Vol.2018-IFAT-131 No.1
2018/7/28

Uncomplicated Procedure for Thesaurus Mapping: Use of Stemming, Edit Distance and Vector Matching

Kazuaki KISHIDA[1,a)]

Abstract: This paper reports the results of an experiment on methods for finding similar terms in another thesaurus when a descriptor in one thesaurus is given, which is usually called thesaurus matching. The experiment used two well-known thesauri in the field of social science, the ICPSR Thesaurus (as the source) and the ERIC Thesaurus (as the target). First, 232 pairs of descriptors found by stem-based matching were examined by a human assessor. As a result, about 51% of them were categorized as 'equivalent' and 40% were judged as 'nearly equivalent'. The other 8.6% were mismatches. Next, by using two measures, an ERIC descriptor that was most similar with a given ICPSR descriptor was selected and evaluated. The measures were the edit distance (Levenshtein Distance) between names of descriptors and the cosine similarity between vectors constructed specially for the descriptors. The vectors consisted of terms extracted from the descriptor and its non-descriptors, broader terms, narrower terms, and related terms. When two descriptors had different stems, the vector matching by cosine similarity specified similar ERIC descriptors more successfully than the edit distance. The experiment thus suggested a three-stage procedure for thesaurus matching: 1) string matching in a case-insensitive manner, 2) stem-based matching, and 3) vector matching based on cosine similarity.

1. Introduction

In information retrieval (IR), *thesauri* have played an important role for many years. Needless to say, web search engines usually do not involve a manual indexing process based on controlled vocabularies, but there still exist situations in which a thesaurus would lead to better search results. First, highly specialized collections such as medical document databases continue to depend on thesaurus functions to enable users to attain better recall; the same is partly true for the retrieval of music, image or video materials not containing explicitly character-based information. Second, even though the descriptors of a thesaurus are not manually assigned to each item as index terms in a database, a so-called 'search thesaurus' can provide useful keywords or phrases for users who are not familiar with the vocabulary of the target collection (see Shiri, 2012 [24]).

To further enhance the effectiveness and usefulness of thesauri in IR, many researchers have focused on thesaurus mapping, which can be defined as "the process of identifying terms, concepts and hierarchical relationships that are approximately equivalent" (Doerr, 2001 [5]) on two or more thesauri. Thesaurus mapping between multiple thesauri, or between thesauri and other types of vocabularies, is indispensable for establishing terminological interoperability. For instance, Isaac et al.(2009) [10] enumerated its four practical roles: (1) reindexing (i.e., supporting the indexing process by a thesaurus based on index terms of the other thesaurus), (2) concept-based search across vocabularies, (3) navigation across thesauri, and (4) thesaurus merging.

Some studies have attempted to perform thesaurus mapping for improving search performance, such as linking between free-text terms and medical subject headings (MeSH) and EMBASE Thesaurus (EMTREE) terms (see McCulloch et al., 2005 [17]). More recently, the HILT project tried to map LCSH (Library of Congress Subject Headings), the UNESCO Thesaurus and MeSH to a 'central' DDC (Dewey Decimal Classification) spine (see Nicholson et al., 2006 [20]). Also, Liang et al.(2005) [14] and Liang & Sini (2007) [15] discussed thesaurus mapping from the Chinese Agricultural Thesaurus (CAT) to the AGROVOC Thesaurus (a controlled vocabulary covering food, nutrition, agriculture, fisheries, forestry, environment, etc.).

McCulloch & Macgregor(2008) [18] reviewed related works on thesaurus mapping and identified research activities on this topic before the 2000s[*1]. Some recent works are mentioned in Section 2. Furthermore, it should be noted that some vocabularies have become available in Linked Open Data (LOD) systems (e.g., DDC, LCSH, AAT, etc.), and semantic interoperability among them is widely recognized as an important issue. Binding & Tudhope(2016) [3] discussed extensively vocabulary matching in a linked data environment.

This paper explores a practical method for automatically finding 'similar' descriptors between two thesauri. ISO 25964-2:2013(E) [11] defines three types of mapping, 'equivalence', 'hierarchical' and 'associative', and the equivalence is more pre-

[1] School of Library and Information Science, Keio University, Minato-ku, Tokyo 108–8345, Japan
[a)] kz_kishida@keio.jp

[*1] Also, they reported an experiment of mapping from LCSH, MeSH, UNESCO Thesaurus and AAT (Art & Architecture Thesaurus) to DDC [18]. Interestingly, the mapping results were categorized into Chaplan's 19 match types (Chaplan, 1995 [4]).

cisely divided into 'exact equivalence', 'inexact equivalence' and 'partial equivalence' in the standard. As discussed below, it is not so difficult to detect automatically two descriptors having the 'exact' equivalence relationship by *string matching* with *stemming* (or computation of an *edit distance*). Therefore, the experiment focused on particularly other relationships such as 'inexact' and 'partial' equivalence, and 'hierarchical' or 'associative'.

For the mapping, the *cosine similarity* between two *vectors* of descriptors was employed in the traditional IR manner where the vector was constructed from a set of the descriptor, corresponding non-descriptors designated by UF, broader terms (BTs), narrower terms (NTs) and related terms (RTs). Actually, the ERIC (Education Resources Information Center) Thesaurus and ICPSR (Inter-university Consortium for Political and Social Research) Thesaurus were used in the experiment, and the effectiveness of the method was empirically examined. Although its implementation is relatively uncomplicated, the *vector matching* by the cosine measure would be able to detect successfully many 'similar' descriptors by applying jointly *stem-based matching*.

This paper is organized as follows. The methods and procedures are explained in Section 3 after related works are reviewed in Section 2. Section 4 describes the data and results of the experiment, then Section 5 discusses the results.

2. Related works

2.1 Automatic mapping

In biomedical fields, thesaurus mapping is feasible through the UMLS (Unified Medical Language System) Metathesaurus, which is a useful tool for inter-terminology mapping. For instance, Fung et al.(2007) [8] examined empirically two approaches using the UMLS Metathesaurus for mapping of terms from SNOMED CT (Systematized Nomenclature of Medicine – Clinical Terms) to ICD-9-CM (International Classification of Diseases, Ninth Revision, Clinical Modification). One of their approaches was based on the MetaMap algorithm (Aronson, 2001 [2]) that enables us to relate given biomedical text to UMLS concepts. Also, Saitwal et al. (2012) [23] developed methods for associating a medication code with concepts in medical vocabularies to improve the process of searching stored clinical records.

For detecting 'exact match links' between two aligned vocabularies, Morshed et al.(2011) [19] tried to match a term in the AGROVOC Thesaurus with those in six thesauri, NALT (National Agricultural Library Thesaurus), GEMET (General Multilingual Environmental Thesaurus), LCSH, RAMEAU (a heading list used at the French National Library), EUROVOC (EU's multilingual and multidisciplinary thesaurus) and STW (Standard-Thesaurus Wirtshaft). When character strings of two terms were perfectly identical in a case-insensitive manner, they were considered to be exactly matched. If not so, a similarity was computed based on a well-known distance measure between two strings, and pairs of terms with high values of similarity were selected as matched terms. The experiment showed good precision of the technique with some failure cases such as complete homonymy, almost-homonymy,'false friends' (i.e., with different meaning), etc.

In Ahn et al.(2014) [1], the mapping task was to detect ARTSTOR concepts containing descriptors in AAT as their components (because ARTSTOR descriptors were a longer phrase consisting of two or more components as a general tendency). A set of heuristic rules for bridging representational gaps of descriptors between the two vocabularies was employed. Basically, the matching operation was based on capitalization, singularization, spelling correction and conversion of British spelling to American.

Lin et al.(2015) [16] proposed a framework of 'taxonomy metamatching' that incorporated four taxonomy matchers (i.e., matching modules) such as string-based (using edit distance), property-based, structure-based (checking superclasses and subclasses) and semantic-based (employing WordNet) matchers.

In the field of linked data, Binding & Tudhope(2016) reported informally experimental results of applying general software tools such as Silk[*2] for mapping between linked data items. Other tools are introduced on the W3C site[*3].

2.2 Ontology Alignment Evaluation Initiative (OAEI)

The Ontology Alignment Evaluation Initiative (OAEI) is an international research project to develop and explore *ontology matching* techniques, which started in 2004, and includes several tracks using thesauri as resources to be matched[*4]. For instance, Library Track in the project tried to match

- GTT and the Brinkman Thesaurus in 2007 and 2008,
- LCSH, RAMEAU, and SWD (a heading list used at the German National Library) in 2009, and
- The STW Thesaurus for Economics and the TheSoz (The Thesaurus for the Social Sciences) in 2012 to 2014.

Also, in Food Track (2006 and 2007)[*5], AGROVOC and NALT were matched, and GEMET was added to them in Environment Track (2007), Furthermore, Large Biomedical Ontologies Track (2012 –) tried to find alignments between the Foundational Model of Anatomy (FMA), SNOMED CT, and the National Cancer Institute (NCI) Thesaurus.

In its research activities, the performance of participants' systems was typically evaluated by F-measure computed based on 'reference alignments' that human experts provided. For instance, the systems with the highest values of F-measure were ODGOMS and YAM++ in Library Track 2013 (Grau et al., 2013 [9]).

The ODGOMS system combined finally the results of several matching modules, some of which were LCSMatcher (finding a best matched entity with highest LCS (Longest Common Subsequence) similarity), SMOAMatcher (based on SMOA similarity, not LCS), PurityMatcher (removing *stopwords* from labels), TFIDFMatcher (computing tf-idf cosine similarity between classes), and NETMatcher (finding a best matched class with highest NET (named-entity transformation) similarity) (Kuo & Wu, 2013 [12]). Note that SMOA (String Metric for Ontol-

[*2] http://silkframework.org/
[*3] https://www.w3.org/wiki/TaskForces/CommunityProjects/LinkingOpenData/EquivalenceMining
[*4] http://oaei.ontologymatching.org/
[*5] Lauser et al.(2008) [13] selected five systems that participated in the track, and analyzed the results of a qualitative evaluation of their mapping by human experts.

ogy Alignment) is a metric of measuring similarity between two strings, which was developed specially for ontology matching (see Stoilos et al., 2005 [25]).

In YAM++[*6], results from three matching modules, Terminological Matcher, Contextual Matcher and Instance-based Matcher, were combined for obtaining final alignments (Ngo & Bellahsene, 2013 [21]), where the Terminological Matcher employed several techniques such as edit distance-based methods, token-based methods, machine learning-based approach, IR-based approach, etc. which were described by Ngo et al. (2013) [22]. Ngo et al. (2013) [22] reported that a machine learning-based approach with J48 decision trees and an IR-based approach using tf-idf weights outperformed the other methods.

2.3 Co-occurrence mapping

If documents were manually indexed by two different vocabularies, then it is possible to identify a linkage between two terms in the vocabularies according to the number of documents including both terms, which is often called *co-occurrence mapping* (see ISO 25942-2:2013(E) [11]). For instance, Zhang et al.(2011) [27] reported an experimental result of automatic mapping from DDC to CLC (Chinese Library Classification) based on the number of USMARC records containing simultaneously the two classification numbers. A similar technique was explored by Du et al.(2017) [6] for establishing interoperability among KOSs (Knowledge Organization Systems) for the purpose of research management.

2.4 Multilingual mapping

Subject (or semantic) interoperability often has to be established between a pair of thesauri represented by different languages. Although such kind of multilingual mapping is important and several attempts have been made, this paper considers thesaurus mapping in a monolingual setting.

3. Uncomplicated methods for finding similar descriptors

3.1 Basic assumptions

This paper explores automatic methods for finding 'similar' descriptors in a 'target' thesaurus when a descriptor in the 'source' thesaurus is given. More precisely, 'inexact' and 'partial' equivalence relationships are supposed to be primary similar relationships between two descriptors. By ISO 25964-2:2013(E) [11], inexact equivalence occurs when "the most closely concepts in two or more vocabularies are not exactly the same", and the partial equivalence means that "the one concept is slightly broader than the other". Note that when character strings of two descriptors are perfectly identical in a case-insensitive manner, it is categorized as an 'exact' equivalence match as defined by Morshed et al.(2011) [19].

Additionally, as discussed below, many descriptors having 'hierarchical' and 'associative' relationships with the source descriptor were specified as candidates of 'similar' ones in the experiment. Ideally, they should have been explicitly discerned from 'inexact' and 'partial' equivalence, but this was beyond the scope of our experiment. Therefore, (1) 'inexact' equivalence, (2) 'partial' equivalence, and (3) 'hierarchical' and (4) 'associative' relationships, other than 'exact' equivalence, are operationally considered to be 'similar' relationships in this paper. The definition of 'similar' may be too broad; further research is needed on automatically categorizing the relationships more precisely.

In this paper, it is assumed that external resources such as UMLS (an inter-terminology mapping tool), WordNet (a general thesaurus) or documents indexed by the thesauri are not available when searching 'similar' descriptors. This means that similar descriptors have to be determined by using only information inherent in thesauri to be matched.

3.2 Methods

Despite not using any external resource, many methods for ontology matching such as those enumerated by Euzenat & Shvaiko (2013) [7] may still be applicable to the problem. However, this paper investigates particularly three methods based on stemming (ST), edit distance (ED) and cosine similarity (CS).

3.2.1 Stemming and edit distance

Among them, ST and ED are relatively easy to implement because it is enough to examine only character strings of the descriptors (i.e., names or representations of the descriptors) by using software tools. More precisely, these methods specify a similar descriptor as follows.

- Stemming (ST): If two character strings of the descriptors become perfectly identical after stemming them, the two descriptors are regarded to be similar.
- Edit distance (ED): Similarity values based on an edit distance (e.g., Levenshtein distance) are computed between the character string of a given descriptor in the source thesaurus and those of all descriptors in the target thesaurus, and a descriptor with the highest value is selected as similar one for the given descriptor in the source thesaurus.

When a descriptor consists of two or more words, individual words are stemmed by the algorithm. For example, "Digital Divide" is converted to "digit divid". On the other hand, in computing ED, descriptors are not decomposed.

3.2.2 Cosine similarity (vector similarity)

Cosine similarity (CS) has already been used for ontology matching (see Kuo & Wu, 2013 [12]). For computing the CS measure between two descriptors, a *vector* of each descriptor has to be constructed according to a rule. As a trial, this paper tests a novel construction of the vector, elements of which correspond to terms that are extracted from the descriptor, its non-descriptors (designated by UF), its broader terms (BTs), its narrower terms (NTs) and its related terms (RTs). By creating a traditional document vector (i.e., one-hot encodings) from the set of terms, a vector similarity between two descriptors can be computed by using the cosine measure based on IR theory, which leads to selection of a descriptor with the highest cosine value as a similar one. If a threshold value of similarity by ED or CS is set, then it would be possible to identify multiple similar descriptors whose similarities are over the threshold, but only a single similar descriptor is specified in this paper due to a difficulty of determining properly

[*6] http://yamplusplus.lirmm.fr/index

the threshold value.

Operationally, the descriptor, non-descriptors, BTs, NTs and RTs are simply concatenated and transformed to a pseudo-document, which is broken down into a set of index terms to be treated as a *bag-of-words* (note that a descriptor may consist of two or more words). In the indexing process, conversion to lower-case, removing stopwords and stemming are applied according to an IR practice. For example, if a descriptor "University Libraries" has "College Libraries" as a non-descriptor, "Libraries" as a BT, "Undergraduate Libraries" as a NT and "Public Libraries" as a RT, then the set of index terms becomes {college, librar, public, undergrad, univers}, and the *term frequency* (TF) of "librar" amounts to four and that of the others is one. Although a tf-idf weighting scheme is available for the process, this paper omits intentionally the IDF factor.

3.2.3 Use of partial tree

Each descriptor can be regarded as a 'node' in a network representing a hierarchical or polyhierarchical structure of descriptors prescribed by a thesaurus. The similarity by ED or CS described in the previous sections is defined between two single nodes of different networks, and is computed only from information contained in these nodes.

It may also be possible to consider a descriptor to be equivalent with a 'partial tree' in which the descriptor is located at the root node and all its subordinate descriptors (i.e., NTs) are included, as shown in Figure 1. Thus, the similarity between two descriptors can be calculated from a comparison between two corresponding partial trees, not single nodes. This implies that substructures inherent in a thesaurus are partly taken into account for thesaurus matching.

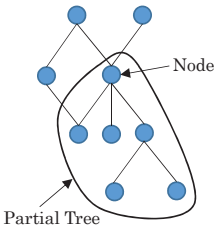

Fig. 1 Example of partial tree

If the degree of similarity between two single nodes, v and u, is denoted by $s(v,u)$, then the value of similarity between two partial trees can be simply defined such that

$$S(T_v, T_u) = \frac{1}{|T_v| \times |T_u|} \sum_{v \in T_v} \sum_{u \in T_u} s(v,u) \qquad (1)$$

where T_v and T_u indicate the set of nodes included in the partial trees, respectively. This paper calculates $s(v,u)$ as cosine similarity (CS) that is defined operationally in this section, which means that CS was doubly computed between a pair of nodes (denoted by CS-N) and between a pair of partial trees (denodted by CS-T) in the experiment. Note that if both the descriptors correspond to a leaf node (i.e., with no child node), then two values of CS-N and CS-T become identical inevitably.

Needless to say, it is possible to apply any formula other than Equation (1), and $s(v,u)$ can be defined as a measure other than CS. However, this paper uses only Equation (1) and CS as a trial.

4. Experiment

4.1 Data

As the source thesaurus, the experiment adopted Subject Thesaurus included in the ICPSR Thesaurus, which covers a wide range of social science areas such as political science, sociology, history, economics, education, criminal justice, gerontology, demography, public health, law, and international relations[*7]. On the other hand, the ERIC Thesaurus was employed as the target thesaurus. Because the ERIC Thesaurus is "a list of terms representing research topics in the field of education"[*8], the ICPSR Thesaurus is more general. Therefore, if similar descriptors are found in the ERIC Thesaurus, it is expected to expand the ICPSR Thesaurus so as to include more specific descriptors that are related to education. This is a kind of thesaurus merging (see Zeng & Chan, 2004 [26] for details), but its technique is not discussed in this paper.

The two thesauri were chosen for the experiment because they are not large and the present author is more familiar with social science terms than those in natural sciences such as medical terms. Machine readable data sets of both the thesauri were downloaded from the web sites of these thesauri in July 2017.

The version of the ICPSR Thesaurus included 3,265 entries of descriptors (i.e., there were 3,265 'preferred terms' other than non-descriptors), which indicates that 3,265 nodes existed in the network, while the ERIC Thesaurus contained 4,520 entries. Therefore, edit distance (ED) or cosine similarity (CS) were computed for 14,757,800 pairs of entries ($= 3,265 \times 4,520$) in total. All terms were in English (i.e., monolingual matching) and because both the thesauri contained some descriptors having two or more parents (i.e., BTs), their structure was polyhierarchical as shown in Figure 1.

4.2 Implementation and evaluation

The matching system in the experiment was basically implemented by using Java language. Before all matching operations, characters included in all terms were converted to lower-case by using a method of String class of Java. Concurrently, parentheses within names of descriptors and non-descriptors were removed. For example, "Educational Equity (Finance)" was transformed into "educational equity finance" [*9].

The well-known Porter's algorithm[*10] was employed for the stemming, and similarity based on edit distance was calculated by LevensteinDistance class of the Apache Lucene project[*11] (i.e., only the Levenshtein distance was used in the experiment). Also, when vectors of descriptors were constructed, stopwords were removed and each word was stemmed by Porter's algorithm.

In the experiment, two methods of evaluation were adopted. First, a human assessor classified each descriptor that was identified by stem-based matching into (a) equivalent term, (b) re-

[*7] https://www.icpsr.umich.edu/icpsrweb/ICPSR/thesaurus/index
[*8] https://eric.ed.gov/?ti=all
[*9] In this case, "(Finance)" is a qualifier for disambiguating the sense of "Educational Equity". A special rule for processing qualifiers may be required for more effective matching.
[*10] http://snowballstem.org/
[*11] https://lucene.apache.org/

lated term, and (c) unrelated term. Second, descriptors specified by ED, CS-N and CS-T, respectively, for each source descriptor were compared by a human assessor, and the better one was manually determined (i.e., comparative evaluation). The details will be explained later.

4.3 Result of exact matching in a case-insensitive manner

Each non-descriptor designated generally by UF in an entry can be considered to be an 'alias' of the descriptor. Exact matching in a case-insensitive manner between two character strings may occur not only between descriptors in the target and source thesauri but also between a descriptor and a non-descriptor or between non-descriptors. For example, an ICPSR descriptor "ability" having an alias "talent" was exactly matched with two ERIC descriptors "Ability" and "Talent" (i.e., the ERIC Thesaurus distinguished "Talent" from "Ability" unlike the ICPSR Thesaurus). In such cases, "ability" was categorized as 'descriptor has one or more equivalents in the target thesaurus', and the number of equivalents in the target thesaurus (i.e., times of exact matching) was counted as two.

Among 3,265 ICPSR descriptors, 1,118 (34.2%) had one or more equivalents in the ERIC Thesaurus, and the total times of exact matching was 1,221.

4.4 Result of stem-based matching

When character strings of two descriptors are perfectly identical, they matched again after stemming. Thus, this section focuses on only the case that an ERIC descriptor with a different form of strings was found by applying the stemming algorithm. For example, "activism" in the ICPSR Thesaurus had an exactly equivalent ERIC descriptor "Activism", and also, matched with "Activities" after stemming (the common stem was "activ"). In this case, only "Activities" is taken into consideration here. Actually, after stemming, just 200 ICPSR descriptors (6.1%) matched with ERIC descriptors having a different form, and the total number of matched ERIC descriptors was 232 (note that some ICPSR descriptors matched two or more ERIC descriptors).

Basically, there is no perfect stemming algorithm, which implies that over-stemming and under-stemming may occur. In thesaurus matching, over-stemming causes errors. For example, "hospitalization" in the ICPSR Thesaurus was improperly matched with "Hospitals" in the ERIC Thesaurus because Porter's algorithm generated the same stem "hospit" from both of them.

A human assessor examined ERIC descriptors found by stem-based matching, and classified each case manually into the following three categories.

- Equivalent: e.g., "arts" and "Art",
- Nearly equivalent: e.g., "budgets" and "Budgeting", and
- Erroneous: e.g., "hospitalization" and "Hospitals".

Actually, the 'Nearly equivalent' matching included several cases in which two descriptors were regarded to be almost identical: one descriptor may be a broader term (BT), or a related term (RT) of the other description, etc. Although these cases should have been ideally discerned into separate categories, it was too difficult to assign a single category to some cases in an objective manner.

Table 1 summarizes the result of categorization where 'Alias' indicates a non-descriptor designated by UF, and say, 'Alias to Desc' means that a non-descriptor in the ICPSR Thesaurus was matched with an ERIC descriptor, not an ERIC non-descriptor. As shown in the table, over half (51.3%) of 232 ERIC descriptors were evaluated to be equivalent with the source descriptors, and almost the same number of ERIC descriptors (40.1%) were categorized as 'Nearly equivalent matching'. In the author's impression, many related terms were found by the stem-based matching such as "Administrators" for "administration". Erroneous matching accounted for only 8.6%, which suggested that stem-based matching can effectively find exact and inexact (or partial) equivalents for thesaurus mapping.

Table 1 Result of stem-based matching

Type	Equiv.	Nearly Equiv.	Error	Total	%
Desc to Desc	53	48	14	115	49.6%
Desc to Alias	52	33	3	88	37.9%
Alias to Desc	4	5	3	12	5.2%
Alias to Alias	10	7	0	17	7.3%
Total	119	93	20	232	100%
%	51.3%	40.1%	8.6%	100%	

Note: "Desc" indicates a descriptor.

4.5 Result of matching by similarity measures – (1)

In the case of matching based on similarity measures, ERIC descriptors specified by the three measures (ED, CS-N and CS-T), respectively, were compared between a pair of measures, and a better descriptor was determined by a human assessor in each case. For example, about "arts funding" in the ICPSR Thesaurus, "Early Reading" and "Arts" were selected by ED and CS-N, respectively, and it was judged that CS-N found a better descriptor. When it was difficult to determine definitely a better one, the case was safely recorded to be even (e.g., for "urban decline" in the ICPSR Thesaurus, "Urban Areas" and "Urban Environment" were specified by CS-N and CS-T, respectively, and it could not be decided which was better).

The comparison between ED and CS-N was limited to only 639 ICPSR descriptors having an ERIC descriptor of which the CS-N value was over 0.75. Inevitably, the evaluation result was biased so that the effectiveness of CS-N was overestimated in comparison to ED. However, an explicit tendency became clear from the limited sample, as shown in Table 2. This table shows the numbers of ICPSR descriptors, which are specially divided into 'Same stem' and 'Diff. stem' cases. If an ERIC descriptor specified by either similar measure matched exactly with the source ICPSR descriptor after stemming, then it was categorized as 'Same stem', and if not, it was counted as 'Diff. stem'.

Cases in which there was almost no difference between descriptors found by ED and CS-N (i.e., the two ERIC descriptors were judged to be even, or in some cases, the same ERIC descriptor was specified by both the measures) are shown under the label 'ED = CS-N' in Table 2, to which about half of ICPSR descriptors (49.8%) belongs. Although the percentages of 'ED > CS-N' (i.e., more similar ERIC descriptors were found by ED) cases and

Table 2 Result of comparison between ED and CS-N

Evaluation	Same stem[1]	Diff. stem[2]	Total	%
ED = CS-N	196	122	318	49.8%
ED > CS-N	166	12	178	27.9%
ED < CS-N	2	141	143	22.4%
Total	364	275	639	100%
%	57.0%	43.0%	100%	

Note: 1) Stems of two descriptors are identical.
2) Stems of two descriptors are different.

'ED < CS-N' cases were not largely different (i.e., 27.9% and 22.4 %), the difference between them becomes clear when taking the grouping of 'Same stem' and 'Diff. stem' into consideration. When stems of the ICPSR and ERIC descriptors were identical, the ED measure identified explicitly more similar or equal ERIC descriptors (166 vs. 2 except for not different cases, as indicated in Table 2).

By considering the mechanism of ED, this result would be natural because a basic function of ED is to detect similar character strings, which is common to stemming operation. On the other hand, when two stems disagreed, ED-based matching tended inevitably to specify dissimilar descriptors. For example, there was no ERIC descriptor having the same stem of "senility" in the ICPSR Thesaurus. In this case, "Sexuality" was specified by ED whereas "Alzheimers Disease", which is closely related to senility, was identified by CS-N.

When character strings of two semantically similar descriptors are largely different, CS-based vector matching may play a key role. Actually, the CS-N measure found more similar descriptors many times for 'Diff. stem' cases (141 vs. 12 except for not different cases, as shown in Table 2). In order to examine what descriptors were detected by CS-N, the 639 ERIC descriptors were manually categorized as (1) perfectly or almost identical string, (2) conceptually almost equivalent, (3) broader than the source descriptor, (4) narrower than the source descriptor, (5) related to the source descriptor, and (6) mis-match. In the case of stem-based matching, such kind of detailed categorization was not possible because the difference between two descriptors was so small in many cases, but it was relatively easy to assign each descriptor to a category for the result of CS-based vector matching.

A result of the categorization is shown in Table 3. The dominant category was 'Related term' (36.3%) which included various relationships between the source and target descriptors. For example, "school attendance" (ICPSR) and "Truancy" (ERIC) are closely related, and also, "Overpopulation " (ERIC) may be a result of "population growth" (ICPSR).

Interestingly, more descriptors were interpreted as a BT than those interpreted as a NT (23.0% vs. 5.0%). Examples are as follows.

- BT - ICPSR:"agricultural services", ERIC:"Agriculture"
- NT - ICPSR: "records", ERIC:"Confidential Records"

As indicated by these examples, CS-based vector matching would be able to specify similar descriptors when the numbers of component words are different, which leads to detection of hierarchical relationships[*12].

[*12] Of course, the hierarchical relationship may be found by a heuristic rule. For example, "records" is a 'head' word of "Confidential Records".

Table 3 ERIC descriptors detected by CS-N

Categories	Same stem[1]	Diff. stem[2]	Total	%
Equiv. string[3]	193	0	193	30.2%
Concept. almost equiv.	1	26	27	4.2%
Broader term	48	99	147	23.0%
Narrower term	17	15	32	5.0%
Related term	102	130	232	36.3%
Mis-match	3	5	8	1.3%
Total	364	275	639	100%
%	57.0%	43.0%	100%	

Note: 1) Stems of two descriptors are equal.
2) Stems of two descriptors are different.
3) This can be identified by the stemming algorithm.

Also, the CS-N measure could identify conceptually almost equivalent ERIC descriptors such as "Undocumented Immigrants" for "illegal immigrants" in the ICPSR Thesaurus although the share was small (4.2%). However, there were some cases where it was difficult to classify a descriptor into 'Conceptually almost equivalent' or 'Related term'. Fortunately, mis-match cases occupied only 1.3% (e.g., "informal economy" (ICPSR) and "Access to Information" (ERIC)) [*13].

4.6 Result of matching by similarity measures – (2)

Table 4 shows a result of comparative evaluation for two samples in which the top-ranked 100 ERIC descriptors with the highest values of CS-N and CS-T are included, respectively (e.g., the descriptors were sorted by CS-N values in descending order and the top-ranked 100 descriptors were selected for the evaluation). As mentioned above, the difference between CS-N and CS-T appears only when either the source or target descriptors are not a leaf node at least. Therefore, in the process of selecting 100 descriptors, the case where both the source and target descriptors were a leaf node was ignored for the selection.

Table 4 Comparison between CS-N and CS-T for the top 100 descriptors

Evaluation	Top 100 by CS-N	Top 100 by CS-T
CS-N = CS-T	60	67
CS-N > CS-T	30	22
CS-N < CS-T	10	11
Total	100	100

Note: Cases where the two descriptors are a leaf node are not included.

In both the samples, there was no explicit difference between two ERIC descriptors specified by the measures in over half of the cases (i.e., 60 and 67), but it seems that CS-N slightly outperformed CS-T (i.e., 30 vs. 10 and 22 vs. 11). It would not have been effective in the experiment to measure the similarity degree between descriptors by interpreting a descriptor as a partial tree of the polyhierarchical structure.

5. Discussion

Table 5 shows the numbers of ICPSR descriptors for which similar ERIC descriptors were specified by stemming and/or CS-N measure with a threshold of 0.75. Similar descriptors were found by both the methods for 72 ICPSR descriptors (2.2%) (note that the two ERIC descriptors of each ICPSR descriptor were not

[*13] Note that some mis-matches may be categorized into 'related term' in a broader sense.

always the same). Also, for 128 descriptors (3.9%), only the stem-based matching detected similar ones, and for 567 descriptors (17.4%), only the CS-N measure with a threshold of 0.75 detected similar ones. On the other hand, a similar descriptor could not be identified for most of the ICPSR descriptors (2,498, 76.5%).

Table 5 ICPSR descriptors for which a similar descriptor was specified

	CS-N > .75	CS-N ≦ .75	Total	%
Stem matching	72	128	200	6.1%
No stem matching	567	2498	3065	93.9%
Total	639	2626	3256	100%
%	19.6%	80.4%	100%	

Needless to say, if a lower threshold is used for the CS measure, then the descriptors for which a similar descriptor is found will increase. Instead, it is expected that dissimilar descriptors will tend to be specified as the threshold becomes smaller. This is a trade off between the number of descriptors that are potentially similar and the degree to which 'correctly similar' descriptors are successfully specified. Unfortunately, this paper can not discuss an optimal value of the threshold.

In order to find exactly identical descriptors, string matching after converting upper-case characters into lower-case (or vice versa) is indispensable. After that, it is effective to compare stems between a given source descriptor and target descriptors although sometimes erroneous matching may occur as exemplified in Table 1. By recording the stems of all target descriptors into a hash table or a binary search tree, stem-based matching can be efficiently executed.

When trying to find further similar descriptors, it would be possible to use similarity measures based on an edit distance between two strings of the descriptors or a cosine value between vectors constructed for the descriptors. The cosine measure may be more effective for specifying similar descriptors that are not found by stem-based matching (see Table 2). As already mentioned, this is easily understood because the edit distance and stem-based matching bear a resemblance in terms of operations on the character string of each descriptor. On the other hand, if the vectors are constructed by using BTs, NTs, RTs, etc., it may be possible to find a similar descriptor having a word form different from that of the source descriptor. Particularly, in this experiment, the CS-N measure specified many descriptors interpreted as a BT of the source descriptor (see Table 3). Note that similarity-based matching involves more computational complexity since all pairs of descriptors have to be treated straightforwardly in the process, and additionally, vectors must be constructed in order to use the CS-measure. However, in this experiment, there was no problem in this regard because both the thesauri were not so large.

All terms (the descriptor, non-descriptors, BTs, NTs and RTs) were equivalently weighted when constructing a vector of each descriptor in this experiment. Better results may be obtained by introducing a special weighting scheme (e.g., increasing the weight of BTs and NTs and decreasing that of RTs) or adding an IDF factor, which is a subject of future research.

In summary, an uncomplicated procedure for automatically finding similar descriptors is as follows.

(1) Matching of character strings in a case-insensitive manner,
(2) Stem-based matching by using a stemming algorithm, and
(3) Selection of top-ranked descriptors whose vectors have the highest cosine similarity (CS) with that of a given source descriptor.

According to the result of this experiment, CS-N is better than CS-T in stage (3). Needless to say, the CS-T measure or an edit distance can be applied as a second method although more unrelated terms may be found.

6. Concluding remarks

This paper reported an experiment for testing some uncomplicated methods of thesaurus matching, which were (1) matching of character strings in a case-insensitive manner, (2) matching of character strings after stemming (stem-based matching) and (3) similarity-based matching. In the similarity-based matching, the Levenshtein distance and cosine similarity were used. The cosine similarity was computed for vectors that were specially constructed from the descriptor, non-descriptors, broader terms, narrower terms and related terms listed in the entry of thesauri according to IR practice. It turned out that this kind of vector matching can specify successfully a linkage between descriptors having different word forms whose stems are not identical.

As already mentioned, several issues require further research. In particular, it may be required to take the structural characteristics of thesauri into consideration. This experiment attempted to do so by regarding a descriptor as a partial tree in the polyhierarchical structure of the thesauri, but the method did not improve the overall effectiveness of cosine similarity-based matching. It would be worth exploring other techniques based on the thesaurus structure.

References

[1] Ahn, J.-W., Soergel, D., Lin, X., Zhang, M., and Ying W.: Mapping between ARTstor Terms and the Getty Art and Architecture Thesaurus, *Proceedings of the Thirteenth International ISKO Conference*, pp. 184 – 191 (2014).
[2] Aronson, A. R.: Effective Mapping of Biomedical Text to the UMLS Metathesaurus: The MetaMap Program, *AMIA Annual Symposium Proceedings, 2001*, pp.17 – 21 (2001).
[3] Binding, C., and Tudhope, D.: Improving Interoperability Using Vocabulary Linked Data, *International Journal on Digital Libraries*, Vol.17, pp. 5 – 21 (2016).
[4] Chaplan, M. A.: Mapping "Laborline Thesaurus" Terms to Library of Congress Subject Headings: Implications for Vocabulary Switching, *Library Quarterly*, Vol. 65, No.1, pp. 39 – 61 (1995).
[5] Doerr, M.: Semantic Problems of Thesaurus Mapping, *Journal of Digital Information*, Vol.1, No.8 (2001).
[6] Du, W., Cheng, X., Yang, C., Sun, J., and Ma, J.: Establishing Interoperability among Knowledge Organization Systems for Research Management: A Social Network Approach, *Scientometrics*, Vol. 112, pp.1489 – 1506 (2017).
[7] Euzenat, J., and Shvaiko, P.: *Ontology Matching*, second edition, Springer-Verlag, Berlin (2013).
[8] Fung, K. W., Bodenreider, O., Aronson, A. R., Hole, W. T., and Srinivasan, S.: Combining Lexical and Semantic Methods of Inter-terminology Mapping Using the UMLS, *Studies in Health Technology and Informatics*, Vol.129, pp. 605 – 609 (2007).
[9] Grau, B. C., Dragisic, Z., Eckert, K., Euzenat, J., Ferrara, A., Granada, R., Ivanova, V., Jiménez-Ruiz, E., Kempf, A. O., Lambrix, P., Nikolov, A., Paulheim, H., Ritze, D., Scharffe, F., Shvaiko, P., Trojahn, C., and Zamazal, O.: Results of the Ontology Alignment Evaluation Initiative 2013, http://oaei.ontologymatching.org/2013/results/oaei2013.pdf (2013).
[10] Isaac, A., Wang, S., van der Meij, L., Schlobach, S., Zinn, C., and Matthezing, H.: Evaluating Thesaurus Alignments for Semantic Inter-

operability in the Library Domain, *IEEE Intelligent Systems*, Vol.24, No.2, pp.76 – 86 (2009).
[11] *ISO 25964-2:2013(E): Information and Documentation – Thesauri and Interoperability with Other Vocabularies – Part 2: Interoperability with Other Vocabularies*, (2013).
[12] Kuo, I. -H., and Wu, T.: ODGOMS: Results for OAEI 2013, http://ceur-ws.org/Vol-1111/oaei13_paper8.pdf (2013).
[13] Lauser, B., Johannsen, G., Caracciolo, C., van Hage, W. R., Keizer, J., and Mayr, P.: Comparing Human and Automatic Thesaurus Mapping Approaches in the Agricultural Domain, *Proceedings of International Conference on Dublin Core and Metadata Applications 2008*, (2008).
[14] Liang, A., Sini, M., Chun, C., Sijing, L., Wenlin, L., Chunpei, H., and Keizer, J.: The Mapping Schema from Chinese Agricultural Thesaurus to AGROVOC, *6th Agricultural Ontology Service (AOS) Workshop on Ontologies*, (2005).
[15] Liang, A. C., and Sini, M.: Mapping AGROVOC and the Chinese Agricultural Thesaurus: Definitions, Tools, Procedures, *New Review of Hypermedia and Multimedia*, Vol.12, pp. 51 – 62 (2007).
[16] Lin, H., Wang, Y., Jia, Y., Xiong, J., Zhang, P., and Cheng, X.: An Ensemble Matchers Based Rank Aggregation Method for Taxonomy Matching, *Web Technologies and Applications: 17th Asia-Pacific Web Conference, APWeb 2015* (Cheng, R. et al. eds.), Springer, pp. 190 – 202 (2015).
[17] McCulloch, E., Shiri, A., and Nicholson, D.: Challenges and Issues in Terminology Mapping: A Digital Library Perspective, *Electronic Library*, Vol.23, No.6, pp. 671 – 677 (2005).
[18] McCulloch, E., and Macgregor, G.: Analysis of Equivalence Mapping for Terminology Services, *Journal of Information Science*, Vol. 34, No. 1, pp. 70 – 92 (2008).
[19] Morshed, A., Caracciolo, C., and Johannsen, G.: Thesaurus Alignment for Linked Data Publishing, *Proceedings of International Conference on Dublin Core and Metadata Applications 2011*, (2011).
[20] Nicholson, D., Dawson, A., and Shiri, A.: HILT: A Pilot Terminology Mapping Service with a DDC Spine, *Cataloging & Classification Quarterly*, Vol. 42, No. 3/4, pp. 187 – 200 (2006).
[21] Ngo, D.-H., and Bellahsene, Z.: YAM++: Results for OAEI 2013. http://ceur-ws.org/Vol-1111/oaei13_paper16.pdf, (2013).
[22] Ngo, D.-H., Bellahsene, Z., and Todorov, K.: Opening the Black Box of Ontology Matching, *The Semantic Web: Semantics and Big Data: Proceedings of 10th International Conference, ESWC 2013*, pp.16 – 30 (2013).
[23] Saitwal, H., Qing, D., Jones, S., Bernstam, E. V., Chute, C. G., and Johnson, T. R.: Cross-terminology Mapping Challenges: A Demonstration Using Medication Terminological Systems, *Journal of Biomedical Informatics*, Vol.45, pp. 613 – 625 (2012).
[24] Shiri, A.: *Powering Search: The Role of Thesauri in New Information Environments*, Information Today, New Jersey (2012).
[25] Stoilos, G., Stamou, G., and Kollias, S.: A String Metric for Ontology Alignment, *The Semantic Web - ISWC 2005 : 4th International Semantic Web Conference, ISWC 2005* (Gil, Y. et al. eds), Springer, pp.624 – 637 (2005).
[26] Zeng, M. L., and Chan, L. M.: Trends and Issues in Establishing Interoperability among Knowledge Organization Systems, *Journal of the American Society for Information Science and Technology*, Vol. 55, No. 5, p.377 – 395 (2004).
[27] Zhang, Y., Peng, J., Huang, D., and Li, F.: Design of Automatic Mapping System between DDC and CLC, *Proceedings of the 13th International Conference on Asia-Pacific Digital Libraries, ICADL 2011* (Xing, C et al. eds.), Springer, pp. 357 – 366 (2011).