)

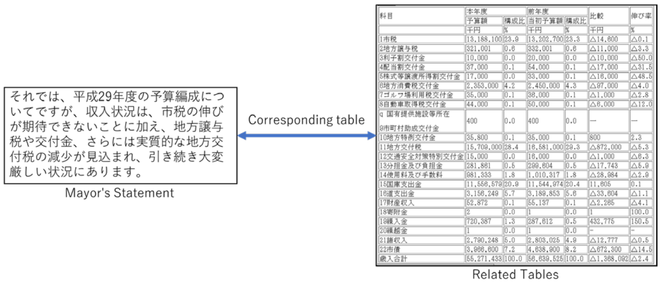
1 INTRODUCTION

The fuys team participated in the Minutes-to-Budget Linking subtask (MBLink) of the NTCIR-17 QA Lab=PoliInfo-4 Task [1]. This task can be thought of as a binary classification task using utterance sentences and table cells. This paper reports our approach to solving the problem and discusses the official results.

2 MBLink

The MBLink task is a task that links statements in the council meeting minutes explaining the mayor's budget with tables dealing with budget information, such as the budget summary table. In this article, we will target the mayor's opening remarks at the first regular meeting of Otaru City, and look for related items in tables dealing with budget information for the same year. The mayor's opening remarks are converted to an HTML file, so they can be read one sentence at a time using <p> tags. In addition, the statements in the mayor's opening remarks that explain the budget have already been assigned an ID, and we can find the table associated with that <p> tag with an ID and link it to the table.

For example, in the first regular meeting of 2017, we will link together the table that describes the increase or decrease in "市税 (city taxes)" in response to the mayor's statement on "予算編成 (budgeting)" (Figure 1). In some cases, there may be more than one related table, at which time it is necessary to link multiple tables together.



それでは、平成29年度の予算編成についてですが、収入状況は、市税の伸びが期待できないことに加え、地方譲与税や交付金、さらには実質的な地方交付税の減少が見込まれ、引き続き大変厳しい状況にあります。

科目	本年度		前年度		比較	伸び率
	予算額	構成比	当初予算額	構成比		
	千円	%	千円	%	千円	%
1市税	13,188,100	23.9	13,202,700	23.3	△14,600	△0.1
2地方譲与税	321,001	0.6	332,001	0.6	△11,000	△3.3
3利子割交付金	10,000	0.0	20,000	0.0	△10,000	△50.0
4配当割交付金	37,000	0.1	54,000	0.1	△17,000	△31.5
5株式等譲渡所得割交付金	17,000	0.0	33,000	0.1	△16,000	△48.5
6地方消費税交付金	2,353,000	4.2	2,450,000	4.3	△97,000	△4.0
7ゴルフ場利用税交付金	35,000	0.1	36,000	0.1	△1,000	△2.8
8自動車取得税交付金	44,000	0.1	50,000	0.1	△6,000	△12.0
q 国有提供施設等所在 9市町村助成交付金	400	0.0	400	0.0	—	—
10地方特例交付金	35,800	0.1	35,000	0.1	800	2.3
11地方交付税	15,709,000	28.4	16,581,000	29.3	△872,000	△5.3
12交通安全対策特別交付金	15,000	0.0	16,000	0.0	△1,000	△6.3
13分担金及び負担金	281,861	0.5	299,604	0.5	△17,743	△5.9
14使用料及び手数料	981,333	1.8	1,010,317	1.8	△28,984	△2.9
15国庫支出金	11,556,579	20.9	11,544,974	20.4	11,605	0.1
16道支出金	3,156,249	5.7	3,189,853	5.6	△33,604	△1.1
17財産収入	52,872	0.1	55,137	0.1	△2,265	△4.1
18寄附金	2	0.0	1	0.0	1	100.0
19繰入金	720,387	1.3	287,612	0.5	432,775	150.5
20繰越金	1	0.0	1	0.0	-	-
21諸収入	2,790,248	5.0	2,803,025	4.9	△12,777	△0.5
22市債	3,966,600	7.2	4,638,900	8.2	△672,300	△14.5
歳入合計	55,271,433	100.0	56,639,525	100.0	△1,368,092	△2.4

Figure 1 Examples of related table.

3 METHOD

We performed learning and inference using BERT[2] and searched for table cells to be linked using binary classification. For the BERT pre-training model, we used the Japanese language model created by the Inui/Suzuki Laboratory at Tohoku University, cl-tohoku/bert-base-japanese-whole-word-masking. The BERT-related part was implemented by "BertForSequenceClassification," a class for document classification in Transformers, an open-source library developed by HuggingFace[1] . The model was created by performing fine tuning on a pre-trained model with the number of epochs as 10, batch size as 32, learning rate as 1e-5, and maximum input length as 512.

We thought that related tables could be found by focusing on the cells of the table. Therefore, the relevant tables were searched for by binary classification using the text of the <p> tag with the ID tag and the text of the table's cells. The table was modified from an HTML file to a two-dimensional array for easy access to the cells. The text of the table was then used to create a list of texts.

We created our list in three different ways. We will now describe them.

3.1 Method 1

We went through the table cell by cell and extracted all cells that were not monetary expressions and created a wordlist (Figure 2). The text of <p> tags with an ID for the same year and words in

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

the wordlist were used, and they were encoded and combined to create the input sentences for the learning inference (Figure 3).

During training, input sentences were created with all combinations of texts of <p> tag with an ID and words from the wordlist. The words were trained with a label of 1 if the wordlist word existed in the table linked to the <p> tag and 0 otherwise.

Input sentences were created using them, and binary classification was performed. If there is at least one combination that produces a result of 1, that table was determined to be the relevant table (Figure 4).

3.2 Method 2

In Method 1, there was a problem that if the result for a word with a high frequency of occurrence in a table, such as "合計(total)" or "科目(subject)," was set to 1, many tables would be linked together. Therefore, we deleted words that appeared 6 or more times from the wordlist created by Method 1 and created a new wordlist (Figure 5).

Using the newly created wordlist, input sentences were created as in Method 1, and learning inference was performed.

3.3 Method 3

The table cells were joined by delimiters to form one sentence for each column, except for the cells in the table with only amounts, unit notations, and symbols, and a sentence list was created (Figure 6). We tried two types of delimiters for connecting table cells: [SEP], a special token in BERT, and ",(comma)", which is not a special token in BERT. Since the amounts are different from year to year, the cells with the amounts were omitted to avoid recognizing them as different when the year changes in the study and inference. We also decided to skip cells with only unit notation or symbols because they are present in many tables and are not a criterion for finding related tables.

We used text of <p> tag with an ID and sentence list sentences for the same year, encoded them, and then combined the two to create the input sentences (Figure 7).

The label at training time was 1 if the sentence in the sentence list was a sentence made in the related table, and 0 otherwise.

The inference used the created sentence list and looked at the text of <p> tag with an ID and the sentence list for each table, one by one.

Using them, input sentences were created and binary classification was performed. If there was at least one combination that produced a result of 1, that table was determined to be the relevant table.

3.4 Feature word extraction

The mayor's statements are often written differently, even if he or she is saying the same thing in different years. Therefore, different fiscal years may not link together tables that we would like to see linked together with statements of the same content. Therefore, we performed feature word extraction using TFIDF with text of <p> tag within the same year. We attempted to improve accuracy by using only the top 10 feature words instead of a single <p> tag text. The top 10 feature words were rearranged in order from the word with the highest TFIDF value and joined by delimiters to create a single sentence. Words that are feature words were defined as nouns and not numerals or verbs. As in Method 3, two types of delimiters were used: [SEP] and ", (comma)". Sentences created using the top 10 feature words were used in place of text of <p> tag, and learning inference was performed in the same manner as the aforementioned method. The method using feature word extraction by TFIDF was applied to Method 3, which had the highest accuracy.

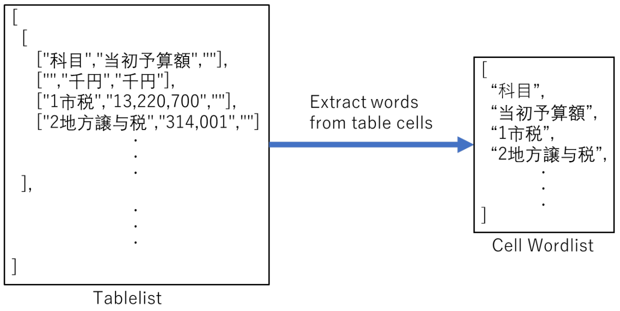

[
[
["科目","当初予算額",""],
["","千円","千円"],
["1市税","13,220,700",""],
["2地方譲与税","314,001",""]
],
]
Tablelist
Extract words from table cells
[
"科目",
"当初予算額",
"1市税",
"2地方譲与税",
]
Cell Wordlist

Figure 2 How to create a Wordlist.

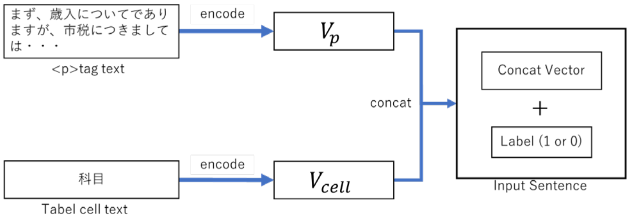


Figure 3 How to create Input Sentences in Methods 1 and 2.

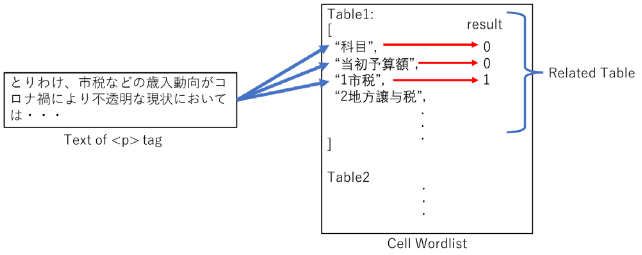


Figure 4 How to find related tables.

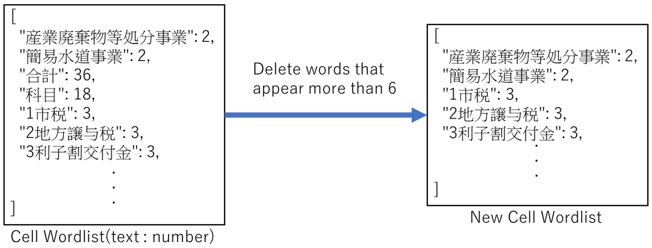


Figure 5 How to create a Wordlist in Method 2.

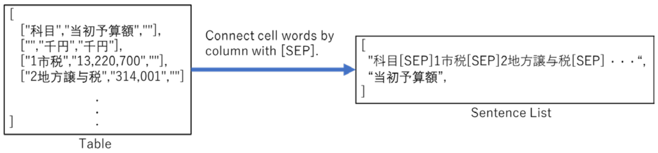

[
["科目","当初予算額",""],
["","千円","千円"],
["1市税","13,220,700",""],
["2地方譲与税","314,001",""]
.
.
.
]
Table
Connect cell words by column with [SEP].
[
"科目[SEP]1市税[SEP]2地方譲与税[SEP]・・・“,
“当初予算額”,
]
Sentence List

Figure 6 How to make a Sentence List in Method 3.

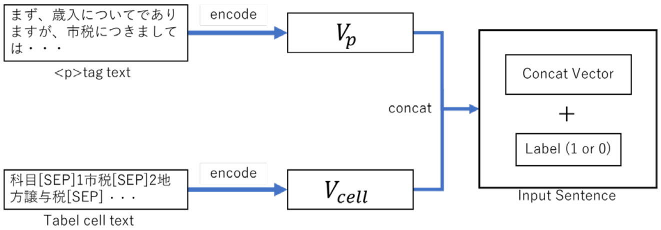


Figure 7 How to create an Input Sentence in Method 3.

4 RESULT

The results were 01.43% for Method 1 and 21.36% for Method 2 (Table 1). The result of Method 3 was 27.23% when [SEP] was used as the delimiter, and 36.66% when ", (comma)" was used as the delimiter (Table 2). The results show that the best results were obtained when Method 3 was applied. When feature words were extracted using TFIDF and [SEP] was used as the delimiter, the rate was 11.63%, and when a comma was used as the delimiter, the rate was 24.99% (Table 3). This indicates that the results are better without feature word extraction by TFIDF. It also shows that the results are better when commas, which are not special tokens, are used as delimiters when joining words.

Method 1 and Method 2 show higher recall and lower precision. Method 3, on the contrary, shows that recall is lower than precision. However, there is a large gap between recall and precision in Method 1 and Method 2, and Method 3 has a better balance between recall and precision. Therefore, it is better to use method 3 to link related tables.

Table 1 Results of Method 1 and Method 2.

	Method 1	Method 2
Result	01.43%	21.36%
Recall	88.79%	70.08%
Precision	00.74%	16.34%

Table 2 Results of Method 3.

	Method 3 ([SEP])	Method 3 (,)
Result	27.23%	36.66%
Recall	31.28%	40.53%
Precision	46.75%	52.06%

Table 3 Results of methods for feature word extraction.

	Feature word extraction ([SEP])	Feature word extraction (,)
Result	11.63%	24.99%
Recall	12.55%	25.10%
Precision	49.58%	46.41%

5 CONSIDERATION

As can be seen from the results, the best results were obtained when a comma was used as the delimiter in Method 3. In BERT, [SEP] is treated as a special token that connects sentences to sentences, so context is no longer taken into account if [SEP] is used to connect them. Also, when a comma that is not a special token is used, BERT recognizes it as a single sentence. Therefore, it is believed that the use of the non-special token comma improved the results of the current method.

One of the reasons for the poor results in feature word extraction using TFIDF is that the word delimiters did not do what was intended. For example, the word "地方交付税(local allocation tax)" was split into the words "地方(local)," "交付(allocation)," and "税(tax)," and existed in pieces when combined (Figure 8). This made it difficult to find the relevant tables, and it is believed that the prospect of being able to deal with different years was missed.

5.1 Problem

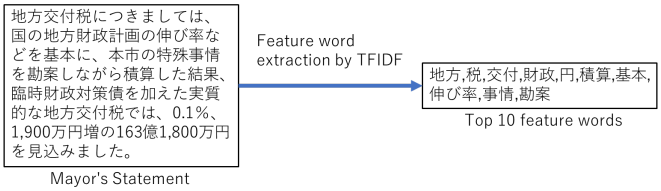
Analysis of the results when a comma is used as a delimiter in Method 3, which has the highest percentage of correct answers, reveals two things.

The first is that there are places where none of the related tables are attached. There were 22 out of 81 <p> tags with IDs in the correct data that had nothing attached. In Method 3, the columns of the table are connected and learning inference is performed. Therefore, as the number of rows in the table increases, the amount of words to be combined increases, and words that are material to human judgment, such as "一般会計(general accounting)," are more likely to be buried. As a result, it is thought that tables that should be linked together are being overlooked.

A possible solution to this problem is to separate the table by half of the rows when the number of rows is large. By doing so, the text created by connecting columns can be shortened, preventing necessary words from being buried. As a result, we believe that related tables will be missed less and accuracy will be improved.

Second, the data may not be well developed. In fact, when I looked at the tables that I had attached by mistake, I found that they were linked together in a way that could not be said to be unrelated. For example, in the result data, a comment about "地方交付税(local allocation tax)" was accompanied by a table containing the details of "地方交付税(local allocation tax)," but in the Gold data, this table was not linked (Figure 9).

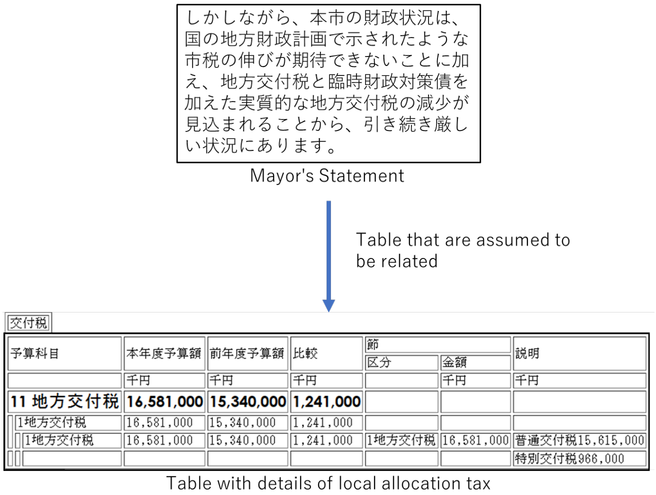
The solution to this problem would be to increase the number of annotators to eliminate table oversights and maintain the data.



地方交付税につきましては、国の地方財政計画の伸び率などを基本に、本市の特殊事情を勘案しながら積算した結果、臨時財政対策債を加えた実質的な地方交付税では、0.1%、1,900万円増の163億1,800万円を見込みました。

地方,税,交付,財政,円,積算,基本,伸び率,事情,勘案

Figure 8 Example of feature word extraction results



しかしながら、本市の財政状況は、国の地方財政計画で示されたような市税の伸びが期待できないことに加え、地方交付税と臨時財政対策債を加えた実質的な地方交付税の減少が見込まれることから、引き続き厳しい状況にあります。

交付税

予算科目	本年度予算額	前年度予算額	比較	節		説明
				区分	金額	
	千円	千円	千円		千円	千円
11 地方交付税	16,581,000	15,340,000	1,241,000			
1地方交付税	16,581,000	15,340,000	1,241,000			
1地方交付税	16,581,000	15,340,000	1,241,000	1地方交付税	16,581,000	普通交付税15,615,000
						特別交付税966,000

Mayor's Statement
Table that are assumed to
be related
Table with details of local allocation tax

Figure 9 Example of related table

6 CONCLUSIONS

We thought that this MBLink task could be used to find related tables by focusing on table cells. We approached this task as a binary classification task using the text of the <p> tags and the text of the table cells. We combined the <p> tags and table cells in three ways to create a classification model. The best classification model resulted in an F-score of 36.66%.

Since this method cannot join tables with a large number of rows, a method to split the table into smaller pieces is possible.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP22K12740, JP21H03769.

REFERENCES

[1] 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

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, & Kristina Toutanova. “BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding”, NAACL-HILT, 2019.