# AKBL at NTCIR-18 U4
# TableRetrieval and TableQA

So Takasago[1], Tomoyosi Akiba[2]
[1,2]Toyohashi University of Technology
[1]takasago.so.ro@tut.jp, [2]akiba@cs.tut.ac.jp

## Approach

Our proposed methods for TableQA and TableRetrieval follow the workflow shown in the right figure.

- Table Preprocessing
- Cell Text Classification
- Table Segmentation
- Table-to-Text Generation
- Add Text Above Table
- Sentence Selection

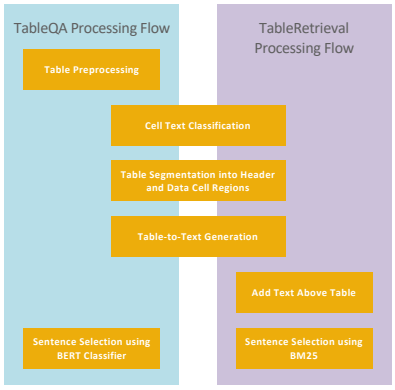


Figure : Processing Flow

## Conclusion

- The TableRetrieval subtask did not yield good results.
- In the TableQA subtask, we achieved scores of 0.7850 for cell_id and 0.6871 for value.
- These scores are higher than the value score of 0.6470 obtained using GPT-4o with TO.
- Achieving these results with DeBERTa-V3, which is relatively lightweight compared to LLMs like GPT-4o, demonstrates the usefulness of our approach.
- The current text generation method cannot handle complex tables well, so further improvements are needed.

## Related Works

- In last year's NTCIR-17 UFO[1] task, the TDE subtask was conducted.
- In this study, we reproduced the method of team OUC[2] and tackled the current task.

[1] Yasutomo Kimura, Hokuto Ototake, Kazuma Kadowaki, Takahito Kondo, and Makoto P. Kato. 2023. Overview of the NTCIR-17 UFO Task. Proceedings of The 17th NTCIR Conference (12 2023).
[2] Eisaku Sato, Keiyu Nagafuchi, Yuma Kasahara, Kazuma Kadowaki, Yasutomo Kimura. "OUC at NTCIR-17 UFO: TDE and TTRE." Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies, December 12-15, 2023, Tokyo, Japan.

## Table Segmentation

| 回次 | 回次 | 第44期 |
|---|---|---|
| 決算年月 | 決算年月 | 2017年2月 |
| 売上高 | (百万円) | 458140 |
| 経常利益 | (百万円) | 75007 |

Figure : Target Table

Using the TDE model, each cell text is converted to $[P(header|cell), P(data|cell)]$. (Cell Text Classification)

| | | |
|---|---|---|
| [0.95, 0.05] | [0.95, 0.05] | [0.95, 0.05] |
| [0.40, 0.60] | [0.40, 0.60] | [0.20, 0.80] |
| [0.90, 0.10] | [0.95, 0.05] | [0.01, 0.99] |
| [0.90, 0.10] | [0.95, 0.05] | [0.01, 0.99] |

Figure : Example output by TDE

Using the equation on the right, the TSS score is calculated for various splits, and the split position with the highest TSS is searched for.

$$\mathrm{TSS}(i,j) = \frac{\sum_{c\in\mathcal{H}(i,j)} P(\mathrm{header} \mid \mathrm{cell})}{|\mathcal{H}(i,j)|} + \frac{\sum_{c\in\mathcal{D}(i,j)} P(\mathrm{data} \mid \mathrm{cell})}{|\mathcal{D}(i,j)|}$$

$$(\hat{i},\hat{j}) = \arg\max_{i,j} \mathrm{TSS}(i,j)$$

| | | |
|---|---|---|
| [0.95, 0.05] | [0.95, 0.05] | [0.95, 0.05] |
| [0.40, 0.60] | [0.40, 0.60] | [0.20, 0.80] |
| [0.90, 0.10] | [0.95, 0.05] | [0.01, 0.99] |
| [0.90, 0.10] | [0.95, 0.05] | [0.01, 0.99] |

$i$–*th* row

$\mathcal{H}(i,j)$ $j$–*th* column $\mathcal{D}(i,j)$

Figure : The split with the highest TSS

The split shown in the left figure is the optimal split. When the score is calculated for this split,

$$\mathrm{TSS}(1,2) = \frac{\sum_{c\in\mathcal{H}(1,2)} P(\mathrm{header} \mid \mathrm{cell})}{|\mathcal{H}(1,2)|} + \frac{\sum_{c\in\mathcal{D}(1,2)} P(\mathrm{data} \mid \mathrm{cell})}{|\mathcal{D}(1,2)|}$$
$$= \frac{7.35}{9} + \frac{2.78}{3} = 1.74$$

## Table-to-Text Generation

We generate sentences from the table using the template「AのBはCです。」.

A

| 回次 | 回次 | 第44期 |
|---|---|---|
| 決算年月 | 決算年月 | 2017年2月 |
| 売上高 | (百万円) | 458140 |
| 経常利益 | (百万円) | 75007 |

B C

Figure : Example of a table after Table Segmentation

「第44期の売上高、(百万円)は458140です。」

When sentence generation is performed on the above table, the following sentences are obtained:

- 「第44期 の 決算年月 は 2017年2月 です。」
- 「第44期 の 売上高、(百万円) は 458140 です。」
- 「第44期 の 経常利益、(百万円) は 75007 です。」

## Results

- TableRetrieval had a low accuracy.
- DeBERTa-V3, which has relatively low computational cost, achieved accuracy comparable to LLM.

Table : Accuracy in TableQA

| Training Dataset | Accuracy | |
|---|---|---|
| | cell_id | value |
| TO (GPT-4o) | - | 0.6470 |
| JRTE-Corpus | 0.3323 | 0.3179 |
| Created Dataset | **0.7850** | **0.6871** |

Table : Accuracy in Retrieval.
Note that the accuracy for the "BM25" method is based on the DryRun and Valid datasets.

| Approach | Accuracy |
|---|---|
| TO (text-embedding-3-small) | 0.187 |
| BM25 | 0.209 |
| BM25(Add text above Table) | 0.329 |