Book Recommendation to Wikipedia Article Readers in a University Library

Keita Tsuji *

Abstract

Wikipedia has emerged as an important source of information for university students. It has been reported that university students tend to read and do research with Wikipedia articles more than they do with books, even when within a library. To encourage students to read library books as a more reliable source of information, a library system was developed for recommending library books to Wikipedia readers within a particular university library. The proposed system assigns a Nippon Decimal Classification (NDC) category to each Wikipedia article and recommends library books in the same NDC category to readers of the article. The recommended books are displayed to the reader with their covers and call numbers within that library. In the test implementation of the system, the precision of assigning NDC categories to Wikipedia articles using a convolutional neural network was as high as 87.4%, while the precision of selecting books for recommendation using a support vector machine reached 99.8%. In addition, 82.0% of the student subjects found at least one book that "I want to read" among three recommended books.

Keywords: Book recommendation, convolutional neural network, university library, Wikipedia

1 Introduction

Wikipedia has become an important source of information for university students. It has been reported that university students tend to read and do research with Wikipedia articles than with books, even when within a library [1]. Within this context, a system was developed for recommending library books to Wikipedia readers when they are within a university library. The system is addable to the Web browser of a library desktop PC. The recommended books are displayed to the Wikipedia article reader with their covers and call numbers within that library. The system is aimed at encouraging students to read library books as a more reliable source of information rather than relying on Wikipedia articles. Although the demonstration of the proposed system in the present study considered readers of Japanese Wikipedia articles in Japanese university libraries, the system is applicable to any language.

The methodology of the proposed library system involves the following three steps:

* University of Tsukuba, Ibaraki, Japan

(1) Assigning a Nippon Decimal Classification (NDC) category[2] to each Japanese Wikipedia article.

(2) Selecting books to recommend to readers of each article.

(3) Obtaining the user's evaluation of the recommended books.

In the context of these three steps, the present study considered three questions: (a) how can NDC categories be automatically assigned to Wikipedia articles? (b) how can books of the same NDC categories as Wikipedia articles be selected? and (c) how can users evaluate the selected books when they are recommended to them?

The reason for assigning NDC categories to Wikipedia articles is as follows. As would be further discussed, 1,070,202 Japanese Wikipedia articles were considered in this study, and the average number of printed books held by Japanese university libraries in 2017 was approximately 227,244 (= 323,595,000 books divided by 1,424 libraries reported by the Japan Library Association in 2018 [2]). If all the books in a library are considered as potential candidates for recommendation for each Wikipedia article, the total possible combination of articles and books would be 1,070,202 × 227,244. A system based on this figure would require a time-consuming and unrealistic operation. It thus becomes rational to adopt a process for reducing the number of candidate books for a particular article. The NDC is very handy for this purpose. Almost all Japanese books in Japanese libraries are assigned an NDC category as part of their respective call numbers. If NDC categories are similarly assigned to Wikipedia articles based on their content and only books of the same category are considered as recommendation candidates for a particular article, the number of possible combinations of books and articles would be significantly reduced. For instance, if a Wikipedia article is assigned NDC category 324 (i.e., *Civil Law*), only books in the same NDC category would be considered as recommendation candidates for that article.

A convolutional neural network (CNN) and support vector machine (SVM) were respectively used to answer the above-mentioned questions (1) and (2) in the development of the proposed system, namely, to automatically assign NDC categories to Wikipedia articles and to select books for recommendation to readers of a particular article. These two tools are representative machine learning methods for text classification [3][4][5][6]. Regarding question (3) above, the student subjects were asked to evaluate the books recommended to them by the proposed system.

There have been some previous works on book recommendation to library users. Mikawa et al. [7] proposed the use of silhouette images captured by cameras installed in libraries and recommended books based on the gender and age of the library users. Jomsri [8] proposed the use of an association rule based on the faculty and profile of the user, their book loaning data, and book categories. The target users of the system are somewhat similar to those of the present system, although the present target users are further narrowed to readers of Wikipedia articles in university libraries. In addition, the information used for book recommendation in the present system mainly consists of the words in the Wikipedia article. This particularly distinguishes the system from those of Mikawa et al. [7] and Jomsri [8]. Other works on book recommendation systems include those of Mooney & Roy [9], Givon & Lavrenko [10], Yang et al. [11], Pera et al. [12], Crespo et al. [13], Vaz et al. [14],

---

[2] NDC is a Japanese library classification system similar to DDC (Dewey Decimal Classification). Its notation is based on three digits, which systematically represent the main class, division, and section, respectively. For example, in NDC category 324, 3, 32, and 324 indicate *Social Sciences*, *Law*, and *Civil Law*, respectively.

Benkoussas & Bellot [15], Pathak et al. [16], Vaz et al. [17], Garrido et al. [18], Pera & Ng [19], Pera & Ng [20], Priyanka et al. [21], Rajpurkar et al. [22], Sase et al. [23], Bhosale et al. [24], Gao et al. [25], Sohail et al. [26], Alharthi et al. [27], Parekh et al. [28], Sohail et al. [29], Thanapalasingam et al. [30], Tsuji et al. [31], Tsuji et al [32] and Tsuji et al. [33]. However, the aims and target users of the systems developed in these previous works differ from those of the present study.

The author has previously investigated the use of books cited in Wikipedia articles and their NDC categories in making book recommendations to readers [34], and attempted to assign NDC categories to Wikipedia articles based on their setting titles, categories, and main texts using one single CNN channel [35].

2 Method

This section explains some key aspects of the proposed library book recommendations system. The first aspect is how Wikipedia articles are accessed and the distributed representation of nouns in them are determined. The second is how an NDC category is assigned to an article. The third is how books are selected for recommendation to an article reader. The fourth is obtaining the user's evaluation of the recommended books. It should be ehphasized that the objective of the second aspect above is to reduce the computational cost of the third aspect.

2.1 Accessing Wikipedia Articles and Determining the Distributed Representation of Nouns

The steps below were used to access Wikipedia articles and determine the distributed representation of nouns.

(a-1) The Japanese Wikipedia dump as of August 1, 2017 was downloaded from https://dumps.wikimedia.org/jawiki/. The file size was approximately 14 GB.

(a-2) Pages (i) (ns ≠ 0, indicating that it is not a "Main/Article") and (ii) (it is only a redirection page) of the articles were removed from the download (here, ns is the namespace tag). The number of pages that remained was 1,070,202, which are implied in subsequent references to Wikipedia articles.

(a-3) Mecab ver. 0.996 [36] with the mecab-ipadic-NEologd dictionary (as of August 10, 2017) [37] was used for a morphological analysis of the downloaded Wikipedia articles. A total of 5,512,620 types of nouns were identified in the articles.

(a-4) Word2vec [38] in gensim [39] was used to obtain a 200-dimensional "distributed representation" of each noun in the Wikipedia articles. The parameters *window* and *min_count* in the Word2Vec module of gensim were set to 5 and 10, respectively.

2.2   Assigning NDC Categories to Each Wikipedia Article

This subsection explains the development of the training and testing data that were used to assign NDC categories to the Wikipedia articles. The data were inputted to the CNN and SVM.

2.2.1 Development of Training and Testing Data

The steps below were used to develop the training and testing data that were used to assign NDC categories to the Wikipedia articles.

(b-1) From the Wikipedia articles extracted in Step (a-2) above, the articles that contained "References" and cited books with their ISBNs were extracted. There were 50,375 such articles, including 45,151 unique ISBNs and 117,219 books (including duplicates). Only books with specified ISBNs were considered because this enabled obtaining the book bibliographies without ambiguity in Step (b-2). A procedure that utilizes book titles or author names is likely to produce differing bibliographies.

(b-2) Based on the identified book ISBNs, the unique and consistent bibliographies of the cited books were obtained using the OpenSearch tool of the National Diet Library of Japan [40], which is the largest and most comprehensive library in Japan. Bibliographies that did not contain three-digit NDCs (for instance, the three digits such as 324 indicate the book's main class 3, division 2, and section 4) or book titles were excluded. Overall, 40,647 bibliographies were obtained.

(b-3) The most popular NDC categories among the books cited in each article were identified. Regarding the level or depth of the NDC categories, the main class and division of the books were employed. For example, if an article cited five books with NDC categories 324, 324, 324, 325, and 369, respectively, the NDC category 32 (main class 3 and division 2) was considered to be the most popular, having a frequency of four. Incidentally, NDC category 32 and 36 indicate *Law* and *Society*, respectively.

(b-4) An article was extracted if the frequency of the most popular NDC category was ≥3 and its ratio relative to the total number of categories were ≥0.8. The threshold values of 3 and 0.8 were chosen empirically. Thus, the example article mentioned in Step (b-3) was extracted because the frequency of the NDC category 32 is 4 (> 3) and its ratio is 0.8 (= 4/5). In this way, 5,385 articles were extracted. This set of articles is henceforth referred to as the LC set.

(b-5) It was assumed that the most popular NDC category in an article in the LC set was the one that should be assigned to that article. For instance, the NDC category 32 should be assigned to the example article in Step (b-4). Accordingly, 5,385 pairs of the above-mentioned articles and NDC categories were extracted from the LC set. These were used as training and testing data for the CNN and SVM, as will be explained in the next subsection. The number of NDC categories that were to be assigned to the LC set of articles are presented in Table 1, where C, N, and T respectively represent the NDC category, number of articles, and their respective total values. It can be seen from Table 1 that, for instance, NDC category 32 should be assigned to 107 articles in the LC set. In other words, 107 articles were obtained for the training and testing of the CNN for

NDC category 32. Although the CNN and SVM could not be assigned NDC categories that were not included in the training data, these were somewhat exceptional cases. Actually, only 16 NDC categories (05, 06,…97) were not included in the LC set.

Table 1. Numbers of NDC Categories to be Assigned to Articles in LC Set

C	N	C	N	C	N	C	N	C	N	C	N	C	N	C	N	C	N	C	N
00	32	10	1	20	4	30	5	40	6	50	8	60	0	70	11	80	4	90	4
01	22	11	1	21	754	31	36	41	139	51	13	61	7	71	0	81	21	91	272
02	10	12	14	22	74	32	107	42	37	52	26	62	2	72	118	82	1	92	8
03	1	13	14	23	114	33	42	43	11	53	57	63	0	73	0	83	2	93	9
04	2	14	52	24	5	34	0	44	13	54	32	64	44	74	14	84	0	94	10
05	0	15	3	25	6	35	0	45	142	55	234	65	20	75	45	85	0	95	6
06	0	16	53	26	0	36	46	46	17	56	4	66	6	76	190	86	0	96	0
07	4	17	17	27	1	37	23	47	114	57	2	67	20	77	576	87	0	97	0
08	0	18	79	28	145	38	73	48	279	58	8	68	101	78	330	88	0	98	1
09	0	19	60	29	159	39	62	49	219	59	81	69	7	79	90	89	1	99	2
T	71	T	294	T	1,262	T	394	T	977	T	465	T	207	T	1,374	T	29	T	312

(b-6) The above-mentioned 5,385 pairs were randomly divided into 4,835 training data and 500 testing data. The former was used to train the CNN and SVM to assign NDC categories to each Wikipedia article, while the latter was used to compare the performances of the CNN and SVM. The better of CNN and SVM was then used to assign NDC categories to all the Wikipedia articles. As will be discussed in subsection 2.3, the assigned NDC categories were used to select books for recommendation to the Wikipedia article readers.

2.2.2 Input to CNN

The input to the employed CNN utilized three channels, which correspond to the title, category, and main text of the Wikipedia articles. The structures of the channels and how the input was implemented are discussed below.

Title Channel

First, the nouns in the title of each Wikipedia article in the LC set were extracted and represented by their distributed representation (i.e., 200-dimensional vectors obtained through Step (a-4) in subsection 2.1). The vectors and other zero vectors constitute a 50 (rows) by 200 (columns) matrix, referred to as the title channel for the CNN. In the title channel matrix, the i-th row contains (a) the 200-dimensional vector for the i-th noun in the title, when $i \leq 5$ (the threshold was chosen empirically), and (b) a zero vector, when $i > 5$. If the number of nouns in a certain title is less than five, the corresponding row would contain a zero vector. If the number of nouns is greater than five, only the first five nouns would be adopted in the title channel. This matrix is henceforth referred to as *tl*.

Category Channel

The first five categories in each LC set article were extracted and represented by a 50×200 matrix. The $\{(i-1)*10+j\}$-th row of the matrix contained the 200-dimensional vector (obtained by Step (a-4) in subsection 2.1) for the j-th noun in the i-th category ($i \leq 5$ and $j \leq 5$). The threshold 5 was chosen empirically. The other rows contained zero vectors. If the number of nouns in a certain category was less than five, the corresponding row was set to a

zero vector. If the number of nouns was greater than five, only the first five nouns in each category was adopted. This matrix is henceforth referred to as *cg*.

Main Text Channel

Finally, ten nouns within the main text of each LC set article with the highest TF-IDFs (term frequency-inverse document frequency) were extracted. The TF-IDF of noun X in article Y is defined as "frequency of X in the main text of Wikipedia article Y" multiplied by "log (number of Wikipedia articles/number of articles with main text containing X)." These were represented by a 50×200 matrix. The $\{(i-1)*10+1\}$-th row contained the 200-dimensional vector (obtained through Step (a-4) in subsection 2.1) of the noun whose TF-IDF was the i-th highest in the main text ($i \le 10$). The other rows were set to zero vectors. This matrix is henceforth referred to as *tx*.

Hence, *tl*, *cg*, and *tx* constituted the input channels to the present CNN. All seven possible combinations of these channels were tested, namely, (1) *tl*, (2) *cg*, (3) *tx* (i.e. only one channel), (4) *tl* and *cg*, (5) *tl* and *tx*, (6) *cg* and *tx* (two channels), and (7) *tl*, *cg*, and *tx* (all three channels).

A pseudo example of a category channel is shown in Figure 1. The title and main text channels can be represented similarly.

1.23	3.54	2.17	— — — — — — —	1.62	1st row
0.24	2.46	−1.38	— — — — — — —	3.21	2nd row
2.15	4.32	3.21	— — — — — — —	1.12	3rd row
3.78	−0.51	1.84	— — — — — — —	5.43	4th row
1.62	1.11	−2.57	— — — — — — —	1.78	5th row
0.00	0.00	0.00	— — — — — — —	0.00	6th row
0.00	0.00	0.00	— — — — — — —	0.00	7th row
0.00	0.00	0.00	— — — — — — —	0.00	8th row
0.00	0.00	0.00	— — — — — — —	0.00	9th row
0.00	0.00	0.00	— — — — — — —	0.00	10th row
3.75	−1.62	2.31	— — — — — — —	2.14	11th row
1.48	2.03	−1.52	— — — — — — —	4.41	12th row
5.15	−3.32	1.21	— — — — — — —	3.62	13th row
3.78	−0.51	2.84	— — — — — — —	1.53	14th row
1.62	1.11	3.57	— — — — — — —	2.73	15th row
0.00	0.00	0.00	— — — — — — —	0.00	16th row
0.00	0.00	0.00	— — — — — — —	0.00	17th row
0.00	0.00	0.00	— — — — — — —	0.00	18th row
0.00	0.00	0.00	— — — — — — —	0.00	19th row
0.00	0.00	0.00	— — — — — — —	0.00	20th row
2.73	3.51	4.21	— — — — — — —	3.42	21th row
1.36	4.01	−1.37	— — — — — — —	2.31	22th row
....	....	....	....	....	—
0.00	0.00	0.00	— — — — — — —	0.00	50th row

Figure 1. Pseudo Example of Category Channel *cg*

2.2.3 CNN Settings

The CNN is well-known for its excellent ability for visual recognition. The above-mentioned matrices can be considered as two-dimensional (2D) images. The present CNN utilized five kinds of filters of heights 1, 2, 3, 4, and 5, respectively. All the filters had a width of 200. Furthermore, for each filter size, six different numbers of filters were used in the tests, namely, 50, 100, 150, 200, 250, and 300.

The Tensorflow ver. 1.0 [41] was used to create and train the CNN. The *strides* and *padding* of the *tf.nn.conv2d* function were set to [1, 1, 1, 1] and *VALID*, respectively. Rectified linear units (*tf.nn.relu*) were used after *tf.nn.conv2d*, and max pooling was

employed. The *ksize*, *strides*, and *padding* of the *tf.nn.max_pool* function were set to [1, IH-FH+1, 1, 1], [1, 1, 1, 1], and *VALID*, respectively, where IH and FH respectively represent the heights of the input channel and filter. The softmax function was used in the final output layer. The size of the mini batches, number of epochs, dropout rate, and learning rate were 100, 300, 0.5, and 0.0001, respectively. The data flow is shown in Figure 2. The pseudo example of the channel in the left is presented in Figure 1.
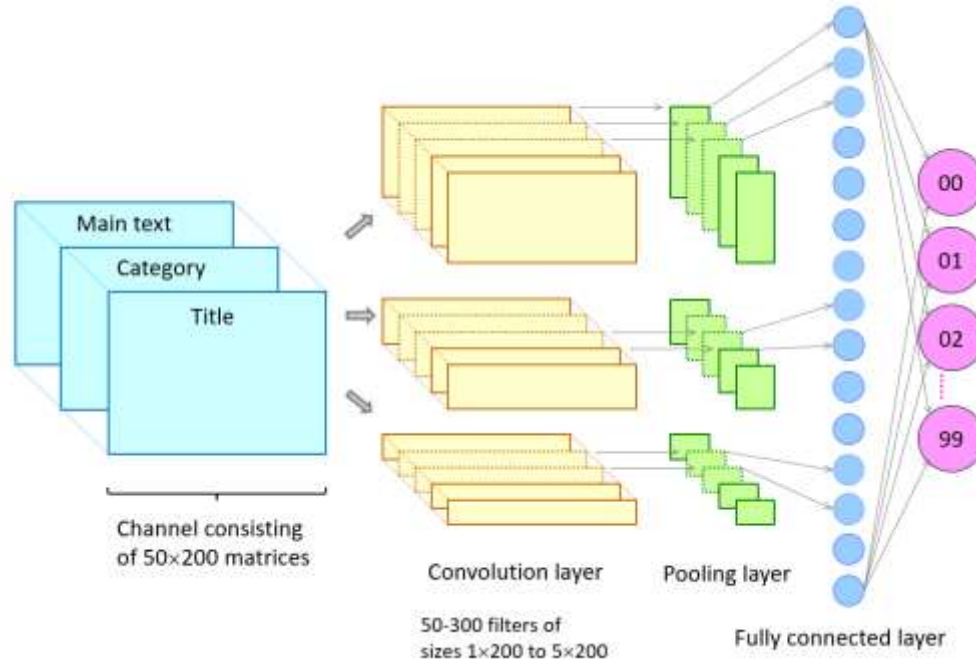

Figure 2. Data Flow of the Developed CNN

2.2.4 Input to SVM

The channels mentioned in the previous subsections were converted into the vector *V*, where the *V*'s $\{(i-1)*200+j\}$-th element is the (i, j) element of the channel and was inputted to the SVM. When multiple channels were used for the input to the SVM in this study, the corresponding vectors were concatenated for the input. For instance, when all the three channels (title, category, and main text) were used, the dimension of the input vector was 30,000 ($= 50 \times 200 \times 3$).

2.2.5 SVM Settings

The LIBSVM ver. 3.23 [42] was used with the SVM to assign NDC categories to the Wikipedia articles. The radial basis function (RBF) kernel of the SVM was employed for this purpose. The easy.py of the LIBSVM package was used to determine the optimal values of the parameters *C* and γ .

2.3 Selecting Books to Recommend

This subsection describes the selection of books to recommend to a Wikipedia article reader. The training and testing data were generated and inputted to the CNN and SVM.

2.3.1 Generation of Training and Testing Data

The following steps were used to generate the training and testing data for selecting books to recommend to Wikipedia article readers.

(c-1) As mentioned in Section 1, the aim of the present study was the development of a system for recommending library books to students who read Wikipedia articles in university libraries. The experimental implementation of the system in this study was conducted at T University.[3] There were 621,129 candidate books for recommendation in the library, the books having been borrowed at least once between January 2, 2006 and March 31, 2017. It was assumed that books that had not been borrowed in more than 10 years would not be suitable for recommendation. The exclusion of such books also had the additional benefit of reducing the computational cost of the system.

(c-2) The distribution of the NDC categories of the 621,129 candidate books is presented in Table 2, where C and N represent NDC category and number of corresponding samples, respectively. It can be seen from the tTable 2 that, for instance, the number of books with NDC category 32 is 21,276. Incidentally, the 621,129 candidate books had 3,830,241 loan records.

Table 2. Distribution of NDC Categories of Books Borrowed from T University Library

C	N	C	N	C	N	C	N	C	N	C	N	C	N	C	N	C	N	C	N
00	14,544	10	3,891	20	5,244	30	10,399	40	6,088	50	8,770	60	4,060	70	10,521	80	8,081	90	3,746
01	8,562	11	2,388	21	21,987	31	18,177	41	26,147	51	8,063	61	7,719	71	1,685	81	7,188	91	19,697
02	3,584	12	5,592	22	8,877	32	21,276	42	11,020	52	8,751	62	1,769	72	15,958	82	2,886	92	5,084
03	572	13	10,362	23	4,356	33	25,639	43	7,002	53	2,774	63	146	73	775	83	4,507	93	5,152
04	827	14	13,305	24	1,355	34	2,696	44	1,495	54	8,672	64	1,011	74	1,774	84	809	94	2,111
05	687	15	1,417	25	1,141	35	2,482	45	6,243	55	569	65	2,391	75	4,135	85	2,330	95	2,538
06	754	16	4,567	26	418	36	25,711	46	8,226	56	1,329	66	617	76	2,317	86	147	96	172
07	1,088	17	1,423	27	225	37	36,168	47	2,089	57	1,764	67	4,002	77	2,920	87	86	97	116
08	24,895	18	4,694	28	4,383	38	11,024	48	2,847	58	1,297	68	2,057	78	15,074	88	675	98	687
09	1,761	19	3,750	29	4,740	39	1,399	49	43,479	59	717	69	793	79	692	89	611	99	388
T	57,274	T	51,389	T	52,726	T	154,971	T	114,636	T	42,706	T	24,565	T	55,851	T	27,320	T	39,691

(c-3) The author, together with others, has previously reported that the recommendation of frequently borrowed books produces better results [33]. To promote the recommendation of such books, the loan frequencies of the library books were adopted as a feature in the CNN and SVM, specifically the number of times that each book had been borrowed between January 2, 2006 and March 31, 2017. This parameter is henceforth referred to as *N*. Each book was then assigned a value $\ln(N/Y+1)$, where *Y* is (a) 2018 minus the publication year of the book, if the book was published in or after 2006, or (b) 12, if the book was published before 2006. For instance, $\ln(20/4+1) = \ln(6)$ was assigned to a book published in 2014 and had been borrowed 20 times during the above-mentioned period. Henceforth, $\ln(N/Y+1)$ is referred to as *LF*. The *LF* is the

---

[3] This name is used for anonymity.

natural logarithm of a rough approximation of the number of times the book had been borrowed per year during the above-mentioned period. The initial idea was to use N/Y, but the above natural logarithm seemed to be more effective in the pilot experiment. Further study is planned to confirm this observation based on the responses of the subjects to questions presented to them.

(c-4) The sets $\{a_i\}$, $\{x_i\}$, $\{y_i\}$, and $\{z_i\}$ were defined as distributed representations (the 200-dimensional vectors obtained in Step (a-4) in subsection 2.1) of the first five nouns in the book titles (A), the first five nouns in the titles of the Wikipedia articles (X), the first five nouns in the first five categories of the Wikipedia articles (Y), and ten nouns in the main texts of the Wikipedia articles (Z) with the highest TF-IDFs, respectively. Calculations were then implemented to obtain a set of values $\{\text{dot}(a_i, x_j)\}_{i,j}$, which were arranged in descending order, where $\text{dot}(p, q)$ is the inner product of vectors p and q. If the number of values was less than a threshold M, 0's were appropriately added to the end of the sequence. This sequence of values (i.e., a vector) is henceforth referred to as S_{AX}. The vectors S_{AY} and S_{AZ} were similarly obtained from $\{\text{dot}(a_i, y_j)\}_{i,j}$ and $\{\text{dot}(a_i, z_j)\}_{i,j}$, respectively.

(c-5) Pairs of Wikipedia articles and the books they cited were extracted as training data (i.e., pairs of Wikipedia articles and books that should/should not be recommended to the article readers). More concretely, from the combination of 50,375 articles and 117,219 books they cited (mentioned in (b-1) in subsection 2.2), pairs of articles and books were extracted if they satisfied all the following conditions: (1) the assigned NDC categories of the former and the NDC categories of the latter were identical, (2) *LF* of the books in Step (c-3) were no less than ln2 (implying that the books were borrowed at least once a year on average), and (3) the maximum values of the elements of S_{AX}, S_{AY}, and S_{AZ} in Step (c-4) were ≥0.9. The threshold 0.9 was chosen heuristically through preliminary experiments. The pairs were regarded as positive examples (i.e., examples of Wikipedia articles that the corresponding books should be recommended for). A total of 5,444 such pairs of articles and books were extracted. Among these were 3,943 articles, which implies that some articles were paired with more than one book. Articles on *Leukemia* and *Galois Theory* were the most represented and appeared in 14 pairs.

(c-6) For each of the 3,943, the books were chosen when (1) their NDC categories were identical to the NDC categories assigned to the articles and (2) they were not cited in the articles. For instance, when a certain article appeared in three positive examples, three books with the same NDC category as that assigned to the article and which were not cited in the article were randomly chosen. In this way, another 5,444 pairs of articles and books were generated and regarded as negative examples. A total of 10,888 training data was thus obtained.

2.3.2 Input to and Settings of CNN and SVM

The scalar *LF* in Step (c-3) and vectors S_{AX}, S_{AY}, and S_{AZ} in Step (c-4) were concatenated and inputted to the CNN and SVM. Regarding M in Step (c-4), values of 10, 20, and 30 (inner products of vectors) were considered in the tests. Therefore, for example, for $M = 30$, a vector of length 91 (= 1 + (30 × 3)) was inputted to the CNN and SVM. In this case, the number of the CNN input channels was one, and the height and width of the filter were 1 and

1+M × 3, respectively. The tests considered 50, 100, 200, and 400 filters, respectively. Regarding the SVM, the easy.py was once again used to determine the optimal values of C and γ for the radial basis function kernel.

In the tests, the present CNN (or SVM) was used to assign the NDC category (main class and division) of each Wikipedia article, as described in subsection 3.2. Subsequently, for each Wikipedia article, all the books in the T University Library with NDCs identical with that of the article were identified. For example, for a Wikipedia article with an assigned NDC of 32, all the books in the T University Library with an NDC of 32 were identified. The three books with the highest probabilities in the CNN and SVM were then recommended for that article.

2.4   Obtaining User Evaluation to Recommended Books

To examine the validity of the proposed system, it was implemented as a Google Chrome Extension[4] and the student subjects were requested to evaluate the books recommended to them by the system. Figure 3 shows an example of the image display of the three books recommended for the Wikipedia article *Data Mining*. These books were displayed as a part of the Wikipedia article at the end of its first <h2> tags.

The tests utilized 10 student subjects, who were third-year information science major undergraduate students registered at the T University Library. Further study is planned to use more subjects and those in other years and faculties, as well as graduate students. Each student was requested to indicate five Wikipedia articles related to their study. Two variants of the proposed system was then used to recommend three books for each article. The two system variants were (1) the system that uses the SVM with $M = 10$ (this value produces the best results, as will be discussed below) and does not show the loan frequencies of each book, and (2) the system that also uses the SVM and shows the loan frequencies. The students were asked to evaluate the books recommended by the two system variants, respectively. If the evaluations of the same books changed with the knowledge of the loan frequencies, the loan frequency was determined to be a factor of the evaluation.

The system displayed the book cover images if they were available in OpenBD [43], which provides free book cover images. It was expected that the display of the book cover would attract a positive evaluation by the students because the images gave the books a kind of *reality*. As will be discussed in subsection 3.3, cover images were available for 88 out of the 150 books that were recommended in the test implementation of the proposed system. Evaluations for both sets of books were compared to assess the effectiveness of displaying book covers.

Each student was requested to evaluate a total of 30 books (5 articles × 3 books × 2 system variants) by choosing one out of the following: (a) I want to read this book very much, (b) I want to read this book if possible, (c) I do not want to read this book if possible, and (d) I do not want to read this book at all.

---

[4] The proposed system (the version that displays loan frequencies) is available at
  https://chrome.google.com/webstore/detail/bookrecommendationchromea/fmndigflkimlc
  nobibkcapjgbhpoglnc.

Figure 3. Example Display of Cover Images of Recommended Books

3 Results

This section presents the results of (1) the assignment of NDC categories to Wikipedia articles, (2) the selection and recommendation of books to articles readers, and (3) the evaluation of the recommended books by articles readers.

3.1 Assigning NDC Categories to Wikipedia Articles

The precision of the NDC category assignment to the Wikipedia articles was defined as the ratio of the number of articles to which correct categories were assigned to the total number of articles, expressed as a percentage. The CNN-based NDC category assignment results are presented in Table 3, where *tl*, *cg*, ..., *tl+cg+bd* respectively represent channel combinations (1)–(7) mentioned in the last paragraph of subsection 2.2.2. It can be seen from the tTable 3 that, for example, when 100 filters and only the titles of the Wikipedia articles (i.e., *tl*) are used, the NDC category assignment precision is 49.0%. The use of article titles, categories, and main texts with 250 filters produces the highest precision of 87.4%.[5] It is also obvious that the use of more filters and channels does not necessarily improve the results, owing to the problem of overfitting.

---

[5] As mentioned in Section 1, Tsuji (2017) attempted to assign NDC categories to Wikipedia articles by setting the titles, categories, and main texts in a single channel of the CNN. For performance comparison, the NDC categories were assigned in the same way in the present study. The highest precision achieved in the previous work was 86.6%, which is lower than that of the present study.

Table 3. Precision of NDC Category Assignment Using CNN

	Number of Filters					
	50	100	150	200	250	300
tl	47.4	49.0	47.4	48.6	49.4	49.2
cg	73.4	67.6	67.8	69.2	63.4	68.0
bd	75.2	80.0	80.0	80.8	80.4	80.8
tl+cg	73.6	75.2	73.0	74.0	75.0	77.2
tl+bd	80.2	80.8	82.0	82.4	81.2	82.2
cg+bd	84.8	84.4	84.8	85.8	86.0	86.2
tl+cg+bd	85.8	85.0	86.2	86.0	87.4	85.4

Table 4 presents the SVM-based NDC category assignment results for the LC set of Wikipedia articles. It can be seen from the Table 4 that, for example, the use of the distributed representation of only the titles produces a precision of just 67.2%, corresponding to *C* and γ (subsection 2.2.5) values of 128 and 0.031250000, respectively. The highest precision of 84.4% is achieved using the distributed representations of the titles, categories, and main texts, with *C* and γ values of 512 and 0.000122070. This precision is lower than the highest value of 87.4% achieved by the CNN (Table 3).

Table 4. Precision of NDC Category Assignment Using SVM

	Precision	C	γ
tl	67.2	128	0.031250000
cg	79.3	32	0.001953125
bd	79.1	8	0.001953125
tl+cg	80.6	128	0.001953125
tl+bd	80.1	512	0.000488281
cg+bd	84.0	512	0.000003052
tl+cg+bd	84.4	512	0.000122070

Based on the above results, the developed CNN was used to assign NDC categories to all the Wikipedia articles using their titles, categories, and main texts and 250 filters. The number of Wikipedia articles with respect to the NDC category obtained by this means is presented in Table 5. Here, C and N represent the NDC category and number of articles, respectively. It can be seen that, for example, the number of Wikipedia articles assigned NDC category 32 is 9,447.

Table 5. Number of Wikipedia Articles with Respect to Assigned NDC Category

C	N	C	N	C	N	C	N	C	N	C	N	C	N	C	N	C	N	C	N	C	N
00	14,434	10	0	20	181	30	3,373	40	963	50	1,522	60	0	70	2,763	80	1,229	90	1,292
01	1,892	11	73	21	68,066	31	12,702	41	7,338	51	2,120	61	23,774	71	0	81	4,614	91	23,539
02	1,735	12	1,945	22	27,584	32	9,447	42	4,308	52	9,379	62	0	72	29,984	82	0	92	1,430
03	0	13	2,594	23	41,070	33	11,263	43	4,570	53	15,671	63	0	73	0	83	1,157	93	4,007
04	98	14	3,599	24	808	34	0	44	8,123	54	13,704	64	4,506	74	2,589	84	0	94	1,988
05	0	15	0	25	902	35	0	45	7,352	55	13,736	65	212	75	1,806	85	0	95	0
06	0	16	3,787	26	0	36	10,642	46	1,841	56	210	66	3	76	95,971	86	0	96	0
07	304	17	4,494	27	0	37	34,331	47	5,493	57	83	67	5,490	77	101,957	87	0	97	0
08	0	18	9,578	28	4,410	38	4,880	48	10,752	58	963	68	49,171	78	127,860	88	0	98	0
09	0	19	7,197	29	92,185	39	32,884	49	19,692	59	5,668	69	22,049	79	22,865	89	0	99	0
T	18,463	T	33,267	T	235,206	T	119,522	T	70,432	T	63,056	T	105,205	T	385,795	T	7,000	T	32,256

3.2 Selecting Books for Recommendation

The precisions of the CNN and SVM for classifying books to recommend or not recommend are presented in Table 6 and Table 7. A comparison of the tables reveals that the SVM produces better results, with its precision reaching 99.8%. The SVM was thus used to select books for recommendation to the Wikipedia articles readers. The three books with the highest probabilities of belonging to the group of books to be recommended for each Wikipedia article were selected and recommended to the subjects of the test implementation of the propose system.[6] [7]

Table 6. Precision of Book Classification Using CNN

M	Number of Filters			
	50	100	200	400
10	98.3	98.4	98.3	98.4
20	98.4	98.7	98.6	98.5
30	98.6	98.6	98.6	98.5

Table 7. Precision of Book Classification Using SVM

M	Precision	C	γ
10	99.8	512	0.03125
20	99.8	32	0.12500
30	99.7	32	0.12500

3.3 Obtaining User Evaluation of Recommended Books

The results of the student subjects' evaluation of the recommended books are presented in Tables Table 8 and Table 9, which correspond to when the loan frequencies of the books were shown and not shown, respectively. In the tables, "evaluation," N, and "ratio" respectively represent the evaluation response, the number of books to which that response was given, and the ratio of that number to the total number of book (150 = 10 students × 5 Wikipedia articles × 3 books), respectively. It can be seen from the tables that the ratios of books that the subjects evaluated as "I want to read this book very much" and "I want to read this book if possible" are 47.3% (= 15.3% + 32.0%) and 45.3% (= 18.0% + 27.3%), respectively. Although not shown in the tables, the subjects found at least one book that "I want to read" among the three recommended books for 41 out of 50 (= 10 students × 5 articles) Wikipedia articles when the loan frequencies were shown. In other words, the proposed system provided a useful book with a rate of 82.0%.[8]

---

[6] When the probabilities were higher than 50%, the books were judged by the CNN and SVM as books that should be recommended. When the probabilities were lower than 50%, the books were judged by the CNN and SVM as books that should not be recommended. The above-mentioned precision was calculated based on the respective numbers of these books.

[7] The option -b was used to show the above-mentioned probabilities in LIBSVM.

[8] When the loan frequencies were not shown, the ratio was 40 out of 50 articles (i.e., 80%).

Table 8. Evaluation of Recommended Books When Loan Frequency is Shown

Evaluation	N	Ratio
(1) I want to read this book very much	23	15.3
(2) I want to read this book if possible	48	32.0
(3) I do not want to read this book if possible	41	27.3
(4) I do not want to read this book at all	38	25.3
Total	150	100.0

Table 9. Evaluation of Recommended Books When Loan Frequency is Not Shown

Evaluation	N	Ratio
(1) I want to read this book very much	27	18.0
(2) I want to read this book if possible	41	27.3
(3) I do not want to read this book if possible	39	26.0
(4) I do not want to read this book at all	43	28.7
Total	150	100.0

Although there are slight differences between the results when the loan frequencies are shown and not shown, they are not statistically significant. However, it should not be concluded that the display of the loan frequencies does not affect the evaluation. It was examined whether extremely low or high loan frequencies affected the evaluation. The results are shown in Table 10, where "up" and "down" indicate that the student increased and decreased the evaluation of the same book, respectively (e.g., changed from "I do not want to read this book if possible" to "I want to read this book if possible") after seeing the loan frequency. It can be seen from the Table 10 that, for example, while only one student increased their book evaluations after seeing the loan frequencies, the same caused four students to decrease their evaluations. Conversely, more students (four vs. one) increased their evaluations after seeing that the loan frequencies of the books were ≥40. Students may be attracted to popular books (in the sense that they are frequently borrowed books). This is consistent with the so-called Matthew effect ("the rich get richer and the poor get poorer") proposed by Merton [44]. Showing loan frequencies that are up to 40 would thus be effective for increasing the interest of students in the books. The display of low loan frequencies should be avoided so that they do not discourage interest in such books.

Table 10. Change in Book Evaluation with Display of Loan Frequency

		Loan Frequencies					
		1	2	3	20–29	30–39	40+
Evaluation	Up	1	0	1	5	8	4
	Down	4	3	1	4	4	1

The evaluations of the students when the book cover images were displayed and not displayed are presented in Table 11 and Table 12, respectively. It can be seen from these tables that the ratio of evaluation (1) significantly decreases when the book cover images are not displayed (from 22.7% to 4.8%), while the ratio of evaluation (4) increases (from 21.6% to 30.6%). It can thus be said that the book cover images increase the interest of the students, at least the ratio of evaluation (1). It would, however, be useful to further investigate the effects of particular images in this regard.

Table 11. Change in Book Evaluation with Display of Book Cover

Evaluation	N	Ratio
(1) I want to read this book very much	20	22.7
(2) I want to read this book if possible	22	25.0
(3) I do not want to read this book if possible	27	30.7
(4) I do not want to read this book at all	19	21.6
Total	88	100.0

Table 12. Change in Book Evaluation with Display of Book Cover

Evaluation	N	Ratio
(1) I want to read this book very much	3	4.8
(2) I want to read this book if possible	26	41.9
(3) I do not want to read this book if possible	14	22.6
(4) I do not want to read this book at all	19	30.6
Total	62	100.0

The student evaluation with respect to NDC category is presented in Table 13. It is noteworthy that the ratio of evaluations (1) or (2) for NDC categories 40–49 is relatively low. This may be due to the fact that only one such article was among those used for the study.

Table 13. Book Evaluation with Respect to NDC Category

NDC	Evaluation	N	Ratio
00–09	(1) or (2)	34	49.3
10–19	(1) or (2)	4	44.4
20–29	(1) or (2)	6	66.7
30–39	(1) or (2)	5	41.7
40–49	(1) or (2)	1	16.7
50–59	(1) or (2)	6	40.0
60–69	(1) or (2)	4	66.7
70–79	(1) or (2)	11	45.8
80–89	(1) or (2)	0	—
90–99	(1) or (2)	0	—

4   Discussion

This section considers some potential error sources in the present work and how such may be addressed for further study. Among the 500 test data that were used for NDC category assignment, 63 Wikipedia articles were wrongly assigned NDC categories by the CNN. Among these 63, articles with proper nouns in their titles were predominant, with 12 containing the names of persons and six the names of organizations. Three articles with the names of historic Japanese personalities in their titles were assigned NDC category 21 (*History of Japan*), although the "correct" category was 28 (*Biography*). In this study, the correct NDC category was determined based on the books cited in the Wikipedia articles. These persons were mentioned in articles that cited their biographies. However, considering that the personalities are actually of historic significance, the above-mentioned assignment of *History of Japan* to the relevant articles is not entirely a failure. Other "incorrect" categorizations were observed for articles that belonged to multiple categories. For example, while the "correct" NDC category of an article on a historic Chinese calligrapher and philosopher is 72 (*Painting and Calligraphy*), the proposed system assigned category 12 (*Oriental Philosophy*) to it. This apparent failure may be difficult to correct, and it may not even be necessary to attempt a correction for the purpose of book recommendation because recommending a book for a specific aspect of the article may be acceptable.

The book selection precision of the SVM reached 99.8% and few errors were observed here. Regarding user evaluation, the 79 (= 41 + 38 in Table 8) books were evaluated as "I do

not want to read this book if possible" and "I do not want to read this book at all." One means of reducing such negative evaluations is to recommend a book only when the probability mentioned in subsection 3.2 is sufficiently high. Table 14 gives the evaluation ratios for recommended books when the probability exceeded 99.99%. It can be seen from the table that the number of books evaluated as "I do not want to read this book if possible" or "I do not want to read this book at all" decreased to 35 (= 17 + 18), while those evaluated as "I want to read this book very much" and "I want to read this book if possible" increased from 47.3% (= 15.3% + 32.0% in Table 8) to 56.3% (= 22.5% + 33.8%).

Table 14. Evaluation of Recommended Books when Probability > 99.99%

Evaluation	N	Ratio
(1) I want to read this book very much	18	22.5
(2) I want to read this book if possible	27	33.8
(3) I do not want to read this book if possible	17	21.3
(4) I do not want to read this book at all	18	22.5
Total	80	100.0

It is also noteworthy that the developed CNN achieved a precision of 87.4% in assigning the two-digit (main class and division) NDC categories. There are actually 100 main class and division categories with notations ranging from 00 to 99. Considering that the probability that the correct category would be randomly chosen is 1%, the precision of 87.4% is very high. Further, although the proposed system uses the NDC categories assigned to Wikipedia articles to recommend books to the article readers, the data can also be inversely utilized, namely, to recommend Wikipedia articles to book readers. In addition, the procedure can be modified to assign NDC categories to Web pages generally and recommend books to readers of the Web pages. Hence, in a broader sense, the proposed system can be regarded as a tool for linking Web content with printed books. This promises to be beneficial in various fields.

5   Conclusions

To encourage students to read library books as a more reliable source of information, a system was developed for recommending library books to Wikipedia article readers in university libraries. The system assigns NDC categories to Wikipedia articles and recommends library books in the same categories as the respective articles. The CNN-based NDC category assignment precision of the system for Wikipedia articles was determined to be as high as 87.4%, while the SVM-based book selection precision reached 99.8%. In 82.0% of the test cases, the student subjects found at least one book that they evaluated as "I want to read this book" among three recommended books. Although the precision of selecting books for recommendation to the article readers was high, the subjects' evaluations of the books were relatively lower. This implies that the system training data obtained from the books cited in the articles were not ideal for book recommendation. Although it would be labor-intensive, training data could be manually generated by requesting many students to evaluate many books. The results could be used to achieve better book recommendation performance of the proposed system. In the present study, the system was implemented as a Google Chrome extension. It would be interesting to further implement the system as an extension of other Web browsers such as Microsoft Edge and Mozilla Firefox, and investigate whether the books recommended to the Wikipedia article readers are actually borrowed from the library.

References

[1] T. Anbiru et al., "Information Seeking Behavior," Proceedings of the Spring Meeting of the Japan Society of Library and Information Science, 2010, pp. 87-90. (Text in Japanese.)

[2] Japan Library Association, Statistics on library in Japan. Japan Library Association. 2018, 515p.

[3] Y. Kim, "Convolutional Neural Networks for Sentence Classification," Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, 2014, pp. 1746-1751.

[4] S. Arai and K. Tsuji, "Automatically Assigning NDC Categories to Reference Service Records by Using Machine Learning Methods," Journal of the Japan Society of Information and Knowledge, vol. 25, no.1, 2015, pp. 23-40. (Text in Japanese.)

[5] R. Johnson and T. Zhang, "Effective Use of Word Order for Text Categorization with Convolutional Neural Networks," Proceedings of Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL, 2015, pp. 103-112.

[6] P. Wang et al., "Semantic Clustering and Convolutional Neural Network for Short Text Categorization," Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, 2015, pp. 352-357.

[7] M. Mikawa et al., "Book Recommendation Signage System Using Silhouette-based Gait Classification," Proceedings or the 10th International Conference on Machine Learning and Applications, 2011, pp. 416-419.

[8] P. Jomsri, "FUCL Mining Technique for Book Recommender System in Library Service," Proceedings of the 11th International Conference Interdisciplinarity in Engineering, 2018, pp. 550-557.

[9] R. Mooney and L. Roy, "Content-based Book Recommending Using Learning for Text Categorization," Proceedings of the 5th ACM conference on Digital Libraries, 2000, pp. 195-204.

[10] S. Givon and V. Lavrenko, "Predicting Social-tags for Cold Start Book Recommendations," Proceedings of the 3rd ACM Conference on Recommender Systems, 2009, pp. 333-336.

[11] X. Yang et al., "ARTMAP-based Data Mining Approach and its Application to Library Book Recommendation," Proceedings of the 2009 International Symposium on Intelligent Ubiquitous Computing and Education, 2009, pp. 26-29.

[12] M.S. Pera et al., "Personalized Book Recommendations Created by Using Social Media Data," Proceedings of the 2010 International Conference on Web Information Systems Engineering, 2010, pp. 390-403.

[13] R.G. Crespo et al., "Recommendation System based on User Interaction Data Applied to Intelligent Electronic Books," Computers in Human Behavior, vol. 27, 2011, pp. 1445-1449.

[14] P.C. Vaz et al., "Improving a Hybrid Literary Book Recommendation System through Author Ranking," Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries, 2012, pp. 387-388.

[15] C. Benkoussas and P. Bellot, "Book Recommendation based on Social Information," Working Notes for CLEF 2013 Conference, 2013, pp. 23-26.

[16] D. Pathak et al., "NOVA: Hybrid Book Recommendation Engine," Proceedings of the IEEE 3rd International Advance Computing Conference, 2013, pp. 977-982.

[17] P.C. Vaz et al., "Understanding Temporal Dynamics of Ratings in the Book Recommendation Scenario," Proceedings of the 2013 International Conference on Information Systems and Design of Communication, 2013, pp. 11-15.

[18] A.L. Garrido et al., "SOLE-R, a Semantic and Linguistic Approach for Book Recommendations," Proceedings of the IEEE 14th International Conference on Advanced Learning Technologies, 2014, pp. 524-528.

[19] M.S. Pera and Y. Ng, "Automating Readers' Advisory to Make Book Recommendations for K-12 Readers," Proceedings of the 8th ACM Conference on Recommender Systems, 2014a, pp. 9-16.

[20] M.S. Pera and Y. Ng, "How Can We Help Our K-12 Teachers?: Using a Recommender to Make Personalized Book Suggestions," Proceedings of the IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies, 2014b, pp. 335-342.

[21] K. Priyanka et al., "Personalised Book Recommendation System based on Opinion Mining Technique," Proceedings of the Global Conference on Communication Technologies, 2015, pp. 285-289.

[22] S. Rajpurkar et al., "Book Recommendation System," International Journal for Innovative Research in Science & Technology, vol. 1, no. 11, 2015, pp. 314-316.

[23] A. Sase et al., "A Proposed Book Recommender System," International Journal of Advanced Research in Computer and Communication Engineering, vol. 4, no. 2, 2015, pp. 481-483.

[24] S. Bhosale et al., "SuggestABook: A Book Recommender Engine with Personality based Mapping," International Journal of Computer Applications, vol. 159, no. 9, 2017, pp. 1-4.

[25] B. Gao et al., "Research on the Application of Persona in Book Recommendation System," Journal of Physics: Conference Series, vol. 910, 2017, pp. 1-8.

[26] S.S. Sohail et al., "A Novel Approach for Book Recommendation Using Fuzzy based Aggregation," Indian Journal of Science and Technology, vol. 10, no. 19, 2017, pp. 1-30.

[27] H. Alharthi et al., "A Survey of Book Recommender Systems," Journal of Intelligent Information Systems, vol. 51, 2018, pp. 139-160.

[28] P. Parekh et al., "Web based Hybrid Book Recommender System Using Genetic Algorithm," International Research Journal of Engineering and Technology, vol. 5, no. 8, 2018, pp. 1536-1539.

[29] S.S. Sohail et al., "An OWA-based Ranking Approach for University Books Recommendation," International Journal of Intelligent Systems, vol. 33, 2018, pp. 396-416.

[30] T. Thanapalasingam et al., "The Smart Book Recommender: An Ontology-driven Application for Recommending Editorial Products," Proceedings of the International Semantic Web Conference 2018, 2018, 5p. (No Pagination).

[31] K. Tsuji et al., "Book Recommendation based on Library Loan Records and Bibliographic Information," Proceedings of the 3rd International Conference on Integrated Information, 2013, 8p. (No Pagination).

[32] K. Tsuji et al., "Book Recommendation Using Machine Learning Methods based on Library Loan Records and Bibliographic Information," Proceedings of the 5th International Conference on E-Service and Knowledge Management, 2014, pp. 76-79.

[33] K. Tsuji et al., "Book Recommendation Using Machine Learning Methods based on Library Loan Records and Bibliographic Information," International Journal of Academic Library and Information Science, vol. 3, no. 1, 2015, pp. 7-23.

[34] K. Tsuji, "Books Cited in Wikipedia: Possibility to Use their Nippon Decimal Classification Categories for Book Recommendation," Proceedings of the 7th International Conference on E-Service and Knowledge Management, 2016, pp. 1196-1197.

[35] K. Tsuji, "Automatic Classification of Wikipedia Articles by Using Convolutional Neural Network," Proceedings of the 9th Qualitative and Quantitative Methods in Libraries International Conference, 2017, 8p. (No Pagination).

[36] Mecab. http://taku910.github.io/mecab/ [Last Access: 2019-01-14]

[37] mecab-ipadic-NEologd: neologism dictionary for MeCab. https://github.com/neologd/mecab-ipadic-neologd [Last Access: 2019-01-14]

[38] Word2vec. https://radimrehurek.com/gensim/models/word2vec.html [Last Access: 2019-01-14]

[39] gensim: topic modelling for humans. https://radimrehurek.com/gensim/ [Last Access: 2019-01-14]

[40] OpenSearch by the National Diet Library of Japan. http://iss.ndl.go.jp/information/api/ [Last Access: 2019-01-14]

[41] Tensorflow. https://www.tensorflow.org/ [Last Access: 2019-01-14]

[42]   LIBSVM: library   for   support   vector   machines. https://www.csie.ntu.edu.tw/~cjlin/libsvm/ [Last Access: 2019-01-14]

[43] OpenBD. https://openbd.jp/ [Last Access: 2019-01-14]

[44] R.K. Merton, "The Matthew Effect in Science," Science, vol. 159, no. 3810, 1968, pp. 56-63.