Integrating Multiple Internet Directories by Instance-based Learning

Ryutaro Ichise, Hiedeaki Takeda, Shinichi Honiden
National Institute of Informatics
2-1-2 Hitotsubashi Chiyoda-ku, Tokyo, 101-8430 Japan
{ichise,takeda,honiden}@nii.ac.jp

Abstract

Finding desired information on the Internet is becoming increasingly difficult. Internet directories such as Yahoo!, which organize web pages into hierarchical categories, provide one solution to this problem; however, such directories are of limited use because some bias is applied both in the collection and categorization of pages. We propose a method for integrating multiple Internet directories by instance-based learning. Our method provides the mapping of categories in order to transfer documents from one directory to another, instead of simply merging two directories into one. We present herein an effective algorithm for determining similar categories between two directories via a statistical method called the κ-statistic. In order to evaluate the proposed method, we conducted experiments using two actual Internet directories, Yahoo! and Google. The results show that the proposed method achieves extensive improvements relative to both the Naive Bayes and Enhanced Naive Bayes approaches, without any text analysis on documents.

1 Introduction

The World-Wide Web (WWW) is now used not only by computer specialists, but by all sorts of people, from children to businessmen and businesswomen, which has resulted in an enormous quantity of web pages available on the Internet. This makes finding pages containing the desired information rather difficult. Search engines are requisite for finding the desired information. In general, current search engines perform their functions via one of two methods: a keyword search and a directory search. A keyword search engine performs searches by means of user-specified keywords. Keyword-based search engines like Google can reliably find pages containing the specified keywords. However, if the user does not have a thorough knowledge of his/her search domain and cannot choose the appropriate keywords, the search engine is useless. In such a situation, directory-based search engines do a good job. In a directory-based search engine, pages are evaluated and organized by humans before being registered in the search engine's archive. Directory-based search engines provide knowledge of WWW navigation. Users reach the desired information by going up and down the directories.

Although pages are carefully selected and well-organized, a single Internet directory is not sufficient because the search engine tends to have some bias in both collecting and categorizing pages. In order to solve these problems, we herein propose a method that coordinates multiple Internet directories by estimating directory similarities. The proposed method does not simply merge multiple directories into a larger directory, but instead determines the relationship between directories. Many such public Internet directories exist. Some are designed to cover wide domains, while others focus on special domains. Such Internet directories are difficult to merge. Moreover, these directories should not be integrated, because the existing differences in concept hierarchies among the directories is important when selecting and using directory-based search engines. In this paper, we propose a method to solve this problem by determining the rules for mapping categories in a directory to those in another directory. Our solution can be applicable not only to the Internet directory problem, but also to the integration of web marketplace catalogs [Agrawal and Srikant, 2001][1] and ontology integration in general.

The remainder of this paper is organized as follows. In Section 2, we define the problem of coordinating multiple Internet directories and discuss related studies. In Section 3, we propose a new machine learning method for the above-mentioned problem. Next, in Section 4, we compare the performance of the proposed method to that of the Enhanced Naive Bayes approach [Agrawal and Srikant, 2001] for an integration problem using real Internet directories. Finally, in Section 5, we present our conclusions.

2 Integration of Multiple Internet Directories

In order to state the problem, we introduce a model for the Internet directories we intend to integrate. We assume there are two Internet directories: a *source* directory and a *target* directory. The documents in the source Internet directory are

[1] For example, the integration of a distributor's catalog and a web marketplace.

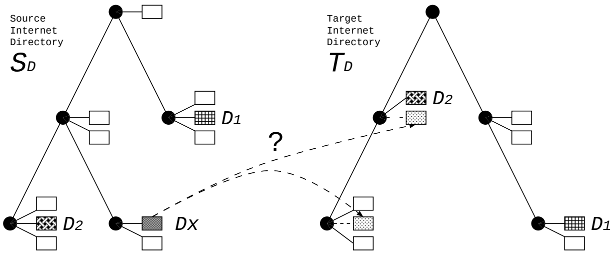

Figure 1: Problem statement

expected to be assigned to categories in the target Internet directory. This produces a *virtually* integrated Internet directory in which the documents in the source directory are expected to be members of both the source and target directories. This integrated directory inherits the categorization hierarchy from the target Internet directory.

The Internet directory model for the source and target is as follows:

- The source Internet directory, S_D contains a set of categories, $C_{s1}, C_{s2}, \ldots, C_{sn}$, that are organized into an "is-a" hierarchy. Each category can also contain documents.
- The target Internet directory T_D contains a set of categories, $C_{t1}, C_{t2}, \ldots, C_{tm}$ that are organized into an "is-a" hierarchy. Each category can also contain documents.

The proposed model permits documents to be assigned to intermediate categories. This model is similar to the catalog model in [Agrawal and Srikant, 2001], except that the categories are organized into a conceptual hierarchy. Since the catalog model ignores hierarchy structure and therefore cannot assign documents to an intermediate category, our Internet directory model is more general than the catalog model.

The problem addressed in this paper is finding an appropriate category C_t in the target directory T_D for each document D_{si} in the source directory S_D. An example is shown as the mapping of a black box D_X in Figure 1, where the black circles indicate categories and the hollow boxes indicate documents. What we need to do is determine an appropriate category in T_D for a document which appears in S_D but *not* in T_D, because mapping is not necessary if the document is included in both the source and the target directories. Documents D_1 and D_2 in Figure 1 are examples of such documents. This mapping can have several possibilities, e.g., D_X can be mapped to an upper left category or to a lower left category, etc.

3 Related Work

One popular approach to this kind of problem is to apply standard machine learning methods [Langley, 1996]. This requires a flattened class space having one class for every leaf node. The problem can then be considered to be a normal classification problem for documents. Naive Bayes(NB) [Mitchell, 1997] is an established method used for this type of document classification framework. A classifier is constructed using the words in the documents. However, this classification scheme ignores the hierarchical structure of classes and, moreover, cannot use the categorization information in the source directory. Enhanced Naive Bayes [Agrawal and Srikant, 2001], hereafter referred to as E-NB, is a method which does use this information. E-NB will be discussed in the next section. GLUE [Doan *et al.*, 2002] is another type of system employing NB. To improve accuracy, GLUE combines NB and a constraint optimization technique called relaxation labeling. However, the general performance of the system depends on that of NB. Unlike NB, the systems in [Koller and Sahami, 1997], [Wang *et al.*, 1999] classify documents into hierarchical categories, and these systems use the words in the documents for classification rules. However, these systems cannot use the categorization information in the source directory.

Another type of approach is ontology merging/alignment systems. These systems combine two ontologies, which are represented in a hierarchal categorization. Chimaera [McGuinness *et al.*, 2000] and PROMPT [Noy and Musen, 2000] are examples of such systems and assist in the combination of different ontologies. However, such systems require human interaction for merging or alignment. In addition to this requirement, these systems are based on the similarity between words, which introduces instability. The dictionaries used for such systems often have word similarity bias. FCA-MERGE [Stumme and Madche, 2001] is another type of ontology merging method. It uses the attributes of concepts to merge different ontologies. As a result, it creates a new concept without regarding the original concepts in both ontologies. Calvanese et al. [Calvanese *et al.*, 2001] also discussed an ontology integration framework with respect to global ontology and local ontology.

As mentioned before, we also tackled the similar problem of catalog integration. An approach, besides E-NB, is to construct the abstract-level structure of two hierarchies [Omelayenko and Fensel, 2001]. This approach does not direct the transformation of source and target information, but transforms via the abstract-level structure. It is relatively easy to transfer information through many hierarchical structures, but it is hard to create a common structure for those hierarchies.

The bookmark-sharing systems of Siteseer [Rucker and Polanco, 1997] and Blink [Blink, 2000] also deal with a similar problem. These systems attempt to share URL information which appears in the source bookmark but not in the target bookmark. These systems flatten the categorization of bookmarks in the same manner as NB and determine the mapping of categories in each bookmark, based on the shared URL information. Such systems are problematic when a given URL does not fit into an exact category. kMedia [Takeda *et al.*, 2000] is another bookmark-sharing system that uses hierarchical structures explicitly, but is dependent on the similarity of the words within pages.

In the context of information retrieval, Stuckenschmidt [Stuckenschmidt, 2002] has coordinated the formal model of hierarchies with multiple classification hierarchies. In our paper, since the upper and lower boundaries of concept mapping can be determined by subsumption relationships, we can infer the boundary of valid mapping and also check whether or not the given mapping is consistent. In our problem, since we have only partial evidence of classes and do not know the definitions of classes, we can not apply this method directly.

4 Learning the Relationship between Categories of Two Internet Directories

In this section, we explain our method to determine the relationship between categories in two Internet directories. One characteristic of the proposed method is to use the hierarchical structure "as is". We use all categories including intermediate categories and leaf categories, because information for categorization can also be obtained from these categories. Another characteristic of the proposed method is the exclusive reliance on the categorization structure of both directories, i.e., with no reliance on the semantic information of the documents. We can determine the relationship between categories of the two directories by statistically comparing the membership of the documents to the categories.

4.1 Basic Concept

Although E-NB was developed for a very similar problem, it is missing an important feature: categorization hierarchy. According to [Agrawal and Srikant, 2001], the initial definition of the problem is identical to that of this paper. However, the previous paper assumes that "any documents assigned to an interior node really belong to a conceptual leaf node that is a child of that node," and concludes from this assumption that "we can flatten the hierarchy to a single level and treat it as a set of categories." However, the conclusion overlooks the categorization hierarchies. The categorizations are not independent of each other. The categorization hierarchies are usually structured using an "is-a" or another relationship. If a document is categorized in a lower category in the categorization hierarchy, then the document should also be categorized in the upper categories [2]. If we flatten the categories, such information regarding the relationships between categories will be lost.

Another problem associated with NB is its sensitivity to the words in documents. For example, if a document contains the word *bank*, the category for the document could be a financial category; however, the category could also be a construction category. The quality of categorization with NB is thus not stable according to the contents of documents, due to the natural language techniques of processing words. It is critical to integrate Internet directories because mixing reliable Internet directories maintained by human editors with less reliable and less stable machine-generated classifiers will confuse users and decrease their confidence for them. In addition to the problem of sensitivity, analysis of a document to extract words is expensive.

Our method focuses on the similarity of the way of categorization, not the similarity of documents. Then, how do we measure the similarity of categorization? We utilize shared documents in both the source and target Internet directories as our measurement standard. If many documents in category C_{si} also appear in category C_{tj} at the same time, we consider these two categories to be similar, because the ways of categorization in C_{si} and C_{tj} are supposed to be similar, i.e., if another document D comes in C_{si}, it is likely that D will be also included in C_{tj}. This method avoids both the keyword extraction process and the handling of word meaning. The example shown in Figure 2 illustrates that documents in the bottom category in the source Internet directory can be transferred to a similar category in the target Internet directory.

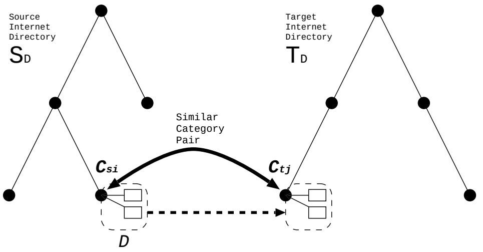

Figure 2: Document transfer from source Internet directory to target Internet directory

		Category C_{tj}	
		belong	not belong
Category C_{si}	belong	N_{11}	N_{12}
	not belong	N_{21}	N_{22}

Table 1: Classification of documents by two categories

4.2 κ-Statistic

The remaining problem is how to determine the pairs of similar categories. We adopt a statistical method to determine the degree of similarity between two categorization criteria. The κ-statistic method [Fleiss, 1973] is an established method for evaluating the similarity between two criteria. Suppose there are two categorization criteria, C_{si} in S_D and C_{tj} in T_D. We can determine whether or not a particular document belongs to a particular category[3]. Consequently, documents are divided into four classes, as shown in Table 1. The symbols N_{11}, N_{12}, N_{21} and N_{22} denote the numbers of documents for these classes. For example, N_{11} denotes the number of documents which belong to both C_{si} and C_{tj}. We may logically assume that if categories C_{si} and C_{tj} have the same criterion of categorization, then N_{12} and N_{21} are nearly zero, and if the two categories have a different criterion of categorization, then N_{11} and N_{22} are nearly zero. The κ-statistic method uses this principle to determine the similarity of categorization criteria.

In the κ-statistic method, we calculate the probability P, which denotes the percentage of coincidence of the conceptual criteria, and the probability P', which denotes the percentage of coincidence of the conceptual criteria by chance.

$$P = \frac{N_{11} + N_{22}}{N_{11} + N_{12} + N_{21} + N_{22}}$$

[2] Note that the reverse is not always true, because documents can be categorized into intermediate categories.

[3] Remember that categorization in a directory is determined using the nodal structure of the categorization hierarchy.

$$P' = \frac{(N_{11}+N_{12})(N_{11}+N_{21}) + (N_{21}+N_{22})(N_{12}+N_{22})}{(N_{11}+N_{12}+N_{21}+N_{22})^2}$$

As such, the value of the κ-statistic is represented as the following equation:

$$\kappa = \frac{P - P'}{1 - P'}$$

Next, we examine whether we can assume the $\kappa = 0$, which indicates that the percentage of coincidence for the two conceptual criteria is zero. We therefore calculate the test statistic Z according to the following equation.

$$Z = \kappa \sqrt{\frac{(N_{11}+N_{12}+N_{21}+N_{22})(1-P')}{P'}} \qquad (1)$$

The value Z follows a normal distribution. A null hypothesis is considered to occur when the percentage of coincidence of the concept criteria is zero. When we assume a significance level of 5% and the following equation is satisfied, we can dismiss the null hypothesis.

$$Z \geq 1.64486$$

When the null hypothesis can be dismissed, the criteria are determined to be the same.

4.3 Determination of Pairs of Similar Categories

The relationship between the two categorization criteria is examined from top to bottom. This algorithm is shown in Figure 3. First, the most general categories in the two categorization hierarchies are compared using the κ-statistic. If the comparison confirms that the two categories are similar, then the algorithm outputs this pair of categories. At the same time, the algorithm generates all possible pairs of their children categories. It generates two lists of categories each of which is a list of the confirmed category and its children categories. Then, the algorithm picks one category from each list and makes a pair, except for the original pair. This new pair is then evaluated recursively using the κ-statistic method. When a similar pair is not generated, the algorithm outputs the pairs of similar concepts between the two categorization hierarchies. They are used as mapping rules from the source directory to the target directory[4].

This top-down comparison approach attempts to reduce the exploration space, using the structure of both directories as a guide. Assuming that similar categories include similar subcategories, we can skip the combination of categories which is likely unnecessary[5]. This is another benefit of using the “as is” hierarchy, in contrast to the flattening approach.

```
main_algorithm
Input:    $C_{s1}$,   // Top category in $S_D$
          $C_{t1}$,   // Top category in $T_D$
Output:   $R$;        // A set of similar category pairs
begin
    $t := 1$;
    $R := \phi$;
    $X_1 := \{[C_{s1}, C_{t1}]\}$;
    while $X_t \neq \phi$
        $X_{t+1} := \phi$;
        while $X_t \neq \phi$
            $[C_1, C_2] := any\ element\ in\ X_t$;
            if $C_1\ and\ C_2$ is similar
                $X_{t+1} := X_{t+1} + make\_comb(C_1, C_2)$;
                $R := R + [C_1, C_2]$;
            fi;
            $X_t := X_t - [C_1, C_2]$;
        end;
        $t := t + 1$;
    end;
    return $R$;
end;
-----------------------------------------
make_comb
Input:    $C_s$,   // Category in $S_D$
          $C_t$,   // Category in $T_D$
Output:   $S$;     // A set of category pairs in $S_D$ and $T_D$
begin
  $S := \{[C_{si}, C_{tj}] \mid C_{si}\ is\ either\ C_s\ or\ a\ child\ of\ C_s,$
          $C_{tj}\ is\ either\ C_t\ or\ a\ child\ of\ C_t\} - [C_s, C_t]$;
  return $S$;
end;
```

Figure 3: Algorithm to determine similar category pair

4.4 Application Policy of Rules

Since the proposed method uses categorization similarity, if a category in the source Internet directory does not have a similar category in the target directory, then those documents cannot be categorized in the target Internet directory. In order to avoid this problem, we once again use the categorization hierarchy. If the system cannot find a similar category pair for the category containing the document, it applies the rule generated for the parent category instead.

4.5 Computational Complexity

In this section, we discuss the computational complexity of the proposed algorithm shown in Figure 3. Suppose that two concept hierarchies S_D and T_D are trees of width b (the number of subcategories for each category) and height d (the number of categories from top to bottom) for the sake of simplicity. In this case, we calculate the total number of categories N using the following equation:

$$N = b^0 + b^1 + \ldots + b^d$$

Comparison of the similarity among all categories requires $N \times N$ comparisons. As a result, the order of computational complexity is $O(b^{2d})$.

[4] The rules do not guarantee the consistency of the mappings, because category hierarchies have the possibility of inconsistency with regard to similar relationships.

[5] This assumption may sometimes be too strong, e.g., a child category of a category is not so relevant, whereas its child category (a grandchild category of the original category) is relevant. In such cases, we should relax this constraint when exploring future candidates.

Next, we consider the cost of our algorithm. Assuming that after the similarity tests using κ-statistic have been conducted, $n\%$ of the tests are confirmed to be similar to other categories. The number of pairs produced by $make\_comb$ is $(b+1)^2 - 1$ because it makes pairs from the two lists, each pair of which consists of the category itself and its child categories, except for the original pair. The number of expected combination pairs after processing of the κ-statistic is $n(b^2+2b) + (1-n) \cdot 0$. The second half of the formula $(1-n) \cdot 0$ is applied because no similar pair is produced when the κ-statistic does not confirm the similarity. We finally obtain the total number of nodes M to be compared as the product of the number of examined nodes and the expected number of nodes. M is given by the following equation:

$$\begin{aligned} M &= b^0 + n^1(b^2+2b)^1 \\ &\quad + n^2(b^2+2b)^2 + \ldots + n^d(b^2+2b)^d \end{aligned}$$

The order of complexity is then obtained as $O(n^d b^{2d}) = O((nb^2)^d)$. Since n denotes the percentage of category pairs confirmed to be similar, $0 \leq n \leq 1$ is satisfied. As a result, we obtain less computational complexity using the algorithm in Figure 3. In other words, although the depth remains the same, the search algorithm can reduce computation by constraining the breadth size.

5 Experiments Using Internet Directories

5.1 Experimental Settings

In order to evaluate the proposed algorithm, we conducted experiments using data collected from the Yahoo! [Yahoo!, 2001] and Google [Google, 2001][6] Internet directories. The data was collected in the fall of 2001. In order to compare the proposed method to that in [Agrawal and Srikant, 2001], we selected the same locations in Yahoo! and Google for the experimental data. The locations are as follows:

- Yahoo! : Recreation / Automotive
  Google : Recreation / Autos
- Yahoo! : Entertainment / Movies_and_Film
  Google : Arts / Movies
- Yahoo! : Recreation / Outdoors
  Google : Recreation / Outdoors
- Yahoo! : Arts / Visual_Arts / Photography
  Google : Arts /Photography
- Yahoo! : Computers_and_Internet / Software
  Google : Computers / Software

Table 2 shows the numbers of categories, the links in each Internet directory and the links included in both Internet directories. Links are considered to be the same when the URLs in both directories are identical.

We conducted ten-fold cross validations for the shared links. The shared links were divided into ten data sets; nine of these sets were used to construct rules, and the remaining set was used for testing. Ten experiments were conducted for each data set, and the average accuracy is shown in the results. In order to compare our proposed method to the E-NB approach, the classifiers are assumed to correctly assign documents when the document is categorized either in the same category as the test data or in the parent categories of the test data. E-NB constructs classifiers for only the first-level categories in the experimental domain, whereas the proposed method uses all of the categories from top to bottom. The significance level for the κ-statistic was set at 5%.

	Yahoo!		Google		shared
	categories	links	categories	links	links
Autos	782	5200	583	7909	1002
Movies	4001	16947	4623	21244	2735
Outdoors	2247	13932	825	13725	716
Photography	375	4173	170	3999	536
Software	997	4686	2142	35628	952

Table 2: Statistics on the experimental data

	Accuracy		
Dataset	E-NB	SBI	Improvement
Autos	76.2	88.0	11.8
Movies	42.6	77.9	35.3
Outdoors	77.8	74.0	-3.8
Photography	72.8	79.7	6.9
Software	62.4	73.4	11.0
Average	64.4	78.4	14.0

Table 3: Results of Yahoo! as the source Internet directory and Google as the target Internet directory

5.2 Experimental Results

The experimental results are shown in Tables 3 and 4. Table 3 shows the results obtained using Yahoo! as the source Internet directory and Google as the target Internet directory, and Table 4 shows the results obtained using Google as the source Internet directory and Yahoo! as the target Internet directory. For comparison, these tables also include the results of [Agrawal and Srikant, 2001]. E-NB denotes the method of [Agrawal and Srikant, 2001] and SBI denotes the similarity-based integration method that is proposed in this paper. The data obtained for the proposed method and that presented in [Agrawal and Srikant, 2001] are not truly comparable because the collection date is different[7].

The proposed algorithm did well compared to E-NB, performing more than 10% better in accuracy on the averages. In the Movies domain, the proposed algorithm performs much better in accuracy than E-NB. One reason for this is that the pages related to movies contain various words that indicate numerous movie settings, making classification using word-based systems difficult. On the other hand, in the Outdoors domain, performance is similar for both methods because the Outdoors domain treats pages using special outdoors-related

[6] Since the data in Google is constructed by the data in dmoz [dmoz, 2001], we collected data through dmoz.

[7] In addition, differences may have occurred due to data selection. For example, Yahoo! has links that use @(the at mark). The treatment of such links was not discussed in [Agrawal and Srikant, 2001]. In the present study, we did not use such links.

Dataset	Accuracy		Improvement
	E-NB	SBI	
Autos	73.1	83.6	10.5
Movies	46.2	68.1	21.9
Outdoors	65.4	68.9	3.5
Photography	51.3	60.4	9.1
Software	58.6	74.0	15.4
Average	58.9	71.0	12.1

Table 4: Results of Google as the source Internet directory and Yahoo! as the target Internet directory

words, and no classification method for the Outdoors domain has been established for human readers.

As mentioned earlier, in the experiments described in [Agrawal and Srikant, 2001], the classifiers induced by the system have the ability to classify the documents into only the first-level categories of the test domain in the target Internet directory. The proposed classifiers can classify the documents into sub-categories as well. For example, we used 4,623 categories for document assignment in the Movies domain of Yahoo! as the source Internet directory and Google as the target Internet directory, whereas their system used only 40 categories. Therefore, the classifiers obtained by the proposed method are more reliable and useful than those obtained by other methods.

Next, we compare our method with GLUE [Doan *et al.*, 2002]. The basic steps of GLUE are as follows. First, the user trains learners for each concept for target taxonomies. Learners can be a combination of different strategies like name- or contents-based learning. Then, they determine that the source concept and target concept are similar, if the application of learners for the source concept yields good enough. Finally, the system applies the relaxation labeling method for similar concept pairs. The method does not use the hierarchical structures in learning very well, and relies mainly on the performance of the learning method, namely NB. On the other hand, our method does use hierarchical structures in learning very well. So, we do not need a semantical analysis in learning. They report an accuracy of 65-95% which is equivalent to our results, but they use only a few categories, i.e., 40-363, while we use 170-4623 categories. We can say from both theoretical and practical viewpoints that our method is more appropriate in learning for large hierarchical structures.

6 Conclusions

In this paper, a statistical-based technique was proposed for integrating multiple Internet directories by determining the relationship between these directories. The proposed method uses the κ-statistics to find similar category pairs, and transfers the document categorization from a category in the source Internet directory to a similar category in the target Internet directory. The proposed method has an advantage in document treatment in that it relies on the category structure only, and not on words or word similarity in a document. The performance of the proposed method was tested using actual Internet directories, and the results of these tests show that the performance of the proposed method was more than 10% better in accuracy than that of previous methods, although the number of categories were by far larger for our proposed method. In addition, our problem modeling is more general than the other models in terms of assignment documents on intermediate categories. From this, we can conclude our approach shows great promise in comparison to systems employing NB, such as E-NB [Agrawal and Srikant, 2001], GLUE [Doan *et al.*, 2002], and so on.

Although the present results are encouraging, much has yet to be done. The limitation of the proposed method is that this method is based on the existence of shared links. In other words, the proposed method is less reliable if there are fewer shared links. To improve our proposed method, various methods might be used to obtain semantic information in order to increase the number of shared links, e.g., to regard similar pages as the same links. Another option for this problem might be to combine semantic information or other information about the documents. If we could a establish good combination of our proposed method and such information, it is possible that we could greatly improve our method. Moreover, the present method can find only one-to-one mapping rules. If the system is able to find one-to-many or many-to-many mappings, it would work more correctly in categorization. For the development of these mappings, we might invent probabilistic mapping representations and also introduce a subsumption technique [Stuckenschmidt, 2002]. Finally, the proposed method should be expanded so that it can apply to more than three concept hierarchies. In such a case, despite the conflict between several concept hierarchies, we would expect more information to be obtained. The aforementioned tasks for improvement will be investigated in future studies.

References

[Agrawal and Srikant, 2001] Rakesh Agrawal and Ramakrishnan Srikant. On integrating catalogs. In *Proceedings of the Tenth International World Wide Web Conference (WWW-10)*, pages 603–612, 2001.

[Blink, 2000] Blink. http://www.blink.com/, 2000.

[Calvanese *et al.*, 2001] Diego Calvanese, Giuseppe De Giacomo, and Maurizio Lenzerini. A framework for ontology integration. In *Proceedings of the First Semantic Web Working Symposium*, pages 303–316, 2001.

[dmoz, 2001] dmoz. http://dmoz.org/, 2001.

[Doan *et al.*, 2002] AnHai Doan, Jayant Madhavan, Pedro Domingos, and Alon Halevy. Learning to map between ontologies on the semantic web. In *Proceedings of the 11th International World Wide Web Conference*, 2002.

[Fleiss, 1973] J. L. Fleiss. *Statistical Methods for Rates and Proportions*. John Wiley & Sons, 1973.

[Google, 2001] Google. http://directory.google.com/, 2001.

[Koller and Sahami, 1997] Daphne Koller and Mehran Sahami. Hierarchically classifying documents using very few words. In Douglas H. Fisher, editor, *Proceedings of the 14th International Conference on Machine Learning*, pages 170–178, Nashville, US, 1997. Morgan Kaufmann Publishers, San Francisco, US.

[Langley, 1996] Pat Langley. *Elements of Machine Learning*. Morgan Kaufmann, 1996.

[McGuinness *et al.*, 2000] Deborah L. McGuinness, Richard Fikes, James Rice, and Steve Wilder. An environment for merging and testing large ontologies. In Anthony G. Cohn, Fausto Giunchiglia, and Bart Selman, editors, *Proceedings of the Conference on Principiles of Knowledge Representation and Reasoning (KR-00)*, pages 483–493, S.F., April 11–15 2000. Morgan Kaufman Publishers.

[Mitchell, 1997] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[Noy and Musen, 2000] Natalya Fridman Noy and Mark A. Musen. Prompt: Algorithm and tool for automated ontology merging and alignment. In *Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-2000)*, pages 450–455, Menlo Park, 2000. AAAI Press.

[Omelayenko and Fensel, 2001] Borys Omelayenko and Dieter Fensel. An analysis of b2b catalogue integration problems. In *Proceedings of the International Conference on Enterprise Information Systems*, pages 945–952, 2001.

[Rucker and Polanco, 1997] James Rucker and Marcos J. Polanco. Siteseer: Personalized navigation for the web. *Communications of the ACM*, 40(3):73–75, 1997.

[Stuckenschmidt, 2002] Heiner Stuckenschmidt. Approximate information filtering with multiple classication hierarchies. *International Journal of Computational Intelligence and Applications*, 2(3):295–302, 2002.

[Stumme and Madche, 2001] G. Stumme and A. Madche. Fca-merge: Bottom-up merging of ontologies. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence IJCAI-01*, 2001.

[Takeda *et al.*, 2000] Hideaki Takeda, Takeshi Matsuzuka, and Yuichiro Taniguchi. Discovery of shared topics networks among people. In R. Mizoguchi and J. Slaney, editors, *Proceedings of the 7th Pacific Rim International Conference on Artificial Intelligence (PRICAI-2000)*, volume 1886 of *LNAI*, pages 668–678, Berlin, 2000. Springer.

[Wang *et al.*, 1999] Ke Wang, Senqiang Zhou, and Shiang Chen Liew. Building hierarchical classifiers using class proximity. In Malcolm Atkinson, Maria E. Orlowska, Patrick Valduriez, Stan Zdonik, and Michael Brodie, editors, *Proceedings of the 25th international Conference on Very Large Data Bases*, pages 363–374, Los Altos, CA 94022, USA, 1999. Morgan Kaufmann Publishers.

[Yahoo!, 2001] Yahoo! http://www.yahoo.com/, 2001.