KSU Systems at the NTCIR-16 Data Search 2 IR Subtask

Taku Okamoto
i2086042@cc.kyoto-su.ac.jp
Kyoto Sangyo University

Tomokazu Hayashi
g1954651@cc.kyoto-su.ac.jp
Kyoto Sangyo University

Hisashi Miyamori
miya@cc.kyoto-su.ac.jp
Kyoto Sangyo University

ABSTRACT

This paper describes the system and results of Team KSU work on the NTCIR-16 Data Search 2 IR subtask. The documents covered by this task consist of metadata extracted from the governmental statistical data and the body of the corresponding statistical data. The metadata is characterized by the fact that its document length is short, and the main body of statistical data is almost always composed of numbers, except for titles, headers, and comments. In the previous studies on ad hoc search for statistical documents, most of the ranking methods used only the metadata of the statistical documents, and there are few methods of using the contents of the tables of statistical data. However, ranking methods using only metadata have not been able to achieve the same or better performance compared to conventional ad hoc search for text documents. Therefore, in this paper, we propose a method that employs features of the table body of statistical data and a re-ranking method based on neural network models used in neural search, and verify how much the ranking results are improved. For the features of the main body of the table, we use eight types of features, four from the main body of the table and four from the whole table. As a neural search method, we use a re-ranking method based on the scores predicted from the features obtained by BERT and MLP. The results of the experiment showed that the method combining category search and BM25 resulted in nDCG@10 of 0.314 for Japanese and that of 0.069 for English. The results showed that Japanese ranked 2nd and English 6th among all teams.

KEYWORDS

Statistical data, Ad hoc search, Tabular data, Neural ranking model, Open data, Information retrieval

TEAM NAME

KSU

SUBTASKS

IR subtask (English, Japanese)

1 INTRODUCTION

In recent years, there has been progress in the development of infrastructures for the effective use of public data held by various organizations as open data, and the importance of ad hoc search infrastructures for statistical data, a type of open data, is increasing. For example, statistical data is considered to play an important role in fact-checking in order to deal with fake news, which is becoming a social problem. In general, statistical data released by the government and industry organizations are considered to be of a certain quality, and by comparing and contrasting these statistical data with information whose authenticity is uncertain, it is possible to check whether the information is consistent or not.

In NTCIR-16 Data Search 2[5] , there are three subtasks to support fact-checking using statistical data: the IR subtask, which takes the text to be fact-checked as input and retrieves relevant statistical data; the QA subtask, which identifies and outputs answers from the retrieved statistical data; and the UI subtask, which intuitively presents the output results to the user. In this paper, we describe the system and results of Team KSU in the NTCIR-16 Data Search 2 IR subtask.

In the previous studies on ad hoc search for statistical documents, most of the ranking methods used only the metadata of the statistical documents[11][6][7], and there are few cases where the contents of the main body of the statistical data are utilized. In addition, ranking methods using only metadata have yet to show equivalent or better performance compared to conventional ad hoc search ranking for text documents. Furthermore, neural search[1], which has been actively researched in recent years as a method to improve the performance of conventional ad hoc search methods for text documents, has so far failed to show much better performance than BM25[4] in ad hoc search for statistical documents.

In this paper, we propose a method that employs features of the main body of a table of statistical data and a re-ranking method based on a neural ranking model, and examine the degree to which the ranking results are improved. As features of the main body of the statistical data table, we use eight types of features, four from the main body of the table and four from the entire table. As a neural ranking model, we adopt a re-ranking method with scores predicted from the features obtained using BERT and MLP. The re-ranking method based on the scores predicted from the features obtained by BERT and MLP has shown relatively high performance in ad-hoc retrieval of Wikipedia tables[2], and we will verify how well it works for statistical documents of various sizes and structures, which are the target of this task.

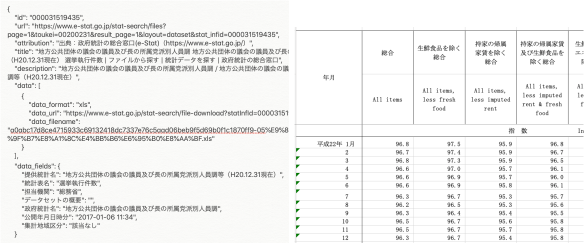
2 DATA COLLECTION

The statistical document dataset used in this task is collected from the government statistics portal site (e-Stat) and Data.gov, and each statistical document consists of a pair of metadata and statistical data, as shown in Figure 1. Examples of metadata and statistical data are shown in Figures 1 (a) and (b), respectively.

Statistical data consists of one or more tables. Figure 2 shows a single statistical data, and the tabs surrounded by a boxed area of (a) indicate that there are multiple tables in a statistical data.

A table is composed of a title, a column header, a row header, and a body. The boxed area of (b) in Figure 2 represents the title. The title may be composed of multiple cells.

The boxed area of (c) in Figure 2 shows the column header. The column headers are sometimes separated by a line break in a cell, and the hierarchical structure of the header contents is expressed by double-byte spaces. In some cases, the hierarchical structure of the header contents is expressed in multiple cells.



```
{
  "id": "000031519435",
  "url": "https://www.e-stat.go.jp/stat-search/files?
page=1&toukei=00200231&result_page=1&layout=dataset&stat_infid=000031519435",
  "attribution": "出典：政府統計の総合窓口(e-Stat)（https://www.e-stat.go.jp/）",
  "title": "地方公共団体の議会の議員及び長の所属党派別人員調 地方公共団体の議会の議員及び長の
（H20.12.31現在） 選挙執行件数 | ファイルから探す | 統計データを探す | 政府統計の総合窓口",
  "description": "地方公共団体の議会の議員及び長の所属党派別人員調 / 地方公共団体の議会の議
調等（H20.12.31現在）",
  "data": [
    {
      "data_format": "xls",
      "data_url": "https://www.e-stat.go.jp/stat-search/file-download?statInfId=00003151
      "data_filename":
"a0abc17d8ce4715933c69132418dc7337e76c5aad06beb9f5d69b0f1c1870ff9-05%E9%8
%9F%B7%E8%A1%8C%E4%BB%B6%E6%95%B0%E8%AA%BF.xls"
    }
  ],
  "data_fields": {
    "提供統計名": "地方公共団体の議会の議員及び長の所属党派別人員調等（H20.12.31現在）",
    "統計表名": "選挙執行件数",
    "担当機関": "総務省",
    "データセットの概要": "",
    "政府統計名": "地方公共団体の議会の議員及び長の所属党派別人員調",
    "公開年月日時分": "2017-01-06 11:34",
    "集計地域区分": "該当なし"
  }
```

(a) metadata

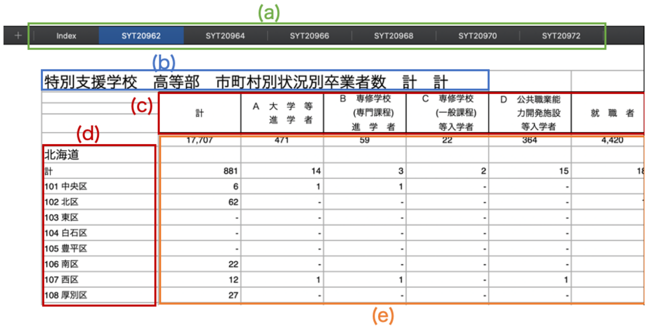
年月	総合	生鮮食品を除く総合	持家の帰属家賃を除く総合	持家の帰属家賃及び生鮮食品を除く総合	生鮮 エネ 除
	All items	All items, less fresh food	All items, less imputed rent	All items, less imputed rent & fresh food	Al le f
			指　数		In
平成22年 1月	96.8	97.5	95.9	96.8	
2	96.7	97.4	95.9	96.7	
3	96.8	97.3	95.9	96.5	
4	96.6	97.0	95.7	96.1	
5	96.6	96.9	95.7	96.0	
6	96.6	96.9	95.8	96.1	
7	96.3	96.7	95.3	95.7	
8	96.2	96.5	95.3	95.6	
9	96.3	96.4	95.4	95.5	
10	96.5	96.7	95.6	95.8	
11	96.5	96.7	95.7	95.8	
12	96.3	96.7	95.4	95.8	

(b)statistical data

Figure 1: Examples of metadata and statistical data that make up a statistical document

The boxed area of (d) in Figure 2 represents a row header. Similar to the column header, the hierarchical structure of the header contents may be expressed in one cell or multiple cells.

The boxed area of (e) in Figure 2 represents the main body. The body consists of multiple cells, and the value of each cell is expressed as a numerical value in most cases.



特別支援学校　高等部　市町村別状況別卒業者数　計　計

	計	A 大学等進学者	B 専修学校(専門課程)進学者	C 専修学校(一般課程)等入学者	D 公共職業能力開発施設等入学者	就職者
	17,707	471	59	22	364	4,420
北海道						
計	881	14	3	2	15	18
101 中央区	6	1	1	-	-	
102 北区	62	-	-	-	-	
103 東区	-	-	-	-	-	
104 白石区	-	-	-	-	-	
105 豊平区	-	-	-	-	-	
106 南区	22	-	-	-	-	
107 西区	12	1	1	-	1	
108 厚別区	27	-	-	-	-	

Figure 2: Examples of components of statistical data

Statistical data corresponds to a file of statistical data itself, saved in a format such as xls, csv or pdf. It contains one or more tabular data, and while the title and the header of the table contain usual non-numeric text, the body of the table is almost always composed of a very large amount of numbers. Table 1 shows the distribution of file formats for statistical data.

The metadata are extracted from the data described in the introduction pages of e-Stat and Data.gov statistical data. The metadata is a JSON format file and consists of the id of the statistical data, the URL of the introductory page of the statistical data, the title of the statistical data, as well as a description describing a brief summary of the statistical data, the URL of the statistical data itself, the file format, the variable name representing the file name, etc., and the corresponding values.

Table 2 shows the mean and standard deviation of the usage ratio of numerical and non-numerical words in the statistical data, as well as the total number of statistical data. To determine the usage rate of Japanese words, all strings in the statistical data were divided into words using MeCab, a Japanese morphological analyzer, and a word was considered to be a numeric word if all the characters constituting a word were ascii numerals, and a non-numeric word otherwise. From Table 2, we can see that numeric words are used at an average rate of about 0.79 and 0.60 in Japanese and English, respectively.

Table 1: Distribution of file formats for statistical data

(a) Japanese (e-Stat)

file format	frequency
xls	686436
csv	568042
pdf	49124
xlsx	34794
xlsm	6

(b) English (Data.gov)

file format	frequency
pdf	47260
x_gzip	17099
html	9296
xml	4443
csv	2919
plain	2761
none	2542
json	1938
rdf+xml	1484
octet-stream	1430
ms-excel	753
sheet	568
14 other formats	948

Similarly, Table 3 shows the mean and standard deviation of the percentage of use of numeric and non-numeric words in the metadata, as well as the total number of metadata. In the metadata, non-numeric words are used at an average rate of about 0.92 and 0.99 in Japanese and English, respectively, indicating that the rate of non-numeric words, which are easy to be useful as search clues, is larger than in the statistical data.

Next, we explain the document length of metadata. In metadata, values corresponding to variables other than title and description variables are not always easy to be useful for retrieval, such as document id or source URL. Therefore, this paper considers a pair consisting of values of title and description variables to be a metadata document. Table 4 shows the mean and standard deviation of the number of words in each of the metadata title and description variables, as well as the mean and standard deviation of the number of words in the values of both variables. Here, the number of words in Japanese is the number of segmented results as words obtained by MeCab, a Japanese morphological analyzer.

Table 2: Ratio of use of numeric and non-numeric words in statistical data

Language	Numeric words		Non-numeric words		Total number of statistical data
	avg.	std.	avg.	std.	
Japanese	0.786	0.221	0.213	0.221	1,338,402
English	0.595	0.082	0.505	0.465	92,930

Table 3: Ratio of use of numeric and non-numeric words in metadata

Language	Numeric words		Non-numeric words		Total number of metadata
	avg.	std.	avg.	std.	
Japanese	0.076	0.023	0.923	0.023	1,338,402
English	0.011	0.022	0.988	0.022	92,930

Table 4: Word length used in metadata

Language	title		description		title and description	
	avg.	std.	avg.	std.	avg.	std.
Japanese	36.2	16.7	20.0	8.2	56.2	21.2
English	11.5	7.6	98.3	84.9	109.9	86.9

In ad hoc retrieval, various documents such as web documents, academic papers, discussion forums, internal documents of governments and companies, news and social media articles, etc. have been used as retrieved documents. These documents are characterized by the fact that they are mainly written in natural language such as sentences used in daily life. For example, about 30,000 words[12] are used for a relatively long document such as a novel, and about 330 words[10] are used for a relatively short newspaper article.

On the other hand, the average number of words in Japanese for the variables title and description, which are described in natural language and are easy to be useful as search clues in the metadata targeted in this paper, is about 56 words, and the average number of words in English is about 109 words, both of which are very small. Therefore, even if the metadata contains a large percentage of non-numeric words, the number of words is small, and it is difficult to obtain search results that adequately satisfy the query by simply applying conventional ad hoc search methods.

3 PROBLEM FORMULATION

In this section, we formulate the problem to be addressed in this paper. First, let the query set Q and the document set to be retrieved D be represented as:

$$Q = \{q_i\}, \quad D = \{d_j\} \tag{1}$$

where a query q_i represents one or more word sequences $w_1^{q_i}$, $w_2^{q_i}$, $\ldots, w_{n_{q_i}}^{q_i}$ given in a single search, and a retrieved document d_j is represented as a pair of metadata m_j and statistical data t_j.

$$d_j = (m_j, t_j) \tag{2}$$

Also, t_j is represented as a tuple of table $\tau_{j,k}$ because it could have multiple tables.

$$t_j = (\tau_{j1}, \ldots, \tau_{jk}) \tag{3}$$

The set of documents in the document set to be retrieved D that are relevant to the query q_i is represented as:

$$D^{q_i,+} = \{d_j^{q_i,+}\} \tag{4}$$

In addition, we denote by $R_{rank(q_i, d_j^{q_i,+})}$ the ranking list of document sets $D^{q_i,+}$ related to the query q_i, sorted in descending order by the ranking function $rank(q_i, d_j^{q_i,+})$.

The goal of the problem we tackle in this paper is to obtain from a set of statistical documents D an appropriate ranking result $result_{q_i}$ of the document set that is relevant to the query q_i. That is

$$result_{q_i} = R_{rank(q_i, d_j^{q_i,+})} \tag{5}$$

4 CATEGORY SEARCH

We propose a method to refine the set of retrieved documents by categories in order to properly reflect the intended search range of user queries[8][9]. During indexing, we assign a category to each document to be retrieved using a text classifier and register it as a new document set to be retrieved with categories. At the time of retrieval, the categories are estimated from the query using a text classifier, and only the retrieved documents belonging to the estimated categories are ranked and the retrieval results are returned. The category set is determined by the following procedure.

First, we collect all search results obtained by nine different queries meaning "e-Stat" in the site search of Yahoo! Chiebukuro, one of the Japanese community question and answer Web services. Since the query may be included in either the question or the answer, or both, we extracted from the collected question-and-answer items those whose answers contained links to e-Stat, and listed the category to which the corresponding question belonged for each of them. In the same way, we extracted questions and answers that contained links to Data.gov from the English community question-and-answer Web service Yahoo! Answers, and listed the categories to which the corresponding questions belonged. In this way, we defined a category set consisting of 10 categories used in both Yahoo! Chiebukuro and Yahoo! answers.

Next, we construct a text classifier to estimate the categories. For each item in the collected question-answer item set, we extracted words with the parts of speech of nouns and verbs, and used the average of the distributed representations by fastText of the corresponding words as the feature vector for each item. Using this feature vector and the correct answer categories, a text classifier was trained by using SVM.

5 DATA AUGMENTATION

In order to compensate for the short document length of the metadata, we adopt a method to extract the header information of a table from the statistical data itself and add it to the document to be retrieved [8][9]. Specifically, we examine the number of non-empty cells in each row or column of the statistical data in order, and extract the column or row header according to the change in the number of non-empty cells. The extraction procedure for extracting column headers is shown in Algorithm 1. First, the input statistical data is denoted by sd, and the variable $prev$, which stores the number of non-empty cells in the previous row, is initialized with 0, and the variable hdr_col, which stores the column headers, is initialized with an empty list. The number of elements of $sd.rows$, which stores the sequence of each row of the statistical data as a list, is stored in max_row.

Then, repeat the following from line 1 to line max_row. Let $unempty\_cells()$ be the method that filters and extracts only the non-empty cells from the list of cells, and returns the result as a list. Apply $unempty\_cells()$ to $sd.rows[i]$, the list of cells in the ith row, and obtain the list of non-empty cells in the ith row. Store the number of elements $length$ of the list of non-empty cells in the ith row in $curr$. The number of non-empty cells in the first $i-1$th row is stored in $prev$.

If $curr$ is greater than $prev$, add a list of non-empty cells in the ith row at the end of the column headerhdr_col using the $append()$

Algorithm 1 Extracting column headers from statistical data

```
Input: statistical data $sd$
Output: column headers $hdr\_col$
  $prev = 0$
  $hdr\_col = []$
  $max\_row = sd.rows.\text{length}$
  for $i = 1, \dots, max\_row$ do
    $curr = sd.rows[i].unempty\_cells().\text{length}$
    if $curr > prev$ then
      $hdr\_col.\text{append}(sd.rows[i].unempty\_cells())$
    end if
    $prev = curr$
  end for
  return $hdr\_col$
```

method which adds another list to the end of a list. Update *prev* with *curr* and move on to the next row.

When the iteration finishes, return the column header *hdr_col* to h_j^{col}. h_j^{col} contains the list of rows that contains non-empty cells corresponding to the column headers. For the row headers, the process is performed as in Algorithm 1, where the rows and columns are interchanged, and the header h_j^{row} is extracted. The extracted header information is added to the metadata m_j to create a document d_j^{m+h} that compensates for the short document length of the metadata.

6 FEATURES USING THE MAIN BODY OF THE TABLE OF STATISTICAL DATA

In order to make effective use of the clues in the main body of a statistical data table, we extract various features that focus on the structure and content of the table.

First, we decided to use the four types of features that can be obtained only from the table body, as shown in Table 5. These are the formal features of the table body and are denoted by F_B. These are the ones used in a previous study [13]. #Rows, #Cells, and #EmptyCells are the average number of rows, columns, and empty cells in each table body in statistical data, respectively. #InLinks is the average number of predicates for each object in DBpedia [3] RDF per statistical data, when each word (mainly numeric) in the table body is considered as an object in RDF. DBpedia is a resource of RDF format LOD (Linked Open Data) based on Wikipedia.

Table 5: Features obtained from the table body only

Table feature items	Table feature description
#Rows	Average number of rows in each table body in statistical data
#Cols	Average number of columns in each table body in statistical data
#EmptyCells	Average number of empty cells appearing in each table body in statistical data
#InLinks	The total number of predicates for the corresponding object in DBpedia when each word in the main body of the table is regarded as an RDF object

Table 6: Features obtained from the entire table

Table feature items	Table feature description
hitsLC	Frequency of occurrence of query tokens in the leftmost column of the entire table
hitsSLC	Frequency of occurrence of query tokens in the leftmost two columns of the entire table of statistical data
hitsB	Frequency of occurrence of query tokens in the entire table of statistical data
qInPgTitle	Ratio of the number of query tokens in the title to the total number of tokens in the title of the metadata

In addition, we decided to use four types of features obtained from the entire table, as shown in Table 6. These are features that use specific zones of the statistical document, and are denoted by F_T. These features were also used in a previous study [13]. hitsLC and hitsSLC are the frequency of query tokens in the leftmost and leftmost two columns of the entire table, respectively. hitsB is the frequency of query tokens in the entire table. qInPgTitle is the ratio of the number of query tokens in the title to the total number of tokens in the metadata title. Both values are averages for one or more tables in a statistical data set.

From now on, we will denote the F_B and F_T features extracted from statistical documents together as F_{tab}.

7 RE-RANKING METHOD USING BERT AND MLP

An overview of the re-ranking method using BERT and MLP is shown in Figure 3. First, the input to BERT is a query and the entire table of statistical data. If we denote the query as q_i and the sequence of values in each cell of the lth row of the table $\tau_{j,k}$ in the statistical data t_j as $\rho_{j,k,l}$, then the input token sequence $S^{q_i,\rho_{j,k,l}}$ to BERT is as:

$$S^{q_i,\rho_{j,k,l}} = [[CLS], q_i, [SEP], \rho_{j,k,l}, [SEP]] \qquad (6)$$

where $[CLS] and [SEP]$ are special tokens that are inserted at the beginning and the end of the token sequence, respectively. Let $BERT_{CLS}()$ be the function that returns the output of the BERT model corresponding to the [CLS] token and $f_{BERT}^{q_i,t_j}$ [1] be the feature vector for the statistical data t_j obtained from BERT.

$$f_{BERT}^{q_i,t_j} = \frac{1}{kl} \Sigma_k \Sigma_l BERT_{CLS}(S^{q_i,\rho_{j,k,l}}) \qquad (7)$$

Next, we use the query, metadata, and the entire table of statistical data as input to the MLP. Here, we use the 8-dimensional feature vectors in Table 5 and Table 6 as input. Let $v_a^{q_i,m_j,t_j}$ be the input feature vector, $MLP()$ be the function that returns the output from the MLP model, and $f_{MLP}^{q_i,m_j,t_j}$ be the feature vector for the

[1] Because of BERT's restriction on the length of the input token sequence to 512 or less, only the leftmost 512 token sequence of each row is used as input.

Table 7: List of methods to be evaluated

Runs	Method name	(Initial) ranking method	Data augmentation	Category search	Table clipping	Features of the table body	Re-ranking method
		BM25	DA	Cat	Clip	F_{tab} ($F_B + F_T$)	BERT +MLP
RUN-{E, J}-{1, 2}	Cat+Clip+F_{tab}+BERT+MLP	✓	✓	✓	✓	✓	✓
RUN-{E, J}-{3, 4}	Cat+F_{tab}+BERT+MLP	✓	✓	✓		✓	✓
RUN-{E, J}-{5, 6}	Clip+F_{tab}+BERT+MLP	✓	✓		✓	✓	✓
RUN-{E, J}-{7, 8}	F_{tab}+BERT+MLP	✓	✓			✓	✓
RUN-{E, J}-{9, 10}	Cat+MB25	✓	✓	✓			

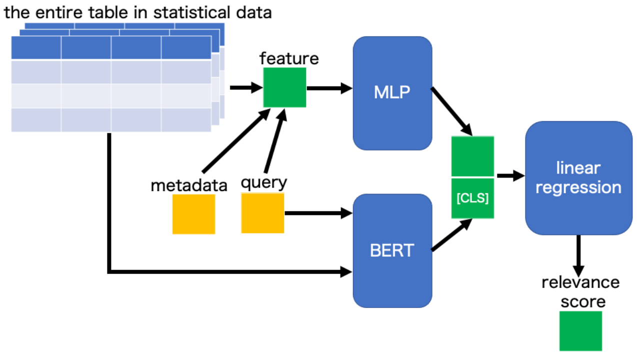


Figure 3: Re-ranking using MLP and BERT

statistical data t_j obtained from the MLP.

$$f_{MLP}^{q_i,m_j,t_j} = MLP(v_a^{q_i,m_j,t_j}) \tag{8}$$

Finally, $f_{BERT}^{q_i,t_j}$ and $f_{MLP}^{q_i,m_j,t_j}$ are concatenated to form a single vector f^{q_i,m_j,t_j} and fed into the fully connected layer to obtain the relevance score $score_{BRT+MLP}^{q_i,m_j,t_j}$.

$$f^{q_i,m_j,t_j} = [f_{MLP}^{q_i,m_j,t_j}; f_{BERT}^{q_i,t_j}] \tag{9}$$

$$score_{BRT+MLP}^{q_i,m_j,t_j} = Linear(f^{q_i,m_j,t_j}) \tag{10}$$

The initial ranking results are re-ranked using the relevance score $score_{BRT+MLP}^{q_i,m_j,t_j}$.

8 EXPERIMENT

8.1 METHODS TO BE EVALUATED

Table 7 shows the list of methods to be evaluated in the experiment. Since previous studies [9] have shown that augmenting metadata with table headers improves the value of nDCG@10, it is assumed in this experiment that all the methods to be evaluated use documents with additional header information in the metadata. In addition, considering that there is important information in the header part, we introduced a method of clipping the area of the table where the features are calculated so that the ratio of the area of the header part becomes relatively high (called "Table clipping"), and compared it with the case without clipping. The table clipping range was set to 15 columns and 5 rows based on the upper left corner.

We also investigate the effect of narrowing down statistical documents with high relevance by category search in re-ranking using BERT and MLP. Since it would take a lot of time to re-rank all the documents retrieved during the initial ranking using BERT+MLP, we will only re-rank the top 100 documents of the initial ranking result by BM25.

8.2 RESULTS AND DISCUSSION

Table 8 shows the experimental results. Compared with the proposed method, the method by the organizer BM25 (ORGJ-J-2, ORGE-E-2) showed the best performance. Even the category search only method, which showed superior results in the previous study [9], showed a difference of 0.123 from the best score for nDCG@10 in Japanese, and a difference of 0.142 for nDCG@10 in English. One of the possible reasons for this is that the organizer's BM25 used parameters that were optimized for the dataset, while the proposed method's BM25 did not optimize the parameters. In addition, the queries used in NTCIR-16 contain more proper nouns than those used in NTCIR-15, which may make it more difficult to estimate appropriate categories and calculate relevance. In fact, the evaluation results of nDCG@10 for the same category search-only methods (RUN-J-10 and RUN-E-9) were 0.448 and 0.255, respectively, when using Japanese and English queries in NTCIR-15 Data Search, while the results for NTCIR-16 Data Search 2 decreased to 0.218 and 0.211, respectively.

Comparing the results in Japanese with and without table clipping (RUN-J-{2,4} and RUN-J-{6,8}), there was no difference in the values of nDCG@10. Even though the values of the features (F_B, F_T) obtained from the main body of the table were different between the methods with and without clipping, there was no difference in the results. Therefore, we investigated whether there was a cause for the difference in the feature vectors input to the MLP. While the scaling by MaxAbsScaler is applied according to the original paper [2], we confirmed that most of the values of the features (F_B, F_T) obtained from the main body of the table were close to zero (Figure 4 - Figure 11). In fact, when checking Figure 4 - Figure 11, we can see that there is a large value that can be considered as outliers as the maximum absolute value of each feature. Due to these large values, the scaling results of most of the features are almost zero, and it is thought that the original, potential features of the table could not be fully utilized. In addition, Table clipping did not make clear differences as a result.

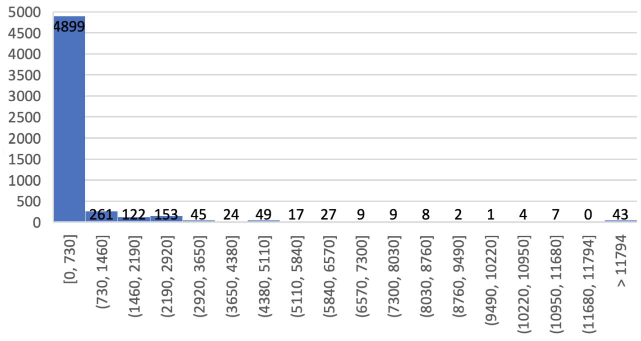


Figure 4: Histogram of #Rows

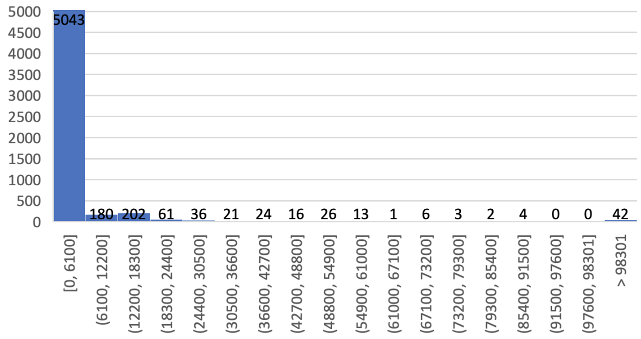


Figure 5: Histogram of #Cols

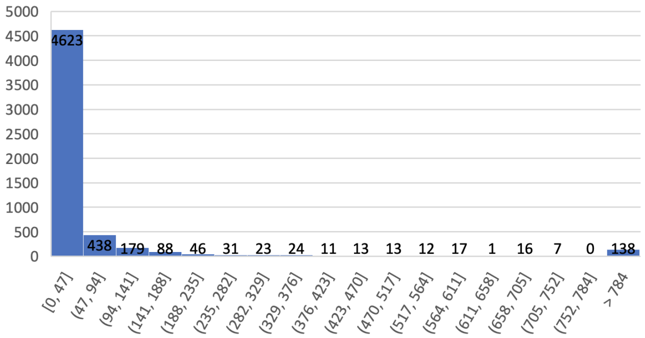


Figure 6: Histogram of #EmptyCells

The value of nDCG@10 by RUN-J-8 is 0.286 lower than that by ORGJ-J-2, and 0.219 lower than that by RUN-J-4 with the addition of category search. Furthermore, the value of nDCG@10 by RUN-J-4 is 0.096 lower than that by RUN-J-10. Thus, at least one of the features of F_{tab} and the re-ranking method of BERT+MLP is not effective in ranking statistical documents. On the other hand, the value of nDCG@10 by RUN-J-4 is improved by 0.067 compared to the value by RUN-J-8, which confirms that the category search is able to narrow down more relevant documents.

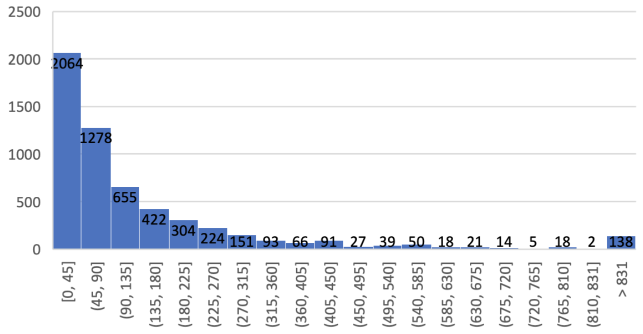


Figure 7: Histogram of #Inlinks

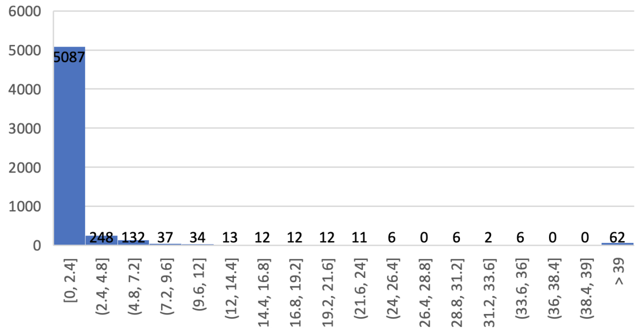


Figure 8: Histogram of hitsLC

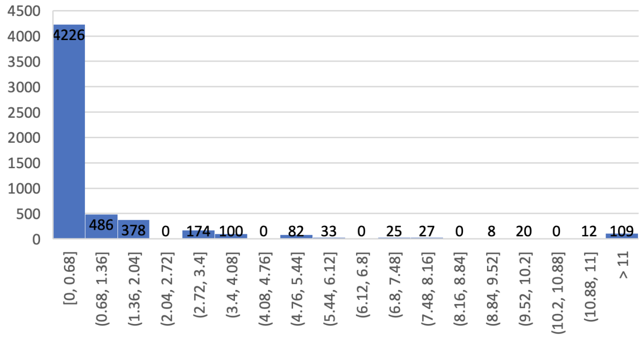


Figure 9: Histogram of hitsSLC

Comparing the case of applying Table clipping in English and the case of not applying table clipping (RUN-E-{1,3} and RUN-E-{5,7}), the value of nDCG@10 increased in the case of using clipping with category search, and decreased in the case of using clipping without category search. The values of nDCG@10 decreased in both cases when category search was used regardless of using clipping. The effect of category search was the opposite of that for Japanese. This may be due to the fact that the number of proper nouns in the NTCIR-16 test query increased compared

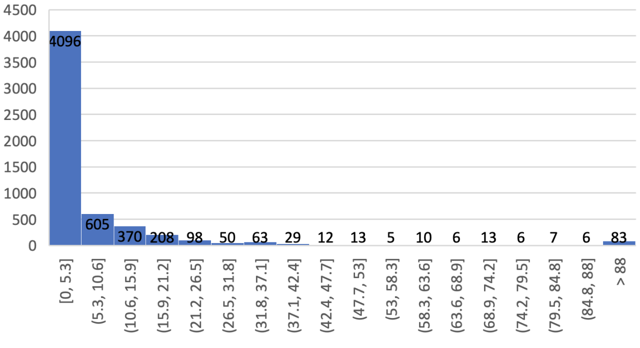


Figure 10: Histogram of hitsB

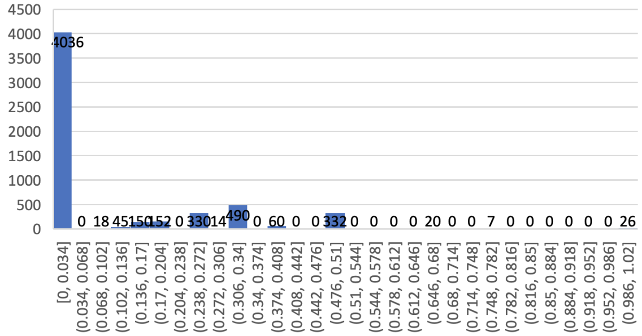


Figure 11: Histogram of qInPgTitle

to the NTCIR-15 test query, making it more difficult to estimate appropriate categories and calculate relevance.

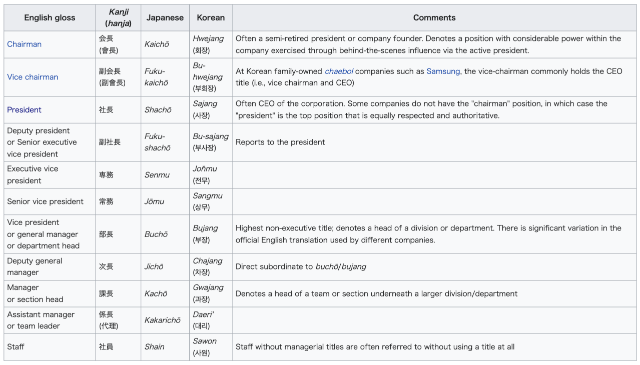
The value of nDCG@10 by RUN-E-7 is only 0.160 lower than that of ORGE-E-2, and even for RUN-E-3 with additional category search, the result is only 0.183 lower. Furthermore, the value of nDCG@10 by RUN-E-3 is 0.041 lower than that by RUN-E-9. Thus, at least either the features of F_{tab} or the re-ranking method of BERT+MLP is not effective in ranking statistical documents.

Table 8: Ranking evaluation results

runs	method name	nDCG@10
ORGJ-J-2	BM25	0.438
RUN-J-2	Cat+Clip+F_{tab}+BERT+MLP	0.218
RUN-J-4	Cat+F_{tab}+BERT+MLP	0.218
RUN-J-6	Clip+F_{tab}+BERT+MLP	0.151
RUN-J-8	F_{tab}+BERT+MLP	0.151
RUN-J-10	Cat + BM25	0.314
ORGE-E-2	BM25	0.211
RUN-E-1	Cat+Clip+F_{tab}+BERT+MLP	0.037
RUN-E-3	Cat+F_{tab}+BERT+MLP	0.028
RUN-E-5	Clip+F_{tab}+BERT+MLP	0.044
RUN-E-7	F_{tab}+BERT+MLP	0.051
RUN-E-9	Cat + BM25	0.069

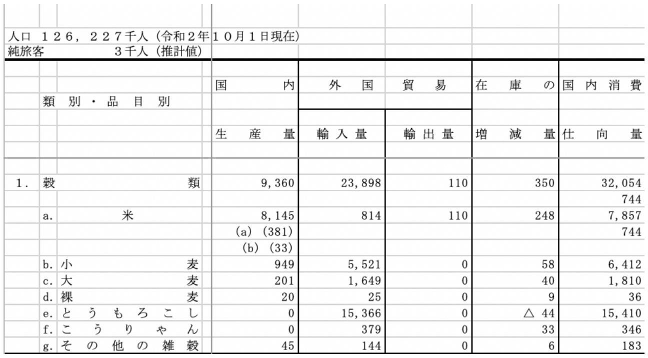
In particular, one possible reason for the results being much lower than nDCG@10 by the organizer's BM25 in both Japanese and English is that the information in the table itself was converted into feature vectors using BERT. The main body of the table used in previous studies contains ordinary word sequences other than numeric values, as shown in Figure 12, while the main body of the statistical document in this task consists mostly of numeric values, as shown in Figure 13. Even if we could obtain some regularity in the ordering of the numbers by applying BERT, it is unlikely that it is an intrinsically important regularity useful for this task. Therefore, in this task, we could not capture the features of the table as in previous studies [13], and the value of nDCG@10 would have been greatly reduced.

In the future, it is necessary to consider the use of features corresponding to the structure of the table in addition to the information in the table itself.



English gloss	*Kanji* (*hanja*)	Japanese	Korean	Comments
Chairman	会長 (會長)	*Kaichō*	*Hwejang* (회장)	Often a semi-retired president or company founder. Denotes a position with considerable power within the company exercised through behind-the-scenes influence via the active president.
Vice chairman	副会長 (副會長)	*Fuku-kaichō*	*Bu-hwejang* (부회장)	At Korean family-owned *chaebol* companies such as Samsung, the vice-chairman commonly holds the CEO title (i.e., vice chairman and CEO)
President	社長	*Shachō*	*Sajang* (사장)	Often CEO of the corporation. Some companies do not have the "chairman" position, in which case the "president" is the top position that is equally respected and authoritative.
Deputy president or Senior executive vice president	副社長	*Fuku-shachō*	*Bu-sajang* (부사장)	Reports to the president
Executive vice president	専務	*Senmu*	*Joňmu* (전무)	
Senior vice president	常務	*Jōmu*	*Sangmu* (상무)	
Vice president or general manager or department head	部長	*Buchō*	*Bujang* (부장)	Highest non-executive title; denotes a head of a division or department. There is significant variation in the official English translation used by different companies.
Deputy general manager	次長	*Jichō*	*Chajang* (차장)	Direct subordinate to *buchō*/*bujang*
Manager or section head	課長	*Kachō*	*Gwajang* (과장)	Denotes a head of a team or section underneath a larger division/department
Assistant manager or team leader	係長 (代理)	*Kakarichō*	*Daeri'* (대리)	
Staff	社員	*Shain*	*Sawon* (사원)	Staff without managerial titles are often referred to without using a title at all

Figure 12: Examples of statistical data used in traditional research



人口　１２６，２２７千人（令和２年１０月１日現在）
純旅客　　　　　３千人（推計値）

		国内	外国貿易		在庫の	国内消費
	類別・品目別					
		生産量	輸入量	輸出量	増減量	仕向量
1.	穀類	9, 360	23, 898	110	350	32, 054
						744
	a. 米	8, 145	814	110	248	7, 857
		(a) (381)				744
		(b) (33)				
	b. 小麦	949	5, 521	0	58	6, 412
	c. 大麦	201	1, 649	0	40	1, 810
	d. 裸麦	20	25	0	9	36
	e. とうもろこし	0	15, 366	0	△ 44	15, 410
	f. こうりゃん	0	379	0	33	346
	g. その他の雑穀	45	144	0	6	183

Figure 13: Examples of statistical data used in this study

9 CONCLUSION

This paper describes Team KSU system and results for the NTCIR-16 Data Search 2 IR task. We proposed a re-ranking method based on the features of the main body of the statistical data table and the neural network model used in neural search. For the features of the main body of the table, we use eight types of features, four from the main body of the table and four from the whole table. As a neural search method, we used a re-ranking method based on the scores predicted from the features obtained by BERT and MLP. The results of the experiment showed that the method combining category search and BM25 resulted in nDCG@10 of 0.314 for Japanese and 0.069 for English. The results showed that Japanese ranked second and English ranked sixth among all teams. However, the proposed features of the table itself and the re-ranking by the neural ranking model did not show any improvement. Even though BERT was applied to the main body of the table, which is mainly composed of numerical values, the feature vectors did not essentially reflect the important regularity of the features, which may have contributed to the deterioration of the ranking results. In the future, it is necessary to consider the use of features corresponding to the structure of the table in addition to the information in the table itself.

REFERENCES

[1] Zeynep Akkalyoncu Yilmaz, Shengjin Wang, Wei Yang, Haotian Zhang, and Jimmy Lin. 2019. Applying BERT to Document Retrieval with Birch. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*. Association for Computational Linguistics, Hong Kong, China, 19–24. https://doi.org/10.18653/v1/D19-3004

[2] Zhiyu Chen, Mohamed Trabelsi, Jeff Heflin, Yinan Xu, and Brian D. Davison. 2020. Table Search Using a Deep Contextualized Language Model. *CoRR* abs/2005.09207 (2020). arXiv:2005.09207 https://arxiv.org/abs/2005.09207

[3] Kato Fumihiro. 2017. DBpedia Linked Data Project (in Japanese). *Journal of Information Processing and Management* 60, 5 (2017), 307–315. https://doi.org/10.1241/johokanri.60.307

[4] Makoto P. Kato, Hiroaki Ohshima, Ying-Hsang Liu, and Hsin-Liang Chen. 2020. Overview of the NTCIR-15 Data Search Task. In *Proceedings of the NTCIR-15 Conference*.

[5] Makoto P. Kato, Hiroaki Ohshima, Ying-Hsang Liu, and Hsin-Liang Chen. 2022. Overview of the NTCIR-16 Data Search 2 Task. In *NTCIR-16*.

[6] Ryota Mibayashi, Pham HuuLong, Naoaki Matsumoto, Takehiro Yamamoto, and Hiroaki Ohshima. 2020. Uhai at the NTCIR-15 Data Search Task. In *Proceedings of the NTCIR-15 Conference*.

[7] Phuc Nguyen, Kazutoshi Shinoda, Taku Sakamoto, Diana Andreea Petrescu, Hung Nghiep Tran, Atsuhiro Takasu, Akiko Aizawa, and Hideaki Takeda. 2020. NII Table Linker at the NTCIR-15 Data Search Task: Re-ranking with Pre-trained Contextualized Embeddings, Data Content, Entity-centric, and Cluster-based Approaches. In *Proceedings of the NTCIR-15 Conference*.

[8] Taku Okamoto and Hisashi Miyamori. 2020. KSU Systems at the NTCIR-15 Data Search Task. In *Proceedings of the NTCIR-15 Conference*.

[9] Taku Okamoto and Hisashi Miyamori. 2021. Ad Hoc Search for Statistical Data Based on Refinement and Augmentation of Retrieved Documents and Query Expansion (in Japanese). *Information Processing Society of Japan* (2021).

[10] Mainichi Shimbun. 1996. CD-Mainichi Shimbun 1995 Data Collection, Nichigai Associates. http://www.nichigai.co.jp/sales/mainichi/mainichi-data.html

[11] Lya Hulliyyatus Suadaa, Lutfi Rahmatuti Maghfiroh, Isfan Nur Fauzi, and Siti Mariyah. 2020. STIS at the NTCIR-15 Data Search Task: Document Retrieval Re-ranking. In *Proceedings of the NTCIR-15 Conference*.

[12] Takuto Watarai and Masatoshi Tsuchiya. 2020. Developing Dataset of Japanese Slot Filling Quizzes Designed for Evaluation of Machine Reading Comprehension. In *Proceedings of the 12th Language Resources and Evaluation Conference*. European Language Resources Association, Marseille, France, 6895–6901. https://www.aclweb.org/anthology/2020.lrec-1.852

[13] Shuo Zhang and Krisztian Balog. 2018. Ad Hoc Table Retrieval using Semantic Similarity. *CoRR* abs/1802.06159 (2018). arXiv:1802.06159 http://arxiv.org/abs/1802.06159