Is It Possible for the First Three-Month Time-Series Data of Views and Downloads to Predict the First Year Highly-Cited Academic Papers in Open Access Journals?

Hiroshi Ishikawa*, Masaki Endo*, Iori Sugiyama**, Masaharu Hirota***, and Shohei Yokoyama****
*Graduate School of System Design, Tokyo Metropolitan University, Hino, Japan
**Faculty of System Design, Tokyo Metropolitan University, Hino, Japan
***Department of Information Engineering, Oita National College of Technology, Oita, Japan
****Graduate School of Science and Technology, Shizuoka University, Hamamatsu, Japan

Abstract- Currently, academic papers and their authors can be mainly evaluated by the statistics in Bibliometrics such as number of citations, h-index, and impact factor. However, it usually takes at least half a year or more for Bibliometrics-based approaches to evaluate academic papers. Open access journals, which do not restrict browsers, are spreading especially in recent years. Further, the number of viewing academic papers published in open access journals and the number of posting articles about the papers to social media continue to increase year after year. Such data can be treated as time-series data with immediacy. Therefore it is thought that if the academic papers as time-series data can be analyzed by proper data mining techniques such as clustering, it will be possible to extract the characteristics of highly-cited scientific papers. Instead of conventional evaluation of scholarly papers based on Bibliometrics, this paper discusses a method for estimating scientific papers with the potential of being highly cited in future based on the associated time-series data.

Keywords: open access journal; time-series data; Dynamic Time Warping; clustering; BIRCH

1 INTRODUCTION

Recently, open access journals (OAJ), which do not restrict viewing, are spreading in the world, especially in US and the ratio of such journal papers over all academic papers is increasing accordingly [1]. In general, since OAJ publish accepted papers worldwide in one week or so, the academic papers can be accessed and viewed without restrictions by any user. Compared to traditional journals, OAJ can ensure the immediacy of the scientific papers by proving shorter periods from submission to publication. On the other hand, traditional surveys about academic papers are based on so-called Bibliometric indices such as the total number of papers published by one author and the number of citations per paper. And such traditional surveys based on Bibliometrics tend to take long time just like submitted papers take long time to be published.

Very recently, as methods to evaluate the scientific papers in the immediate term, alternative Bibliometrics called Altmetrics, which use the posts to social media, such as Twitter (micro blogging), Facebook (blogging), and Mendeley (social bookmarking), and analyze the contents and numbers, are gathering attention. Typical services for evaluation of scientific papers by using Altmetrics include altmetric.com and ImpactStory. In Japan, Ceek.jp Altmetrics, a university-originated venture has begun to provide Altmetrics-based services [2]. In this paper, we mean by Altmetrics both methods for quantitative measurement of impacts of research products such as journal papers and data sets using the social media responses and activities as to measuring the future influences of emergent researches based on the results [3].

So we have thought that it is necessary to clarify the relationships between Altmetric indices such as traffic data (i.e., number of views) and social media posts and Bibliometric indices such as citation data. In the preliminary experiment, we have found weak or very weak correlations between the number of views or downloads as of the first month after publication and that of citations of the scientific papers. This detail will be explained in Section 3 of this paper. Since these correlations are negligible, we have clustered academic papers published in OAJ with immediacy by paying attention to the counts of views and downloads so as to predict Bibliometric indices such as the counts of citations. We have used time-series data consisting of the numbers of views and downloads of papers per month for three or six months in clustering. Our final objective, which we have not fully obtained yet, is to estimate papers with the possibility of being highly cited in the future by performing machine learning, that is, learning classifiers for such papers based on clustering results. If the objective is fully attained, it will be possible to know the technological trends at the very early stage. Figure 1 illustrates the big picture of our research as the flow of the associated processes. In this paper, we focus on the clustering of the academic papers based on the time- series data. Section 2 de-

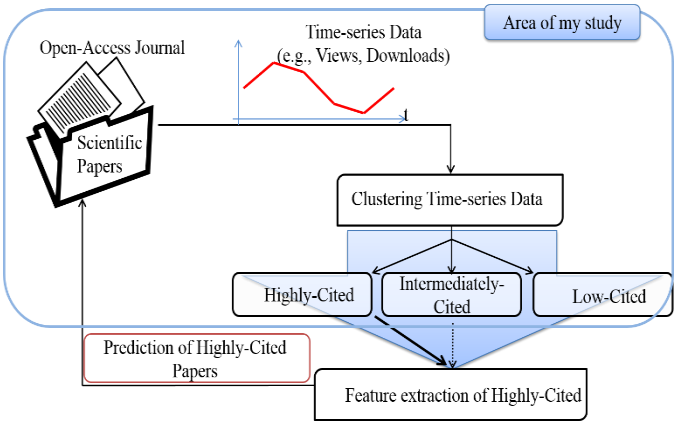

Figure 1: Big Picture of Our Research.

ISSN1883-4566 ©2016 - Informatics Society and the authors. All rights reserved.

Table 1: Correlation between Numbers of Citations and Numbers of Views as of 1st Month.

	Cited	Views
Cited	1	---
Views	0.0695	1

Table 2: Correlation between Numbers of Citations and Numbers of Downloads as of 1st Month.

	Cited	Downloads
Cited	1	---
Downloads	0.2852	1

scribes the relevant works. Section 3 explains the normalization of the time series data. Section 4 and section 5 describe our clustering method and experiments using the method, respectively. Section 6 summarizes the contribution of our work and describes future challenges.

2 RELATED WORK

Most of works that cluster time-series data focus on the frequency. As clustering time-series data based on the frequencies as features, subsequence time-series clustering [4] is often used. So as to exactly handle the frequency of time-series data as the features, time-series data longer than a certain length are required. However, as our work focuses on the immediacy of the scientific papers published in OAJ, we use only data recorded every month for at most 12 months from publication, that is, 12 pieces of time-series data. Therefore, we use the shape of a wave as features of the time-series data. Then we can calculate dissimilarity (i.e., distance) of the time-series data using Dynamic Time Warping DTW [5].

Recently, studies on Altmetrics are becoming very active and are expected to complement Bibliometric evaluation of scientific papers. Posts to Twitter (i.e., Tweets) referring to academic papers published in OAJ are observed for several days just after publication, according to Gunther [6], thereby enabling the prediction of the number of citations. However, the number of collected Tweets mentioning scholarly papers is not so large, partially because it is rather difficult to exhaustively find correspondences between academic papers and tweets. Nakahashi et al [7] have done automatic mapping between academic papers and Tweets and have attained better performance than the previous works. However, their research have used only tweets referring to papers presented in the traditional academic meetings. There still remain a lot of works to do so as to automatically find correct correspondences between Tweets and scholarly papers on the web, such as OAJ.

3 NORMALIZATION OF TIME-SERIES DATA

We used the data of academic papers published in Public Library Of Science (PLOS) [8], one of US-based OAJ. We used the API provided by the PLOS and obtained the data of academic papers for two times, on 7/20/2013 and 12/22/2013. While the numbers of views are equal to those of accesses to web pages provided to individual academic papers by PLOS

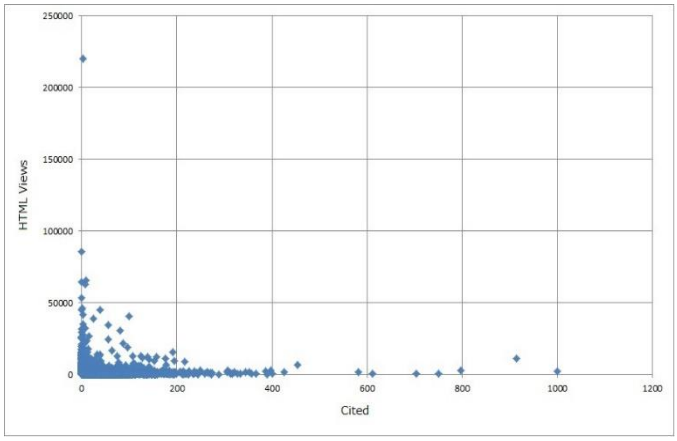

Figure 2: Number of Citations vs. Number of Views.

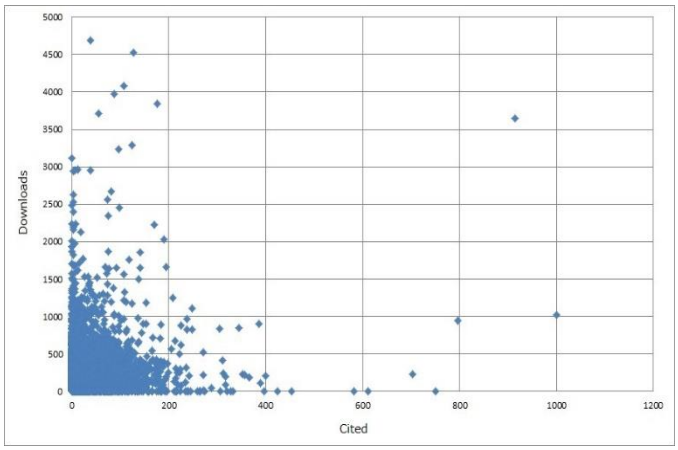

Figure 3: Number of Citations vs. Number of Downloads.

(html views), the numbers of downloads are equal to those of the saved PDF (pdf views). The numbers of views and downloads are monthly calculated from the published month. They constitute time-series data. The numbers of citations as to some papers are also calculated at fixed intervals from the published month. The numbers of citations as to others, however, are those of citations accumulated from the published date to the collected date. Then we used the accumulated numbers of citations in a uniform fashion.

We acquired academic papers on 7/20/2013 whose number n = 52,386 and calculated the correlation coefficient r between the numbers of citations and those of views as of the first month of papers since publication and we obtained a very weak correlation ($r = 0.0695$) (See Table I). As to the same set of papers, we also calculated a correlation coefficient r between the numbers of citations and those of downloads as of the first month and obtained a weak correlation ($r = 0.2852$) (See Table II). The corresponding scatter charts are illustrated in Fig. 2 and Fig. 3. Then we judged these correlations negligible and decided to use time-series data of views and down-

loads instead. We normalized these time-series data by dividing all the values by those as of the first month. Figure 4 shows examples of normalized time-series data as to the numbers of downloads of 14 samples published in 2/2004 for the first 12 months.

4 CLUSTERING

We examined two different clustering techniques for time-series data. First, we made vectors out of the time-series data and conducted k-means clustering based on the cosine measure as to the vectors. Next, we calculated dissimilarities between two pieces of the time-series data by Dynamic Time Warping (DTW) and conducted clustering based on the dynamic time warping squashed trees, an extension of CF tree.

4.1 K-Means Clustering

4.1.1 Vectorizing Time-Series Data

First, let a_i be the number of views or downloads (exactly, the ratio over the value for the first month) recorded for i-th month as to the scientific papers A. As a whole, the time series data for the paper A is represented as A = a_1, a_2,..., a_I. Next we vectorize the time-series data as follows. So as to focus on the rate of change of each component of the time series data along the time line, we consider a sub-vector for the change of two consecutive elements. The x-component of the sub-vector is constantly 1 because the value is always one month. Each y-component of the sub-vector is represented as (a_{k+1}-a_k). Therefore, vectorized time-series data for the paper A is represented as a set of sub-vectors by the following formulas:

$$V_A = \{\overrightarrow{v_{A(1)}}, \overrightarrow{v_{A(2)}}, \dots, \overrightarrow{v_{A(k)}}, \dots, \overrightarrow{v_{A(kI-1)}}\} \quad (1)$$

$$\overrightarrow{v_{A(k)}} = \{1, a_{k+1} - a_k\} \quad (2)$$

Figure 5 shows two examples of vectorized time series data.

4.1.2 K-Means Method

Here we consider the similarity between the academic papers A and B. The similarity between corresponding sub-vectors is calculated by using the cosine measure. Let the angle between them be θ_k. Then the cosine similarity between two vectors $\overrightarrow{v_{A(k)}}$ and $\overrightarrow{v_{B(k)}}$ is represented by the formula (3) (See Fig. 5). Note that as $0 \leq \cos\theta_k \leq 1$, two vectors is similar if the cosine similarity is close to 1 or dissimilar if close to 0. Further, by using the total sum $\cos_{sum}$ and the total product $\cos_{prod}$ of the cosine similarity between corresponding sub-vectors, which are defined in the formulas (4) and (5), respectively, the similarity of the academic papers A and B (i.e., the vector sets $\boldsymbol{V_A}$ and $\boldsymbol{V_B}$) s is expressed as s = ($\cos_{sum}$, $\cos_{prod}$). Suppose that I and J are the lengths of the time-series data A and B, respectively. Every piece of time series data has the same length, that is, $I = J$.

$$\cos\theta_k = \frac{\overrightarrow{v_{A(k)}} \cdot \overrightarrow{v_{B(k)}}}{|\overrightarrow{v_{A(k)}}| \cdot |\overrightarrow{v_{B(k)}}|} \quad (3)$$

$$\cos_{sum} = \sum_{k=1}^{I-1} \cos\theta_k \quad (4)$$

$$(0 \leq \cos_{sum} \leq I - 1)$$

$$\cos_{prod} = \prod_{k=1}^{I-1} \cos\theta_k \quad (5)$$

$$(0 \leq \cos_{prod} \leq 1)$$

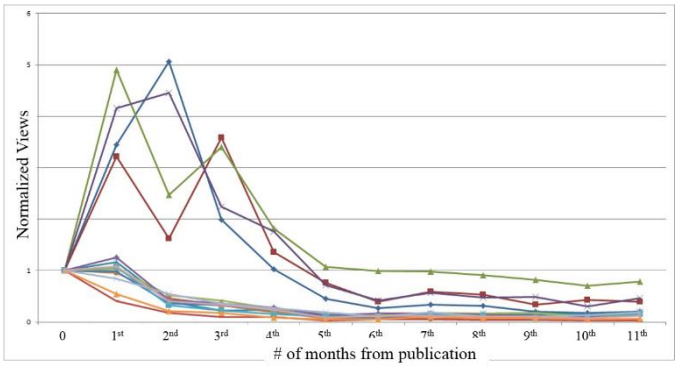

Figure 4: Examples of Normalized Time-Series Data.

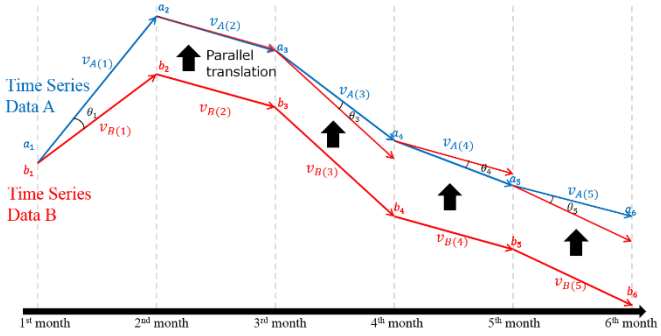

Figure 5: Vectorization of Time-Series Data.

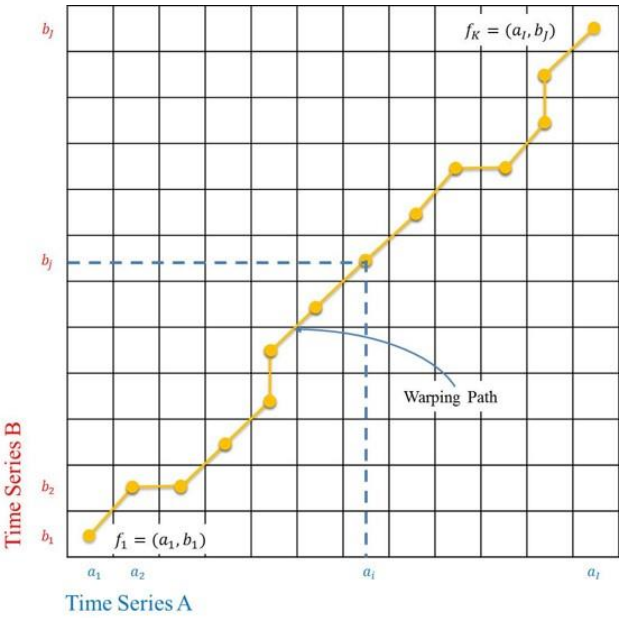

Figure 6: Example of Warping Path in DTW.

Here we cluster academic papers by using k-means clustering based on this similarity measure s. Beforehand, we sorted data for the whole academic papers according to the descending order of DOI (Digital Object Identifier) and divided into k groups as initial clusters. The centroid of the cluster is calculated as the vector set $\boldsymbol{Vm}$ expressed by the formula (6). Let

N be the total number of the papers. The initial centroids are calculated by using the initial clusters.

$$V_m = \frac{\sum_{n=1}^{N} V_n}{N} = \{\frac{\sum_{n=1}^{N} \overrightarrow{v_{n(1)}}}{N}, \ldots, \frac{\sum_{n=1}^{N} \overrightarrow{v_{n(I-1)}}}{N}\} \quad (6)$$

In one repetition, the similarities between each paper and all the centroids are calculated. If there is another cluster to which the paper is more similar than the current cluster, the paper is moved to the former cluster. In case of any movement at the end of the repetition, the centroids of the clusters are updated. This process is repeated until the number of academic papers to move is less than a certain threshold, that is, stable.

4.2 Dynamic Time Warping Squashed Tree Clustering

4.2.1 Dynamic Time Warping

The Dynamic Time Warping (DTW) algorithm can calculate the dissimilarity between a pair of time-series data. The algorithm can map multiple points in one piece of time- series data to a single point in another piece of time series data, thereby allowing non-linear transformation as to the time axis.

Let us consider a pair of 12-month time-series data for two academic papers A and B. The time-series data are represented as $A = (a_1,a_2,\ldots,a_{11}, a_{12})$ and $B = (b_1,b_2,\ldots,b_{11}, b_{12})$. Then we construct a 12×12 matrix (i.e., table) by corresponding A and B to the row and column, respectively. In this table, grid points $f_k = (a_{ik}, b_{jk})$ represent correspondences between A and B. The series $F = (f_1,f_2,\ldots, f_k,\ldots, f_K)$ is called warping path. An example is shown in Fig. 6. The distance between a_{ik} and b_{jk} is denoted by $\delta(f_k)$ and is calculated by the formula (7). Using this distance, the evaluation function $\Delta(F)$ for the warping path F is calculated by the formula (8). Here w_k is a positive weight to f_k and is calculated by the formula (9).

$$\delta(f_k) = |a_{ik} - b_{jk}| \quad (7)$$

$$\Delta(F) = \frac{1}{I+J}\sum_{k=1}^{K} w_k \cdot \delta(f_k) \quad (8)$$

$$w_k = (i_k - i_{k-1}) + (j_k - j_{k-1}) \quad (9)$$
$$i_0 = j_0 = 0$$

Here i_k and j_k which are integers between 1 and 12 denote subscripts of components a and b of a grid point f_k, respectively. I and J are the lengths of the time-series data A and B, respectively. The smaller $\Delta(F)$ is, the smaller the dissimilarity between A and B is. In other words, mapping is better in that case. The warping path F with the minimum $\Delta(F)$ is the shortest warping path of A and B. In that case $\Delta(F)$ is used as the dissimilarity between the two time series data. This is calculated by using the function mlpy.dtw_std provided by the machine learning library mlpy [9] in the programming language Python.

4.2.2 Dynamic Time Warping Squashed Trees

Dynamic time warping squashed tree (DTWS tree) [10] is a height-balanced binary tree, a variant of CF tree, when BIRCH, a typical hierarchical clustering method is adapted to time-series data. This method clusters time-series data based on DTW-dissimilarities by exhaustive searching, doing data compression.

First, let us consider the node vector CF of the original CF tree. Generally using N_0 as the number of elements belonging to the node, the linear sum of vectors $\boldsymbol{LS}_0 = \Sigma_{k=1,\ N_0} \boldsymbol{X_k}$, and the squared sum $SS_0 = \Sigma_{k=1,\ N_0} (\boldsymbol{X_k})^2$, the node CF of the CF tree is represented as $CF = (N_0, \boldsymbol{LS}_0, SS_0)$. On the other hand, the node vector $DTWS$ of the DTWS tree corresponds to the CF vector but the squared sum SS_0 is omitted from the CF and is represented as $DTWS = (N, \boldsymbol{ATW})$. Here N is the number of time-series data belonging to the node $DTWS$ and $\boldsymbol{ATW}$ is the average vector of time-series data. $\boldsymbol{ATW}$ is calculated as follows: First the average of two time-series data A and B is calculated as $\boldsymbol{ATW}_\beta = (x_1, x_2, x_k,\ldots, x_K)$. Here x_k correspond to f_k, calculated by the formula (10). As the average vector $\boldsymbol{ATW}_\beta$ is, in general, longer than the original time-series data A and B ($K \geq I = J$). Thus $\boldsymbol{ATW}_\beta$ is un-normalized with respect to the length. Then we compress it in order to obtain $\boldsymbol{ATW} = (y_1, y_2,\ldots, y_{k'},\ldots, y_I)$ in accordance with the original time-series data. If the path for x_k extends diagonally on the warping path,

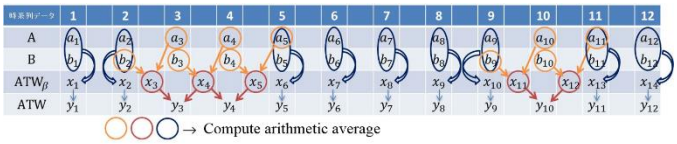

Figure 7: Computation of ATW.

Table 3: Data Used in the Experiments.

Length of Time-series Data	Published	# of Papers	Data1		Data2	
			Intermediately- and Higly-Cited	Highly-Cited	Intermediately- and Higly-Cited	Highly-Cited
3 months	Before 2013/4	48261	10978	398	13890	579
6 months	Before 2013/1	43363	10978	398	13885	579
12months	Before 2012/7	33934	10978	398	13715	579

Table 4: Results of K-Means Clustering (K=10).

	Data1		Data2		Difference	
# of Papers in a cluster	Cited≥10	ratio(%)	Cited≥10	ratio(%)	Cited≥10	ratio(%)
13,659	2,586	18.93	3,298	24.15	712	5.21
4,542	1,459	32.12	1,790	39.41	331	7.29
4,174	516	12.36	760	18.21	244	5.85
3,392	1,256	37.03	1,476	43.51	220	6.49
1,765	809	45.84	911	51.61	102	5.78
1,707	231	13.53	346	20.27	115	6.74
1,460	538	36.85	640	43.84	102	6.99
1,045	190	18.18	269	25.74	79	7.56
955	58	6.07	115	12.04	57	5.97
801	341	42.57	397	49.56	56	6.99
780	54	6.92	94	12.05	40	5.13
762	234	30.71	296	38.85	62	8.14
751	65	8.66	113	15.05	48	6.39
721	124	17.20	182	25.24	58	8.04
545	195	35.78	219	40.18	24	4.40
524	143	27.29	211	40.27	68	12.98
510	106	20.78	151	29.61	45	8.82
491	39	7.94	65	13.24	26	5.30
485	17	3.51	36	7.42	19	3.92
469	198	42.22	220	46.91	22	4.69
436	17	3.90	39	8.94	22	5.05
432	122	28.24	155	35.88	33	7.64
429	177	41.26	196	45.69	19	4.43

Table 5: Results of *K*-Means Clustering (*K*=25).

	Data1		Data2		Difference	
# of Papers in a cluster	Cited>=10	ratio(%)	Cited>=10	ratio(%)	Cited>=10	ratio(%)
4,290	647	15.08	848	19.77	201	0.05
3,997	504	12.61	697	17.44	193	0.05
3,761	1,108	29.46	1,365	36.29	257	0.07
2,126	384	18.06	538	25.31	154	0.07
1,732	635	36.66	759	43.82	124	0.07
1,575	620	39.37	718	45.59	98	0.06
1,359	371	27.3	440	32.38	69	0.05
1,066	73	6.85	123	11.54	50	0.05
906	411	45.36	473	52.21	62	0.07
854	183	21.43	252	29.51	69	0.08
789	84	10.65	131	16.6	47	0.06
721	303	42.02	344	47.71	41	0.06
633	173	27.33	212	33.49	39	0.06
553	253	45.75	293	52.98	40	0.07
542	60	11.07	99	18.27	39	0.07
458	166	36.24	189	41.27	23	0.05
433	73	16.86	104	24.02	31	0.07
402	91	22.64	125	31.09	34	0.08

Table 6: Kruskal-Wallis Test of the Results by *K*-Means Clustering (*K*=10).

	χ^2	# of degree of freedom	p-value
Data1	3283.75	96	$p_1<2.2\times10^{-16}$
Data2	2962.84	96	$p_2<2.2\times10^{-16}$

its length will be 1; If the path for x_k extends either horizontally or vertically, its length will be 0.5. Based on this calculation, $\boldsymbol{ATW}_\beta$ is compressed to $\boldsymbol{ATW}$. The value of $y_{k'}$ is calculated, that is, equal to that of x_k by the formula (11) if the length to x_k from the starting point of the time-series data is integer, otherwise it is calculated as linear interpolation between the two consecutive elements by using the formula (12). As to the time warping path in Fig. 6, the processes of the above calculation are shown in Fig. 7.

$$x_k = \frac{a_{ik}+b_{jk}}{2} \tag{10}$$

$$y_{k'} = \begin{cases} x_k & (11) \\ \dfrac{x_k + x_{k+1}}{2} & (12) \end{cases}$$

The procedures of the DTWS tree are similar to those of the CF tree. The dissimilarity between the time-series vector to be added and the average time-series vector of the node is calculated based on DTW. If there exists a node with the minimum dissimilarity, less than a prescribed threshold, the new vector is added to the node.

5 EXPERIMENTS

5.1 Outline of the Experiments

For the experiments, we used data that were collected from the scientific open access journal PLOS twice on 7/20/2013 and 12/22/2013, respectively. The data for the scientific papers include the numbers of views and those of downloads of each paper as of each month as time-series data starting just after the publication and the number of citations of each paper as of the time of collection. We collected data as to 52,555 scientific papers on 7/20/2013, among which 52,386 papers have information as to citations. Further, from the collection, we selected scientific papers which have 3-month, 6-month, and 12-month numbers of views and of downloads as of 7/20/2013. Data collected on 7/20/2013 and 12/22/2013 are called Data1 and Data2, respectively (See Table III). We used two clustering methods described in Sections 4.1 and 4.2 for comparison. In one run of experiments, we clustered the same collection of academic paper data separately based on the numbers of views and of downloads. As a result, we obtain two sets of clusters. By making intersections of two sets of clusters, we obtained one set of clusters as a final result. We evaluated the final set of clusters in terms of the number of citations. The average number of citations is 11.81 and the median is 5 in Data1. The low-cited papers are defined as those with less than 10 citations. The highly-cited papers are assumed to belong to the top 10 % of the whole collection with respect to citations. In the top 10% subset of Data1, the minimum number of citations is 87. For the simplification of the judgment when evaluating clustered results, the threshold for the highly-cited papers is set to 90 in our experiments. Therefore, the intermediately-cited papers are defined as those with 10≤citations<90. Table III shows some statistics about our scientific paper collections prepared for the experiments. The "intermediately- and highly-cited" papers and the "highly-cited" papers denote the items cited times ≥10 and those cited times≥90", respectively.

Table 7: Results of DTWS Tree Clustering of Three-Month Time-Series Data.

Threshold		# of clusters		Data1		Data2	
X_{html}	X_{pdf}	All	10 papers or more	Intermediately- and Higly-Cited Ratio>=90%	Num	Intermediately- and Higly-Cited Ratio>=90%	Num
32	47	123	43	3	8,851	3	8,874
42	22	161	49	3	8,818	3	8,819
42	42	96	33	2	8,818	2	8,820
42	40	114	38	4	8,809	5	8,848
42	25	150	51	3	8,806	3	8,801
42	32	118	37	2	8,803	2	8,797
42	15	254	90	5	8,779	7	10,020
42	37	112	40	2	8,778	3	8,822
42	12	295	92	4	8,774	4	8,767
42	10	301	80	4	8,773	5	8,802
42	50	101	36	2	8,761	2	8,755
42	47	100	36	2	8,760	2	8,754
42	20	185	65	3	8,757	4	8,785
42	35	118	40	2	8,757	3	8,799
42	7	395	101	7	8,754	7	8,747
42	27	143	52	3	8,753	4	8,827
42	30	148	57	2	8,751	3	8,760
42	45	101	38	2	8,750	3	8,791
42	5	634	139	9	8,733	11	8,778
42	17	194	57	3	8,721	4	8,817

5.2 Results

5.2.1 *K*-Means Clustering

We conducted k-means clustering on 3-month time-series data with respect to both views and citations as k = 10 and 25. Because each result consists of 10 or 25 clusters, the academic

Table 8: Recall, Precision, and *F*-Value of Intermediately- and Highly-Cited Papers.

Length of Time-series Data	Threshold *X*		# of Papers in a Cluster	Data	# of Intermediately- and Highly-Cited Papers in a Cluster	R (%)	P (%)	F (%)
	Views	Downloads						
3	42	42	8,470	Data1	8,460	77.06	99.88	78.42
3	42	32	8,353	Data1	8,347	76.03	99.93	77.43
3	45	42	9,447	Data1	8,385	76.38	88.76	77.19
3	47	42	12,766	Data1	8,539	77.78	66.89	76.84
3	22	42	9,486	Data1	8,338	75.95	87.90	76.74
3	42	40	8,276	Data1	8,265	75.29	99.87	76.71
3	42	30	8,250	Data1	8,245	75.10	99.94	76.54
6	50	42	14,464	Data1	8,565	78.02	59.22	76.19
3	45	32	9,524	Data1	8,275	75.38	86.89	76.14
3	45	30	8,311	Data1	8,175	74.47	98.36	75.86

Table 9: Number of Clusters in Top Clusters Ranked by *F*-Value for Intermediately- and Highly-Cited Papers.

Length of Time-series Data	# in Top 100 clusters	# in Top 200 clusters	# in Top 300 clusters
3 months	70	119	201
6 months	30	79	93
12 months	0	2	6

Table 10: Recall, Precision, and *F*-Value of Highly-Cited Papers.

Length of Time-series Data	Threshold		# of Papers in a Cluster	Data	# of Highly-Cited Papers in a Cluster	R (%)	P (%)	F (%)
	Views	Downloads						
12	47	42	506	Data1	396	99.50	78.26	97.50
12	47	50	516	Data1	396	99.50	76.74	97.32
12	47	40	496	Data1	393	98.74	79.23	96.94
12	47	30	494	Data1	392	98.49	79.35	96.73
3	25	35	500	Data1	389	97.74	77.80	95.88
12	47	22	459	Data1	379	95.23	82.57	94.14
12	47	12	466	Data1	375	94.22	80.47	93.02
6	25	40	473	Data1	375	94.22	79.28	92.90
6	25	32	473	Data1	375	94.22	79.28	92.90
6	25	47	479	Data1	375	94.22	78.29	92.80

Table 11: Number of Clusters in Top Clusters Ranked by *F*-Value for Highly-Cited Papers.

Length of Time-series Data	# in Top 100 clusters	# in Top 200 clusters	# in Top 300 clusters
3 months	48	87	110
6 months	24	71	122
12 months	28	42	68

papers are divided into 10 × 10 or 25 × 25 clusters. However, there also exist clusters which contain no elements. Then clusters with more than 400 elements are picked up and described in Table IV and Table V for k = 10 and k = 25, respectively. In the tables, "# of papers", "Cited≥10", and "ratio(%)" denote the number of elements in the cluster, the number of elements cited for 10 or more times, and the ratio over the cluster (%), respectively. By k-means clustering, no clusters with only intermediately- and highly-cited academic papers could be found. However, the Kruskal-Wallis test, a nonparametric method, was conducted on 100 clusters constructed by k-means clustering (i.e., k=10) using Kruskal.test function of the R language [11]. Table VI shows the results of the Kruskal-Wallis test. As p_1, $p_2 \leq$ significance level $\alpha = 0.01$ from Table VI, it has been confirmed that there exist significant differences between the median numbers of citations of clusters. Note that, because there exist three clusters with less than two elements in the results and the tests were performed on the rest and thus the number of degrees of freedom was 96.

5.2.2 DTWS Tree Clustering

DTWS tree clustering was performed on 3-month, 6-month, and 12-month time-series data with respect to the number of views and of citations. DTWS tree is a clustering method using exhaustive searching based on thresholds. Then, the size of clusters and the number of clusters change, depending on the given thresholds. For both the numbers of views and of downloads, the threshold *X* takes one of 26 different values:{0.3, 0.5, 0.7, 1, 1.5, 2, 3, 5, 7, 10, 13, 15, 17, 20, 22, 25, 27, 30, 32, 35, 37, 40, 42, 45, 47, 50}. And time-series data have 3 different lengths. Then we get 2,028 clusters as a final result. From the result, we excluded clusters whose size were less than 10. Thus, we evaluated the result, focusing on clusters in the result whose size is larger than or equal to 10. Table VII shows the results of 3-month time-series data. In Table VII, "X_{html}" and "X_{pdf}" denote the thresholds for the numbers of views and of downloads, respectively. "*All*" and "10 papers or more" denote the total number of clusters and the number of clusters which contain 10 or more elements, respectively. Further, "intermediately- and highly-cited" and "*Num*" denote the number of and the total paper cardinality of clusters, respectively, where 90 or more percent of the elements are intermediately- and highly-cited papers.

Table VII describes only results containing mostly intermediately- and highly-cited academic papers. In DTWS tree, no clusters with only highly-cited academic papers could be found, either. However, using 3-month time-series data, with 32 as the *view*-threshold and 47 as the *download*-threshold, approximately 80% of the intermediately- and highly-cited papers in Data1 could be successfully extracted. Similarly, approximately 72% of the intermediately- and highly-cited papers in Data2 could be extracted with 42 as the *view*-threshold and 15 as the *download*-threshold. As for 12-month time-series data, approximately 93% of the intermediately- and highly-cited papers in Data1 could be extracted with 37 as the *view*-threshold and 50 as the *download*-threshold. Approximately 79% of the intermediately- and highly-cited papers in Data2 could be extracted with 50 as the *view*-threshold and 40 as the *download*-threshold.

Further, evaluation of the clustering results was done focusing on one of the clusters using the *F*-value calculated from

the precision ratio P and recall ratio R of the results. R, P, and F are calculated by the formulas (13), (14), and (15), respectively. The precision ratio is expected to increase as the numbers of citations increase in a course of time. The weight β for the recall ratio is set to $\beta = 3.5$ in the formula (15).

$$R(\%) = \frac{relevant\ papers\ in\ the\ cluster}{relevant\ papers} \times 100 \quad (13)$$

$$P(\%) = \frac{relevant\ papers\ in\ the\ cluster}{papers\ in\ the\ cluster} \times 100 \quad (14)$$

$$F(\%) = \frac{(\beta^2+1)\times P\times R}{\beta^2\times P+R} \quad (15)$$

All the resultant clusters are sorted in the descending order of F-values with respect to the intermediately- and highly-cited scientific papers. Among them, the top 10 clusters are shown with precision ratios, recall ratios, and F-values in Table VIII. By inspecting Table VIII, it is known that there exist clusters with recall ratios over 75% and precision ratios over 90% for the intermediately- and highly-cited scientific papers. It is also known that 9 out of 10 clusters are based on 3-month time series data. The numbers as to the top 100 clusters, top 200 clusters, and top 300 clusters sorted by the F-values for the 3-month-, 6-month-, and 12-month time-series data are shown in Table IX. From Table IX, it is known that clusters from 3-month time-series data are relatively highly ranked with respect to F-values. Similarly, the top 10 clusters for the highly-cited papers sorted by F-values are shown in the descending order in Table X. From Table X, it is known that there exist clusters with recall ratios over 90% and precision ratios over 75%. The numbers as to the top 100 clusters, top 200 clusters, and top 300 clusters sorted by the F-values with respect to highly-cited papers for the 3-month-, 6-month-, and 12-month time-series data are shown in Table XI. From Table XI, it is known that clusters based on 3-month- and 6-month time-series data are relatively highly ranked.

5.2.3 Discussion

By k-means clustering, only highly-cited academic papers could not be extracted. In other words, at least under the k-means clustering, vectors made from time-series data of numbers of views and of downloads cannot effectively represent the characteristics of only highly-cited academic papers. However, by the results of the Kruskal-Wallis test, it is confirmed that there exist significant differences among the medians of numbers of citations as to clusters.

By merging multiple clusters with 90% or more precision ratios as to intermediately- and highly-cited papers based on DTWS tree clustering, 80% or more of the papers could be recalled. Further, it was found that there existed clusters containing intermediately- and highly-cited papers with the recall ratios of 75% or more and the precision ratios of 90% or more. It was also found that there existed clusters containing highly-cited papers with the recall ratios of 90% or more and the precision ratios of 75% or more. This indicates that vectors made from time-series data consisting of numbers of views and downloads can predict intermediately- and highly-cited papers under the DTWS tree clustering. The numbers of citations in the Data2 represent those of the scientific papers published 8 or more months ago. Therefore, by clustering 3-month time-series data by DTWS tree clustering, at best approximately 77% of the intermediately- and highly-cited papers published 8 or more months ago could be detected. Also, at best approximately 98% of the highly-cited papers academic papers could be detected. Further, clusters containing rather many low-cited scientific papers could be found. In summary, by clustering the 3-month time-series data by using DTWS tree clustering, each of low-, intermediately- and highly-, and highly-cited academic papers could be detected with high likelihood.

In this experiment, the number of days from publication was not sufficiently considered. Thus, the numbers of citations of papers with different publication dates were equally treated. It is expected that a clustering scheme considering exactly the number of days from publication can cluster the academic papers with higher accuracy.

6 CONCLUSION

We have clustered academic papers published in open access journals by using time series data of the numbers of views and downloads. We used k-means clustering and clustering DTWS tree clustering and compared the performances. As a result, clusters mostly containing intermediately- and highly-cited papers could be discovered. Indeed, the result described in this paper cannot directly confirm that highly-cited papers in the first year in open access journals can be predicted by at least the first three-month time series data of views and downloads based on the extracted features of these clusters. However, the obtained observation at least suggests that creation of classifiers based on the time-series data of downloading and browsing is promising. The discovery that clustering based on the *time-series data* can isolate highly-cited papers is significant for the relevant researchers and practitioners. Therefore, the following approaches to classification are among our future plan. To be more concrete, one possible approach is to directly use clusters created in the way proposed in this paper as a kind of classifier and classify papers into "will-be"-highly-cited papers if they are most similar to the clusters mostly consisting of highly-cited papers with respect to the first three-month time-series data. Another approach is to create a classifier based on the time-series data by using clusters (i.e., highly-cited, intermediately-cited, or low-cited) obtained by the proposed way as the training set. Of course, the latter approach will draw heavily upon further researches about how to represent features of time-series data and which methods to choose as a classifier.

DTWS tree clustering experiments in 3 different lengths of time-series data were compared. However, in k-means clustering, k was uniquely fixed and only 3-month time-series data were used. For this reason, there remain possibilities that the results were not properly compared among the two clustering methods. It is thought that other advanced methods such as x-means can remedy some of the above problems and improve the comparison results.

We used the numbers of views and downloads as of every month. This is because PLOS releases time-series data every month. However, if finer-grained time-series data, for example, of every day, are available, more accurate clustering of scholarly papers may be possible. Further, a wide range of

academic papers have been posted on PLOS. Therefore, by clustering papers in restricted areas, more specialized characteristics may be detected.

Further, as using 12-month time-series data for clustering lacks the immediacy of estimated papers, clustering based on data provided by Altmetrics such as posts in social media and links in social bookmarks is expected to overcome the problem. From our experimental results, the clustering results are expected to provide the features required for machine learning, that is, classification of highly-cited papers.

ACKNOWLEDGEMENTS

This research was supported by Grant-in-Aid for Research on Priority Areas, Tokyo Metropolitan University, "Research on social big data."

REFERENCES

[1] M. Laakso, et al., "The Development of Open Access Journal Publishing from 1993 to 2009," PLoS ONE 6(6): e20961. doi:10.1371/journal.pone.0020961 (2011).
[2] Ceek.jp Altmetrics http://altmetrics.ceek.jp/ Accessed (2013).
[3] J. Priem, H. A. Piwowar, and B. M. Hemminger, "Altmetrics in the wild: Using social media to explore scholarly impact," arXiv preprint arXiv:1203.4745 (2012).
[4] T. Ide,"Why does Subsequence Time-Series Clustering Produce Sine Waves?" Knowledge Discovery in Databases: PKDD 2006, Lecture Notes in Computer Science, Vol. 4213, pp. 211-222 (2006).
[5] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," Readings in speech recognition, A. Waibel and K. -F. Lee (Eds.). Morgan Kaufmann Publishers Inc., pp.159-165 (1990).
[6] G. Eysenbach, "Can Tweets Predict Citations? Metrics of Social Impact Based on Twitter and Correlation with Traditional Metrics of Scientific Impact," Journal of Medical Internet Research, Vol.13, No.4-e123 (2011).
[7] H. Nakahashi, et al.,"Automatic Alignment of Tweet," IEICE Technical reports, Vol.113, No.105, .pp.65-70 (2013) (in Japanese).
[8] PLOS http://www.plos.org/ Accessed (2014).
[9] Mlpy http://mlpy.sourceforge.net/ Accessed (2014).
[10] K. Nakamoto, et al., "Fast Clustering for Time-series Data with Average-time-sequence-vector generation Based on Dynamic Warping," Trans. JSAI, Technical papers, vol.18, No.3, pp.144-152 (2003) (in Japanese).
[11] R http://www.r-project.org/ Accessed (2014).

(Received September,28,2015)
(Revised January 21,2016)

Hiroshi Ishikawa received the B.S. and Ph.D degrees in Information Science from the University of Tokyo. After working for Fujitsu Laboratories and being a full professor of Shizuoka University, he is now a full professor of Tokyo Metropolitan University from April, 2013. His research interests include database, data mining, and social big data. He has published actively in international, refereed journals and conferences, such as ACM TODS, IEEE TKDE, VLDB, IEEE ICDE, and ACM SIGSPATIAL. He has authored some books, *Social Big Data Mining* (CRC Press). He is fellows of IPSJ and IEICE (The Institute of Electronics, Information and Communication Engineers) and members of ACM and IEEE.

Masaki Endo received his B.E. degree from Polytechnic University, Tokyo and graduated from the course of Electrical Engineering & Computer Science, Graduate School of Engineering Polytechnic University and received M.E degree from NIAD-UE, Tokyo. He is currently an assistant professor of Polytechnic University, Tokyo. His research interests include Web service and Web mining. He is also members of DBSJ, IPSJ and IEICE.

Iori Sugiyama received his B.E. degree from Tokyo Metropolitan University, Japan in 2014. He entered Tokyo Institute of Technology, Japan in 2014 and now he is a Master course student. His research interests include educational technologies and educational data mining. He is a member of Japan Society of Educational Technology (JSET).

Masaharu Hirota received his Doctor of Informatics in 2014 from Shizuoka University. Since April 2015, he has been working as assistant professor in National Institute of Technology, Oita College, Department of Information Engineering. His main research interests include geo social data, multimedia, and visualization. He is a member of ACM, IPSJ and IEICE.

Shohei Yokoyama is a lecturer of Informatics at Shizuoka University, Japan, and he was previously with National Institute of Advanced Industrial Science and Technology (AIST) for two years. His research concerns data engineering, and its focus is on georeferenced user-generated contents on Social Networking Service