IIAI Open Conference Publication Series
IIAI Letters on Institutional Research
Vol. 003, LIR152
DOI: https://doi.org/10.52731/lir.v003.152

Entity Linking among Categorized Knowledge Resources for Computer Science Curricula

Michiko Yasukawa [*] , Koichi Yamazaki [†]

Abstract

In educational information processing, both humans and computers require more effective information access than ever. However, there is one serious issue in text processing in higher education. The quantity of digitized text resources is not sufficient to develop large language models in specialized areas. General purpose language models often create unreliable results when the training data do not cover the target domain. To deal with this problem, our proposed method is designed to establish links between lecture course information and entities for real world knowledge. Owing to this entity linking, the meaning of text data can be more easily understood by humans and computers. Our method employs curriculum standards and university syllabus information to associate important keywords with text entities defined in Wikipedia articles. The results of evaluation experiments suggest the effectiveness of the proposed method in feature selection and entity linking for educational information. The contributions of this study include detailed comparison among dictionaries in Japanese morphological analysis. Our findings are expected to provide useful insights for researchers engaging in educational data analysis.

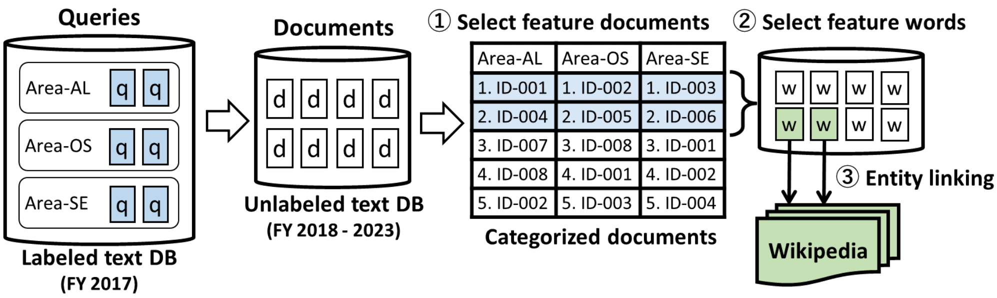
Keywords: data accessibility, institutional research, faculty development, knowledge graph

1 Introduction

Recent developments in digital technology have made the utilization of computers crucial in educational information processing at universities. While skills implemented using computers, such as machine translation and dialogue with chatbots are increasingly comparable to those exhibited by humans in some fields, educational information processing at universities still requires extra efforts to make effective use of computers' abilities.

State-of-the-art AI technology, such as generative AI, uses training data to create language models. For this reason, results (output) produced by language models are dependent on the training data (input) and can be unpredictable and unreliable. Specifically, computer-based assistant tools using large language models may not be practical in specialized areas

[*] Gunma University, Gunma, Japan
[†] Tokyo Denki University, Tokyo, Japan



Area-AL	Area-OS	Area-SE
1. ID-001	1. ID-002	1. ID-003
2. ID-004	2. ID-005	2. ID-006
3. ID-007	3. ID-008	3. ID-001
4. ID-008	4. ID-001	4. ID-002
5. ID-002	5. ID-003	5. ID-004

Figure 1: Feature selection and entity linking

with few digitized data. Therefore, it is necessary to enhance accessibility to relevant educational information on an individual basis, rather than depending on general purpose language models. For example, it would be useful to create customized language models for specific objectives using the text data at hand. To build a language model for practical usage, entities in the text data should be properly marked up so that humans and computers can recognize their meaning. Here, an **entity** is a thing or concept that has a **link** to its definition in the knowledge resources[1].

For Japanese language models, developers and research groups provide trained language models[2]. Reportedly, LUKE [1] has achieved high performance in varied Japanese tasks such as relation extraction and question answering. This language model incorporates not only linguistic knowledge, but also wide-ranging knowledge about the real world that is accumulated in Wikipedia. However, using the entire Wikipedia is computationally too difficult. For this reason, its approach currently focuses on learning only high priority areas for the developer and the provided models are not accurately created in less prioritized areas. Also, the publicly available trained models have the problem of not being able to handle entities that are not part of the training vocabulary or entities whose meanings have changed after the training process. Taking into account these current issues, this study proposes a method to identify entities in university syllabus information and link them to Wikipedia. Our motivation is to explore text analysis techniques as a new fundamental technology for university educational information processing. The outcome of this study is expected to bring useful insights from the viewpoint of text analysis in institutional research since we report the results of comprehensive experiments using three system dictionaries (IPADIC, UniDic, Neologd) for the most popular Japanese morphological analyzer, MeCab.

The remainder of this paper is organized as follows. In Section 2, the proposed method is introduced. In Section 3 and 4, the experimental data and results are described, respectively. Then, we provide a discussion in Section 5 and future work in Section 6.

2 Method

The outline of our method is illustrated in Figure 1. The method consists of three steps: feature document selection (Step-1), feature word selection (Step-2), and entity linking

[1] Let us consider the following example sentence. "[Institutional research] is a broad category of work done at schools, colleges and universities. " In this sentence, the marked-up word "Institutional research" is an entity that has a link to its definition and it is differentiated from other ordinary words, such as "category" and "work."

[2] `https://huggingface.co/models?search=japanese`

(Step-3). For feature document selection in Step-1, similar document searches for identifying feature documents is performed. In these document searches, the queries are pieces of text data in a knowledge base maintained by experts, and the documents to be searched for are syllabus documents used in university lecture courses.

In our previous study on text analysis of grant application documents, a basic method for feature document selection was proposed [2]. The main points of the feature document selection are summarized as follows.

(1-1) Each labeled document is used as a search key to search for similar documents in an unlabeled text database.

(1-2) The N best search results from each query are grouped by categories to select the M best search results for each category. The selected N multiplied by M documents are used as a candidate set of feature documents. (E.g., If N is set at 100 and M is set at 20, then 2,000 documents in total belong to a candidate set.)

(1-3) Automatic text classification is applied to the feature documents using a machine learning algorithm. The categorized documents are evaluated by using the classification accuracy to choose an optimal set of feature documents among all candidate sets.

In (1-3) above, it is necessary to determine the optimal combination of N and M from all possible combinations. Hence, this method becomes infeasible when the computation is too large to find parameter values. In our previous study [2], we reported that using a large text database was beneficial in realizing feature document selection. When a collection of documents is very large, an overwhelmingly small number of high-frequency documents (head) and a huge number of low-frequency documents (tail) exist. This tendency of text data is called the long-tail phenomenon, and it inevitably occurs in large text databases since it is an intrinsic nature of natural languages [3]. Thus, similar documents are gathered in the top ranks in the search results[3].

While the data in the previous study consisted of 900,000+ documents, our target data for this study contained only 30,000+ documents. The small scale of the text database makes it difficult for parameters to converge in the top ranks. Therefore, our technological challenge in the current study is how to determine the optimal M and N values within the permissible computation time. While the disadvantage for this study is that the small data size results in a larger area for parameter search, the benefit is that each attempt is shorter as the search index is smaller[4]. Based on these preliminary considerations, the following improvements to the previous method were made in this study.

(2-1) Upper and lower limits are set for parameter values N and M to perform the grid search. Similar documents are searched and classified by applying all combinations of N and M. Then, the classification accuracy is obtained for each attempt.

(2-2) The average value of accuracy for each N among all M is calculated. Then, the optimal value for N with the highest average accuracy is selected.

[3] Figure 4(b), which we discuss later, is a typical long tail distribution. When the quantity of text data is large, a tall head and a long tail are observed. If we divide the items into 3 groups: high frequency, medium frequency, and low frequency, and then select only high frequency, we can narrow the range of item variation.

[4] Figure 4(a), which we discuss later, is an example of a small dataset. Since it does not show a long tail distribution, there are no overwhelmingly dominant items. However, there are only three items and the range of the items is small. When the search index is sufficiently small, brute-force searches can be applied.

(2-3) According to the application's need, a preferable number for the M value is selected to construct a group of documents. For applications that require higher accuracy, M is set at a smaller value, and if recall is more important, the value for M is increased.

To reduce the computational cost per attempt in parameter search, the morphological analysis of queries and documents was performed outside the parameter search. In addition, since the process of selecting N documents to aggregate by area is a computationally expensive process, the aggregated results are cached in memory after calculation, and are called from the cache data for the second and subsequent times.

After determining a set of feature documents, feature word selection is conducted by applying the Mutual Information (MI)[5] measure, which is a common techniques for information retrieval [4]. Once feature words are obtained, entity linking is performed. In this study, the following markup is used to ensure that the syllabus is highly understandable for both humans and computers.

(3-1) The top page of our wiki system presents the list of areas (such as AL, OS, and SE). The page is used as the table of contents page. Feature documents (the syllabus) categorized in each area are linked from the table of contents page and can be easily referred to.

(3-2) All nouns that appear in the syllabus are extracted and arranged on a page as a list of keywords in alphabetical order. This is used as an index page for humans, as in the case of the index of a printed book.

(3-3) Each word in the keyword list is also used as a subject heading, just like the cataloging tools at a library[6]. For each feature word, an internal link to the syllabus page is placed under the heading. If the index word has a Wikipedia link, an external link is placed under the feature word. Feature words that appear in both the knowledge base and Wikipedia are marked with a special symbol (e.g., an asterisk) near their anchor strings to gather attention from the viewer.

The above markup should increase the readability of syllabus information for both humans and computers to interpret the meaning of the lecture content. Furthermore, it is expected that faculty members who write syllabus information will be helped in scrutinizing the content of the syllabus. If a concept described with a keyword is supposedly defined in Wikipedia but is not linked to any Wikipedia articles, there is a possibility that the keyword contains misspellings of foreign words or Kanji conversion errors in Japanese.

3 Data

For this study, we used the latest educational resources (the latest standard guideline for curricula and syllabus documents) and Wikipedia. The details of the data used for this study are described below.

[5] `https://nlp.stanford.edu/IR-book/html/htmledition/mutual-information-1.html`

[6] For example, subject headings defined by the National Diet Library, Japan are used for university book collections. `https://www.ndl.go.jp/en/data/classification_subject.html`

Table 1: Pieces of text in BOK (body of knowledge) for CS curricula

Area	Description	Unit	Word	Summary	Notice
AL	Algorithms and Complexity	7	90	58	0
AR	Architecture and Organization	8	110	85	26
CN	Computational Science	6	118	49	5
DS	Discrete Structures	6	102	56	0
GV	Graphics and Visualization	6	139	57	15
HCI	Human-Computer Interaction	10	151	65	0
IAS	Information Assurance and Security	11	89	78	0
IM	Information Management	12	83	59	23
IS	Intelligent Systems	12	91	60	22
MR	Media Representation	5	39	45	20
NC	Networking and Communication	7	29	58	9
OS	Operating Systems	12	126	86	32
PBD	Platform-Based Development	5	3	43	3
PD	Parallel and Distributed Computing	9	76	77	9
PL	Programming Languages	17	118	91	0
SDF	Software Development Fundamentals	4	80	71	0
SE	Software Engineering	10	102	104	42
SF	Systems Fundamentals	10	70	55	0
SP	Social Issues and Professional Practice	11	51	67	15

3.1 Dictionaries

As a knowledge resource, we use J17-CS, which is the CS curriculum standard [5]. This curriculum standard is provided by the Information Processing Society of Japan (IPSJ). We extracted pieces of text from it to use them as search queries (See, Figure 1). The dataset consists of the areas (category labels), description of the areas, and a word list. As the entire dataset is generally referred to as BOK (Body of Knowledge), we follow this custom and call it BOK, hereinafter.

The areas in BOK are abbreviations with two or three letters in the English alphabet, and the areas (e.g., AL) are corresponding to the area description (e.g., Algorithm and Computational Complexity). The concepts to be learned in each area are called knowledge units, and each unit is listed as compulsory or elective, as well as the estimated learning time. The concepts and knowledge units to be studied in each area are accompanied by a written description, called a summary, in which various topics are mentioned in the text. In some areas, notes are given to explain obsolete topics that were removed, or other matters requiring special attention. The list of technical terms used in BOK is organized as a word list. Table 1 shows the quantity of text data included in J17-CS. In the table, Area indicates the category label for the area data. Description is the corresponding description of the area. Unit is the knowledge unit. Word is the headword of the word list. Summary is a paragraph in the report giving the outline of the area and its knowledge units. Notice is an additional notice for the area. As can be seen from the numbers in the table, the text data for areas vary in text size. Some areas have more than 100 pieces of text, while others have fewer than 50. The pieces of text in Word for PBD are very few in number, so the method in this study was not applicable. Based on the explanation in Summary and Notice for PBD, we learned that it is closely related to SDF. Hence, PBD is merged into SDF and

Table 2: Intersection and difference between BOK and dictionary tokens

Dict.	Description	Ver.	Surface	Distinct	Inter.	Diff.
BOK	Body of Knowledge	J17-CS	2,040	1,827	n/a	n/a
IPA	IPAdic for MeCab	2.7.0-20070801	392,126	325,871	277	1,550
UNI	Unidic for MeCab	2.1.2	756,463	570,139	285	1,542
NEO	NEologd (mecab-ipadic)	10-Sep-2020	5,572,307	5,179,042	411	1,416
Jawiki	Japanese Wikipedia	03-Apr-2023	2,213,757	2,202,659	811	1,016

18 areas in total are used as our target areas. In the search for similar documents for feature document selection, Description, Unit, Word, and Summary in Table 1 are used. For the similar document search, Area (the alphabetic characters) and Notice are not used as queries because they do not contain sufficient query terms.

Pieces of text in Table 1 contain duplications and stylistic variations (upper/lower alphabets, ASCII/UTF-8 alphanumerals, and so on). Hence, in order to use them as search queries, we needed to perform text normalization and tokenization. To apply tokenization, we use the Japanese morphological analyzer, MeCab [6]. For MeCab, three widely-used system dictionaries are provided: IPAdic [7], UniDic [8], and Neologd [9]. The versions of the dictionaries used in this study are shown in the middle rows of Table 2. In the table, IPA, UNI, and NEO in the left column are shortened names, derived from the names of the dictionaries listed in the Description. The three dictionaries will be referred by the abbreviations (IPA, UNI, NEO), hereinafter. In Table 2, Surface indicates the defined headwords, and Distinct indicates the number of headwords without duplications. The columns on the right-hand side indicate Inter and Diff that are the intersection and difference between BOK and the dictionary's tokens. Note that the sum of intersection (for example, 277 for IPA) and difference (for example, 1,550 for IPA) is equal to the number of distinct tokens in BOK, which is 1,827. As can be seen, the number of distinct tokens in IPA is smaller than that for UNI. While IPA contain common compound words in Japanese, UNI separate unknown compound words into single known words. For example, a compound word "Database System" can be a single token by applying a dictionary for compound words (such as, IPA) while the two tokens, "Database" and "System" can be recognized by applying a dictionary for single words (such as, UNI). NEO is an extension of IPA and it contain both words in IPA and many neologisms (new words) that are common entities in Japanese Wikipedia. For example, longer compound words, such as "Database Management System," is included in NEO. Jawiki in the bottom row of the table indicates the number of tokens in Japanese Wikipedia. Jawiki contained more common tokens with BOK than the three system dictionaries for morphological analysis.

3.2 Syllabus documents

We constructed a corpus of syllabus documents for this study. The syllabus documents in the corpus were downloaded from websites of 10 Japanese national universities. These universities were identified by manual Google searches. We confirmed that these websites were open to the public for information disclosure purposes and did not disable downloading syllabus HTML files. We also confirmed that lectures at these universities included one or more lectures on CS curricula. To construct the corpus, we downloaded 9,887 syllabus HTML files. The breakdown of the collected files is as follows: University A (170

Table 3: Number of syllabus documents and tokens

Dict.	#Doc.	#Token	max(n|D)	max(f|D)	max(n|T)	max(f|T)	count(n)	total(f)
IPA	30,846	38,837	871	4,584	22,691	160,938	1,595,490	3,073,823
UNI	*do.*	49,413	872	4,717	22,958	163,259	1,622,272	3,206,750
NEO	*do.*	37,828	870	3,763	22,670	158,244	1,551,609	2,863,773

files), University B (56 files), University C (227 files), University D (1303 files), University E (133 files), University F (182 files), University G (154 files), University H (699 files), University I (175 files), University J (295 files). The files from University A, University I, and University J were the syllabi for the year 2023, and the other files were for the year 2022. Since the latest data alone was not sufficient for text data analysis, we also used the undergraduate and graduate syllabi for the year 2018 at University J (6,493 files).

Some of the syllabus HTML files contained announcements not directly related to the course content, such as week of days and the name of the lecture room building. Hence, we segmented each HTML file into paragraphs. Then, only paragraphs that were associated with relevant headings were used in similar document searches. The relevant headings included course objectives, course overviews, course topics, weekly schedules, keywords, and messages to students. We refer to the obtained paragraphs as syllabus documents, hereinafter. The number of obtained syllabus documents is shown in the Doc column of Table 3. In addition, the documents were divided into tokens by the morphological analyzer, and the number of counted tokens is shown in the Token column of Table 3. The number of tokens in UNI was larger than those in other dictionaries. NEO had fewer tokens than the others. This stylistic tendency of syllabus documents is similar to that of BOK. Details on the occurrence of tokens are shown in columns 4 to 9 of Table 3. The columns represent the following values.

- max(n|D) ··· the maximum number of distinct tokens in a document
- max(f|d) ··· the maximum sum of token frequency in a document
- max(n|T) ··· the maximum number of documents in which a token appears
- max(f|t) ··· the maximum sum of document frequency for a token
- count(n) ··· the number of non-zero elements in the word-document matrix
- total(f) ··· the total frequency of words in the corpus

If we look closer at a single document in the corpus, differences among different tokenization dictionaries appear to be trivial. However, when we are concerned with the total sum of token frequencies in the entire corpus, the differences among dictionaries becomes non-negligible, as can be seen in the right-most column in Table 3.

4 Experiments

Using the data described in the previous section, we conducted evaluation experiments as follows.

Table 4: F1-scores for feature documents

Dict.	Source	AL	AR	CN	DS	GV	HCI	IAS	IM	IS	MR	NC	OS	PD	PL	SDF	SE	SF	SP	Avg.
IPA	BOK	.38	.33	.26	.31	.47	.30	.50	.34	.62	.33	.40	.29	.38	.37	.39	.23	.00	.35	.351
	Syll.	.59	.27	.44	1.0	.92	.82	.91	.75	.44	.77	.92	.63	.77	.57	.52	.88	.00	.92	.711
UNI	BOK	.36	.23	.25	.35	.49	.34	.55	.39	.56	.40	.36	.27	.34	.38	.39	.26	.00	.42	.356
	Syll.	.63	.13	.75	.75	.88	.80	1.0	.80	.73	.80	.86	.42	.50	.60	.38	.62	.00	.77	.659
NEO	BOK	.26	.35	.13	.24	.43	.34	.33	.23	.50	.40	.18	.31	.38	.28	.37	.21	.00	.30	.295
	Syll.	.75	.52	.75	.88	.72	.63	.83	.93	.67	.55	.20	.71	.62	.86	.80	.71	.00	.92	.715

4.1 Feature document selection

To confirm the effectiveness of feature document selection, we performed a parameter search based on the accuracy in document classification. We used Linear SVC [10] for the machine learning algorithm for document classification.[7] Then, the N value with the best average accuracy among all M values was identified. We refer to this as the optimal N value, which depend on the dictionary used for text tokenization. Specifically, the optimal N value was 36, 44, and 116 for IPA, UNI, and NEO, respectively. By using the optimal N value for each dictionary, we evaluated feature documents with an arbitrarily chosen M value. While the optimal value of N varies depending on the dictionary, M was set to a constant value of 20 to consolidate the experimental conditions among the three system dictionaries. By using the obtained feature document set, the f1-score was calculated. The results are shown in Table 4. In the table, the f1-score for all areas and the average f1-score are presented for the baseline and proposed methods. BOK indicates the initial feature document set that is our baseline method. Syll indicates outcomes of our proposed method. The results shown in a lightly gray-colored cells in the table indicate cases wherein one of the three methods outperformed the other two. The darker gray cells indicate the cases wherein two of the methods outperformed the other methods.

As can be seen, the proposed method (Syll) outperformed the baseline method (BOK) in all cases for all three dictionaries. In addition, a comparison of the three dictionaries shows that UNI had the lowest f1-score. In more detail, the f1-score for IPA is between that for UNI and NEO, and NEO has the highest value. It should be noted that the f1-score with the baseline (BOK) has the lowest value with NEO and the highest value with UNI. Possible reasons for this result are as follows. The size of BOK (see, Table 2, BOK) is much smaller than that of Syll (see, Table 3, Doc). In addition, NEO includes many headwords. Hence, recognition of many compound words cannot be achieved using small text data (BOK), while it can for large text data (syllabus documents). For the area SF (software fundamental), the f1-scores obtained by the baseline and proposed methods were zero for all dictionaries (Table 4). This CS area is designed for intensive learning of basic common concepts that are included in other areas. Hence, feature documents in this area are not suitable for automatic document classification. This result suggests that even for reasonable categories defined by human experts, automatic document classification cannot be applied when categories contain concepts that overlap each other.

[7] `https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html`

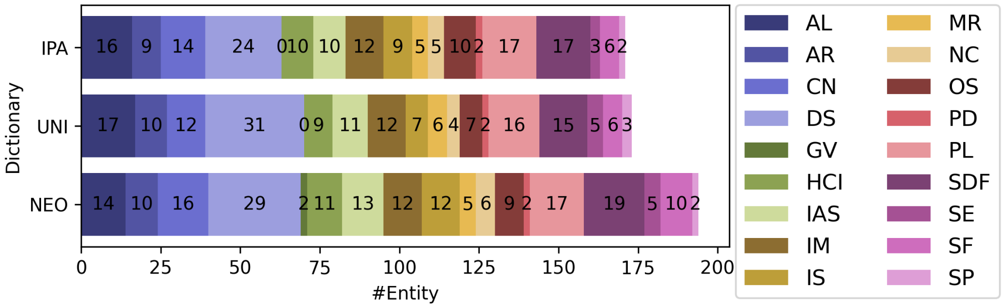


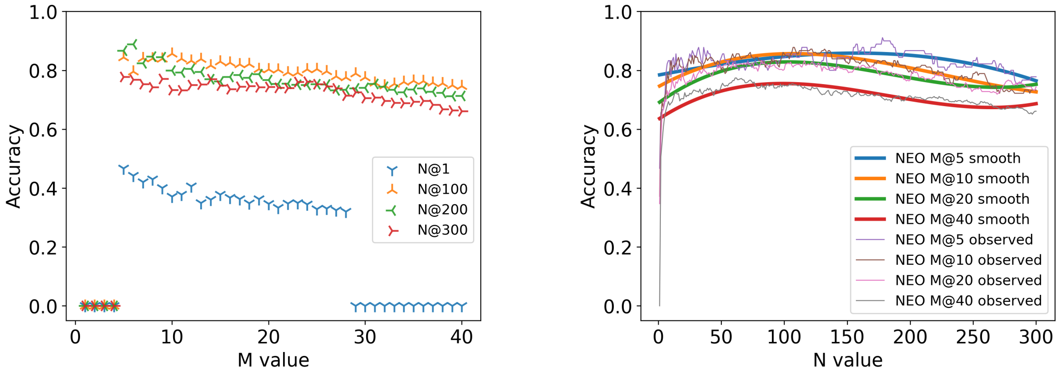
Figure 2: Number of obtained entities for syllabus documents

4.2 Feature word selection

Next, we evaluated the effectiveness of feature word selection. In this experiment, the number of obtained entities was investigated. Here, the obtained entities were the feature words in the feature documents (i.e., the syllabus documents). The feature words were tokens in the syllabus documents that were segmented by using MeCab with one of the three dictionaries (IPA, UNI, and NEO). The number of feature words depended on the dictionaries. Specifically, we obtained 171, 173, and 194 feature words with IPA, UNI, and NEO, respectively. Figure 2 visualizes the breakdown of the obtained results, with different colors indicating different areas in BOK. The numbers indicate the number of obtained feature words, which can be linked to the Wikipedia entities. As can be seen, NEO was able to identify the largest number of feature words. This suggests that our method of combining MeCab and NEO is effective for entity linking for educational information processing. The obtained results are considered to be reasonable since NEO is a dictionary that includes many new words defined in Wikipedia. However, as shown in Table 2, the latest version of NEO was released on September 10, 2020, and no new words have been added since then. When a large number of compound words are registered in the dictionary, the added words are more likely to be recognized as tokens. However, there is a side effect that dictionary entries may be unexpectedly matched with unrelated tokens when many words are randomly added. As can be seen in Table 2, there are 8 cells with IPA grayed out. IPA contained the smallest number of words. While IPA has more unknown words than the others, it produces the next best result after NEO. In Japanese text analysis, what words should be added to the morphological dictionary is a non-trivial question, and this will be investigated further in a future study.

4.3 Brute-force parameter search

To investigate the parameter search in more detail, the relationship among the N and M values and classification accuracies in the case of the dictionary NEO is visualized in Figure 3(a). The graph on the right side shows the accuracy for N values of 1, 100, 200, and 300 when the M value varied from 1 to 40. For any N value, when the M value was too small (i.e., less than five), the classification failed and the accuracy was zero. When the N value was set at one, the classification failed when the M value was 29 or larger. The reason for this result is explained as follows. Classification failed when the number of documents in each class was less than five, since the default value for cross-validation in the machine



(a) M-dependence of accuracy for different N values

(b) N-dependence of accuracy for different M values

Figure 3: Accuracies obtained with different parameter values

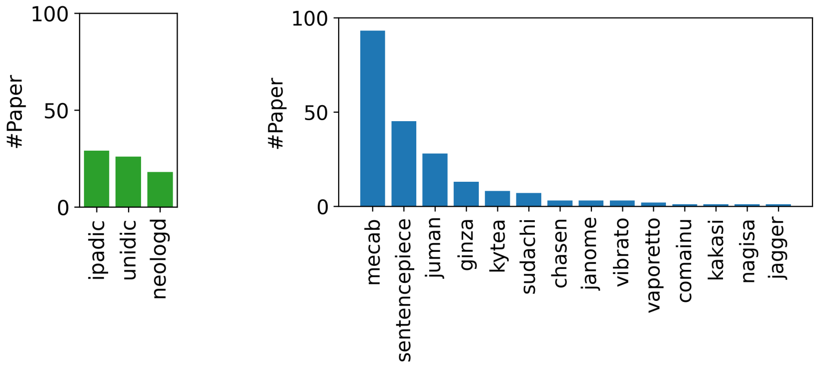
learning library in the experiments was set at five. Classification failed when the number of feature documents obtained by a similar document search was too small to predict unknown features. Overall, the classification was successful when the M and N values were optimally chosen. To summarize the results of this experiment: (i) feature selection fails when the values for N and M are too large or too small, and (ii) when feature selection is performed for smaller values of M, the quality of the obtained feature documents is higher.

A more detailed visualization of the dependence of accuracy on M and N is shown in Figure 3(b). The thin lines indicate the observed feature selection accuracies with NEO, for M values of 5 to 40. The thick lines present smoothed values obtained using Savitzky-Golay filter [11]. For an M value of five, the accuracy is a maximum for N values between 150 and 200. When the M value is 10, 20, or 40, the accuracy peaks between 150 and 200, although the peak values are lower. This shows that even when the same data and the same dictionary are used, different N and M values may affect accuracy, and there can be multiple peak values.

The efficiency tactics described in Section 2 greatly reduced the computation time, allowing the parameter search to complete in 8 hours on a desktop PC (Ubuntu 20.04, Intel Core i9-9820X CPU @ 3.30GHz, 256 GB RAM). If human experts (e.g., members in the task force) have to peruse all syllabus documents in the corpus to define the knowledge base, it would take too much time to be feasible. If knowledge base maintenance takes several years by humans, technological advances in computer science curricula would surpass the abilities of humans for document document processing. On the other hand, our proposed method can make full use of computation power to automatically select feature documents and feature words in the syllabus corpus for entity linking. Our method is expected to be helpful in promoting the digitization of university education.

5 Discussion

Recently, morphological analysis has been utilized in various fields, including institutional research (IR). As there are multiple choices for software and dictionaries, the current standards need to be clarified. Hence, we surveyed the trends among 579 papers in the pro-



(a) Dictionaries

(b) Morphological analyzers

Figure 4: Number of research papers mentioning morphological analysis

ceedings of the 29th Annual Meeting of the Association for Natural Language Processing[8]. Figure 4 shows the number of research papers mentioning morphological analysis. The counts for dictionaries and morphological analyzers are presented in Figure 4(a) and 4(b), respectively. The labels on the X-axis indicate the names mentioned in the proceedings. The three dictionaries, ipadic, unidic, and neologd respectively correspond to IPA, UNI, and NEO in this study. While IPA was mentioned more frequently than the other dictionaries, there was no remarkable difference observed among the three. On the other hand, morphological analyzers had more mentions than dictionaries. Among other things, MeCab was mentioned more than twice as often as Sentencepiece. This is a typical long tail phenomenon, demonstrating the overwhelming advantage of the first rank (MeCab). Therefore, this study, which reports experiments using MeCab and the three dictionaries, is expected to provide timely findings for Japanese text analysis.

Although text analysis for Japanese is a developing technology, there are previous studies in the field of IR that have applied text analysis technology to obtain beneficial results. For example, in a study conducted by Aihara et al., text analysis was performed on university mid-term plans to obtain co-occurring words and word graphs [12]. In a previous study conducted by Tsumagari et al., text analysis was applied to a student free-text questionnaire to identify important topics related to students' interests and future job careers [13]. Regarding the analysis of curricula, Oishi described a survey of IR curricula and provided a discussion based on the survey results [14]. Our study used Computing Curriculum Standard J17 [5], which was developed with reference to the ACM's Curricula Recommendations [15]. Peng et al. [16] reviewed literature on knowledge graphs. They reported that the technological limitations for knowledge graphs included the acquisition of knowledge from multiple resources and their integration into an existing knowledge graph. Our study is expected to be utilized for adding new nodes to existing knowledge graphs by establishing entity links between curriculum standards, university course syllabi, and Wikipedia.

Much of the previous research on data mining in education was concerned with numerical data analysis. In the study conducted by Kondo et al., [17] multi-objective optimization was used to analyze students' academic performance. In the study conducted by Mohamad et al., [18] literature on educational data mining before April 2013 was reviewed. They reported that popular techniques for educational data mining included clustering, classifi-

[8] NLP2023, `https://www.anlp.jp/nlp2023/`

cation, and prediction. Asif et al., [19] investigated a data mining method for predicting students' academic achievement. Costa et al., [20] evaluated the effectiveness of educational data mining techniques. They reported that (i) preprocessing for text data analysis and (ii) algorithm tuning were important for predicting students' academic failure. In a study conducted by Angeli et al., [21] association rules mining and analysis of questionnaire data were addressed from the viewpoint of classroom research in Europe and Australia. Misuraca et al., [22] applied opinion mining to students' feedback comments in university education.

While the present paper focuses on educational information processing in computer science, Akoka et al. [23] analyzed research projects in design science. Specifically, they identified knowledge contributions in order to make paths of knowledge types in the existing body of knowledge. Mukherjee et al. [24] reviewed literature on Industry 5.0, and reported that the most prominent barriers in emerging economies included (a) funding system, (b) capacity scalability, (c) upskilling, and (d) reskilling of human labor. In Japan, MEXT (Ministry of Education, Culture, Sports, Science and Technology) has been promoting a scheme for student-centered higher education ecosystem through digitalization[9]. In this scheme, university faculty, staff, students, and industry are encouraged to share information about university education. In the future, more and more business matching will be required among researchers, students, digital engineers, companies, reskilling participants, and others. We believe that our study will contribute to improving the accuracy and efficiency of such matching.

6 Conclusion

In this paper, we proposed a method of entity linking for educational information in higher education. The proposed method uses the knowledge base of the core curriculum in computer science to identify relevant syllabus information and perform feature extraction for applying entity linking with Wikipedia articles. It is expected that by identifying important feature words in syllabus documents and creating links to the definition of those words, it will help both humans and computers to more easily understand the meaning of text data in highly specialized areas. We conducted evaluation experiments using 1,827 pieces of text data obtained from the body of knowledge in computer science (CS BOK) and 30,846 paragraphs in syllabus documents, which were downloaded from the websites of 10 national universities in Japan. As a result of the experiment, we confirmed that the proposed method is more effective than the baseline method in terms of feature selection for entity linking. We also confirmed that the performance of the proposed method was affected by the parameter values for feature extraction and vocabulary in morphological analysis dictionaries. Our proposed method is considered to be effective when (i) Japanese tokenization is performed using a dictionary with a rich vocabulary in contemporary written Japanese and (ii) appropriate parameter values are set for feature selection. What types of compound words should be registered in the morphological analyzer's system dictionary is an open question, especially for text data containing less frequent and highly specialized terms in university lecture courses. We will attempt to refine the vocabulary selection for the text preprocessing in a future study.

[9] `https://scheemd.mext.go.jp/`

Acknowledgments

For this study, we used syllabus information obtained from the websites of the following universities: Gunma University, Kanazawa University, Kyoto University, Nagoya University, Saitama University, Shiga University, The University of Tokyo, Tokushima University, Wakayama University, Yokohama National University. Note that university names are in alphabetical order. This study was supported by the ISM Cooperative Research Program (2023-ISMCRP-0006) and JSPS KAKENHI Grant Number JP23K11764.

References

[1] I. Yamada, A. Asai, H. Shindo, H. Takeda, and Y. Matsumoto, "LUKE: Deep Contextualized Entity Representations with Entity-aware Self-attention," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. ACM, 2020, pp. 6442–6454.

[2] M. Yasukawa and K. Yamazaki, "Feature Selection by Thematic and Temporal Distinction in Research Grant Applications," *IIAI Letters on Institutional Research*, vol. 001, no. LIR019, pp. 1–13, 2022.

[3] G. K. Zipf, *Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology*. Addison-Wesley, 1949.

[4] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[5] Information Processing Society of Japan (IPSJ), "Computing Curriculum Standard J17," 2018. [Online]. Available: https://www.ipsj.or.jp/annai/committee/education/j07/curriculum_j17.html

[6] T. Kudo, "MeCab: Yet Another Part-of-Speech and Morphological Analyzer." [Online]. Available: https://taku910.github.io/mecab/

[7] M. Asahara and Y. Matsumoto, "ipadic version 2.7.0 User's Manual." [Online]. Available: https://ja.osdn.net/projects/ipadic/

[8] Center for Language Resource Development, NINJAL, "Electronic Dictionary with Uniformity and Identity." [Online]. Available: https://clrd.ninjal.ac.jp/unidic/

[9] S. Toshinori, "Neologism dictionary based on the language resources on the Web for Mecab," 2015. [Online]. Available: https://github.com/neologd/mecab-ipadic-neologd

[10] C. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[11] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures." *Analytical chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[12] S. Aihara, M. Mori, S. Hirokawa, K. Kanekawa, and T. Sugihara, "Text Analysis to the Preambles of the 4th Medium-term Goals / Plans of National University Corporations," *IIAI Letters on Institutional Research*, vol. 001, no. LIR024, pp. 1–6, 2022.

[13] T. Tsumagari, N. Nakazato, and T. Tsumagari, "Student's Interests and Career Understanding: A Topic Analysis of First-year Career Courses," *IIAI Letters on Institutional Research*, vol. 001, no. LIR013, pp. 1–8, 2022.

[14] T. Oishi, "What is the Essential Curriculum for IR in Japan?" *IIAI Letters on Institutional Research*, vol. 001, no. LIR009, pp. 1–5, 2022.

[15] Association for Computing Machinery (ACM), "Curricula Recommendations," 2020. [Online]. Available: https://www.acm.org/education/curricula-recommendations

[16] C. Peng, F. Xia, M. Naseriparsa, and F. Osborne, "Knowledge Graphs: Opportunities and Challenges," *Artificial Intelligence Review*, pp. 1–32, 2023.

[17] N. Kondo, T. Hatanaka, and T. Matsuda, "Evaluation of Predictive Models in Institutional Research Based on Multi-Objective Optimization," *IIAI Letters on Institutional Research*, vol. 001, no. LIR018, pp. 1–9, 2022.

[18] S. K. Mohamad and Z. Tasir, "Educational data mining: A review," *Procedia - Social and Behavioral Sciences*, vol. 97, pp. 320–324, 2013.

[19] R. Asif, A. Merceron, S. A. Ali, and N. G. Haider, "Analyzing undergraduate students' performance using educational data mining," *Computers & Education*, vol. 113, pp. 177–194, 2017.

[20] E. B. Costa, B. Fonseca, M. A. Santana, F. F. de Araújo, and J. Rego, "Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses," *Computers in Human Behavior*, vol. 73, pp. 247–256, 2017.

[21] C. Angeli, S. K. Howard, J. Ma, J. Yang, and P. A. Kirschner, "Data mining in educational technology classroom research: Can it make a contribution?" *Computers & Education*, vol. 113, pp. 226–242, 2017.

[22] M. Misuraca, G. Scepi, and M. Spano, "Using Opinion Mining as an educational analytic: An integrated strategy for the analysis of students ’ feedback," *Studies in Educational Evaluation*, vol. 68, pp. 100 979.1–100 979.9, 2021.

[23] J. Akoka, I. Comyn-Wattiau, N. Prat, and V. C. Storey, "Knowledge contributions in design science research: Paths of knowledge types," *Decision Support Systems*, vol. 166, pp. 113 898.1–113 898.14, 2023.

[24] A. A. Mukherjee, A. Raj, and S. Aggarwal, "Identification of barriers and their mitigation strategies for industry 5.0 implementation in emerging economies," *International Journal of Production Economics*, vol. 257, pp. 108 770.1–108 770.15, 2023.