韓國 컴퓨터情報學會 論文誌
第11卷 第5號, 2006. 11.

2006-11-5-1-13

상위어 관계를 이용한 개념 계층의 생성

신 명 근*

Concept Hierarchy Creation Using Hypernym Relationship

Shin, Myung Keun *

요 약

개념 계층은 지식을 그룹화하여 다단계로 표현하며, 이는 자료의 분류, 저장 및 검색을 효율적으로 지원해 준다. 일반적으로 도메인 전문가의 수작업을 통해 개념 계층이 생성되었으며, 이는 생성과 유지에 많은 비용이 소요되면서도 일관성 유지가 어려운 단점이 있다. 본 논문은 미리 정의된 상위어 관계를 이용하여 문자형 자료의 개념 계층을 자동으로 생성하는 방법에 대한 연구이다. 개념 계층의 자동 생성을 위해서는, 다중 의미로 사용되는 단어에서 적절한 의미를 찾아 내어 모호성을 제거해야 하며, 외부 정보를 이용하여 모호성이 제거된 단어를 그룹화하고 계층을 생성하는 작업이 필요하다. 우리는 워드넷 (WordNet)의 다중 의미에 대한 설명 및 상위어 관계를 이용하여 모호성을 제거하고 개념 계층을 생성하는 방법을 제안 한다.

Abstract

A concept hierarchy represents the knowledge with multi-level form, which is very useful to categorize, store and retrieve the data. Traditionally, a concept hierarchy has been built manually by domain experts. However, the manual construction of a concept hierarchy has caused many problems such as enormous development and maintenance costs and human errors such as inconsistency. This paper proposes the automatic creation of concept hierarchies using the predefined hypernym relation. To create the hierarchy automatically, we first eliminate the ambiguity of the senses of data values, and construct the hierarchy by grouping and leveling of the remaining senses. We use the WordNet explanations for multi-meaning word to eliminate the ambiguity and use the WordNet hypernym relations to create multi-level hierarchy structure.

▸ Keyword : 계층 생성(Hierarchy Creation), 상위어(Hypernym), 지식표현(Knowledge Representation).

• 제1저자 : 신명근
• 접수일 : 2006.10.16, 심사일 : 2006.10.22, 심사완료일 : 2006.11.18
* (주)투이

Ⅰ. Introduction

A concept is the symbolic representation of data or objects that occur in the business activities. By virtue of wide spread database systems, concepts can be more effectively manageable through various database-driven applications. However, in current database systems, managing concepts is restricted to only raw data values and management itself is also limited to storing, querying, and joining. If a more intelligent interface can be developed to the current database systems, and thus more understandable knowledge together with the raw data values can be extracted from the database systems, the current application systems become more useful and valuable to the users [1, 2, 3, 4, 5]. In this regard, a concept hierarchy becomes a good candidate interface, represented by tree-structured frameworks in which lower-level concepts represent specific data values while higher-level concepts represent their generalizations, leading to the semantic relationships between raw data values of the database and higher-level concepts as a whole [1, 2, 5]. Traditionally, a concept hierarchy has been built manually by knowledge engineers or domain experts. However, the manual construction of a concept hierarchy has caused many problems such as enormous development and maintenance costs incurred by the involvement of large number database engineers and human errors such as inconsistency and subjectivity in the concept classifications [6,7].

For text data from the relational databases, there exist at least two major difficulties in the automatic generation of a concept hierarchy: disambiguation of concepts and concept hierarchy generation. As for disambiguation of concepts, unlike a numerical value, a word, used as the nominal data in a relational database, has several senses (i.e., meanings) according to the context, which causes a word-sense ambiguity problem. For example, a table has two senses: a piece of furniture and a tabular array. Also, concept hierarchy generation requires some external information sources because a database does not contain abstraction and generation concepts as well as the relations among concepts. Therefore, without such external sources, the automatic generation of a concept hierarchy is almost impossible.

There have been studies focusing on the automatic generation of a concept hierarchy for the numerical data from the database [1, 8, 9]. However, relatively little is known about the automatic generation of a concept hierarchy for the nominal data from the database because of the aforementioned reasons. There are three kinds of approaches on the automatic generation of a concept hierarchy for nominal data including conceptual abstraction hierarchy, attribute-oriented induction (AOI), and neural network (NN).

The conceptual abstraction hierarchy approach [1, 2, 5, 9] represents attribute values of a database, including attribute values, tuples, and relations, in a tree form, and it uses the tree to retrieve approximate answers when an ordinary query does not produce exact answers from the underlying database. Huh and Lee [5] suggest the construction of a knowledge abstraction hierarchy (KAH) on a relational data model, and explore cooperative query answering using the query generalization and specialization process in the KAH. However, there have been no attempts at the automatic generation of the KAH.

The AOI approach [3, 4]is a data reduction method which can extract hidden patterns from a relational database using concept hierarchies, and represents them as generalized concise rules or patterns. The limitation of the AOI approaches lie in their focus only on the numeric value of attributes; they rarely mention the automatic generation of the concept hierarchy for nominal data, let alone the word-sense ambiguity problem.

The neural network approach, particularly the

self-organizing map (SOM) [10, 11, 12], is used for determining the similarities between documents, and presents its relation as a two-dimensional topography that is analogous to a map. But the focus of the SOM approach is to analyze the documents and visualize them as the map form. Therefore, the SOM approach can not be directly applied in generating a concept hierarchy of a database in which only individual data words exist.

WordNet [13, 14] is an on-line English lexical reference system that has been developed and maintained by the Cognitive Science Laboratory at Princeton University in the 1990s. As an ordinary online dictionary, WordNet lists alphabetically concepts or names important to a particular subject along with discussion of their meanings. Additionally, as a machine-readable thesaurus with a semantic network, it has a rich array of structures showing semantic relations among words. Specifically, WordNet heavily utilizes an is-a relation called hypernym. For example, furniture is the hypernym of table, which becomes a superordinate and subordinate relation. This feature can mitigate the effort to draw higher level concepts in a hierarchy, and it presents a potential basis to build a concept hierarchy.

This paper focuses on automatic hierarchy construction for nominal data in an underlying database using Wordnet. The existing approaches for the automatic generation of a concept hierarchy for nominal data just shows the clustering results as a tree form and does not contain the semantic relation among groups. Unlike the existing approaches, we extend the automatic generation of concept hierarchies to nominal data using two prominent features of WordNet, definitions and hypernyms, by disambiguating senses of a word and combining a one-line hierarchy of correct senses. We specifically propose this framework to solve the two problems that occur in automatic generation of a concept hierarchy, disambiguation of concepts and concept hierarchy generation with WordNet. This study has important implications for the database designers who need to build and maintain concept hierarchies because it allows the database designers to build and maintain hierarchies at a low cost in the situation of continuous inflows of information, and to minimize human intervention for adjustment.

II. The formalism of the concept hierarchy and WordNet

This section first provides the formal definition and properties of a concept hierarchy. Second it delineates the major features of WordNet for building a hierarchy.

2.1 The formal definition and properties of the concept hierarchy

A concept hierarchy, H, can be defined by a set of concepts (i.e., words or data values in records), S, constituting a concept hierarchy and a precedence relation R on the set S. The precedence relation between any two concepts implies that the two can be compared in terms of generalization or specialization and thus be classified into an abstract value and specific value. On the other hand, not all pairs of concepts can be ordered as abstract and specific values; only partial sets of paired concepts can be ordered. In such capacity, a natural scheme to define H with S and Ruses a partially ordered set.

An alternative scheme uses a quasi ordered set that removes the equality relation as a special partially ordered set. More precisely, a concept hierarchy H with S and R can be defined by a quasi ordered set that satisfies the following three properties: (1) we have $a \not\prec a$ for any $a \in A$ (*ir-reflexive*), (2) if $a \prec b$, and $b \prec c$, then $a \prec c$ and (*transitive*), and (3) if $a \prec b$ and $a \prec c$, then $b \prec c \vee c \prec b$ (*disjoint*).

Here we notice two features of a sound hierarchy: non-circularity and comparability. First, closed loop relations are to be avoided when making a hierarchy. Second, every concept in the set has to be comparable to a certain different concept so that all the concepts are to be partially ordered. More precisely, if two distinct concepts in a partially ordered set, A and B, can be compared from the viewpoint of generalization as either $A \prec B$ or $B \prec A$, they are said to be comparable. That is to say that if two are comparable, one is to precede the other, and thus, one is to be an abstract value of the other. However, in the presence of multi-level abstraction, comparison of concepts by a human expert may not guarantee identical comparison.

The concept hierarchy is advantageous since it is simple to understand, machine readable, self-constructive due to its set-theory base, and thus is automatically constructible.

2.2 The features of WordNet

WordNet was developed to provide an on-line dictionaries not merely in an alphabetical order but in a more conceptual way showing semantic relationships among words and concepts such as similar meanings, subsumption relations, part-of relations. WordNet combines features of both a traditional dictionary and a thesaurus with a semantic network concept.

Hazard noun, verb

* *noun* (to sb/sth) a thing that can be dangerous or cause damage: a fire/safety hazard

* *verb* 1. to make a suggestion or guess which you know may be wrong 2. to risk sth or put it in danger

(a) dictionary

Hazard n 1. **syn see** chance 2 2. **syn see** danger

(b) thesaurus

Overview of noun **hazard**
The noun hazard has 3 senses (first 2 from tagged texts)

1. (3) **hazard**, jeopardy, peril, risk -- (a source of danger; a possibility of incurring loss or misfortune; "drinking alcohol is a health hazard")
2. (1) luck, fortune, chance, **hazard** -- (an unknown and unpredictable phenomenon that causes an event to result one way rather than another; "bad luck caused his downfall"; "we ran into each other by pure chance")
3. **hazard** -- (an obstacle on a golf course)

(c) WordNet

Fig. 1. Comparison of the definition of hazard.
그림 1. hazard의 정의 비교

Fig. 1 shows comparison of the definition of hazard definition in a dictionary, thesaurus, and WordNet. A word in WordNet is separated into senses that have the synonym sets with similar meanings and explanations. In this respect, WordNet is more similar to a thesaurus than a dictionary.

However, unlike a thesaurus, WordNet provides additional semantic network structure with four prominent word relations: Hyponym/Hypernym and Meronym/ Holynym. Specifically, the Hyponym/ Hypernym relation is also called subordination/ superordination, subset/superset, or is-a relation. For example, table is the hyponym of furniture, while furniture is the hypernym of table, which becomes a totally ordered relation.

Fig. 2 provides a partial output of the synonyms and hypernyms of the noun, Java in WordNet, which shows three different senses, island, coffee, and programming language. In each sense, Java has the definition (i.e., word form) and explanation for the corresponding synonym set, and its hypernym.

Sense 1
Java -- (an island in Indonesia south of Borneo; one of the world's most densely populated regions)
=> island -- (a land mass (smaller than a continent) that is surrounded by water)
=> land, dry land, earth, ground, solid ground, terra firma
=> object, physical object
=> entity
Sense 2
coffee, **java** -- a beverage consisting of an infusion of ground coffee beans
=> beverage, drink, drinkable, potable
=> food, nutrient
=> substance, matter
=> entity
=> liquid
=> fluid
=> substance, matter
=> *entity*
Sense 3
Java -- simple platform-independent object-oriented programming language
=> object-oriented programming language, object-oriented programing language
=> programming language, programing language
=> artificial language
=> language, linguistic communication
=> communication
=> social relation
=> relation
=> abstraction

Fig. 2. An example of the java hypernym.
그림 2. java에 대한 상의어 예

A hypernym relationship provided in WordNet presents a potential basis to build a concept hierarchy by making the hypernym a component concept hierarchy, and it can take the part of a larger concept hierarchy. If we collect the hypernyms of all the data values (i.e., concepts) of a subject field and remove the ambiguity among senses, we can construct a concept hierarchy subsuming all the concepts in the subject field by unfolding the individual hypernyms. In the subsequent section, we provide the details of the automatic generation of a concept hierarchy out of the component concept hierarchies in Fig. 2.

III. Automatic generation framework

In this section, we propose a framework that leads to the automatic generation of a concept hierarchy. The framework involves four steps as shown in Fig. 3. As a first step, collecting the full set of the word concepts requires the extraction of words, used for the nominal data in a relational database, from the database attributes. This step identifies all the bottom-level nodes in the hierarchy. Such a task starts with users' selection of a table or a view from the target database, and extraction of distinct data values from the subject attributes.

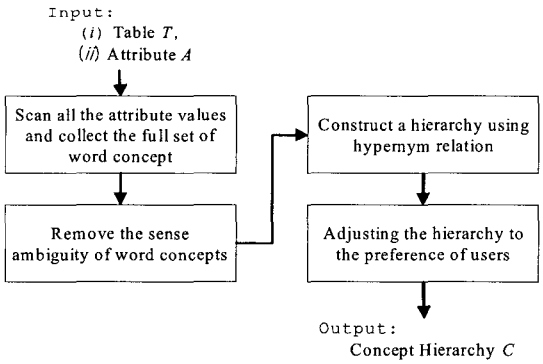

Fig. 3. Skeleton of the automatic creation of a hierarchy.
그림 3. 개념 계층의 자동 생성 개요

The second step is to remove sense ambiguity of word concepts. If a word concept has only one sense in the context, the concept hierarchy can be made simply by overlapping WordNet hypernyms. However, in general, a word has several senses according to the domain context causing the sense ambiguity problem.

For example, consider the computer skill attribute of a personnel database that would have such word data values as JAVA, PASCAL, BASIC, DBMS, etc. In terms of WordNet, these words individually have multiple senses that are not related to the computer at all as shown in Table. 1. We propose a more widely-applicable approach. Details of the approach will be explored further with a vector space model in the subsequent section.

Table 1. A set of senses and their associated domain contexts.
테이블 1. 다중 의미 집합 및 각 의미와 연관된 도메인

Concepts	Domain Contexts	Senses
Java	an island	java_0
	a beverage	java_1
	a programming language	java_2
Pascal	unit of measure	pascal_0
	a mathematician	pascal_1
	a programming language	pascal_2
Basic	a programming language	basic_0
	a necessary commodity	basic_1
DBMS	a software system	dbms_0

In the third step, once the sense ambiguity of a word concept is removed by excluding the unrelated WordNet senses and thus a set of related senses having an identical context are elicited, a concept hierarchy can be constructed from the hypernyms of those individual senses. Assuming only the computer related senses are elicited, the JAVA, PASCAL and BASIC concepts will have identical abstract concepts from the programming language domain while the JAVA, PASCAL, BASIC and DBMS concepts share identical abstract concepts from the communication domain. Thus, if all the hypernyms are collected under a valid set of senses, a concept hierarchy of the attribute can be constructed by overlapping those hypernyms.

Lastly, the resultant hierarchy obtained by overlapping the hypernyms is still premature for use as a complete sound concept hierarchy. The hierarchy could contain non-value adding intermediate nodes like a simple chain type of hierarchy. Or, the result may not be a hierarchy but a collection of separated hierarchies if hypernyms used for overlapping do not contain any common part because they do not share any superordinate concept. This may happen partly as a natural output where no common component exists among senses and partly due to the errors caused by the human inconsistency in the process of creating and maintaining the hypernyms based on human

subjective judgement. In either case, the resultant hierarchy needs to be adjusted by removing non-value adding nodes or by manually connecting superordinate concepts.

IV. Automatic generation of a concept hierarchy

In this section, steps ranging from the second to the fourth in the framework are discussed in more detail.

4.1 Excluding the unrelated senses with a similarity measurement

In the presence of WordNet, because of the sense ambiguity problem as seen in JAVA, PASCAL, and BASIC of Table1, it is necessary to discriminate the meaning or sense of a concept to fit the domain context [13]. For effective discrimination of sense according to domain context, a text categorization method is utilized using a similarity measurement in automatic indexing for texts [15]. Among the text categorization methods including Latent Semantic Indexing, the Baysian Probablistic model, and the Vector Space Model (VSM), a method similar to VSM is specifically adopted since it is the representative methodology in text categorization and can be extended to a more complicated methods like the Baysian model.

A concept usually has multiple senses that have different meanings with one another. Consequently, these senses would share fewer words and such heterogeneity among senses of a concept results in lower similarity values in the VSM analysis. Meanwhile, if similar concepts share more words in their definitions, we can use the number of shared words as a basis to group similar concepts. This is also to say that if a concept has few shared words with a certain group of concepts, this concept can be excluded from the group. When data values are collected from a specific attribute of a database, those values individually form the concepts and a certain set of concepts would make a homogeneous group by sharing similar words since the concepts are from some domain context. For instance, the concepts of programming skill, such as JAVA, PASCAL, BASIC, and DBMS, might have high probability of containing programming related words like language, object, websites, and so on. Namely, they have homogeneous characteristics from the perspective of attributes in a database. Accordingly, if the senses of each concept should have mutually exclusive words for its explanation, and similarity values among all senses are measured based on the cosine coefficient of VSM, these values are zero among senses of a concept for the heterogeneous features. However, they differ from zero value among senses of other concepts according to the degree of similarity. Therefore, first, the shared words used for defining the senses of a concept should be removed and thus each sense of a concept contains mutually exclusive words for the definition.

After excluding shared words used for explaining a concept, all concepts are paired to get the similarity value by applying the VSM model. This similarity value matrix would show the degree of similarity between the senses. We include all senses at least above zero similarity value because incorrect senses of a concept are later adjusted by human control.

The vector-space model is a standard way of representing texts through the words they comprise. If we consider documents as concepts, and terms for documents as words used for explaining a concept, the vector-space model can also be applied to disambiguate senses of a concept. Because words which occur very frequently over concepts are unlikely to discriminate sense words sufficiently, words that are used only for a few concepts should have more weights. Let the word frequency tf_i be defined as the number of occurrence for explaining of a concept. If the concept frequency df_j is defined as the number of concepts in a collection of N concepts

in which the word occurs, the typical inverse concept frequency (*idf*) factor is given by log *N/dfj*. A combined word importance indicator is the product (*tf*idf*) where the importance, or weight, *Wij* of a word *Tj* in the concept *Di* is defined as the word frequency multiplied by the inverse concept frequency. That is,

$$W_{ij} = tf_{ij} \cdot \log \frac{N}{df_j} \quad \cdots\cdots (1)$$

Through the *tf* * *idf* values, the word-concept matrix, which is the basis for the similarity measurement of a pairs of concepts, can be made. The pairwise comparison of a matrix produces N(N-1)/2 different pairs of similarity coefficients for the concepts. The similarity between concepts is measured by the following cosine coefficient measure. That is,

$$\text{sim}(X, Y) = \frac{\sum_{i=1}^{N} x_i y_i}{\sqrt{\sum_{i=1}^{N} x_i^2 \cdot \sum_{i=1}^{N} y_i^2}} \quad \cdots\cdots (2)$$

where x and y are vectors and N is the dimension of the vector space. The cosine coefficient has a value between 0 and 1. The more the concepts have overlapping words, the higher the cosine coefficient will be. If there are no overlapping words between the concepts, the cosine coefficient is zero. After the pair wise concepts for each sense are listed by descending order based on the cosine coefficient value, the definitions which have value zero should be removed because this sense of the concept has no connection with other senses of concepts. For illustration, an example is given in Table 2 that assumes the distinct records of curriculum attribute have Pascal, Basic, Java, and DBMS values.

In Table 3, we can exclude the senses, *pascal_0*, *basic_1*, and *java_0*, *java_1* because they do not share any words with other concepts, and have zero CSV value. We can also exclude the sense *pascal_1*, because a mathematician *pascal_1* has no hypernym relation. But dbms_0 should not be excluded since dbms_0 have a single sense for *dbms*. Now that we have elicited the correct senses of the concepts, the next step is to build a concept hierarchy through WordNet hypernyms of these concepts.

Table 2. A word-concept matrix of a curriculum attribute.
테이블 2. 커리큘럼 속성의 단어-개념 관계표

Senses/ Word	1623	meter	Borneo	Fermat	...	Writing
pascal_0	0	2.3	0	0	...	0
pascal_1	2.3		0	2.3	...	0
pascal_2	0	0	0	0	...	0
java_0	0	0	2.3	0	...	0
java_2	0	0	0	0	...	2.3
...	...	...	...	...	...	...
dbms_0	0	0	0	0	...	0

Table 3. The cosine value matrix of coefficients between the senses of all concepts.
테이블 3. 모든 개념의 의미 사이의 코사인 값 표

	pascal_0	pascal_1	pascal_2	basic_0	basic_1	java_0	java_1	java_2...	dbms_0
pascal_0	-	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
pascal_1		-	0.0000	0.0000	0.0000	0.0000	0.0000	0.0369	0.0000
pascal_2			-	0.0629	0.0000	0.0000	0.0000	0.0570	0.0000
basic_0				-	0.0000	0.0000	0.0000	0.0430	0.0000
basic_1					-	0.0000	0.0000	0.0000	0.0000
java_0						-	0.0000	0.0000	0.0000
java_1							-	0.0000	0.0000
java_2								-	0.0000
dbms_0								...	-

4.2 Building the concept hierarchy through WordNet hypernm

Through the disambiguating process of senses of a concept dealt with in the preceding section, some incorrect senses of concepts can be excluded from the set of concepts. Like in the previous section, database records have no higher domain names so it is necessary to get them from other sources. Fortunately, WordNet has the feature of hypernyms which have superordinate names having generalization relationships for concepts as well as subordinate names having specialization relationships for concepts. In addition, the semantic networks of WordNet are constructed by linguistic experts manually, so the semantic networks are not just mechanical interpretations but meaningful explanations of concepts. These features guarantee that the concepts used for making a concept hierarchy are reliable sources. Through these features of WordNet, we can collect the hypernyms of the remaining senses of concepts elicited by the disambiguating process from WordNet. After that, we overlap each hypernym of senses of concepts on the basis of shared superordinate names and generate a hierarchical semantic structure.

Fig. 4 shows the automatic generation of 4 concepts. After excluding incorrect senses, 4 senses remain correct senses for each concept. Each remaining sense has its own hypernym in WordNet, and we can generate the hierarchy shown in Fig. 4.

But as shown in Fig. 4, sense dbms_0 has no connection with other senses in the lower level of the hierarchy. This may happen partly due to natural output where there are no common words, partly due to the errors caused by human inconsistency involved in the process of creating and maintaining hypernyms based on human subjective judgment, and partly due to incorrect input data. In any case, adjustment is required of the resultant hierarchy by humans to complement these weaknesses. In the following section, we will discuss human control.

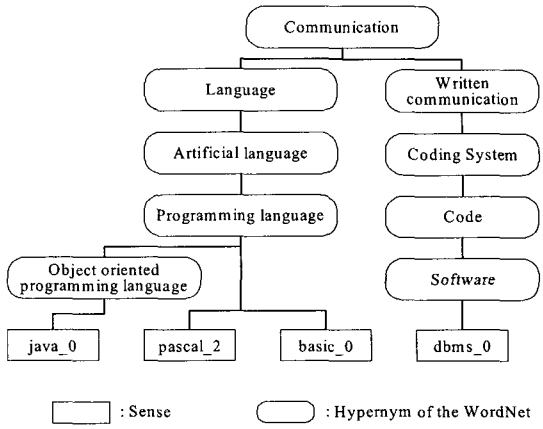

Fig. 4. A concept hierarchy using hypernyms.
그림 4. 상위어를 이용한 개념 계층

4.3 The adjustment of the resultant concept hierarchy

Perfection cannot be expected in the generation of the concept hierarchy because of different purposes of users, inappropriate hypernyms, inherent erroneous structure in WordNet, and the shortage of domain knowledge. In spite of these weaknesses in the resultant concept hierarchy, it is still helpful for domain knowledge experts because the defective hierarchy will still save time and effort in the complete generation of the concept hierarchy by small adjustments by humans. The adjustment can be divided into two parts; automatic and non-automatic adjustable regions. The automatic adjustable regions include the one-line path over two levels. This region can be reduced into one connection. But if one-line connection is useful to comprehend the structure of an underlying database, it should not be removed from the resultant hierarchy.

There are several parts that need to be adjusted manually. First, the failed senses to disambiguate correct sense, such as DBMS in Fig. 4, required to human control that puts DBMS in the proper location of the resultant hierarchy. Otherwise, domain knowledge experts can force the

appropriate sense of DBMS to be included in the automatic generation process of the concept hierarchy, or can make the hypernyms of these senses manually. In addition, the highest parts of the resultant hierarchy may usually have node names that are too wide range such as abstraction, entity, phenomenon, location because WordNet has fixed the highest node names at these branches. Therefore, it is necessary for hierarchy designers to replace these names with the proper concepts or to prune them. Lastly, the duplicated node names and erroneous senses should be removed from the resultant hierarchy.

V. Experiment

We test the automatic generation of a concept hierarchy using an academic background table of the faculty of KAIST. The table contains the user identification number, starting and ending years of employment, nationality, specialty, and other attributes. We choose the specialty attribute as a target for automatic generation of a concept hierarchy. For convenience of access and user friendly interface, we carry out the automatic generation of the concept hierarchy with C programming language in the Internet environment.

As for the disambiguation of senses, the specialty attribute has 79 concepts from accounting to zootechny which deals with the breeding and taming of animals. However, since WordNet does not define specific concepts like zootechny, these concepts cannot be included in the automatic generation of a concept hierarchy. Therefore, we obtain 69 concepts after excluding 10 undefined concepts, and elicit 142 senses for these concepts from WordNet (See Table 4). Among these concepts, 35 concepts have more than two senses causing ambiguity problems, and the number of senses is 108 senses. When we set the threshold at zero to elicit correct senses, a situation where concepts are not connected to the others, we are able to discriminate correctly 21 senses from 108 senses. But, setting the threshold at zero is extreme, and it does not consider the connection numbers to other concepts. For instance, biology_0 has 24 connections, biology_1 to 6, and biology_2 to 1. Like this, one or two senses of a concept are highly connected to other senses, and generally those senses are correct. When we set the threshold at the maximum connection number, about 70% of the senses are correctly discriminated. Thus, there is the trade-off relationship between the threshold and the number of correct senses.

Table 4. Summary on the specialty attribute
테이블 4. 전문분야 속성에 대한 요약

Type	# of concepts	# of senses
Concepts undefined in WordNet	10	-
Concepts defined in the WordNet	79	142
One sense concepts	34	34
More than two senses concepts	35	108

After eliciting the correct senses, we get hypernyms of these senses from WordNet. By overlapping each hypernym, we are able to make 6 hierarchies including abstraction, act, entity, group, phenomenon, and psychological feature because of topmost concepts of WordNet. Then, we combine 6 hierarchies into one hierarchy. The results are shown in Fig. 5. But this hierarchy is incomplete because 10 concepts are excluded, the higher levels are meaningless for the specialty domain, and it is necessary to adjust some parts. Therefore, we manually add 10 concepts into the resultant concept hierarchy, remove the higher levels, and alter the positions of the concepts to the proper places. Fortunately, these manual adjustments are relatively easy compared to a construction of concept hierarchies as a whole. In addition, the resultant hierarchy gives us an opportunity to understand new aspects of the specialty domain, and helps us save time in the generation of the concept hierarchy.

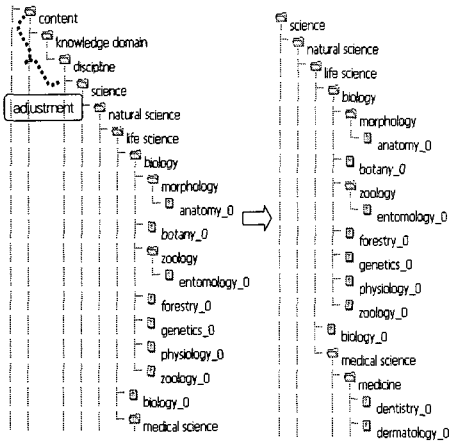

Fig. 5. The resultant hierarchy of the specialty attribute at a university.
그림 5. 대학의 전문 분야 속성에 대한 계층 생성 예

VI. Conclusions

In this paper, we propose the automatic generation of a concept hierarchy through WordNet for nominal data. For this, we first choose a target database, select an attribute from which a concept hierarchy is to be built, and get a set of concepts. Second, we discriminate senses of a concept considering other concepts with similarity values measured by VSM. Third, we generate the concept hierarchy through hypernyms of concepts in WordNet, an existing rich lexical resource, by welding each hypernym to construct the concept hierarchy. Lastly, we automatically adjust some parts of the resultant concept hierarchy and other parts of it with human control.

Initially, among the contributions, the following two points are worthy of attention. First, we overcome the manual generation of a concept hierarchy which is an enormous and expensive cost work. Second, we apply the vector space model to discern senses of a concept and elicit the correct senses of a concept by similarity values. We use WordNet explanation of concepts, measure similarity values between the pairs of concepts, and exclude the senses having zero similarity value.

Reference

[1] Chu, W.W., Liu, Z., and Mao, W., Techniques for Textual Document Indexing and Retrieval Knowledge Sources and Data Mining. Clustering and Information Retrieval, 135-160, 2003.

[2] Dierbach, C., Application of the Abstractional Concept Mapping Theory for the Interpretation of Novel Metaphor, SNPD, 157-162, 2005.

[3] Grzymala-Busse. J. W., Discretization of Numerical Attributes, Handbook of Data Mining and Knowledge Discovery, Oxford University Press, 218-225, 1999.

[4] Han, J., and Fu, Y., Dynamic Generation and Refinement of Concept Hierarchies for Knowledge Discovery in Databases. AAAI Workshop on Knowledge Discovery in Databases, 157-168, 1994.

[5] Han, J., Cai, Y., and Cercone, N., Knowledge Discovery in Database: An Attribute-Oriented Approach, In Proceedings of the 18th VLDB conference, 547-559, 1992.

[6] Hinrich Schutze, Automatic Word Sense Discrimination. Association for Computational linguistics, 24(1), 97-123, 1998.

[7] Huh, S., and Lee, J., Providing Approximate Answers Using a Knowledge Abstraction Database. Journal of Database Management, April-June, 14-24, 2001.

[8] Jing, Y., and Croft, W.B., An Association Thesaurus for Information Retrieval (Technical Report #94-17). Amherst, MA: University of Mass at Amherst, 1994.

[9] Lagus, K., Kaski, S., and Kohonen, T., Mining massive document collections by the WEBSOM method. Information Science, 163(1-3), pp. 135-156, 2004.

[10] Lagus, K., Text Retrieval Using Self-Organized Document Maps. Neural Processing Letters, 21-29, 2002.

[11] Lam, W., Keung, C., and Liu, D., Discovering Useful Concept Prototypes for Classification Based on Filtering and Abstraction. IEEE Trans. Pattern Anal. Mach. Intell., 24(8), 1075-1090, 2002.

[12] Merkl, D. and Rauber, A., Uncovering the Hierarchical Structure of Text Archives by Using an Unsupervised Neural Network with Adaptive Architecture, In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, 384-395, 2000,

[13] Miller G. A., Beckwith R., Felbaum C., Gross D., and Miller K., Introduction to WordNet: An On-line Lexical Database. International Journal of Lexicography, 3(4), 235-244, 1990.

[14] Zhu, S., Yang, C.C., and Lam, W., CatRelate: A New Hierarchical Document Category Integration Algorithm by Learning Category Relationships, ICADL, 280-289, 2004.

[15] Salton G., Automatic text processing: the transformation, analysis, and retrieval of information by computer, Addison-Wesley. 1989.

저 자 소 개

신 명 근

2006년 2월 : KAIST 정보및통신 공학 박사

1994년 ~ 1999년: 정보시스템연구소 선임연구원

2002년 ~ 현재 : (주)투이

관심분야 : 협력적 질의 처리, 데이터베이스, 웹구조화