International Journal of Data Science and Analytics (2018) 6:351–361
https://doi.org/10.1007/s41060-018-0152-8

REGULAR PAPER

Inferring variable labels using outlines of data in Data Jackets by considering similarity and co-occurrence

Teruaki Hayashi[1] · **Yukio Ohsawa**[1]

Received: 11 May 2017 / Accepted: 17 August 2018 / Published online: 1 September 2018

Abstract
The Data Jacket (DJ) is a technique for sharing information related to data, where the data are hidden, by summarizing them in natural language. In DJs, variables are described by variable labels (VLs), which are the names/meanings of variables, and the utility of data is estimated through combinations of VLs. However, DJs do not always contain VLs because the rules describing DJs cannot compel data owners to enter all relevant information. Owing to a lack of VLs in some DJs, even if the DJs can be combined, their combinations cannot be implemented through the string matching of the VLs. In this paper, we propose a method for inferring VLs in DJs using the text in their outlines. We focus on similarity among the outlines of DJs and create two models for inferring VLs, i.e., based on the similarity of the outlines and the co-occurrence of the VLs. We implemented our models on a similarity and a co-occurrence matrix and applied the proposed method to two types of test data: the DJs of public data and business data. The results of experiments show that our method is significantly superior to the technique that uses only the string matching of the VLs.

Keywords Data Jacket · Variable label · Meta-data · Co-occurrence

1 Introduction

The potential benefits of reusing and analyzing massive quantities of data have been discussed by various stakeholders from diverse domains [1,2]. However, the discussion has focused on the privacy and security of data. Acquisti and Gross [3] have pointed out that combining public databases may lead to a serious violation of privacy. Xu et al. [4] reviewed privacy issues related to data mining by differentiating the responsibilities of different users. From an overview of the prevalent scenario of data utilization and exchange, the cost of data management and security issues discourage private companies and individuals from opening or sharing their datasets. To solve these problems, the Data Jacket (DJ) has been developed as a technique for sharing information on data and considering their potential value, with the data themselves hidden, by summarizing them in natural language [5]. The idea of a DJ is to share "a summary of data" as meta-data without sharing the data themselves, which reduces the cost of data management and the risk of violation of privacy, and enables stakeholders to discuss combinations of data.

In discussions on data utilization and exchange using DJs, stakeholders start by discussing variable labels (VLs). A VL is the name/meaning of variables in data. Variables and values in data are summarized as VLs in DJs. The VLs are summaries of variables and values in a given dataset. For example, the dataset "Purchase history data in the supermarket" contains the variables "date," "customer ID," "customer name," "purchase items," and "purchase amount," and each variable contains values (Fig. 1). Even if the data themselves are not open, we can learn and determine the items useful for decision making from the summary in the DJs. Some data include private information, such values and variables as "customer name" and "purchase amount." The values cannot be shared, but the names of the variables "customer name" and "purchase amount" may be shared. By introducing DJs using VLs, stakeholders can learn the meanings of variables in data by posing and testing hypotheses about possible com-

This paper is an extended version of "Matrix-based Method for Inferring Variable Labels Using Outlines of Data in Data Jackets," read at the PAKDD'2017 Long Presentation.

✉ Teruaki Hayashi
teru-h.884@nifty.com; hayashi@sys.t.u-tokyo.ac.jp

Yukio Ohsawa
ohsawa@sys.t.u-tokyo.ac.jp

[1] Department of Systems Innovation, School of Engineering, The University of Tokyo, Tokyo, Japan

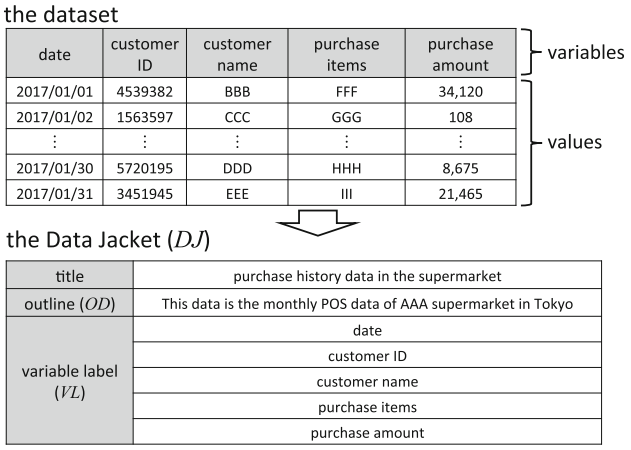

Fig. 1 An example of DJ (actually there are 12 items in the description of the DJs, but we show the core parts (title, outline, and variable labels) in the example)

binations of VLs to reduce the risk of data management and privacy.

Workshop-style methods using DJs have been proposed for the generation of feasible plans of data analysis. Once different stakeholders recognize the utility of the data, they can negotiate conditions for exchanging them. In the gamified workshops Innovators Marketplace on Data Jackets (IMDJ) [6,7] and Action Planning (AP) [8], data owners provide DJs that represent their data, and data analysts create solutions to solve the problems of data users, stated as requirements. In the processes of IMDJ and AP, participants negotiate for data exchange or buying/selling to create new businesses. As a result of this discussion and evaluation among participants, data owners are expected to learn how to use their data using a possible combination of DJs proposed by data analysts. Users are expected to learn how their requirements can be satisfied by the proposed plans. However, the DJs do not always contain VLs because the rules describing the DJs do not force data owners to enter all the information concerning their data. In other words, only information written by data owners is registered in DJs. Therefore, owing to a lack of description of VLs, DJs related to one another may not be linked, which makes it difficult to plan for data analysis and combinations. In this paper, we propose a method for inferring VLs not explicitly included in the outline of data. By focusing on similarities between outlines of data and the co-occurrence of VLs, we construct models according to the following two features:

1. When the similarity between the outlines of a pair of datasets is high, the two datasets are considered similar, and should have similar VLs.
2. When a pair of VLs (vl_i and vl_j) frequently appears in datasets, if vl_i appears, vl_j is considered to have appeared as well.

By modeling the features of VLs and using stored DJs as training data, even if a new DJ misses VLs, it is possible to infer them from the outlines. In the previous study on DJs, the co-occurrence of words in the outline of data [5,6] was used to discuss the combination of DJs, e.g., by using a visualization tool, such as KeyGraph [9]. Our method provides possible connections between DJs whose VLs are missing by inferring VLs. The contributions of our paper are many. It is the first approach to infer VLs by focusing on the similarity among outlines of datasets and the co-occurrence of VLs using DJs. The method to reveal related VLs from outlines of the data in DJs can encourage data utilization. In particular, it is important not only for knowledge discovery from data, but also for decision makers who want to acquire new data. Our method can show them a possible set of variables for decision making. Furthermore, the proposed models to connect DJs with missing VLs are extendable to various methods of calculation. In this paper, in addition to the similarity of data, we show the performance of the model by considering the co-occurrence of VLs.

There are a lot of co-occurrences in the world such as the frequency of words in documents and the coauthorship of articles. In the previous studies of natural language processing, various kinds of measures of contextual similarity based on co-occurrence statistics have been proposed. For example, McDonald and Lowe [10] used the co-occurrence statistics for calculating the semantic relatedness, and Matsuo and Ishizuka [11] proposed the method for extracting keywords using the co-occurrence of the words in a single document. On the other hand, Sarkar et al. [12] considered dynamic co-occurrence data of author-word links in papers published in successive years of the same conferences. However, the previous studies focused on the specific data such as articles or words in the documents. The point of this paper is not focusing on the co-occurrence of data in the specific domains but focusing on the co-occurrence of variables in data stored in different domains. In a previous study [13], we applied the proposed method to public data available on the Web, such as open data provided by local governments. In this paper, we adapt the method to DJs collected from different domains and discuss its performance.

2 Inferring variable labels

2.1 Our approach

The purpose of this study is to infer the VLs of DJs when they are unknown. Because the data are not open, it is impossible to determine the VLs by observing them. Therefore, we consider solving the problem using information about the data described in the DJs. We assume that 1) the datasets are similar when the information used to explain them is similar, and

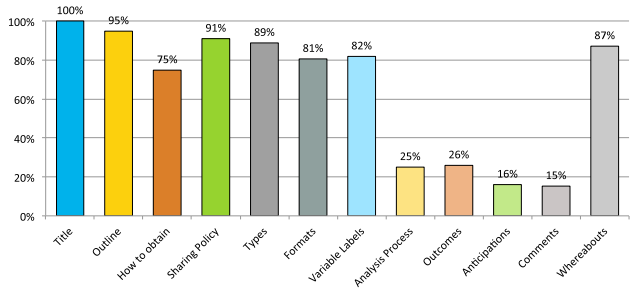

Fig. 2 The registration rates of 12 items

2) the datasets should have similar VLs when the similarity between them is high.

In this study, we introduce the outline of data (OD) as an indicator of the similarity of DJs. OD represents a description in natural language explaining the data. For example, the OD of the data shown in Fig. 1 is "These data are the monthly POS data of AAA supermarket in Tokyo." Although items of the sharing policy or the types of data may be suitable as ODs as well, these are selective entries not written in natural language. Items pertaining to ways to obtain data mostly consist of URLs. Although there are 12 items in the description of the DJs (the title, the outline, variable labels, the sharing policy of data, the format, and so forth), we consider ODs appropriate as characteristic of datasets because they provide text data, the registration rate of which is higher than that of other items. Figure 2 shows the registration rate of each item of the DJs (799 DJs were used in this experiment). The registration rate of the ODs was 95%, second only to the rate of titles, as the title of a data item is required upon DJ registration. Since 5% of DJs do not contain ODs, if DJs do not have ODs, we use the titles of data instead of their outlines. Thus, to infer unknown VLs of data, we propose a method to obtain a set of likely VLs from outlines of data.

2.2 Models

The aim is to obtain sets of likely VLs stored in training data of the DJs and VLs (L) by entering as a query an OD (OD_x) for DJs whose VLs are unknown. The sets of likely VLs are given by $\{vl \in L | f_n(vl, OD_x)\}$. $f_n(vl, OD_x)$ represents a condition whereby a set of the top n VLs (vl) is associated with a query (OD_x). To achieve the above aim, models were created as below.

2.2.1 Similarity of DJs based on ODs (Model 1)

This model is based on the assumption that when a pair of datasets have similar ODs, they have similar VLs. In this model, a scored set of VLs are obtained by considering the similarity between the ODs whose VLs are unknown.

Table 1 Notations used in this paper

Dimensions	
L	A unique set of VLs in training data
D	The number of ODs
V	The number of VLs
W	The number of terms
Matrices	
M	A Term-OD matrix ($W \times D$)
R	A VL-OD matrix ($V \times D$)
C	A VL co-occurrence matrix ($V \times V$)
E	A Term-VL matrix ($W \times V$)
Elements	
v_{ij}	The number of ith terms occurring in the jth OD
r_{ij}	The number of ith VLs occurring in the jth OD
e_{ij}	The number of ith terms linked with the jth VL
c_{ij}	The number of DJs containing a pair of VLs vl_i and vl_j
Performance measures	
Pr	Precision: the fraction of relevant VLs among the retrieved VLs
Re	Recall: the fraction of relevant VLs that have been retrieved over the total amount of relevant VLs
F	F-measure: the harmonic mean of precision (Pr) and recall (Re)

2.2.2 Co-occurrence of VLs (Model 2)

This model considers the co-occurrence of VLs, a feature whereby there may be a frequent pair of VLs appearing at the same time, e.g., "year" and "day," or "name" and "gender." By introducing this model along with Model 1, a scored set of VLs is obtained from the similarity between DJs using ODs when their VLs are unknown.

2.3 Inference process for obtaining VLs

We show the process of inference of VLs from ODs. The basic notations used in this paper are shown in Table 1. We use the bag-of-words and vector space models [15,17]. In the pre-processing steps, we conduct a morphological analysis of the text of ODs by (1) extracting words, (2) removing stop words and (3) restoring words to their original forms. The brief process is shown below, and the details of each step are provided in the following sub-subsections. In the last subsection, we show two examples of outputs (sets of inferred VLs) using the ODs of DJs whose VLs are unknown.

1. Carrying out pre-processing steps to ODs from training data and converting them into a Term-OD matrix M;
2. Extracting VLs from the training data and converting them into a VL-OD matrix R;

3. Obtaining a Term-VL matrix E by considering the similarity of the ODs (Model 1);
4. Creating a VL co-occurrence matrix C by considering the co-occurrence of VLs (Model 2); and
5. Obtaining a Term-VL matrix EC by considering the similarity of the ODs and the co-occurrence of the VLs (Models 1 and 2).

2.3.1 Term-VL matrix *E* (Model 1)

Based on Model 1, we develop an algorithm to calculate similarity among the training data of ODs. Following the pre-processing steps, in the first step, the ODs are converted into a matrix. Using the outlines of data as corpus, a Term-OD matrix M ($W \times D$) is obtained consisting of D-dimensional term vectors as rows and W-dimensional OD vectors as columns, with each element v_{ij} in an OD vector ($\boldsymbol{od_j}$) corresponding to the frequency with which a term (a row i) occurs in an OD (a column j), as shown in (1) and (2). Note that superscript T at the upper-right corner of the vectors represents transposition, the vectors are highlighted in bold.

$$M = (\boldsymbol{od_1}, \ldots, \boldsymbol{od_j}, \ldots, \boldsymbol{od_D}) \tag{1}$$

$$\boldsymbol{od_j} = (v_{1j} \ldots v_{ij} \ldots v_{Wj})^{\mathrm{T}} \tag{2}$$

In the second step, set of VLs in the DJs is converted into a VL-OD matrix. In the training data for the DJs, ODs and VLs are linked when they appear in the same DJ. A VL-OD matrix R ($V \times D$) consists of V-dimensional VL vectors as rows and D-dimensional OD vectors as columns, with each element r_{ij} in the jth OD vector ($\boldsymbol{od'_j}$) corresponding to the frequency (zero or one) with which the ith VL occurs in the jth OD, as shown in (3) and (4).

$$R = \left(\boldsymbol{od'_1}, \ldots, \boldsymbol{od'_j}, \ldots, \boldsymbol{od'_D}\right) \tag{3}$$

$$\boldsymbol{od'_j} = (r_{1j} \ldots r_{ij} \ldots r_{Vj})^{\mathrm{T}} \tag{4}$$

In the third step, we create a Term-VL matrix $E(= MR^{\mathrm{T}})$ ($W \times V$) from a Term-OD matrix M ($W \times D$) and the VL-OD matrix R ($V \times D$) obtained in the second step. This process is equivalent to mapping the ith ($1 \leq i \leq V$) D-dimensional VL vector in the OD space into W-dimensional term space using the Term-OD matrix M. The Term-VL matrix E is represented as follows:

$$\begin{aligned} E &= MR^{\mathrm{T}} \\ &= (\boldsymbol{vl_1}, \ldots, \boldsymbol{vl_j}, \ldots, \boldsymbol{vl_V}) \end{aligned} \tag{5}$$

$$\boldsymbol{vl_j} = (e_{1j} \ldots e_{ij} \ldots e_{Wj})^{\mathrm{T}} \tag{6}$$

$$e_{ij} = \sum_{k=1}^{D} v_{ik} r_{kj} \tag{7}$$

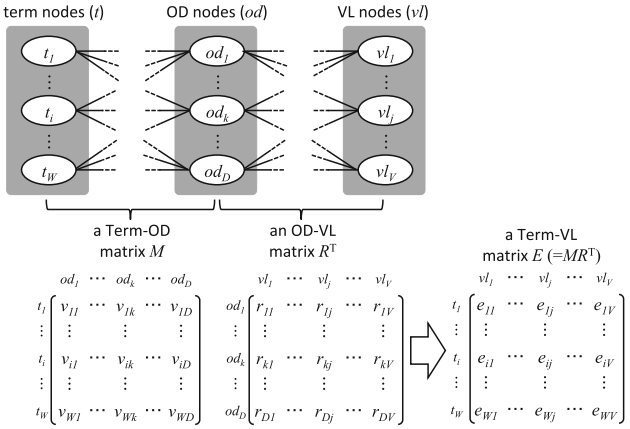

Fig. 3 Term-VL matrix E in the tri-partite graph

The element e_{ij} means that the sum of the product of the frequency (v_{ik}) with which the ith term (t_i) occurs in the kth OD (od_k) and the frequency (r_{kj}) with which the jth VL (vl_j) links with the kth OD (od_k). In other words, e_{ij} represents the number of ODs related to both a term (t_i) and a VL (vl_j). Moreover, the Term-VL matrix E is equivalent to the adjacency matrix of the tri-partite graph consisting of three disjoint sets of nodes, i.e., terms, ODs, and VLs (Fig. 3). The element e_{ij} of the Term-VL matrix E represents the number of paths from the ith term (t_i) to the jth VL (vl_j) by way of OD nodes.

Through the above process, Model 1 is implemented as the Term-VL matrix E. Using this matrix, a scored set of VLs is obtained considering the similarity between ODs in matrices E and OD_x, the VLs of which are unknown. When OD_x is given, a W-dimensional feature vector of OD_x ($\boldsymbol{od_x}$) is obtained following pre-processing consisting of morphological analysis. By comparing the similarity of $\boldsymbol{od_x}$ and each W-dimensional feature vector of VL ($\boldsymbol{vl_j}$ ($1 \leq j \leq V$)) in matrix E, a scored set of VLs is obtained. Figure 4 is the example of the Term-VL matrix E. When we send a free text query "local facility according to age," the algorithm of Model 1 extracts the terms which are included in the training data and calculates the similarity of the feature vector of the query and the feature vectors of each VL. Then, it returns "address," "region," "population," and "age" as a set of VLs with similarity scores.

2.3.2 Term-VL matrix *EC* (Models 1 and 2)

We combine Model 2 with Model 1 by considering the co-occurrence of VLs. We assume that any pair of VLs in the same DJ occurs once. To combine this with the Term-VL matrix E created in Model 1, we create a VL co-occurrence matrix C ($= RR^T$ ($V \times V$)) the element c_{ij} of which represents the number of DJs containing a pair of VLs vl_i and vl_j (8). In other words, an element c_{ij} in the VL co-occurrence

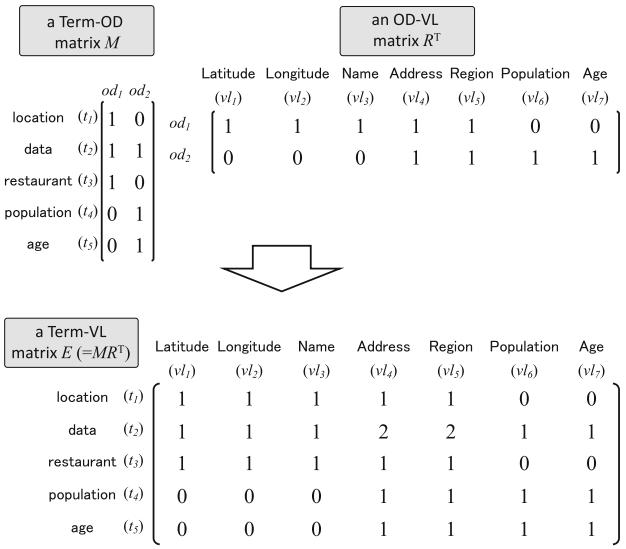

Fig. 4 An example of the Term-VL matrix E

matrix C is represented as in (9), where $|vl_i|_{od_s}$ represents the frequency of vl_i in od_s $(1 \leq s \leq D)$.

$$c_{ij} = \sum_{k=1}^{D} r_{ik} r_{kj} \tag{8}$$

$$= \sum_{s=1}^{D} |vl_i|_{od_s} |vl_j|_{od_s} \tag{9}$$

Finally, a Term-VL matrix EC is generated by the product of the Term-VL matrix E (5) and the VL co-occurrence matrix C. The Term-VL matrix EC consists of V-dimensional term vectors as rows and W-dimensional VL vectors as columns, and has the same structure as Term-VL matrix E. The difference between E and EC is in whether the co-occurrence of VLs (Model 2), i.e., the elements of the matrices, is considered.

The element e_{ij} of matrix E is given as (7), which represents the number of ODs related to both a term (t_i) and a VL (vl_j). On the contrary, element g_{ij} of matrix EC is given as follows:

$$g_{ij} = \sum_{m=1}^{V} \left(\sum_{k=1}^{D} v_{ik} r_{km} \right) \left(\sum_{l=1}^{D} r_{ml} r_{lj} \right) \tag{10}$$

The element g_{ij} represents the value of the similarity between ODs and queries (the function of matrix E), and the co-occurrence of VLs (the function of matrix C). In other words, the Term-VL matrix EC is equivalent to the adjacency matrix of the five-partite graph, which consists of five disjoint sets of nodes, i.e., terms, ODs, VLs, ODs, and VLs (Fig. 5). Element g_{ij} represents the number of paths from the ith term

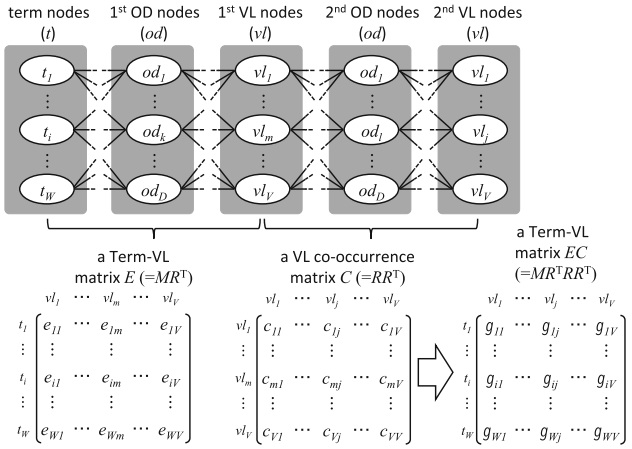

Fig. 5 Term-VL matrix EC in the 5-partite graph

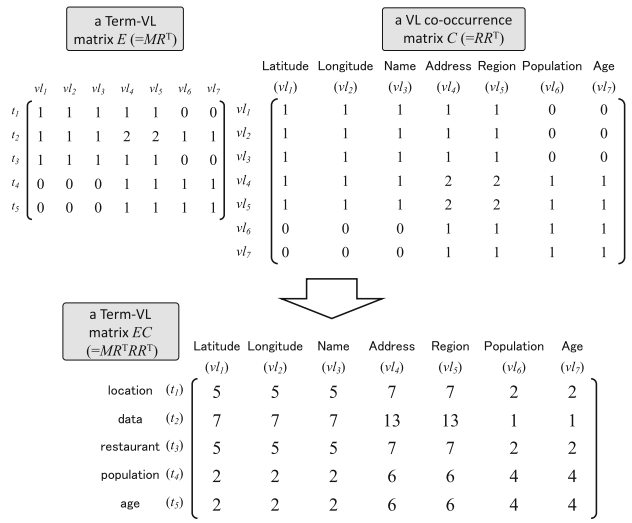

Fig. 6 An example of the Term-VL matrix EC

(t_i) to the jth VL (vl_j) in the second VL node, by way of the first OD node, the first VL node, and the second OD node.

When OD_x is given, a W-dimensional feature vector of OD_x $(\boldsymbol{od_x})$ is obtained. By comparing the similarity between $\boldsymbol{od_x}$ and each W-dimensional feature vector of VL $(\boldsymbol{vl_j}$ $(1 \leq j \leq V))$ in matrix EC, a scored set of VLs is obtained. Figure 6 is the example of the Term-VL matrix EC. When we send a free text query "local facility according to age," the system extracts the terms which are included in the training data and calculates the similarity of the feature vector of the query and the feature vectors of each VL. Then, it returns "latitude," "longitude," "name," "address," "region," "population," and "age" as a set of VLs with similarity scores. Compared with the example in Fig. 4, "latitude," "longitude," and "name" can be retrieved by considering the co-occurrence of VLs, which cannot be found only with the OD similarity.

Table 2 Example 1 using matrix E

Inferred VL	Similarity
Languages that can be offered	0.381758
Languages understood by foreigners	0.381758
National origin	0.317921
Attractions in Tokyo	0.277441
Number of visits by foreigners	0.277441
Number of visitors	0.277441
Experience with or without activity	0.272505
Attribute of visitors (age)	0.272505
Consumed amount	0.272505
Purchase	0.272505

Table 3 Example 1 using matrix EC

Inferred VL	Similarity
Languages that be offered	0.381758
Languages understood by foreigners	0.381758
Satisfaction level of visit	0.277441
Attractions in Tokyo	0.277441
Number of visits by foreigners	0.277441
Number of visitors	0.272505
Experience with or without activity	0.272505
Attribute of visitors (age)	0.272505
Consumed amount	0.272505
Purchase	0.272505

Table 4 Example 2 using matrix E

Inferred VL	Similarity
Total population of farmers	0.313810
Total agricultural workforce	0.313810
Number of births	0.313131
Number of deaths	0.313131
Agricultural workforce (male)	0.312155
Agricultural workforce (female)	0.312155
Number of full-time farmers	0.312155
Number of part-time farmers	0.312155
Every 5 years	0.311423
Number of increases and decreases	0.311423

Table 5 Example 2 using matrix EC

Inferred VL	Similarity
Number of births	0.349185
Number of deaths	0.349185
In-migrants	0.334844
Fatalities	0.334844
Out-migrants	0.334844
Population	0.321476
Number of households	0.317914
Population (male)	0.317914
Population (female)	0.317914
Fertility	0.317914

2.4 Example

Let us show the two examples using following ODs.

- *Example 1*: "Data on amount of beer consumed by foreign tourists visiting Japan at a restaurant"
- *Example 2*: "These data represent the transition of population each year in Japan"

Tables 2 and 3 show the top 10 inferred VLs using example 1 and the VLs of which were unknown (the experimental conditions for obtaining the inferred results are explained in detail in the following section). Moreover, the OD did not exist in the training data of the DJs. The inference from matrix E, using only the similarity of ODs, and from matrix EC, considering both the co-occurrence of VLs and the similarity of ODs, seems highly related to VLs in the OD.

On the contrary, the inferred results are sometimes different between matrices E and EC when using an OD such as example 2. Tables 4 and 5 show different results. The list using matrix E shown in Table 4 indicates that some VLs were not related to the OD, e.g., "total population of farmers" or "total population of agricultural workforce," because of the influence of the highly similar training data for "agricultural population." The list using matrix EC shown in Table 5 yielded better results. It may be possible to infer related VLs in ODs whose VLs are unknown by introducing the models based on the similarity of ODs and the co-occurrence of VLs.

3 Experimental details

3.1 Purpose

The purpose of the experiment was to assess the system's capability to infer VLs from ODs whose VLs were unknown by using the similarity of ODs and the co-occurrence of the VLs. We introduce the string matching (TSM) a comparative method with Term-VL matrices E and EC. This is because when someone retrieves data from a description thereof, a method using the string matching with VLs by employing the outlines of data as query can be considered. The function of TSM ($f'_n(vl, OD_x)$) is to obtain sets of VLs ($\{vl \in L | f'_n(vl, OD_x)\}$) stored in the training data of VLs

Table 6 Training data (corpus) statistics

Number of Data Jackets	799
Average number of terms in each OD	39.5
Average number of VLs in each Data Jacket	5.34
Unique terms in ODs	1935
Total number of VLs	4160
Unique variable labels	3216

Table 7 Test data statistics

	Public data	Business data
Number of Data Jackets	50	50
Average number of terms	36.7 ± 8.80	50.7 ± 43.2
Average number of VLs	4.70 ± 1.71	6.60 ± 4.28
Total number of VLs	398	2605
Unique VLs	131	1862

that match the terms in the ODs by entering OD (OD_x), the VLs of which are unknown, as a query. The entered ODs are converted into a bag of words in the same manner as in our proposed method. The obtained VLs are scored in descending order of the number of acquisitions.

3.2 Training and test data

In this paper, we used 799 DJs containing both ODs and VLs collected from business persons, researchers, and data holders interested in data utilization in various domains. The same training data were used in a previous study [13]. Each DJ was constructed from an OD and several VLs. There were 3215 unique VLs in total. The corpus and the dictionary were constructed from all words in text of the ODs. We removed punctuation marks and symbols in texts as stop words, restored words to their original forms, and extracted nouns, verbs, adverbs, and adjectives appearing more than once. The corpus consisted of approximately 2000 unique words. We used MeCab[1] for morphological analysis [14], as it is a common tool for analyzing morphemes of Japanese texts. Detailed information concerning the training data is shown in Table 6. To weight the discriminative terms in the DJs, we introduced tf–idf to the weighting scheme [16], which is reliable at identifying distinctive terms in each DJ. Term frequency (tf) is the number of times a term appears in a document, and the inverse document frequency (idf) diminishes the weight of frequent terms in all documents, and increases the weights of terms appearing rarely.

As test data, we collected two types of DJs, public data and business data. Public data refers to data available to the public, including datasets already published on the Web or those that are disclosed when requested. In this experiment, we used 50 DJs of public data from the Open Data of Shizuoka prefecture in Japan,[2] which publishes governmental records on the Web. Business data refers to data unavailable in the public domain, including private data. We collected 50 DJs of business data from businesspeople who had registered their data as DJs. All DJs for public data and business data contained ODs and VLs. The detailed information concerning the test data is shown in Table 7. We compared different test datasets because the experimental conditions using business data were stricter than those using public data. The public formed DJs of open data collected from a public organization, but the business data were DJs including the private data of companies from diverse domains. Of course, the variety of VLs and the number of terms in ODs in business data were larger than those in public data. The average numbers of terms and VLs in business data were significantly larger than those in public data according to an unpaired t-test assuming unequal variance (OD: $t(98) = 1.86$, $p < 0.05$, VL: $t(98) = 2.60$, $p < 0.01$). In this study, because we aim to infer VLs to encourage cross-disciplinary data collaboration, we conducted experiments using business data as test datasets.

3.3 Method and evaluation

We prepared 100 DJs as test data (public data and business data) and extracted ODs from them. Using these ODs as queries, we compared each of their feature vectors with feature vectors of the VLs in the Term-VL matrices E and EC, and obtained the sets of VLs in descending order of similarities. The similarity scores of OD_x and vl_j were calculated as cosine similarities $sim(\boldsymbol{od_x}, \boldsymbol{vl_j}) = \boldsymbol{od_x} \cdot \boldsymbol{vl_j} / |\boldsymbol{od_x}||\boldsymbol{vl_j}|$. For the evaluation of the results, we used precision, recall, and the F-measure. We define precision as $Pr = TP/(TP + FP)$ and recall as $Re = TP/(TP + FN)$, using the top 10 VLs returned as the inferred results scored by similarities, where TP = true positives, FP = false positives, and FN = false negatives. The F-measure is defined as $F = 2 \cdot Pr \cdot Re/(Pr + Re)$. Finally, by calculating the average F-measure of each query, we compared the performance of matrix E, matrix EC, and TSM.

We defined average similarity (AS) as in (11) by considering the relationships between ODs and VLs according to similarity to compare the performance of matrices E and EC. L_{od_q} represents the set of correct VLs in od_q, and $rel(od_q, vl_p)$ is an indicator function equivalent to one if vl_p is the correct VL, i.e., $vl_p \in L_{od_q}$, and is zero otherwise. As well as the F-measure, by calculating AS for each query, we compared the performance of matrices E and EC using a paired t-test. Although MAP (mean average precision) was

[1] http://taku910.github.io/mecab/.

[2] http://open-data.pref.shizuoka.jp/.

Table 8 Top 10 ranks using public data (Average Scores ± Standard Deviation)

	F-measure	Precision	Recall
TSM	0.110 ± 0.091	0.082 ± 0.068	0.185 ± 0.173
Matrix *E*	0.235 ± 0.178	0.174 ± 0.131	0.401 ± 0.331
Matrix *EC*	0.196 ± 0.183	0.146 ± 0.133	0.332 ± 0.337

Table 9 Average similarity of public data (Average Scores ± Standard Deviation)

	Mean *AS*	Mean $\overline{AS}$
Matrix *E*	0.329 ± 0.113	0.069 ± 0.014
Matrix *EC*	0.399 ± 0.095	0.111 ± 0.016
p-value	**	**

$^{**}p < 0.01$; $^{*}p < 0.05$, *n.s.* non significance

available to assess the ranking of the inferred results [18], our method *AS* focused on their similarity. In a DJ, each VL is equally linked to an OD, i.e., there is no order among the VLs in the DJs. For example, the VLs "day," "month," and "weather" formed part of weather data. Therefore, in this experiment, we did not evaluate the inferred results using MAP but *AS*.

$$AS_{od_q} = \frac{1}{|L_{od_q}|} \sum_{p=1}^{V} \left(sim(\boldsymbol{od_q}, \boldsymbol{vl_p}) \cdot rel(od_q, vl_p) \right) \tag{11}$$

4 Result and discussion

4.1 Results using public data

We obtained the top 10 VLs as inferred results scored by similarities from each query using matrices *E* and *EC*, and obtained the top 10 VLs from the string matching of ODs and VLs. Comparing the *F*-measure from the precision and recall of each method, the inferred results using matrices *E* and *EC* yielded better performance than when only TSM was used (Table 8). Matrices *E* and *EC* were 2.14 times and 1.78 times better, respectively, in terms of scores *F*-measure than TSM. This shows that although the outline of data is an important attribute for characterizing them, it do not always contain information about VLs. In other words, the string matching of ODs and VLs is not sufficient to infer VLs in data. Moreover, when comparing the *F*-measures of matrices *E* and *EC* with the paired *t*-test, no significant difference was found between the Term-VL matrices *E* and *EC* ($t(98) = 1.61$, $p = 0.110$). This shows that the inferred sets of VLs were almost identical in the Term-VL matrices *E* and *EC*.

On the contrary, comparing the average value of *AS* for evaluating similarities among the inferred VLs, we found significant differences between matrices *E* and *EC* as shown in Table 9 ($t(98) = 9.52$, $p < 0.01$). Although the inferred results were almost identical in terms of *F*-measure, the values of *AS*, the criterion of similarities of the VLs of ODs, obtained higher scores with matrix *EC* than *E*. In terms of number, the similarity of correct sets of VLs increased in 48 of 50 test data items when matrix *EC* was used.

However, it is possible that the similarity among incorrect VL sets with queries also increased by using matrix *EC*. Therefore, we define the average similarity of incorrect VL sets as $\overline{AS}$ (12). $|L \cap \overline{L_{od_q}}|$ represents the number of VLs not included in od_q, and $unrel(od_q, vl_p)$ is an indicator function equivalent to one if vl_p is the incorrect VL, i.e., $vl_p \notin L_{od_q}$, and zero otherwise.

$$\overline{AS_{od_q}} = \frac{\sum_{p=1}^{V} \left(sim(\boldsymbol{od_q}, \boldsymbol{vl_p}) \cdot unrel(od_q, vl_p) \right)}{|L \cap \overline{L_{od_q}}|} \tag{12}$$

Applying (12) to 50 public test data, we compared $\overline{AS}$ values of matrices *E* and *EC*, and found that $\overline{AS}$ of Term-VL matrix *EC* were significantly higher than those of matrix *E* ($t(98) = 30.7$, $p < 0.01$). This shows that the similarity among incorrect VL sets with queries increased when matrix *EC* was used. However, comparing the points of increase of *AS* (the correct VL sets) and $\overline{AS}$ (the incorrect VL sets) in each query with a paired *t*-test, those of *AS* were significantly higher than the points of increase of $\overline{AS}$ ($AS : 0.0707$, $\overline{AS} : 0.0422$, $t(98) = 4.47$, $p < 0.01$). This result shows that similarities among correct VL sets with ODs significantly increase by introducing Term-VL matrix *EC*.

4.2 Results using business data

We compared performances using the DJs of business data, which are more practical test data than public data. In the same manner as with the public data, we obtained the top 10 VLs using matrix *E*, matrix *EC*, and TSM. Comparing the *F*-measures calculated from precision and recall, the scores decreased with matrix *E*, matrix *EC*, and TSM compared with scores using public data (Table 10). However, the inferred results using matrices *E* and *EC* still yielded better performance than those obtained using TSM. The main reason for the decline in performance is the variety of VLs. The fact that most VLs in business data appeared only once in the training data might have made it difficult to infer VLs from ODs, even if the numbers of ODs in each DJ were larger in the business data than in public data, as shown in Table 7. In case of cross-disciplinary data collaboration, it may be the case that even if a pair of variables has the same definition, two variables are represented by different words, such

Table 10 Top 10 ranks using business data (Average Scores ± Standard Deviation)

	F-measure	Precision	Recall
TSM	0.039 ± 0.079	0.024 ± 0.047	0.153 ± 0.331
Matrix *E*	0.124 ± 0.119	0.078 ± 0.078	0.403 ± 0.411
Matrix *EC*	0.097 ± 0.102	0.060 ± 0.063	0.324 ± 0.386

Table 11 Average similarity of business data (Average Scores ± Standard Deviation)

	Mean *AS*	Mean $\overline{AS}$
Matrix *E*	0.190 ± 0.100	0.025 ± 0.008
Matrix *EC*	0.230 ± 0.111	0.037 ± 0.013
p-value	**	**

$^{**}p < 0.01$; $^{*}p < 0.05$, *n.s.* non significance

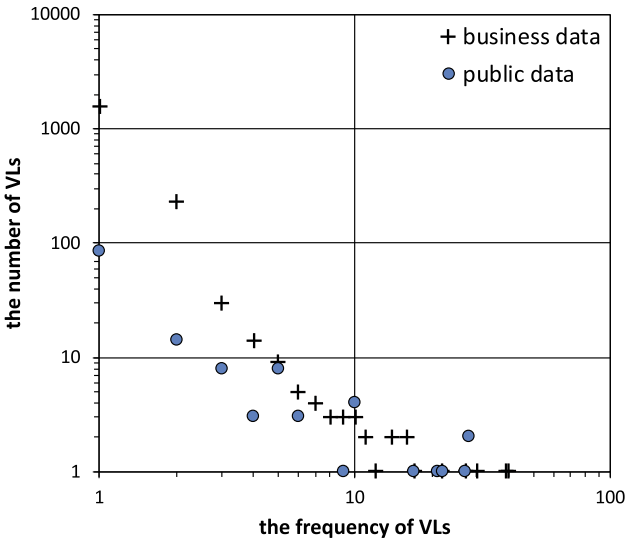

Fig. 7 Logarithmic graph of the distribution of the frequency of appearance of VLs

as "location" and "address" when they are stored in different domains. On the contrary, most VLs in public data appeared more than once in the training data. Moreover, because the meanings of variables were mostly unified in the datasets collected from a single domain, such as local governments, the accuracy of inference using public data was higher than that using business data.

We compared *F*-measure scores of matrices *E* and *EC* with a paired *t*-test, and found no significant difference in the Term-VL matrices *E* and *EC* ($t(98) = 1.55$, $p = 0.123$). Moreover, comparing the average values of *AS* and $\overline{AS}$ for evaluating similarities in the inferred VLs, we found significant differences between matrices *E* and *EC* as shown in Table 11 (*AS*: $t(98) = 6.35$, $p < 0.01$, $\overline{AS}$: $t(98) = 10.5$, $p < 0.01$). In terms of number, the similarity among the correct sets of VLs increased in 47 of 50 test data items when introducing matrix *EC*. Moreover, comparing the points of increase of *AS* and $\overline{AS}$ in each query with a paired *t*-test, those of *AS* were significantly higher than the points of increase of $\overline{AS}$ (*AS*: 0.0402, $\overline{AS}$: 0.0122, $t(98) = 4.43$, $p < 0.01$). These results show that the model considering the co-occurrence of VLs works well to improve the similarity of VLs for business data as well, and although the scores were lower than when using public data, the evaluations of inferred sets of VLs were almost identical in matrices *E* and *EC* even when using business data.

4.3 Summary of discussion

The results of our experiments suggest that the model where "the similarity of outlines of a pair of datasets is high, the datasets are considered similar, and should have similar VLs," works well for inferring VLs. In other words, the information concerning other datasets (the relationship between ODs and VLs) may compensate for missing terms to explain data and work well to discover VLs from outlines of data whose VLs are unknown. These results show that the model that considers the co-occurrence of VLs when a pair of VLs (vl_i and vl_j) frequently appears in datasets, and if vl_i appears, considers vl_j to have appeared, may work well to improve the similarity of VLs in ODs.

Although the similarity of VLs to the query improves by the model considering the VL co-occurrence, the reason why the performances are inferior to the model considering only the OD similarity is that the VLs follow the power distribution. Figure 7 shows the distributions in which the frequency of appearance of the VLs is the horizontal axis, and the number of VLs for that frequency of appearance is contained on the vertical axis. As shown in the figure, approximately 100 VLs in public data and 1000 VLs in business data had a frequency of 1, which made it difficult to infer VLs accurately. The exponent of public data is 2.03 (coefficient of the determination 0.922) and that of business data is 3.05 (coefficient of the determination 0.773), and these show that they follow the power distribution. To avoid the influence of low frequent VLs, it is better to remove low frequent VLs as the noises, which may improve the performance. However, there are also low frequent VLs which are important as elements constituting data. For example, the VL "amount of electricity generation from imported natural gas per year" in the data of "Turkey's dependence on imported natural gas in the electricity generation" or the VL "normalized running distance" in the data of "Football Player Stats" are the core VLs in each data. Therefore, we consider that the approach of removing VLs by their frequency may be avoided. In other word, rather than reducing VLs in the test data and training data, it will be important to focus on the meaning and the semantics of variables to improve performance. Moreover, in the future study,

it is important to the degree distribution and the centralities of VLs in the co-occurrence network of VLs.

5 Conclusion

In this paper, we proposed a method for inferring variable labels from outlines of data whose were missing or unknown. Focusing on the similarity among outlines of data in DJs and the co-occurrence of variable labels, we constructed two models according to the features of the DJs. By modeling the features of variable labels and outlines of data, we found that even if a query DJ misses variable labels, it is possible to infer variable labels from outlines of the DJ. The results of experiments suggested that the model where "when similarity between the outlines of a pair of datasets is high, the datasets are considered similar, and should have similar variable labels," works well for inferring variable labels. Moreover, when we consider not only the similarity of outlines, but also the co-occurrence of variable labels, the similarity of variable labels to the outlines of data improves. When someone retrieves variable labels from a description of data, a method using the string matching with variable labels may be considered. However, outlines of data do not always include terms corresponding to variable labels. There is a problem whereby decision makers who want to acquire new data cannot discover information about the kinds of data (set of variables) that should be obtained. Our proposed method may be helpful in encouraging data acquisition and utilization for the purpose of knowledge discovery.

In this study, because the outlines of data were small but contained a certain number of terms, it was possible to discuss and compare similarities in the vector space model by creating a term-document matrix. However, a variable label is a very small element composed of one or several words. Because the description of DJs allows variable labels written in natural language, even if the variable labels have the same meaning, they are sometimes presented differently in descriptions, e.g., "location" and "address," "the number of births" and "fertility," or "the number of deaths" and "fatalities." This affected to the results using business data. In future work, we aim to construct a model that considers the meanings of variable labels and their synonyms, even if they have brief descriptions or appear only once. Moreover, in our models, it is impossible to retrieve related variables from the free text queries if there are no connections between variable labels and the terms in outlines. It is important to consider the term sense ambiguity in the future. lexical processing and latent semantic indexing using singular value decomposition may be helpful for improving the performances. This study has been developed as a technique for supporting decision making in data utilization and exchange. It is important to validate the performance of the application using our proposed method in workshops on IMDJ or AP.

Acknowledgements This study was partially supported by JST-CREST Grant Number JPMJCR1304, and JSPS KAKENHI Grant Numbers JP16H01836 and JP16K12428. We also thank all staff members of KKE (Kozo Keikaku Engineering Inc.) for supporting our research. We would like to thank Editage (www.editage.jp) for English language editing.

References

1. Rabinovich, E., Cheon, S.: Expanding horizons and deepening understanding via the use of secondary data sources. J. Bus. Logist. **32**(4), 303–316 (2011)
2. Ellram, M.T., Tate, L.W.: The use of secondary data in purchasing and supply management (P/SM) research. J. Purch. Supply Manage. **22**(4), 250–254 (2016)
3. Acquisti, A., Gross, R.: Predicting social security numbers from public data. Proc. Natl. Acad. Sci. **106**(27), 10975–10980 (2009)
4. Xu, L., Jiang, C., Wang, J., Yuan, J., Ren, Y.: Information security in big data: privacy and data mining. IEEE Access **2**, 1149–1176 (2014)
5. Ohsawa, Y., Kido, H., Hayashi, T., Liu, C.: Data Jackets for synthesizing values in the market of data. In: 17th International Conference in Knowledge Based and Intelligent Information and Engineering Systems, vol. 22, pp. 709–716 (2013)
6. Ohsawa, Y., Liu, C., Suda, Y., Kido, H.: Innovators Marketplace on Data Jackets for externalizing the value of data via stakeholders' requirement communication. In: Proceedings of AAAI 2014 Spring Symposium on Big Data Becomes Personal: Knowledge into Meaning, AAAI Technical Report, pp. 45–50 (2014)
7. Ohsawa, Y., Kido, H., Hayashi, T., Liu, C., Komoda, K.: Innovators Marketplace on Data Jackets, for valuating, sharing, and synthesizing data. In: Knowledge-Based Information Systems in Practice, Smart Innovation, Systems and Technologies, vol. **30**, pp. 83–97. Springer (2015)
8. Hayashi, T., Ohsawa, Y.: Processing combinatorial thinking: Innovators Marketplace as role-based game plus Action Planning. Int. J. Knowl. Syst. Sci. **4**(3), 14–38 (2013)
9. Ohsawa, Y., Benson, N.E., Yachida, M.: KeyGraph: automatic indexing by co-occurrence graph based on building construction metaphor. In: Proceedings of Advanced Digital Library Conference, pp. 12–18 (1998)
10. McDonald, S., Lowe, W.: Modeling functional priming and the associative boost. In: Proceedings of the Twentieth Annual Conference of the Cognitive Science Society, pp. 675–680 (1998)
11. Matsuo, Y., Ishizuka, M.: Keyword extraction from a single document using word co-occurrence statistical information. Int. J. Artif. Intell. Tools **13**(1), 157–159 (2004)
12. Sarkar, P., Siddiqi, M.S., Gordon, J.G.: A latent space approach to dynamic embedding of co-occurrence data. In: Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics, vol. **2**, pp. 420–427 (2007)
13. Hayashi, T., Ohsawa, Y.: Matrix-based method for inferring variable labels using outlines of data in Data Jackets. In: Proceedings of PAKDD (2017)
14. Kudo, T., Matsumoto, Y.: Japanese dependency structure analysis based on support vector machines. In: Proceedings of EMNLP, pp. 18–25 (2000)

15. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975)
16. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Inf. Process. Manage. **24**(5), 513–523 (1988)
17. Turney, P.D., Pantel, P.: From frequency to meaning: vector space models of semantics. J. Artif. Intell. Res. **37**, 141–188 (2010)
18. Buckley, C., Voorhees, E.M.: Evaluating evaluation measure stability. In: Proceedings of SIGIR, pp. 33–40 (2000)