Discovery of Characteristic Subgraph Patterns using Relative Indexing and the Cascade Model

Takashi Okada and Mayumi Oyama

Center for Information & Media Studies, Kwansei Gakuin University
1-1-155, Uegahara, Nishinomiya 662-8501, Japan
{okada,oyama}@kwansei.ac.jp

Abstract: Relational representation of objects using graphs reveals much information that cannot be obtained by attribute value representations alone. There are already many databases that incorporate graph expressions. We focus on two types of database, one for syntactic trees in language sentences and one for chemical compound structures. We attempt to mine characteristic subgraph patterns from these databases using a common framework. This mining process employs two methods: relative indexing of graph vertices and the cascade model. The former extracts many linear subgraphs from the database. An instance is then represented by a set of items, each of which indicates whether a specific linear subgraph is contained within the graph of the instance. The cascade model is a rule induction method that uses levelwise expansion of a lattice. If the distribution of the attribute values along a link in the lattice shows a sudden change, then that link is represented as a rule, whose strength is measured by the *BSS* value of the link. The basic assumption of this mining process is that characteristic subgraphs may be well represented by the concurrent appearance of linear subgraphs. The resulting rules are shown to be a good tool for obtaining valuable knowledge in linguistics and toxicology.

1 Introduction

Structured objects can be represented very effectively by using graphs. Graphs can express general relationships in data that cannot be obtained by the usual attribute value expressions, and many databases therefore incorporate graph representations. For example, the structural formulae in chemistry, syntactic trees in natural language, and circuits in engineering all use graphs. We put our focus on the mining method applicable to all of these graph-structured objects.

In chemistry, a graph is the most fundamental language of representation to explain the structure of compounds. Studies of SAR (structure activity relationship) and SPR (structure property relationship) have been among the core research subjects in the field of chemical information research. Though the main stream of these studies has concerned itself with statistical methods treating a relatively narrow range of compound classes, there have also been several attempts to mine knowledge from graph databases with diverse structures [1-4].

Recently, interest in graphing structured objects has increased in the fields of machine learning and data mining; there has been work on GBI (graph based induction) [5], ILP (inductive logic programming) [6], and the association rule for graphs [7, 8]. ILP has been applied to SAR problems and shown useful [9]. However, these methods have not sufficiently considered all respects of universal validity, applicability to a variety of problems, and required computational resources, and there is a need for a new, efficient method.

The principal aim of this paper is to propose a mining scheme that is generally applicable to different kinds of graph-structured objects. We demonstrate that it is applicable to investigation of both syntactic trees and chemical structures. Section 2 explains the new mining process, which consists of item generation and application of the cascade model. In Section 3, the procedure is applied to the analysis of syntactic parse trees. Section 4 discusses the results obtained from an SAR study of the mutagenicity data of aromatic nitro compounds.

2 Mining Methods

2.1 General scheme

We propose a mining scheme that consists of two steps. In the first, we generate thousands of attributes from a set of instance graphs; each attribute denotes whether a specific subgraph is contained within the graph. The method of relative indexing of vertices restricts the subgraphs to linear types, and provides an affordable number of attributes. Each graph can then be described as a tuple in a table with thousands of columns. The whole graph object property is also included as an attribute of the table.

The second step is to find dependencies among attributes. There are many possible methods; we employ a decision tree to derive classification rules for some attributes. Alternatively, the subgraph patterns of a graph could be regarded as items in a basket

and the association rules method could be applied. In this paper, we employ the cascade model to derive rule expressions. We chose this model because it is able to derive characteristic, and/or classification, rules in a single unified framework, with a pruning method that can suppress combinatorial explosion of lattice size, even with high item density.

The resulting rules can act as a guide in extracting valuable knowledge from a database. The rules are expressed not by the target subgraph, but by the concurrent appearance of plural linear subgraphs that are interpreted to provide knowledge.

2.2 Relative Indexing of Graph Vertices

Our method can be explained as follows, using a syntactic tree as an example. Suppose that we wish to find characteristic patterns in the syntactic tree associated with the verb "think". An example of the tree is shown in Figure 1; it contains 8 leaf vertices and 6 non-leaf vertices.

If we extract all possible subgraphs from this tree, the number of attributes will be too large for most mining methods to handle, and we therefore need to impose some restrictions on the subgraph pattern. To do so, we introduce a new scheme: the relative indexing of graph vertices. This scheme assumes that a subgraph is linear and consists of two parts.

- Two meaningful vertices.
- The relationship between the two vertices.

We can fix one of the two meaningful vertices to the leaf vertex, [VT: think], as our aim is to analyze the syntactic pattern based on its usage. As the non-leaf vertices in this tree possess no valuable information except topology, we can restrict the source of the other meaningful vertex to the leaf vertices. Therefore, the attributes employed are the 7 subgraphs between "think" and 7 leaf nodes, as shown in Table 1.

The next problem is the expression of the relationship between the selected leaf vertices. As the resulting rules are depicted using these expressions, we expect the original graph structure to recover as much as possible from the attributes' expression. The syntactic tree is an ordered tree and the edges branching from a vertex can be numbered. Therefore, we assign a relative index to each leaf node, as shown in the

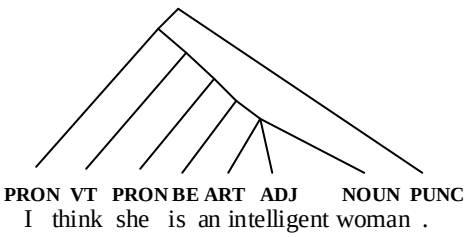


Fig. 1. Sample syntactic tree.

Table 1. Leaf vertices and relative indices.

Word	Part of speech	Index
I	PRON	1.2./.1
she	PRON	1./.2.1
is	BE	1./.2.2.1
an	ART	1./.2.2.2.1
intelligent	ADJ	1./.2.2.2.2
woman	NOUN	1./.2.2.2.3
. (period)	PUNC	1.2.1./.2

last column of Table 1. The relative index of the word "I" is given by "1.2./.1", where "/" indicates the root vertex of the minimum subtree containing the two words, as shown in Figure 2. Starting from the position of "/", the numbers on the left (right) side, delimited by periods, indicate the sequence of edge indices to the word "think" ("I"). Here, we have assigned the edge index 1 to the leftmost edge. The resulting relative index is given by concatenating the two indices to "think" and "I".

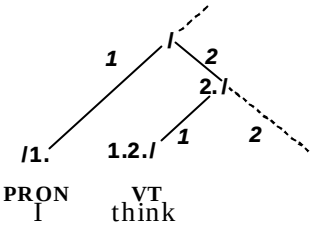


Fig. 2. Relative indexing between "think" and "I".

We can define a unique relative index for any vertex. Consequently, we can recover the relative positions of the words from the index unambiguously. However, this indexing scheme may require modification, depending on the problem considered. For example, when treating chemical compounds, the edges in a graph are not ordered, and therefore we cannot give an unambiguous index between a pair of vertices.

The characteristic subgraph that is to be mined may very well be a general graph that cannot be represented as a linear graph. There is then the question as to whether a set of generated linear subgraphs can stand in for a general subgraph in the representation of a rule, and this is the core point by which to judge the current method. We anticipate that in most cases the concurrent appearance of linear subgraphs in a rule can substitute for a general subgraph. We inspect this hypothesis using two kinds of problems later in this paper.

2.3 The cascade model

The cascade model was originally proposed by Okada [10]. It can be considered as an extension of association rule mining. The method creates an itemset lattice, where [attribute: value] pairs are employed as items that constitute itemsets. Links in the lattice are selected and expressed as rules, by examining the distribution of the RHS attribute values along all the links. A sudden change of distribution along a link will bring the two terminal nodes of that link into focus. Suppose that the itemset at the upper end of a link is [A: y], and that an item [B: n] is added along the link. If a sharp increase in [C: y] is then found along this link, we can write a rule with the following expression:

IF [B: n] added on [A: y] THEN [C: y].

where the added item [B: n] is the main condition of the rule, and the items on the upper end of the link ([A: y]) are considered as the preconditions. Any number of items can be put into the RHS of a rule if its distribution shows a strong interaction with the main condition.

Subsequently, the sum of squares criterion for categorical data was introduced to improve the definition of rule strength [11, 12]. The formulation of the model was also extended to cover the mining of classification rules and characteristic rules in a unified framework [13]. The problem of combinatorial explosion in the number of

lattice nodes was also resolved by a new pruning methodology [14]. The cascade model is implemented as DISCAS software using lisp, and it is used in this work.

3. Application to Syntactic Trees

3.1 Problem Definition and Computation

Mining from corpus data may lead to new knowledge in linguistics, which may be reflected in improvements in natural language processing. We used the Electronic Dictionary Research (EDR) English corpus, which contains 160,000 sentences, with syntactic tree data for each [15]. As an example, we extracted sentences containing the verb "think" and tried to find characteristic patterns that were associated with this word. Among the 1,001 sentences retrieved, there were 134 and 867 sentences that contained VI and VT verbs, respectively.

The corpus treats all blanks between words in a sentence as a special kind of word; we omitted these blanks to simplify the trees. There is a linguistic tag on every non-leaf vertex in the tree of this corpus, but it proved too difficult to interpret these tags and we were forced to omit them. After preprocessing, the resulting tree had the structure shown in Figure 1. The details of the corpus data, including definition of parts of speech, can be accessed over the Internet [15].

Generating an attribute using the scheme in Section 2.2 provides the option of using another indexing scheme through numbering the edges from right to left. As there is no reason to prefer one indexing scheme to the other, we employed both schemes to generate relative indices. The attribute format was set to the concatenation of the index and the part of speech columns in Table 1.

Using the two indexing schemes, every word except "think" generates two attribute records. The number of records created from 1001 sentences was 28010, of which 10469 were recognized as different. The verb class, VI or VT, was also added as an attribute.

The cascade model was used to mine for characteristic rules, using the parameter values (*minsup*: 0.05, *thres*: 0.05, *thr-BSS*: 0.05) [14]. The pruning condition defined by the *thres* value can eliminate most attributes from the actual computation, and in this case left only 29 attributes for construction of the lattice. That is, if the two values *y* and *n* of an attribute have a very unbalanced distribution they do not contribute to forming the characteristic subgraph patterns.

The lattice construction took 7 seconds, giving 359 nodes, using a 266MHz Pentium II computer. The first rule set gave us 5 rules, which explained about half of the total sum of squares in the problem.

3.2 Interpretation scheme

The strongest rule is the first rule of the first rule set and has the expression shown in Figure 3. The main condition of this rule indicates the existence of AUX (auxiliary verb) at the position [1-/-2-2], where hyphens are delimiters among edge indices

```
IF [1-/-2-2AUX: y] added on []
THEN [2./.1.1AUX: y]     11.7%->100.0%; BSS: 91.2
THEN [1-/-2-2AUX: y]     11.7%->100.0%; BSS: 91.2
THEN [1-/-2-1ADV: y]     11.6%-> 98.3%; BSS: 88.0
THEN [2./.1.2ADV: y]     11.6%-> 98.3%; BSS: 88.0
THEN [1-2-1-/-2PRON: y]  14.3%-> 84.6%; BSS: 57.9
THEN [2.1.2./.1PRON: y]  14.3%-> 84.6%; BSS: 57.9
    Cases: 1001 -> 117              Sum_BSS:640.
```

Fig. 3. A rule expression for the verb "think" by the cascade model

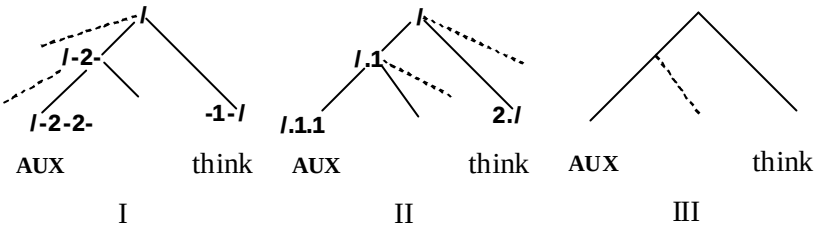


Fig. 4. Subtree expression of the main condition of a rule

numbered from the right. Six RHS clauses are shown in decreasing order of *BSS* values. The underlined row is included to show the information of the main condition item. The last line shows that among 1,001 sentences , 117 satisfy the main condition, and the sum of *BSS* values for all attributes is 640 along this link.

The position of AUX indicated by the main condition is illustrated as **I** in Figure 4, where dashed lines denote the possibility of edges at the indicated locations. The first line of the RHS part indicates the existence of the subgraph **II**. The percentage of subgraph **II** increases from 11.7% to 100% along this rule link, and the associated *BSS* value is 91.2. We can therefore say that the appearance of subgraph **I** is always accompanied by that of **II**. As the frequencies of these two subgraphs are the same, they will always appear together, so the actual main condition of the rule can be expressed by subgraph **III** in Figure 4.

The lines 3-4 and 5-6 indicate the high confidence for concurrent appearance of subgraphs **IV** and **V** when the main condition is satisfied. In conclusion, the overall rule interpretation is shown by **VI** where the subgraph pattern, depicted by the solid lines, appears very frequently when the auxiliary verb is located at the position shown by the bold lines. Also indicated in **VI,** by the dotted lines, is the punctuation symbol that appears with the confidence of 61.5%.

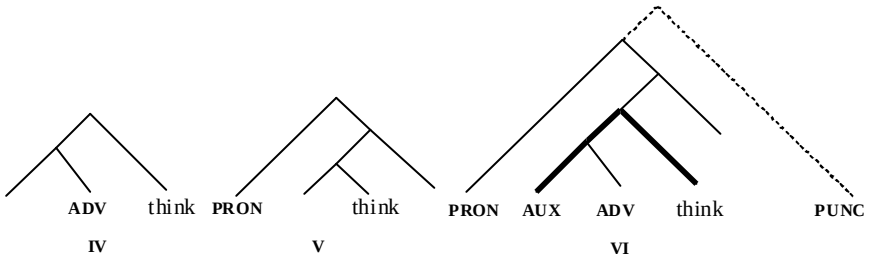


Fig. 5. Characteristic subgraph patterns found in a rule.

Another example of a RHS clause with a large *BSS* value is shown below,

```
THEN [1.2./.1 PRON: n]  57.6% -> 100.0%; BSS:21.0
```

The item of this RHS clause indicates the nonexistence of the specified pattern. Two interpretations are possible for this description. One suggests the existence of words other than PRON at this location; the other leads to the nonexistence of the location itself in the tree, since either no words exist, but rather a subtree, or the existence of the location is incompatible with the main condition. We can see that the location and subgraph **VI** are contradictory in this rule, and therefore this clause does not add useful information.

3.3 Characteristic patterns

The first rule set contained 5 characteristic rules, from which we constructed 3 characteristic patterns, **VI – VIII**, shown in Figure 6, following the procedure given

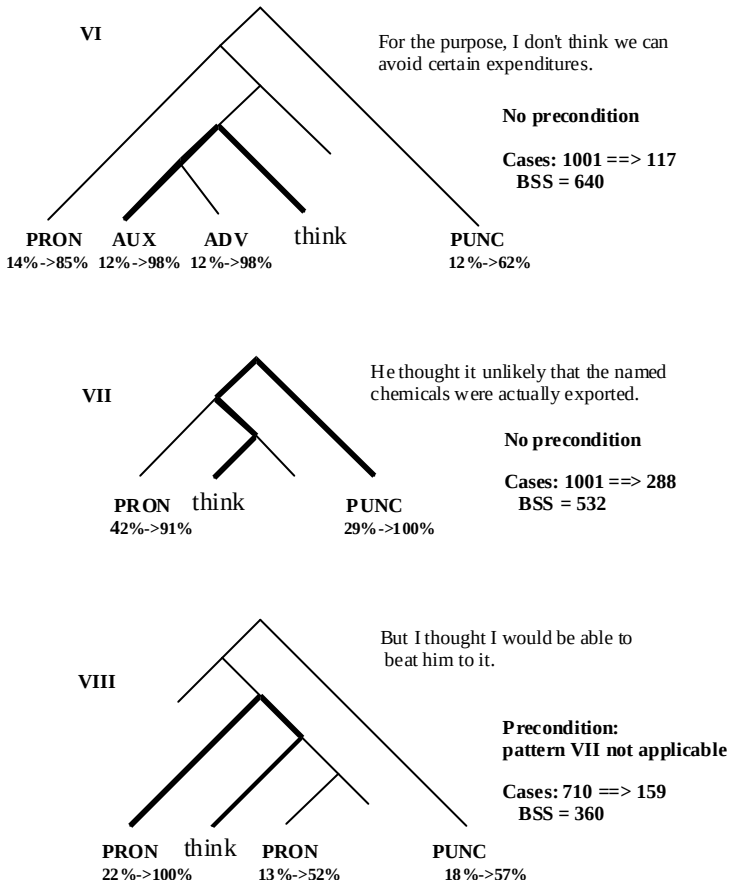


Fig. 6. Characteristic subgraph patterns for the verb “think”.

in the previous section. One rule has few supporting sentences and the other only discriminates a group of sentences from those characterized by the three patterns. Therefore, we can conclude that these are the major patterns associated with the usage of "think". These patterns are exclusive to each other, and cover 56% of all sentences.

The patterns in Figure 6 are shown with an example sentence, the precondition description, the number of cases, and a *BSS* value. The sub-pattern shown by bold lines indicates the main condition, while solid lines are concurrent ones. No significant changes in the VI/VT ratio were observed in these patterns.

In fact, we can see these patterns frequently in various media. How are we to understand the absence of nouns at the locations of pronouns in these patterns? We have to be careful in the interpretation of the patterns. Actually, we can expect proper nouns and noun phrases at the same locations, but proper nouns are less frequent than pronouns and the noun phrase is not recognized in the corpus. Incorporating these kinds of items should result in more impressive patterns.

The results obtained here can be regarded as a type of statistic on syntactic pattern. Extensive application of this method is expected to lead to new knowledge in the field of linguistics.

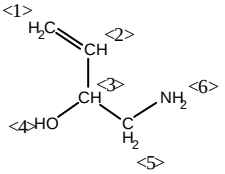
4. Application to Mutagenicity of Chemical Compounds

4.1 Problem Definition and Computation

The objective in this section is to discover valuable SAR rules for the mutagenicity of chemical compounds. Debnath et al. compiled the mutagenicity data of 230 nitro aromatic compounds [16]. An ILP study examined this data [9]. The SDF dataset of these compounds, prepared for the KDD challenge 2000 at PAKDD-2000, was used for this SAR study.

The method of relative indexing of vertices is applied to chemical structural formulae. Here, we cannot restrict one of the meaningful vertices to a particular atom, but rather have to employ all pairs of non-hydrogen atoms as meaningful vertices. However, if both atom pairs are alkyl carbon then it can be excluded from the attribute set by chemical knowledge. We employ the sequence of elements and the bond types in the shortest connecting path between two vertices as the relationship between a pair of atoms,.

Figure 7 shows an example of a structural formula and the set of linear subgraph patterns derived from it. For example, the pattern of the bottom left column employs



C3H=C3H	C3H-C4H	C4H-N3H
C3H=C-C4H	C3H-C-O2H	C4H-C-N3H
C3H=C-C-O2H	C3H-C-C4H	N3H-C-C-O2H
C3H=C-C-C4H	C3H-C-C-N3H	C4H-O2H
C3H=C-C-C-N3H		C4H-C-O2H

Fig. 7. A structural formula and its linear subgraph patterns.

the atoms <1> and <6> as meaningful vertices, and the relationship is described by "=C-C-C-", showing the sequence of the bond types and element symbols along the path, <1>=<2>-<3>-<5>-<6>. An aromatic bond is denoted as "r". Description of an atom includes the coordination number (number of adjacent atoms) and the existence of attached hydrogen atoms. Atom types like C3H and O2H have also been added to members of subgraph patterns.

We note that the scheme employed generates a unique index, but the interpretation of the index is ambiguous. However, as the patterns are written in the language of chemistry they should still prove useful. Items based on these patterns can be regarded as constituting a kind of molecular fingerprint, similar to the descriptor set employed in Klopman's CASE system [1].

The mutagenic activity (*y*) was categorized into 4 classes (inactive, low: $y < 0.0$, medium: $0.0 \leq y < 3.0$, high: $y \geq 3.0$), and the *BSS* value of the categorized activity was calculated on the assumption that each category is nominal.

The item generation process generated 17995 linear subgraph patterns from 230 graphs, in which we found 2044 different patterns. Item datasets were analyzed by DISCAS software; the mutagenic activity was employed as the only attribute to be placed in the RHS. The pruning conditions were set to *minsup* = 0.05 and *thres* = 0.1, which excluded most of the 2044 attributes from the mining process, as they did not have sufficient discrimination power. The lattice was constructed using 77 subgraph attributes. DISCAS generated a lattice containing 1, 91, 1910, and 4937 nodes at the lattice level with 0 to 3 items. It took 109 seconds to create all 6939 nodes, using a 266MHz Pentium II computer.

A link was selected and expressed as a rule if its *BSS* value for the activity was larger than 2.3 (1% of cases). This resulted in 209 rules, which were then grouped into 10 rule sets to facilitate inspection. The rules in a rule set were selected so that their supporting instances did not overlap and the rules were mutually independent. The first rule set contained ten rules. The interpretation of the strongest rule will be given in the next section, while the overall results will be reviewed from a chemistry standpoint and submitted to a suitable scientific journal.

4.2 Interpretation of Rules

The strongest rule, which is the first rule of the first rule set, has the following expression:

IF [C3HrCrC-CrCrCrC-N3: y] added on [C3rCrCrCrC3: n]
THEN Activity = low
40.8% -> 14.0%; BSS: 3.25; Cases: 157 -> 43
0.10 0.41 0.41 0.08 ==> 0.00 0.14 0.58 0.28

The precondition states that there are no 4 consecutive aromatic bonds like **IX** in Figure 8, while the main condition reveals the importance of substructure **X**. The RHS denotes that a large decrease is observed in the percentage of compounds with [Activity: low]. The third line of this rule shows that only 43 compounds satisfy the main condition, among the 157 compounds selected by the precondition. The percentage of [activity = low] decreases from 40.8% to 14.0%, and the *BSS* value of

this rule is 3.25. The bottom line shows the detailed distribution of the activity levels (inactive, low, medium, high) among the compounds, before and after the application of the main condition. We can see that the main condition has shifted the distribution to higher activity levels.

DISCAS can write an optional RHS by request. That is, an attribute-value pair and its change in percentage are depicted as shown below, if it has high correlation with the main condition.

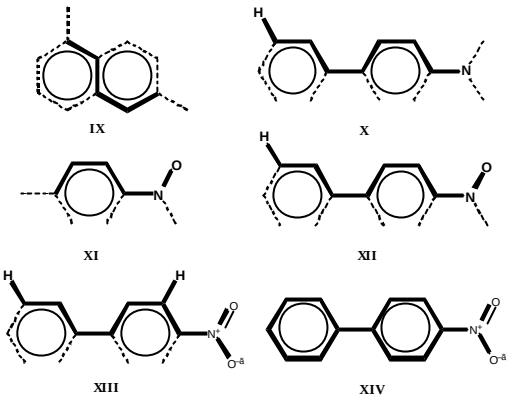


Fig. 8. Subgraph patterns in the strongest rule.

THEN C3rCrCrC-N-O1 = y 68.2% -> 100.0%; BSS: 4.36

This substructure pattern is shown as **XI**. Since its percentage becomes 100% after application of the main condition, the superposition of **X** and **XI** should be the real main condition. Retrieval of the dataset has shown that these patterns can be unified to give a larger pattern, **XII**. Consideration of other optional RHS's has led us to the conclusion that **XIII** should be the substructure pattern for the main condition.

Retrieval of substructure **XIII** shows that all 43 compounds supporting this rule share substructure **XIV**, while none of the 114 compounds excluded contain it. Therefore, this rule can be stated as "After we exclude highly polycyclic aromatic compounds like **IX**, IF a 4-nitrobiphenyl (**XIV**) substructure exists in a compound, THEN the compound is expected to be more mutagenic than otherwise."

5. Concluding Remarks

Combination of the relative indexing of graph vertices and the cascade model has led to successful data mining in two entirely different fields, linguistics and chemistry. The ordered directed tree of sentence syntactic structure presents a clear contrast to the unordered undirected graph of chemical structure formulae. Both applications generate thousands of subgraph patterns as attributes, from which the efficient pruning strategy of the cascade model is able to select less than one hundred attributes to construct a lattice. The whole computation process is very efficient. Moreover, the search is exhaustive, using the given pruning parameters with the mining process. All of these points show the excellence of this mining method.

The basic strategy, the representation of a characteristic subgraph by the superposition of linear subgraphs, seems to work well, at least in the two applications examined. However, the rule interpretation process by individuals requires future development. Specifically, the negation item can be interpreted in several ways, and

constant consultation with the database is required. Further work and research of this mining process should yield positive results in various applications.

References

[1] Klopman, G.: Artificial Intelligence Approach to Structure-Activity Studies, *J. Amer. Chem. Soc.* **106**, pp.7315–7321 (1984).

[2] Okada, T., Wipke, W.T.: CLUSMOL: A System for the Conceptual Clustering of Molecules, *Tetrahedron Computer Methodology*, **2**, pp.249-264 (1989).

[3] Okada, T., Kawai, T.: Analogical Reasoning in Chemistry. 1. Introduction and General Strategy, *Tetrahedron Computer Methodology*, **2**, pp.327-336 (1989).

[4] Brown, R.D., Martin, Y.C.: Use of Structure-Activity Data to Compare Structure-Based Clustering Methods and Descriptors for Use in Compound Selection, *J. Chem. Inf. Comput. Sci.*, **36**, 572-584 (1996).

[5] Yoshida, K., Motoda, H.: CLIP: Concept Learning from Inference Patterns, *Artificial Intelligence*, **75**, pp.63-92 (1995).

[6] Muggleton, S., Raedt, L.D.: Inductive Logic Programming:Theory and Methods. *J. Logic Programming*, **19**, pp.629–679 (1994).

[7] Dehaspe, L., Toivonen, H., King, R.D.: Finding Frequent Substructures in Chemical Compounds, *Proc. KDD-98*, pp.30–36, AAAI (1998).

[8] Inokuchi, A., Washio, T., Motoda, H.: Derivation of the Topology Structure from Massive Graph Data, *Discovery Science*, pp.330-332 LNAI 1721, Springer-Verlag (1999).

[9] King, R.D., Muggleton, S.H., Srinivasan, A., Sternberg, M.J.: Structure-Activity Relationships Derived by Machine Learning. *Proc. Natl. Acad. Sci. USA*, **93**, pp.438–442 (1996).

[10] Okada, T.: Finding Discrimination Rules Using the Cascade Model, *J. Jpn. Soc. Artificial Intelligence*, **15**, pp.321-330 (2000).

[11] Gini, C.W.: Variability and Mutability, contribution to the study of statistical distributions and relations, *Studi Economico-Giuridici della R. Universita de Cagliari* (1912). Reviewed in Light, R.J., Margolin, B.H.: An Analysis of Variance for Categorical Data, *J. Amer. Stat. Assoc.* **66,** pp.534-544 (1971).

[12] Okada, T.: Sum of Squares Decomposition for Categorical Data, *Kwansei Gakuin Studies in Computer Science*, **14**, pp.1-6 (1999). http://www.media.kwansei.ac.jp/home/kiyou/kiyou99/ kiyou99-e.html

[13] Okada, T.: Rule Induction in Cascade Model based on Sum of Squares Decomposition, *Principles of Data Mining and Knowledge Discovery* (*Proc. PKDD'99*), pp.468-475, *LNAI* 1704, Springer-Verlag (1999).

[14] Okada, T.: Efficient Detection of Local Interactions in the Cascade Model, *Knowledge Discovery and Data Mining* (*PAKDD 2000*) LNAI 1805, pp.193-203, Springer (2000).

[15] Japan Electronic Dictionary Research Institute: EDR Electronic Dictionary, http://www.iijnet.or.jp/edr/index.html.

[16] Debnath, A.K. et al.: Structure-Activity Relationship of Mutagenic Aromatic and Heteroaromatic Nitro compounds, *J. Med. Chem*. **34**, pp.786–797 (1991).