IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Metadata-based Clustering and Selection of Metadata Items for Similar Dataset Discovery and Data Combination Tasks

TAKESHI SAKUMOTO[1], TERUAKI HAYASHI[2], HIROKI SAKAJI[3], HIROFUMI NONAKA[4,5]

[1]Nagaoka University of Technology, Nagaoka, Niigata, 940-2188 Japan (e-mail: s183353@stn.nagaokaut.ac.jp)
[2]The University of Tokyo, Bunkyo, Tokyo, 113-8656 Japan (e-mail: hayashi@sys.t.u-tokyo.ac.jp)
[3]Hokkaido University, Sapporo, Hokkaido, 060-0814 Japan (e-mail: sakaji@ist.hokudai.ac.jp)
[4]Aichi Institute of Technology, Toyota, Aichi, 470-0392 Japan
[5]mayolab Co., Ltd., Nagaoka, Niigata, 940-2137, Japan

Corresponding author: Takeshi Sakumoto (e-mail: s183353@stn.nagaokaut.ac.jp).

This work was supported by JSPS KAKENHI Grant Numbers JP20H02384 and JP19K12116.

ABSTRACT Data integration, which aims to solve problems and create new services by combining datasets, has attracted considerable attention. The discovery of similar datasets that can be combined is critical. In the literature on similar dataset discovery, it is important to select an appropriate discovery method for each information need, such as the domain. However, conventional studies have evaluated discovery methods in different ways, such as domains, test datasets, and evaluation metrics. This factor prevents the appropriate method selection for each situation. Furthermore, the specific effects of the combination of different methods are not well known despite conventional studies arguing the importance of the combination. This study attempts to understand (1) the similarity indicators that should be employed for each domain and (2) the effects of a combination of different indicators on performance. We evaluated 16 inter-dataset clustering models based on different metadata-based similarity indicators, using unified evaluation metrics and datasets for 15 domains. Our results (1) suggest that similarity indicators should be used for each domain and (2) demonstrate that most of the combinations of different methods can improve clustering performance.

INDEX TERMS dataset discovery, dataset similarity, clustering, data exchange platform, metadata

I. INTRODUCTION

DATA integration, which aims to solve problems and create new services by combining datasets, has attracted significant attention [1], [2]. This places the discovery of combinable datasets as one of the most critical issues. With improvements in computer processing power and the development of cloud data services, platform providers have developed business ecosystems in which large and diverse datasets are distributed [3]–[6]. There are some tools to discover appropriate datasets from such enormous candidates by applying information-retrieval techniques [7], [8]. Furthermore, there has recently been growing interest in approaches based on dataset similarity rather than simple query matching. It is impractical for users to understand the perfect query words that represent the required datasets in advance [9]. This allows users to avoid constructing appropriate query words and to carry richer semantics [10]. Therefore, conventional studies have proposed numerous similarity indicators for datasets.

Similarity evaluation for open datasets primarily relies on metadata-based indicators [10]–[19]. Metadata is the summarized information about datasets, such as titles, description texts, and tags. Metadata has a unified data structure independent of the modalities and domains of the datasets. It allows similarity evaluation between datasets with different data structures, contrary to approaches based on actual dataset contents. Metadata-based dataset similarity indicators have various directions, such as vector similarity between word embeddings [10], [12], [20] and graph distance between ontology concepts [14], [15], [17].

Although conventional studies have argued the importance of selecting appropriate methods [10], more practical discussions should be conducted. This research field is still in its infancy. Therefore, each study used different evaluation datasets and metrics. For example, Bernhauer et al. [10] used datasets from the Czech National Open Data Catalog, and evaluated their results using Precision@k and PR-AUC. On the other hand, Wang et al. [14] used gold standards in Elsevier DataSearch and evaluated their results mainly using nDCG and F-measure. Consequently, it is difficult to understand an appropriate method by quantitatively comparing their experimental results. In addition, many conventional studies have limited the evaluation domain to one [14], [15], [18] or evaluated the domain-agnostic performance [10]–[12], [17], [19], [20]. These facts make it difficult to determine which metadata-based similarity indicator should be employed for each domain. Furthermore, the specific effects of the combination of different methods are not well known despite conventional studies pointing to the importance of the combination [10].

This paper attempts to understand (1) similarity indicators that should be employed for each domain and (2) the effects of a combination of different indicators on performance. We compared and evaluated the clustering performance of 16 metadata-based similarity indicators using 1500 metadata datasets (15 domains $\times$ 100 datasets) from the Kaggle data platform. As a result, we found the following.

(1) Clustering performance varies significantly depending on the domain. Our experimental results provide the appropriate similarity indicators for each domain.
(2) Most of the combinations can improve clustering performance. We demonstrated what combinations that lead to improved clustering performance.

Table 1 shows a comparison between our study and conventional studies.

II. TASK DEFINITION

Our research objectives are to understand (1) similarity indicators that should be employed for each domain and (2) the effects of a combination of different indicators on performance. We attempted to understand these through inter-dataset clustering. Whereas general clustering takes a single dataset to generate clusters of instances in an input dataset [21], [22], the inter-dataset clustering takes multiple datasets as input to obtain groups of similar datasets. In the former, each cluster element follows the same data format and scheme, although the latter does not necessarily satisfy them. For example, some conventional studies have investigated the discovery of speech recognition corpora sets [23], clustering of geographic dataset series [18], categorization of similar research papers [19].

We work on inter-dataset clustering based on metadata-based dataset similarity in the same manner as [18] and [19]. A more procedural definition of our objective is to find clustering models that can reproduce a true dataset cluster domain-by-domain. Clustering models predict the cluster of each dataset given the distance values between all dataset pairs. To evaluate the metadata-based similarity indicators, we only took different distance values between datasets as inputs to each model. We unified the other elements of the models, such as the clustering algorithms and random parameters.

Our experiments quantitatively evaluated how well each metadata-based similarity indicator rebuilds true dataset clusters for each domain defined based on 15 types of Kaggle dataset tags. When the performance of similarity indicators is exceedingly different for each domain, the results of this study can provide productive insights for future studies to discover similar datasets.

III. RELATED WORKS

A. DISCOVERY OF SIMILAR DATASETS

There are several approaches for discovering combinable datasets. The most typical method is based on information retrieval. Kato et al. [24] indicated the retrieval difficulty depends on user needs from experiments on government open datasets. Wang et al. [14] showed an ontology-based retrieval method is effective for biomedical datasets of Elsevier DataSearch. Sakaji et al. [13] evaluated similarity searches using Word2Vec and BERT vectors on Kaggle datasets. Bernhauer et al. [10] compared some similarity search methods and organized each advantage of similarity search and full-text ad hoc retrieval. They claimed the importance of combining different methods because ad hoc retrieval performs better, whereas a similarity search can improve recall.

In cases where the discovery target is limited to a scientific dataset attached to research papers, information recommendation approaches are predominant. In contrast to retrieval-based approaches, it does not necessarily depend on text information because it mainly uses link structures, such as citations [25]–[27] and co-author relationships [28]. As mentioned above, such types of information are only available on datasets linked to research papers.

There have been some attempts at inter-dataset clustering. Whereas the general clustering takes a single dataset to generate clusters of instances in an input dataset [21], [22], this approach takes multiple datasets as input to obtain groups of similar datasets. Siegert et al. [23] attempted to discover the subsets of the most similar ones from 6 speech emotion recognition corpora. They used acoustic features in each dataset for clustering. Against such an approach based on dataset contents, there are also efforts based on metadata. Sajid et al. [19] attempted to classify research papers using metadata instead of the paper body. They claimed as a contribution that metadata-based methods are also applicable to non-open access papers. Lacasta et al. [18] focused on detecting related dataset series using geographic metadata. They indicated that clustering methods based on text-metadata effectively work to cluster geospatial datasets. This is because such datasets have no unified data modality, and content-based discovery methods cannot be applied.

TABLE 1. A comparison among this paper and conventional studies.

	Task	Compared approaches			Evaluation	
		Text-based	Ontology-based	Variable-based	Multiple domains	Domain-specific
Zhang and Balog. 2018 [11]	Dataset retrieval			✓	✓	
Zhang et al. 2019 [12]	Dataset retrieval	✓		✓	✓	
Škoda et al. 2019 [17]	Dataset retrieval		✓		✓	
Sakaji et al. 2020 [20]	Dataset retrieval	✓		✓	✓	
Wang et al. 2020 [14]	Dataset retrieval	✓	✓			✓
Wang et al. 2021 [15]	Dataset retrieval	✓	✓			✓
Bernhauer et al. 2022 [10]	Dataset retrieval	✓			✓	
Lacasta et al. 2022 [18]	Inter-dataset clustering	✓				✓
Sajid et al. 2022 [19]	Inter-dataset clustering	✓			✓	
Ours	Inter-dataset clustering	✓	✓	✓	✓	✓

In summary, metadata-based methods are effective for datasets with mixed different modalities or limited access to their contents, whereas content-based methods are not. Our study is similar to Lacasta et al. [18] in that it compared metadata-based methods for clustering datasets with mixed different data modalities but differs in not limited the domains of target datasets. Furthermore, we also consider methods based on ontology and non-text metadata, which are not compared in these conventional studies.

B. METADATA-BASED SIMILARITY INDICATOR BETWEEN DATASETS

Conventional studies have proposed three approaches to compute the dataset similarity based on metadata: text-based, ontology-based, and variable-based. Text-based approaches calculate the similarity between two text metadata as the dataset similarity. They have been especially evaluated and discussed in many studies. There have been proposals for Jaccard coefficients and cosine similarity based on text vectors [29]–[31]. Sakaji et al. [13] evaluated Word2Vec and BERT for open domain datasets in Kaggle. Wang et al. [15] found the cosine similarity between BERT vectors was the best performance of Precision@k to retrieve biomedical datasets from Elsevier's DataSearch. In addition, they also observed that BM25, the classical indicator, showed the best performance in some cases. Skopal et al. [16] introduced data-transitive similarity using intermediator datasets between non-similar datasets. Bernhauer et al. [10] evaluated this indicator and widely compared it with other similarity indicators based on text metadata. They indicated that TF-IDF-based data-transitive similarity achieved higher Precision-Recall AUC than the Jaccard coefficient, Word2Vec, and BERT for six similarity search experiments using the Czech Open Data Catalogue. Our work is similar to Bernhauer et al. [10] in comparing diverse similarity indicators. However, we focused on inter-dataset clustering instead of similarity-based dataset search. In addition, we evaluated for fifteen domains and compared not only text-based but also ontology- and variable-based similarity indicators.

Ontology-based indicators calculate the similarity between terms mapped into an ontology, such as Wikidata and WordNet. While text-based similarity indicators reflect all the contents of text metadata, this approach only uses the information contained in the ontology. Therefore, its main characteristic is that it can remove noise information contained in text metadata. Škoda et al. [17] proposed Navigational Distance that can provide a structured explanation of dataset similarity. They proofed of their concepts using the Wikidata ontology. Wang et al. [14] applied Wu-Palmer Similarity and Resnik similarity to ad hoc retrieval of biomedical datasets. They showed Wu-Palmer Similarity overtook Google distance and cosine similarity between Word2Vec vectors. Conventional efforts have been quantitatively evaluated and compared with other approaches in only a few domains. This paper evaluates these indicators for fifteen domains and compares them text-based and variable-based approaches.

While both of the previous two approaches have used text metadata, another approach employs variables instead of text metadata. Variables correspond to the set of strings that refer to the dataset contents, e.g., attribute names of the database and column names of the tabular dataset. Bogatu et al. [32] proposed D3L (Dataset Discovery in Data lakes) as a series of dataset similarity indicators for data lakes. It includes several similarity indicators, one of which used the Jaccard coefficient between variables of data lakes. Several efforts by Zhang et al. [11], [12] also employed variables. They proposed an embedding method for variables based on skip-grams and applied the cosine similarity. Sakaji et al. [20] applied Dice coefficient between variables to datasets of D-Ocean, a Japanese data platform. Their experimental results indicated that it could represent the content and geographic similarity of datasets, as well as Word2Vec and BERT. Although this effort is important in that it verified the variable-based indicators for open data platforms, they have not conducted quantitative evaluations based on objective label information. This study provides a quantitative evaluation based on objective label information. Furthermore, we verified the effectiveness of the variable-based approach in deeper by comparing it with other similarity indicators.

These studies evaluated similarity indicators based on different domains, datasets, and evaluation metrics. It is difficult to compare their evaluation results and determine which similarity indicators should be selected for each domain. In addition, there are no efforts that exhaustively compare similarity indicators for each approach presented above. Efforts to evaluate and compare feature extraction or similarity

methods on the same metrics are also flourishing in many research fields [33]–[35]. Although Bernhauer et al. [10] is a typical example of such studies in the dataset discovery field, they only focused on text-metadata-based similarity indicators. This study provides a more extensive comparison of representative similarity indicators.

IV. METHODOLOGY

A. EXPERIMENTAL PROCESS

Figure 1 provides an overview of our experimental process, which consists of three steps: preparing input data, clustering, and evaluation. The first step involves metadata acquisition, preprocessing, and conversion into the input data for clustering models. We will describe the details of metadata acquisition, preprocessing, and similarity calculation methods in Sections IV-B2, IV-B3, and IV-B4 .

In the clustering step, we predict the domain label of each dataset group by applying clustering to the similarity matrix built in the previous step. We employed the K-Medoids as a clustering method that allows a matrix based on the dataset similarity or distance as an input (in Section IV-C).

The evaluation step indirectly evaluates metadata-based similarity indicators based on the performance of clustering models. We used three evaluation metrics (in Section IV-D) and fifteen domain labels based on Kaggle dataset tags (in Section IV-B1).

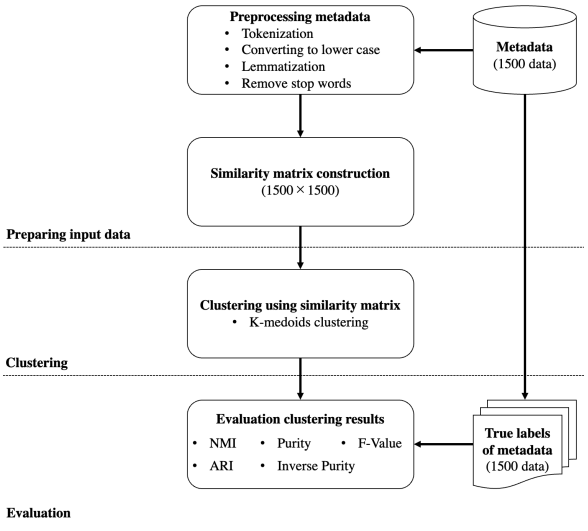

FIGURE 1. An overview of our experimental process

B. PREPARING THE INPUT DATA

1) Input and Evaluation dataset selection

We acquire sets of similar datasets for the evaluation of inter-dataset clustering models. We exploited "subject" tags in Kaggle to sample similar datasets in the domain. Subject tags represent the main contents of datasets, e.g., *Covid-19* and *Investing*. These tags have a hierarchical structure; for example, *Investing* tag is a child of *Finance* tag. We first selected five tags, consisting of *Finance*, *Health Conditions*, *Sports*, *Government*, and *Internet*. We followed the number of datasets belonging to each tag to select the tags. Next, we extracted three child tags for each selected tag based on their frequencies. In summary, we used the following 15 tags in Table 2 as the dataset domains for evaluation. We randomly selected 100 datasets for each of the 15 domains, for a total of 1500 datasets. Then, we acquired the metadata of each dataset. We present more details on metadata and the acquisition process in the next subsection. Note that we preliminarily excluded datasets related to two or more different domains. This preprocess prevents our experiment and evaluation from being complicated by multi-label.

TABLE 2. The list of Kaggle dataset tags used as the dataset domains for evaluation.

Tag name as evaluation domain	Parent tag name
Covid-19	Health Conditions
Cancer	
Heart Conditions	
Investing	Finance
Currencies and Foreign Exchange	
Banking	
Football	Sports
Cricket	
Basketball	
Crime	Government
Public Safety	
Military	
SNS	Internet
E-Mail	
Mobile	

2) Metadata overview and acquisition process

This study defines metadata as data that satisfies at least one of the following conditions: data contained in the Meta Kaggle dataset [36] or acquired from Kaggle API[1]. The following Figure 2 illustrates a sample pair of metadata and the actual contents of the dataset.

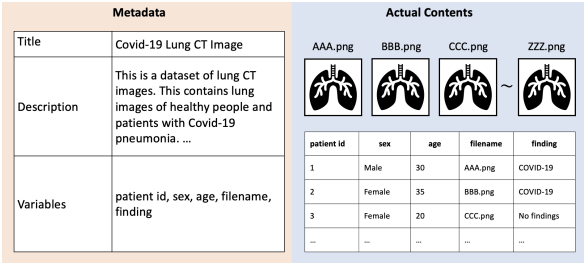

FIGURE 2. An example of metadata and actual contents in the dataset.

We employ three types of metadata: title, descriptions, and variables. The title and descriptions are natural language sentences that describe the dataset contents. The term "text metadata" in conventional studies mainly refers to these metadata. The title is relatively short and abstractly expresses the essential information. In contrast, descriptions often con-

[1]https://github.com/Kaggle/kaggle-api

tain more detailed and specific information. We obtained this metadata from the Meta Kaggle dataset [36] on June 1, 2023.

The variable is a logical set of short strings, such as column names or database attribute names. For example, there are "patient_id", "sex", and "age" in the medical dataset. Because the Meta Kaggle dataset excludes variables, we acquired them using the Kaggle API on June 12, 2023.

3) Preprocessing metadata and making input data

We applied lowercasing, tokenization, lemmatization, and stopword removal for metadata preprocessing for text metadata. We used the en_core_web_sm model of spaCy [37] to tokenize and lemmatize. We have removed stopwords using NLTK and frequent words in descriptions of Kaggle datasets. The following list shows a stopword list that we added manually.

> ∗, #, data, datum, dataset, context, content, acknowledement

For the variables, we applied lowercasing, tokenization, and lemmatization. To tokenize and lemmatize, we employed the same tool used for text metadata.

We build similarity matrices from the preprocessed metadata as input data for clustering. It is a matrix in which each (i, j) element corresponds to the similarity between datasets i and j. We compute the similarity of each dataset pair on the basis of metadata-based similarity indicators described in Section IV-B4.

4) Metadata-based similarity indicator

The term "metadata-based similarity indicator" in this paper refers to inter-dataset indicators that consist of metadata, a data processing method, and a distance function. We use them to compute the similarity between all dataset pairs. We then generate similarity matrices to obtain input data for clustering models. This paper compares 16 similarity indicators. There are three approaches to classifying these indicators: text-based, ontology-based, and variables-based.

A **text-based approach** directly computes the similarity between text metadata. The most primitive is the Jaccard coefficient between word sets of text metadata. It follows the equation below:

$$\text{Jaccard}(T_i, T_j) = \frac{|\text{Unique}(T_i) \cap \text{Unique}(T_j)|}{|\text{Unique}(T_i) \cup \text{Unique}(T_j)|}, \quad (1)$$

where T_i and T_j are text metadata of the corresponding datasets i, j. Unique(T) is a set consisting of unique words of the document T.

The more advanced text-based approaches use vector similarity. Following conventional studies [10], [13]–[15], we employ TF-IDF, Word2Vec, and BERT as vectorizers of text metadata. The following equation (2) show the TF-IDF-based conversion of text metadata T to the dataset vector $\boldsymbol{V}$:

$$\boldsymbol{V_{tfidf}}(T) = (\text{tf}(w_1, T)\text{idf}(w_1), \ldots, \text{tf}(w_n, T)\text{idf}(w_n))^\top, \quad (2)$$

where w is a word in the text metadata T, and n is equal to the vocabulary size of the entire text metadata set. The function $\text{tf}(w_i, T)$ is the frequency of a word w, and $\text{idf}(w_i)$ is the inverted number of documents containing a word w. The definition of the Word2Vec-based conversion function is as follows:

$$\boldsymbol{V_{w2v}}(T) = \frac{1}{|T|} \sum_{w \in T} \text{Word2Vec}(w), \quad (3)$$

where w is a word in text metadata T, Word2Vec(w) is a function that converts a word to the corresponding vector. The BERT-based function is as follows:

$$\boldsymbol{V_{bert}}(T) = \frac{1}{|\text{BERT}_{tok}(T)|} \sum_{\boldsymbol{v} \in \text{BERT}_{emb}(\text{BERT}_{tok}(T)))} \boldsymbol{v}, \quad (4)$$

where $\text{BERT}_{emb}(\text{BERT}_{tok}(T))$ is a function that generates the document-level vector from tokenized text metadata $\text{BERT}_{tok}(T)$. These similarity indicators regard the cosine similarity between these vectors as the dataset similarity.

Another text-based approach is data-transitive similarity, proposed by Skopal et al. [16] It assumes that datasets x and y are transitively similar when they are similar to the same dataset i. The following Equation (5) shows the definition of data-transitive similarity $\text{DT}(x, y)$:

$$\text{DT}(x, y) = \bigodot_{i \in D - \{x, y\}} \uplus(\text{distance}(x, i), \text{distance}(i, y)), \quad (5)$$

where x, y, and i are different datasets, and D is a set of datasets. The operator $\bigodot$ is outer aggregation over all datasets in D, such as min, max, and avg. Another operator $\uplus$ is an inner aggregation over two distance values, e.g., sum, minus, and multiply. This paper uses the cosine similarity between TF-IDF vectors as distance following Bernhauer et al. [10], $\bigodot$ is max and $\uplus$ is multiply.

An **ontology-based approach** maps words in text metadata to an ontology. It computes the graph similarity as the dataset similarity. We adopt WordNet as an ontology that can be applied to various dataset domains. We employ the Navigational distance and Wu-Palmer similarity to compare with other similarity indicators. Navigational distance is an ontology-based indicator contained in the framework proposed by Škoda et al. [17], as follows:

$$\text{Navigational}(T_i, T_j) = 1 - \bigodot_{C_i \in \text{mapD}(T_i), C_j \in \text{mapD}(T_j)} \text{Path}(C_i, C_j), \quad (6)$$

where C_i and C_j are concepts on an ontology mapped from text metadata T_i and T_j, mapD is a function that maps words in input text metadata into an ontology, and Path means path similarity between two concepts. The aggregation operator $\bigodot$ performs over all concept pairs, e.g., sum, max, and avg. Following Škoda et al. [17], we adopt the average function as an aggregation.

The definition of another similarity indicator based on Wu-Palmer similarity is as follows:

$$\mathrm{Sim}(T_i, T_j) = \bigodot_{C_i \in \mathrm{mapD}(T_i), C_j \in \mathrm{mapD}(T_j)} \mathrm{WP}(C_i, C_j), \quad (7)$$

where C_i, C_j are concepts in text metadata T_i, T_j, mapD(T) is a function to map each word to a corresponding concept, and WP(C_i, C_j) is Wu-Palmer similarity. The definition of WP(C_i, C_j) is as follows:

$$\mathrm{WP}(C_i, C_j) = \frac{2\mathrm{Dep}(\mathrm{LCS}_{C_i,C_j})}{\mathrm{Dep}(C_i) + \mathrm{Dep}(C_j) + \mathrm{Dep}(\mathrm{LCS}_{C_i,C_j})}, \quad (8)$$

where LCS_{C_i,C_j} is the lowest one of the common ancestors between C_i and C_j, and Dep(C) returns the number of the hierarchy of an input concept C. The other elements are the same as Equation (6). To reduce the computational cost, we applied TF-IDF-based keyphrase extraction [38] into text metadata. We used limited words contained in the top five keyphrases to map to the ontology.

A **variables-based approach** uses variables to evaluate dataset similarity instead of text metadata. As described in Section IV-B2, variables consist of attribute names of databases or column names of tabular datasets. Following the definition of Sakaji et al. [20], we use a variables-based similarity indicator using the Dice coefficient as follows:

$$\mathrm{Dice}(V_i, V_j) = 2\frac{|V_i \cap V_j|}{|V_i| + |V_j|}, \quad (9)$$

where V_i and V_j are the variables of the corresponding datasets i, j. In addition, we discuss another variables-based similarity indicator using TF-IDF vectorization. For variables, several elements appear across many datasets, such as "id" and "date." Therefore, we expect to obtain a better similarity representation by reducing the influence of such factors. We compute this similarity by replacing text metadata T with variables V in Equation (2).

To summarize this section, we show sixteen similarity indicators compared in this study in the following Table 3. It shows the components of each similarity indicator, including the metadata, data processing method, and distance function. It also shows the approach type to which each similarity indicator belongs. In addition, we have added the name of the clustering model using each similarity indicator in the "Model name" column. We use these names again in Section V.

C. CLUSTERING

This study employed the K-Medoids algorithm as the clustering method. The K-Medoids algorithm is a non-hierarchical clustering method and is suited for community detection of a network. While the basic concept of K-Medoids is similar to that of K-Means, it differs in using the medoid instead of the centroid. The centroid is the average vector of the samples in the cluster. The K-Means allocates each sample into new cluster to minimize the distance with centroid for each cluster. In contrast, the medoid is not the average vector but one of the samples contained in the cluster. The definition of medoid is as follows:

$$\mathrm{medoids}_{X_i} = \underset{\boldsymbol{m} \in X_i}{\mathrm{argmin}} \sum_{\boldsymbol{x} \in X_i - \{\boldsymbol{m}\}} \mathrm{distance}(\boldsymbol{x}, \boldsymbol{m}), \quad (10)$$

where X_i is a cluster and distance$(\boldsymbol{x}, \boldsymbol{m})$ is the distance between samples $\boldsymbol{x}$ and $\boldsymbol{m}$. This method allows a matrix of the dataset similarity and distance as input data in contrast to the other methods. We compare similarity indicators that do not generate a feature vector for each dataset. Therefore, the K-medoids method is suitable for our experiment.

This study aims to evaluate different metadata-based similarity indicators for each dataset domain. However, the construction of the best method for inter-dataset clustering is outside the scope of our study. Therefore, we do not compare the different clustering algorithms.

D. EVALUATION METRICS

To evaluate the clustering performance of each similarity indicator, we used **NMI**, **ARI**, and **Purity**. These are typical quantitative evaluation metrics for clustering models using true cluster labels. NMI (Normalized Mutual Information) is a metric based on mutual information and evaluates the global structural similarity between a clustering result and true clusters. Equation (11) is the definition of NMI.

$$\mathrm{NMI}(X, Y) = \frac{2\mathrm{I}(X; Y)}{\mathrm{H}(X) + \mathrm{H}(Y)}, \quad (11)$$

where X and Y are different clustering results, one of which is a partition based on true labels. $\mathrm{I}(X; Y)$ is mutual information and $\mathrm{H}(X)$ is the entropy of X.

ARI (Adjusted Rand Index) is a metric that also considers global structural similarity. The following Equation (12) shows the definition of ARI.

$$\mathrm{ARI}(X, Y) = \frac{\mathrm{RI}(X, Y) - \mathrm{ExpRI}(X, Y)}{\mathrm{maxRI}(X, Y) - \mathrm{ExpRI}(X, Y)}, \quad (12)$$

where $\mathrm{RI}(X, Y)$ is the rand index without adjusting for chance, and each of $\mathrm{ExpRI}(X, Y)$ and $\mathrm{maxRI}(X, Y)$ refers to the expected and maximum values of the rand index. Equations (13), (14), and (15) are the definitions of these three values.

$$\mathrm{RI}(X, Y) = \sum_{x \in X, y \in Y} \binom{n_{xy}}{2}, \quad (13)$$

$$\mathrm{ExpRI}(X, Y) = \frac{1}{\binom{n}{2}} \sum_{y \in Y} \binom{\sum_{x \in X} n_{xy}}{2} \sum_{x \in X} \binom{\sum_{y \in Y} n_{xy}}{2}, \quad (14)$$

$$\mathrm{maxRI}(X, Y) = \frac{1}{2} \sum_{x \in X} \binom{\sum_{y \in Y} n_{xy}}{2} + \sum_{y \in Y} \binom{\sum_{x \in X} n_{xy}}{2}, \quad (15)$$

where n_{xy} is the number of samples belonging to clusters x and y.

TABLE 3. Sixteen types of dataset similarity indicators based on metadata used in our experiment. In "Model name" column, each term **T**, **D**, and **V** means the title, the description, and the variables. In "Data processing method" column, "Keyphrase + WordNet" means mapping extracted keyphrases to the WordNet ontology. In "Distance function" column, "DT similarity" means the data-transitive similarity.

Model name	Approach type	Metadata	Data processing method	Distance function
Jaccard(T)	Text-based	Title	-	Jaccard coefficient
Jaccard(T+D)		Title + Description	-	Jaccard coefficient
Cosine(Word2Vec(T))		Title	Word2Vec	Cosine similarity
Cosine(Word2Vec(T+D))		Title + Description	Word2Vec	Cosine similarity
Cosine(BERT(T))		Title	BERT	Cosine similarity
Cosine(BERT(T+D))		Title + Description	BERT	Cosine similarity
Cosine(TF-IDF(T))		Title	TF-IDF	Cosine similarity
Cosine(TF-IDF(T+D))		Title + Description	TF-IDF	Cosine similarity
DT(Cosine(TF-IDF(T)))		Title	TF-IDF	Cosine similarity + DT similarity
DT(Cosine(TF-IDF(T+D)))		Title + Description	TF-IDF	Cosine similarity + DT similarity
Wu-Palmer(T)	Ontology-based	Title	Keyphrase + WordNet	Wu-Palmer similarity
Wu-Palmer(T+D)		Title + Description	Keyphrase + WordNet	Wu-Palmer similarity
Navigational(T)		Title	Keyphrase + WordNet	Navigational distance
Navigational(T+D)		Title + Description	Keyphrase + WordNet	Navigational distance
Dice(V)	Variables-based	Variables	-	Dice coefficient
Cosine(TF-IDF(V))		Variables	TF-IDF	Cosine similarity

Purity aggregates the precision of each cluster. The definition of purity is as follows.

$$\begin{aligned}\text{Purity}(X,Y) &= \frac{\sum_{x\in X}\max_{y\in Y}\text{Precision}(x,y)|x|}{\sum_{x\in X}|x|},\\ &= \frac{\sum_{x\in X}\max_{y\in Y}|x\cap y|}{\sum_{x\in X}|x|},\end{aligned} \tag{16}$$

where X is the clustering results and Y is a partition based on true labels. Precision(x, y) follows the usual definition of precision.

E. PARAMETER SETTINGS

For Word2Vec and BERT, we used publicly available pre-trained models and did not fine-tune them. We used the wiki-news-300d-1M model [39][2] from Meta Research to obtain Word2Vec word vectors. This model is based on Wikipedia 2017, the UMBC webbase corpus, and statmt.org news dataset. It converts words to vectors with a dimension of 300. We adopted the bert-base-uncased model from Devlin et al. [31] to obtain BERT embeddings. It is based on English Wikipedia and BookCorpus. This model generates vectors with dimensions of 768.

In ontology-based approaches, we applied a keyphrase extraction method based on TF-IDF. We used the implementation by boudin et al. [38] to extract the top five ranked keyphrases from each text metadata. We used NLTK [40] to map words to the ontology and compute distances between WordNet concepts.

For clustering using K-Medoids, we set the number of clusters to 15, which is the same as the total number of dataset domains. We applied random assignments to the initial clusters and set the maximum number of iterations to 100, following the default settings of the library[3]. In addition, we set the random seed value to a fixed integer of 2023.

TABLE 4. A comparison of clustering performances over 15 domains. Bold values indicate the highest among the same metrics.

Model	NMI	ARI	Purity
Jaccard(T)	0.261	0.064	0.322
Jaccard(T+D)	0.173	0.049	0.250
Cosine(TF-IDF(T))	**0.383**	0.108	0.419
Cosine(TF-IDF(T+D))	0.351	0.175	0.423
Cosine(BERT(T))	0.353	0.184	**0.427**
Cosine(BERT(T+D))	0.171	0.062	0.233
Cosine(Word2Vec(T))	0.297	0.141	0.382
Cosine(Word2Vec(T+D))	0.238	0.062	0.253
DT(Cosine(TF-IDF(T)))	0.324	**0.204**	0.419
DT(Cosine(TF-IDF(T+D)))	0.315	0.164	0.405
Wu-Palmer(T)	0.192	0.072	0.275
Wu-Palmer(T+D)	0.189	0.078	0.275
Navigational(T)	0.222	0.070	0.291
Navigational(T+D)	0.223	0.070	0.295
Dice(V)	0.187	0.030	0.243
Cosine(TF-IDF(V))	0.199	0.058	0.271

TABLE 5. A comparison of clustering performances for each medical domain.

Model	Covid-19	Cancer	Heart Disease
Jaccard(T)	0.288	0.569	0.170
Jaccard(T+D)	0.348	0.471	0.211
Cosine(TF-IDF(T))	**0.611**	0.582	0.280
Cosine(TF-IDF(T+D))	0.391	0.557	0.330
Cosine(BERT(T))	0.290	0.500	**0.406**
Cosine(BERT(T+D))	0.173	0.707	0.189
Cosine(Word2Vec(T))	0.205	0.691	0.400
Cosine(Word2Vec(T+D))	0.158	0.626	0.177
DT(Cosine(TF-IDF(T)))	0.417	0.559	0.307
DT(Cosine(TF-IDF(T+D)))	0.504	0.489	0.252
Wu-Palmer(T)	0.220	**0.763**	0.128
Wu-Palmer(T+D)	0.146	0.738	0.167
Navigational(T)	0.139	0.658	0.169
Navigational(T+D)	0.145	0.494	0.174
Dice(V)	0.238	0.181	0.150
Cosine(TF-IDF(V))	0.292	0.262	0.177

TABLE 6. A comparison of clustering performances for each financial domain.

Model	Investing	Currency	Banking
Jaccard(T)	0.414	0.163	0.542
Jaccard(T+D)	0.148	0.168	0.434
Cosine(TF-IDF(T))	0.208	**0.703**	0.601
Cosine(TF-IDF(T+D))	0.562	0.169	**0.646**
Cosine(BERT(T))	0.373	0.252	0.331
Cosine(BERT(T+D))	0.208	0.206	0.281
Cosine(Word2Vec(T))	0.538	0.470	0.464
Cosine(Word2Vec(T+D))	0.265	0.212	0.333
DT(Cosine(TF-IDF(T)))	0.500	0.592	0.318
DT(Cosine(TF-IDF(T+D)))	**0.578**	0.201	0.565
Wu-Palmer(T)	0.370	0.232	0.527
Wu-Palmer(T+D)	0.465	0.210	0.517
Navigational(T)	0.484	0.202	0.390
Navigational(T+D)	0.497	0.180	0.451
Dice(Variables)	0.429	0.311	0.308
Cosine(TF-IDF(Variables))	0.419	0.273	0.211

TABLE 7. A comparison of clustering performances for each sporting domain.

Model	Football	Cricket	Basketball
Jaccard(T)	0.233	0.597	0.567
Jaccard(T+D)	0.263	0.425	0.292
Cosine(TF-IDF(T))	0.267	**0.662**	0.660
Cosine(TF-IDF(T+D))	0.353	0.621	0.671
Cosine(BERT(T))	**0.705**	0.589	**0.815**
Cosine(BERT(T+D))	0.249	0.271	0.296
Cosine(Word2Vec(T))	0.348	0.351	0.512
Cosine(Word2Vec(T+D))	0.379	0.407	0.431
DT(Cosine(TF-IDF(T)))	0.224	0.445	0.654
DT(Cosine(TF-IDF(T+D)))	0.277	0.567	0.580
Wu-Palmer(T)	0.177	0.215	0.177
Wu-Palmer(T+D)	0.178	0.271	0.148
Navigational(T)	0.266	0.226	0.218
Navigational(T+D)	0.280	0.223	0.203
Dice(V)	0.179	0.374	0.560
Cosine(TF-IDF(V))	0.167	0.544	0.556

TABLE 8. A comparison of clustering performances for each governmental domain.

Model	Crime	Public Safety	Military
Jaccard(T)	0.366	0.231	0.149
Jaccard(T+D)	0.125	0.158	0.142
Cosine(TF-IDF(T))	0.573	0.188	**0.388**
Cosine(TF-IDF(T+D))	0.587	0.191	0.173
Cosine(BERT(T))	0.516	**0.327**	0.218
Cosine(BERT(T+D))	0.222	0.211	0.115
Cosine(Word2Vec(T))	0.503	0.183	0.167
Cosine(Word2Vec(T+D))	0.158	0.167	0.166
DT(Cosine(TF-IDF(T)))	0.512	0.306	0.206
DT(Cosine(TF-IDF(T+D)))	0.328	0.246	0.246
Wu-Palmer(T)	0.256	0.150	0.304
Wu-Palmer(T+D)	0.263	0.157	0.258
Navigational(T)	**0.595**	0.221	0.184
Navigational(T+D)	0.566	0.152	0.154
Dice(V)	0.234	0.152	0.154
Cosine(TF-IDF(V))	0.261	0.161	0.126

TABLE 9. A comparison of clustering performances for each Internet domain.

Model	SNS	E-Mail	Mobile
Jaccard(T)	0.193	0.358	0.493
Jaccard(T+D)	0.148	0.234	0.397
Cosine(TF-IDF(T))	0.199	0.371	0.488
Cosine(TF-IDF(T+D))	0.213	0.463	0.583
Cosine(BERT(T))	**0.336**	0.388	0.556
Cosine(BERT(T+D))	0.222	0.194	0.381
Cosine(Word2Vec(T))	0.248	0.259	0.470
Cosine(Word2Vec(T+D))	0.170	0.140	0.376
DT(Cosine(TF-IDF(T)))	0.171	0.373	0.497
DT(Cosine(TF-IDF(T+D)))	0.263	**0.482**	**0.590**
Wu-Palmer(T)	0.207	0.179	0.256
Wu-Palmer(T+D)	0.180	0.201	0.225
Navigational(T)	0.222	0.184	0.439
Navigational(T+D)	0.181	0.196	0.433
Dice(V)	0.145	0.206	0.181
Cosine(TF-IDF(V))	0.186	0.303	0.166

V. RESULTS

Table 4 indicates an evaluation results over 15 domains shown in Table 2. First, we found that indicators based on cosine similarity and text metadata achieved high performance. A BERT-based indicator using the title showed consistently high performance for all evaluation metrics, followed by TF-IDF-based and Word2Vec-based indicators. Other indicators, including ontology-based and variable-based, performed comparatively poorly, and had no significant differences among their performances.

One interesting tendency from Table 4 is most of the similarity indicators using both the title and the description showed a poorer performance than only using the title. We considered that the main factor was the low ratio of useful information in Kaggle dataset descriptions. For example, descriptions often contain unneccessary information for discovery datasets, such as credits, acknowledgements, and template sentences. As an exception, the ontology-based indicators improved some evaluation metric values by adding a description. These indicators employed keyword extraction and ontology mapping. We considered that such data processes increased the influence of the useful information in descriptions rather than noise. From these facts, indicators based on the cosine similarity between dataset title vectors are better choices in terms of domain-agnostic performance.

Tables 5, 6, 7, 8, and 9 show the clustering performances for each domain, including the medical, financial, sporting, governmental, and Internet domains. Each table shows the highest F-value based on Precision-Recall for each similarity indicator and each domain, as described in Table 2. We can see from each table that the best indicator is completely different for each domain. As one interesting observation, although some similarity indicators using the WordNet ontology performed poorly in almost all domains, each showed the best performance in only one domain. Wu-Palmer similarity using the dataset title in the cancer domain and Navigational distance using the title in the crime domain outperformed

[2]https://fasttext.cc/docs/en/english-vectors.html
[3]https://github.com/kno10/python-kmedoids

BERT and TF-IDF. In contrast, some similarity indicators performed exceedingly poorly in some limited domains. Although the cosine similarity between title vectors using BERT was the best indicator for the highest number of domains, it showed significantly lower performance in the three financial domains. The F-values in these domains are equal to or less than those by the Dice coefficient between variables, which perform poorly in many domains. These results provide a more straightforward explanation of why an appropriate discovery method should be selected for each domain. In addition, these results revealed the appropriate similarity indicators for each domain, such as BERT performing well for sporting domains but not for financial domains.

VI. DISCUSSION

From the results, we found that the most effective similarity indicator was different for each domain. Similarity indicators based on vectors of text metadata were highly versatile, and each ontology-based indicator worked optimally on only one domain. While text-metadata-based similarity indicators worked effectively, variable-based ones have not performed satisfactiorily in any experimental conditions. However, text and variable metadata might complement each other because they contain significantly different information. Conventional studies have also argued for the importance of combining different methods [10]. In this section, we discuss the following two points: (1) whether discoverable datasets differ between text-metadata and variables, and (2) whether combining two similarity indicators based on different types of metadata can improve clustering performance.

TABLE 10. The structural differences between two clustering results for each comparison condition. The smaller the values, the greater the structural differences between the two clustering results.

Comparison condition	NMI Avg.	ARI Avg.	No. of pairs
Different metadata	**0.108**	**0.016**	28
Same metadata	0.210	0.078	92
Different distance functions	0.169	0.057	75
Same distance functions	0.216	0.075	45
Different data processings	0.182	0.062	112
Same data processings	0.247	0.096	8

TABLE 11. The local overlaps between two clustering results for each medical domain and each comparison condition. We used the Jaccard coefficient to measure local overlaps.

Comparison condition	Covid-19	Cancer	Heart Disease
Different metadata	**0.204**	**0.444**	**0.136**
Same metadata	0.297	0.547	0.246
Different distance functions	0.250	0.522	0.196
Same distance functions	0.317	0.525	0.260
Different data processings	0.275	0.514	0.219
Same data processings	0.286	0.645	0.238

A. INFLUENCE OF SIMILARITY INDICATORS ON CLUSTER STRUCTURES

We expect structural differences in clustering results if the discoverable datasets differ depending on the metadata used.

TABLE 12. The overlaps between the two clustering results for each financial domain and each comparison condition. We used the Jaccard coefficient to measure local overlaps.

Comparison condition	Investing	Currency	Banking
Different metadata	**0.379**	0.168	**0.238**
Same metadata	0.415	0.165	0.508
Different distance functions	0.395	**0.138**	0.460
Same distance functions	0.426	0.213	0.421
Different data processings	0.404	0.160	0.441
Same data processings	0.445	0.248	0.498

TABLE 13. The overlaps between the two clustering results for each sporting domain and each comparison condition. We used the Jaccard coefficient to measure local overlaps.

Comparison condition	Football	Cricket	Basketball
Different metadata	**0.109**	0.239	0.344
Same metadata	0.206	0.290	0.364
Different distance functions	0.159	**0.211**	**0.304**
Same distance functions	0.225	0.390	0.451
Different data processings	0.179	0.263	0.361
Same data processings	0.245	0.496	0.341

Therefore, we quantitatively compared the global structural differences and local overlaps between the two clustering results. First, we computed the NMI and ARI between clustering results for all possible pairs of models. There was a significant difference between the two clustering results in their global structure when these metrics took low values. In addition, we evaluated local overlaps by calculating the Jaccard coefficient between each cluster pair in the two clustering results. Note that each cluster used to calculate the Jaccard coefficient has the highest F-value based on Precision-Recall for each domain, same as in the Tables 5 to 9. To identify the influence of metadata on discoverable datasets, we defined the six conditions shown in Table 10.

The following Table 10 shows global structural differences and Tables 11, 12, 13, 14, and 15 indicates local overlaps between two clustering results for each condition. According

TABLE 14. The overlaps between the two clustering results for each governmental domain and each comparison condition. We used the Jaccard coefficient to measure local overlaps.

Comparison condition	Crime	Public Safety	Military
Different metadata	**0.178**	**0.094**	**0.156**
Same metadata	0.414	0.273	0.167
Different distance functions	0.341	0.233	0.163
Same distance functions	0.388	0.228	0.166
Different data processings	0.353	0.228	0.164
Same data processings	0.444	0.271	0.164

TABLE 15. The overlaps between the two clustering results for each Internet domain and each comparison condition. We used the Jaccard coefficient to measure local overlaps.

Comparison condition	SNS	E-Mail	Mobile
Different metadata	**0.208**	0.262	**0.248**
Same metadata	0.257	0.303	0.517
Different distance functions	0.239	**0.247**	0.394
Same distance functions	0.257	0.371	0.554
Different data processings	0.235	0.281	0.440
Same data processings	0.391	0.468	0.644

TABLE 16. The ratio of combinations with clustering performance improvement.

Combination condition	% of combinations with improved NMI	% of combinations with improved ARI	Total number of combinations
Different metadata	**0.643**	**0.964**	28
Same metadata	0.424	0.707	92
Different distance functions	0.440	0.720	75
Same distance functions	0.533	0.844	45
Different data processings	0.494	0.747	99
Same data processings	0.381	0.857	21

to Table 10, we can observe that both NMI and ARI between clustering results based on different metadata took lowest values. Tables 11 to 15 shows the pair with different metadata have the minimum overlap between their clusters in eleven domains. These results suggest that discoverable datasets differ between text-metadata-based and variable-based similarity indicators.

B. INFLUENCE OF COMBINING SIMILARITY INDICATORS ON CLUSTERING PERFORMANCE

On the basis of the results of the previous subsection, we discuss whether the combination of text-metadata-based and variable-based similarity indicators can improve clustering performance. In addition, we evaluated 120 clustering models obtained by combining two different similarity indicators in the manner described in Section IV. We computed the ratio of combinations improved in the evaluation metric values compared to that before combining under the same comparison conditions as in Table 10. Table 16 shows the results. From the table, each ratio of combinations with improved metric value was the highest when combining text-metadata-based and variable-based indicators. In particular, we observed an improvement in 27 of 28 combinations for the ARI value. The ratios in this condition exceeded by more than 0.1 points compared with the second-best result. From this result, we conclude that variable-based similarity indicators are more valuable when combined with text-metadata-based indicators than when used alone.

VII. LIMITATION

The first limitation of this study is the dependencies on the nature of the data platforms. We should be aware of some effects due to the characteristics of the data platform on our experimental results because we collected all datasets from Kaggle. For example, almost all similarity indicators using text metadata showed a decrease in clustering performance when employing the dataset description. We considered that the low quality of the dataset description in Kaggle is one of the main factors. Dataset descriptions in the Kaggle frequently include credits, acknowledgments, template text, and other information unrelated to dataset contents. Therefore, there is scope for verification using other data platforms to reduce the influence of each platform.

The second limitation is the dependencies on the ontologies. We compared and evaluated Wu-Palmer Similarity and Navigational Distance as ontology-based similarity indicators. This study adopted the WordNet ontology for generalizability to diverse domains, whereas other studies used other ontologies. For example, Wang et al. adopted the MeSH (Medical Subject Headlines) terminology to discover biomedical datasets. What ontology should be selected for each domain is beyond the focus of this paper. Therefore, it is one of the future issues.

The third limitation was the nature of the metadata used. Each similarity indicator that we evaluated relies on one of the following metadata: dataset titles, descriptions, and variables. As shown in Section VI, we can improve clustering performance by combining two similarity indicators based on variables and text metadata. The experimental results suggested that such an improvement is due to the differences between discoverable datasets caused by metadata. We except that the additional combination of these metadata and other types of metadata leads to further improvement. One important future direction is to verify metadata beyond the scope of this paper, such as relationships based on citations and co-authorships in datasets related to research papers.

VIII. CONCLUSION

This study evaluated 16 metadata-based similarity indicators for the inter-dataset clustering task for each dataset domain. We selected 15 domains and acquired 1500 dataset metadata from the Kaggle data platform for evaluation. We found that the most effective similarity indicator differs with respect to each domain. While a BERT-based indicator performed the best for the most number of domains, it showed poor performance for financial domains. In contrast, some indicators were effective for only one domain. For example, the indicator based on Wu-Palmer Similarity showed the best performance for only a cancer domain. A similar result was obtained for the Navigational distance-based indicator for a crime domain.

Further analysis indicated that the combination of two different similarity indicators can improve the clustering performance. Although the performances of variable-based similarity indicators were remarkably poorer, 96.4% (27/28) of combinations between variable- and text-metadata-based indicators improved their ARI values. We concluded that the vast separation in the distributions of discoverable datasets between variable-based and text-metadata-based indicators led to such an improvement. One reason is that the NMI and ARI between variable-based and text-metadata-based cluster sets took exceedingly lower values.

Conventional studies have asserted the difficulty of selecting the optimal method for all situations and the im-

portance of combining different methods. Our results added more practical details to these assertions for future studies. This study presented which similarity indicator should be employed for each of the 15 dataset domains from the experimental results. In addition, we identified specific requirements for combining the two indicators to improve clustering performance. Points left to study in future work are an analysis on other data platforms and development of a novel similarity indicator from optimizing ratio in combination of indicators based on our results.

REFERENCES

[1] Magdalena Balazinska, Bill Howe, and Dan Suciu. Data Markets in the Cloud: An Opportunity for the Database Community. PVLDB, 4:1482–1485, August 2011.

[2] Florian Stahl, Fabian Schomm, and Gottfried Vossen. Data Marketplaces: An Emerging Species. Databases and Information Systems VIII, pages 145–158, 2014.

[3] Fan Liang, Wei Yu, Dou An, Qingyu Yang, Xinwen Fu, and Wei Zhao. A Survey on Big Data Market: Pricing, Trading and Protection. IEEE Access, 6:15132–15154, 2018.

[4] Markus Spiekermann. Data Marketplaces: Trends and Monetisation of Data Goods. Intereconomics, 2019(4):208–216, 2019.

[5] Teruaki Hayashi, Gensei Ishimura, and Yukio Ohsawa. Structural Characteristics of Stakeholder Relationships and Value Chain Network in Data Exchange Ecosystem. IEEE Access, 9:52266–52276, 2021.

[6] Raul Castro Fernandez, Pranav Subramaniam, and Michael J. Franklin. Data market platforms: trading data assets to solve data problems. Proceedings of the VLDB Endowment, 13(12):1933–1947, September 2020.

[7] Dan Brickley, Matthew Burgess, and Natasha Noy. Google Dataset Search: Building a search engine for datasets in an open Web ecosystem. In The World Wide Web Conference, WWW '19, pages 1365–1375, New York, NY, USA, May 2019. Association for Computing Machinery.

[8] Omar Benjelloun, Shiyu Chen, and Natasha Noy. Google Dataset Search by the Numbers. In Jeff Z. Pan, Valentina Tamma, Claudia d'Amato, Krzysztof Janowicz, Bo Fu, Axel Polleres, Oshani Seneviratne, and Lalana Kagal, editors, The Semantic Web – ISWC 2020, Lecture Notes in Computer Science, pages 667–682, Cham, 2020. Springer International Publishing.

[9] Martin Nečaský, Petr Škoda, David Bernhauer, Jakub Klímek, and Tomáš Skopal. Modular framework for similarity-based dataset discovery using external knowledge. Data Technologies and Applications, 56(4):506–535, January 2022.

[10] David Bernhauer, Martin Nečaský, Petr Škoda, Jakub Klímek, and Tomáš Skopal. Open dataset discovery using context-enhanced similarity search. Knowledge and Information Systems, 64(12):3265–3291, December 2022.

[11] Shuo Zhang and Krisztian Balog. Ad Hoc Table Retrieval using Semantic Similarity. In Proceedings of the 2018 World Wide Web Conference, WWW '18, pages 1553–1562, Republic and Canton of Geneva, CHE, April 2018. International World Wide Web Conferences Steering Committee.

[12] Li Zhang, Shuo Zhang, and Krisztian Balog. Table2Vec: Neural Word and Entity Embeddings for Table Population and Retrieval. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'19, pages 1029–1032, New York, NY, USA, July 2019. Association for Computing Machinery.

[13] Hiroki Sakaji, Teruaki Hayashi, Yoshiaki Fukami, Takumi Shimizu, Hiroyasu Matsushima, and Kiyoshi Izumi. Retrieving of Data Similarity using Metadata on a Data Analysis Competition Platform. In 2021 IEEE International Conference on Big Data (Big Data), pages 3480–3485, February 2021.

[14] Xu Wang, Zhisheng Huang, and Frank van Harmelen. Evaluating Similarity Measures for Dataset Search. In Zhisheng Huang, Wouter Beek, Hua Wang, Rui Zhou, and Yanchun Zhang, editors, Web Information Systems Engineering – WISE 2020, Lecture Notes in Computer Science, pages 38–51, Cham, 2020. Springer International Publishing.

[15] Xu Wang, Frank van Harmelen, and Zhisheng Huang. Biomedical dataset recommendation. In Proceedings of the 10th International Conference on Data Science, Technology and Applications - Volume 1: DATA,, pages 192–199. INSTICC, SciTePress, 2021.

[16] Tomáš Skopal, David Bernhauer, Petr Škoda, Jakub Klímek, and Martin Nečaský. Similarity vs. Relevance: From Simple Searches to Complex Discovery. In Nora Reyes, Richard Connor, Nils Kriege, Daniyal Kazempour, Ilaria Bartolini, Erich Schubert, and Jian-Jia Chen, editors, Similarity Search and Applications, Lecture Notes in Computer Science, pages 104–117, Cham, 2021. Springer International Publishing.

[17] Petr Škoda, Jakub Klímek, Martin Nečaský, and Tomáš Skopal. Explainable Similarity of Datasets Using Knowledge Graph. In Giuseppe Amato, Claudio Gennaro, Vincent Oria, and Miloš Radovanović, editors, Similarity Search and Applications, Lecture Notes in Computer Science, pages 103–110, Cham, 2019. Springer International Publishing.

[18] Javier Lacasta, Francisco Javier Lopez-Pellicer, Javier Zarazaga-Soria, Rubén Béjar, and Javier Nogueras-Iso. Approaches for the Clustering of Geographic Metadata and the Automatic Detection of Quasi-Spatial Dataset Series. ISPRS International Journal of Geo-Information, 11(2):87, February 2022.

[19] Naseer Sajid, Munir Ahmad, Atta Rahman, Gohar Zaman, Mohammed Ahmed, Nehad Ibrahim, Mohammed Basheer Ahmed, Gomathi Krishna, Reem Alzaher, Mariam Alkharraa, Dania Alkhulaifi, Maryam Alqahtani, Asiya Abdus Salam, Linah Saraireh, Mohammed Gollapalli, and Rashad Ahmed. A Novel Metadata Based Multi-Label Document Classification Technique. Computer Systems Science and Engineering, 46:2195–2214, February 2023.

[20] Hiroki Sakaji, Teruaki Hayashi, Kiyoshi Izumi, and Yukio Ohsawa. Verification of Data Similarity using Metadata on a Data Exchange Platform. In 2020 IEEE International Conference on Big Data (Big Data), pages 4467–4474, December 2020.

[21] Srinivasan Parthasarathy and Mitsunori Ogihara. Exploiting Dataset Similarity for Distributed Mining. In José Rolim, editor, Parallel and Distributed Processing, Lecture Notes in Computer Science, pages 399–406, Berlin, Heidelberg, 2000. Springer.

[22] Ishfaq Ali, Atiq Ur Rehman, Dost Muhammad Khan, Zardad Khan, Muhammad Shafiq, and Jin-Ghoo Choi. Model selection using k-means clustering algorithm for the symmetrical segmentation of remote sensing datasets. Symmetry, 14(6), 2022.

[23] Ingo Siegert, Ronald Böck, and Andreas Wendemuth. Using a PCA-based dataset similarity measure to improve cross-corpus emotion recognition. Computer Speech & Language, 51:1–23, September 2018.

[24] Makoto P. Kato, Hiroaki Ohshima, Ying-Hsang Liu, and Hsin-Liang Chen. A Test Collection for Ad-hoc Dataset Retrieval. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 2450–2456, Virtual Event Canada, July 2021. ACM.

[25] Basmah Altaf, Uchenna Akujuobi, Lu Yu, and Xiangliang Zhang. Dataset Recommendation via Variational Graph Autoencoder. November 2019.

[26] Basmah Altaf, Shichao Pei, and Xiangliang Zhang. Scientific Dataset Discovery via Topic-level Recommendation, June 2021.

[27] Xu Wang, Frank van Harmelen, Michael Cochez, and Zhisheng Huang. Scientific Item Recommendation Using a Citation Network. In Gerard Memmi, Baijian Yang, Linghe Kong, Tianwei Zhang, and Meikang Qiu, editors, Knowledge Science, Engineering and Management, Lecture Notes in Computer Science, pages 469–484, Cham, 2022. Springer International Publishing.

[28] Ayush Singhal and Jaideep Srivastava. Research dataset discovery from research publications using web context. Web Intelligence, 15:81–99, May 2017.

[29] Anand Rajaraman and Jeffrey David Ullman. Data Mining, page 1–17. Cambridge University Press, 2011.

[30] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. arXiv:1301.3781 [cs], September 2013.

[31] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs], May 2019.

[32] Alex Bogatu, Alvaro A. A. Fernandes, Norman W. Paton, and Nikolaos Konstantinou. Dataset Discovery in Data Lakes. In 2020 IEEE 36th International Conference on Data Engineering (ICDE), pages 709–720, April 2020.

[33] Anne Humeau-Heurtier. Texture feature extraction methods: A survey. IEEE Access, 7:8975–9000, 2019.

[34] Fatima Khan, Mukhtaj Khan, Nadeem Iqbal, Salman Khan, Dost Muhammad Khan, Abbas Khan, and Dong-Qing Wei. Prediction of recombination

spots using novel hybrid feature extraction method via deep learning approach. Frontiers in Genetics, 11, 2020.
[35] Dhivya Chandrasekaran and Vijay Mago. Evolution of semantic similarity—a survey. ACM Computing Surveys, 54(2), February 2021.
[36] Megan Risdal and Timo Bozsolik. Meta kaggle, 2022.
[37] Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spaCy: Industrial-strength Natural Language Processing in Python. 2020.
[38] Florian Boudin. pke: an open source python-based keyphrase extraction toolkit. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations, pages 69–73, Osaka, Japan, December 2016.
[39] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. Advances in pre-training distributed word representations. In Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018), 2018.
[40] Steven Bird, Ewan Klein, and Edward Loper. Natural Language Processing with Python. O'Reilly Media, 2009.

TAKESHI SAKUMOTO is a Ph.D. candidate at Nagaoka University of Technology and a student of Doctoral Program for World-leading Innovative and Smart Education. He received the B.S. degree in information and management systems engineering from Nagaoka University of Technology, Japan, in 2020. He was awarded at the 36th Annual Conference of the Japanese Society of Artificial Intelligence (2022).

TERUAKI HAYASHI is a lecturer at The University of Tokyo. He received his Ph.D. degree in engineering from The University of Tokyo (2017). His research topics are knowledge structuring, data management, retrieval systems, and human behavior modeling, focusing on cross-disciplinary data collaboration in the data ecosystem. He is the coauthor of the book Market of Data (Kindaikagakusha, 2017), and Tools for Activating Data Marketplace (Springer, 2022). He was awarded the Dean's Award by the School of Engineering, The University of Tokyo (2017), an Excellence Award at the 32nd Annual Conference of the Japanese Society of Artificial Intelligence (2018), etc.

HIROKI SAKAJI is an associate professor at Hokkaido University. He received the Ph.D. degree in engineering from Toyohashi University of Technology (2012). His research interests are related to natural language processing, text mining concerning economics and finance. He served as an area chair of ACL and as PC members of many conferences (ACL, AAAI, IEEE BigData).

HIROFUMI NONAKA is an associate professor at Aichi Institute of Technology, Japan. He received the Ph.D. degree in Electrical and Electronic Information Engineering from Toyohashi University of Technology, Japan, in 2011.