Building a Knowledge-Base for Protein Function Prediction using Multistrategy Learning

Takashi Ishikawa [1] takashi@j.kisarazu.ac.jp
Shigeki Mitaku [2] mitaku@cc.tuat.ac.jp
Takao Terano [3] terano@gssm.otsuka.tsukuba.ac.jp
Takatsugu Hirokawa [2] hirokawa@cc.tuat.ac.jp
Makiko Suwa [2] suwa@cc.tuat.ac.jp
Seah Boon Ching [2] seah@cc.tuat.ac.jp

[1] Kisarazu National College of Technology
2-11-1 Kiyomidai-higashi, Kisarazu, Chiba 292, Japan

[2] Tokyo University of Agriculture and Technology
2-24-16 Naka-cho, Koganei-shi, Tokyo 184, Japan

[3] The University of Tsukuba
3-29-1 Otsuka, Bunkyo-ku, Tokyo 112, Japan

Abstract

Conventional techniques for protein function prediction using similarities of amino acid sequences enable us to only classify the protein functions into function groups. They usually fail to predict specific protein functions. To overcome the limitation, in this paper, we propose a method for protein function prediction using functional feature analysis and a multistrategy learning approach to building the knowledge-base. By "functional feature", we mean a feature of an amino acid sequence characterizing the function of a protein with the amino acid sequence. They are secondary and/or tertiary structures of amino acid sequences that corresponds to functional elements comprising the functions of a protein. The functional features are extracted from amino acid sequences using Abductive inference, Inductive inference, and Deductive inference. In this paper, we show the effectiveness of the method by an example problem to classify functions of bacteriorhodopsin-like proteins.

[1]石川　孝：木更津工業高等専門学校　情報工学科，〒 292 千葉県木更津市清見台東 2-11-1
[2]美宅成樹，広川貴次，諏訪牧子，謝　文清：東京農工大学工学部，〒 184 東京都小金井市中町 2-24-16
[3]寺野隆雄：筑波大学社会工学系　経営システム科学，〒 112 東京都文京区大塚 3-29-1

1 Introduction

A major issue in genome informatics is to predict functions of proteins coded in DNA fragments. Conventional computational methods for protein function prediction are based on an empirical principle: "*proteins with similar functions should have similar amino acid sequences*". They use the concepts of homology or motif as the similarity measures of amino acid sequences [11] [10]. These methods, however, are only able to classify protein functions into function groups. They usually fail to predict specific protein functions. Furthermore, only few motives corresponding to specific functions have been discovered so far. Therefore, we need a new method for protein function prediction.

An alternative approach to protein function prediction is to apply machine learning techniques to find unknown features and rules to classify protein functions [2]. However, usual inductive inference methods using statistical measures to select features require so many training examples that they are hard to apply because there are a very few examples applicable. From the reason, we explore new machine learning techniques to find classification features and rules using *knowledge intensive multistrategy learning approaches*.

Multistrategy learning is a machine learning technique that integrates multiple inference methods, which generally include *Abductive inference*, *Inductive inference*, and *Deductive inference* [7]. In the proposed method, we apply the algorithm of analogical reasoning, *Analogy by Abstraction* (ABA) [4] [6], to *Abductive inference* to find features characterizing protein functions. In Abductive inference to find new classification features, the method generates hypotheses for a functional model of a target protein by transforming a functional model of a base protein similar to the target protein. Then the method generates classification rules to discriminate the target protein functions from the other proteins using Inductive inference.

2 Protein Function Prediction System

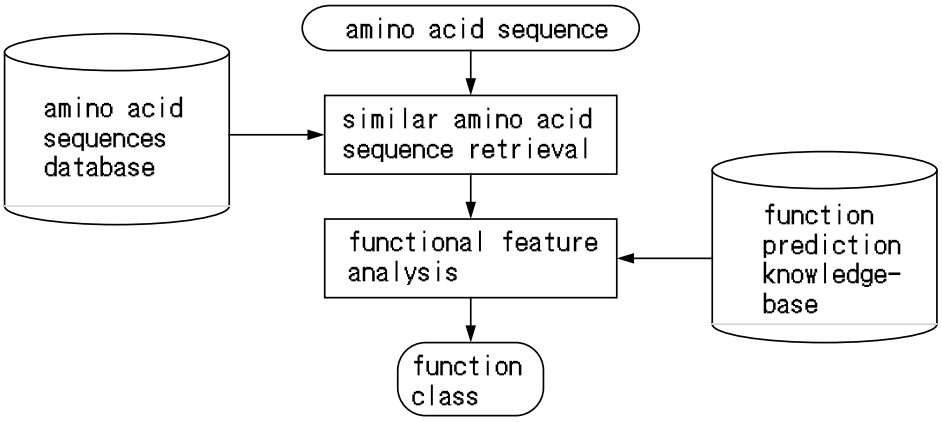

Figure 1: An overview of the protein function prediction system

The protein function prediction system outputs a function class of a protein from its amino acid sequence of the protein (Figure 1). The system comprises two processes: *similar amino acid*

sequence retrieval and *functional feature analysis*. The similar amino acid sequence retrieval finds proteins having amino acid sequences similar to the inputted amino acid sequence from the amino acid sequences database. If the function classes of retrieved proteins fall into one function class, the system outputs the function class. Otherwise, the system executes the next functional feature analysis to decide which class the target protein belongs to.

For a given amino acid sequence of unknown protein functions, we can find candidates of the protein functions by retrieving amino acid sequences similar to the target amino acid sequence. The protein function of retrieved amino acid sequences generally falls into a function group of the protein. Even if the retrieved amino acid sequences have a same protein function, it is not sure that the target amino acid sequence has the same function. To overcome the limitation of the similar amino acid sequence retrieval, the *functional feature analysis* refines candidate protein functions to a specific protein function. The process finds classification features, generates classification rules, and apply the rules to determine a specific protein function using the function prediction knowledge-base. In the process, the system learns *new* knowledge and extends its knowledge-base.

3 Functional Feature Analysis

The functional feature analysis is to analyze features of an amino acid sequence that corresponds to functional elements which constitute the functions of a protein (Figure 2). A functional feature is defined as a feature of an amino acid sequence that characterizes the function of a protein. Typical examples of functional features are secondary structures of proteins (e.g., alpha-helices) and specific amino acid residues corresponding to certain physico-chemical interactions of proteins [5] [9]. In the functional feature analysis we assume 1) that the whole function of a protein is decomposed into functional elements, and 2) that the functional elements are characterized by the functional features of its amino acid sequence. Therefore, we consider the problem of protein function prediction as the recognition of functional features corresponding to the functional elements.

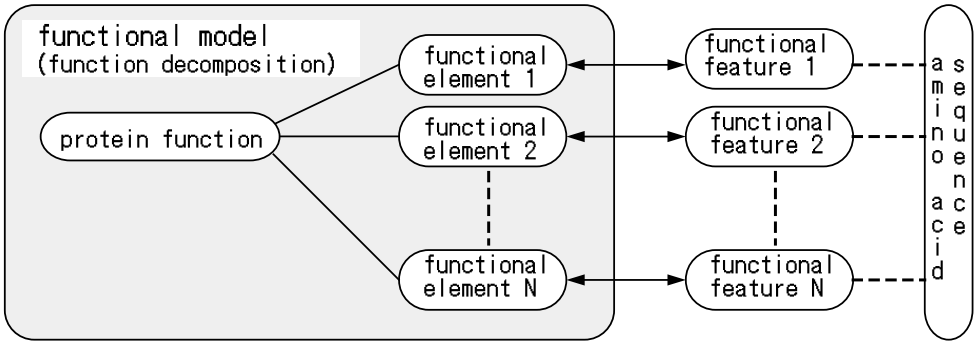

Figure 2: A scheme of functional feature analysis

In Figure 2, a functional model is a description of a relationship between a protein function and constituting functional elements. If a whole function of a protein is characterized with

its constituent functional elements, the problem of recognizing a protein function is reduced to a problem of recognizing a set of functional elements. Functional models have hierarchical structures when the functional elements are comprised from functional elements of the lower levels. In functional feature analysis, we use relationships between a protein function and its amino acid sequence by decomposing the protein function into functional elements such that these elements can be corresponded to some features of the amino acid sequence. In our model, a functional element can be corresponded to multiple functional features.

4 Multistrategy Learning Architecture

A unique feature of our method is its multistrategy learning architecture consisting of *Abductive inference*, *Inductive inference*, and *Deductive inference* shown in Figure 3. Outputs of each inference are respectively used in the next step of the inference. *Abductive inference* generates hypotheses for a functional model of a target protein using an initial functional model of a base protein. The initial functional model explains functions of the base protein using the functional features of its amino acid sequence. *Inductive inference* makes classification rules to discriminate the function of the target protein from functions of the similar proteins. Inputs of Inductive inference are the generated functional model of the target protein and a set of similar amino acid sequences with known functions. *Deductive inference* predicts a function class of the target protein from the amino acid sequence of the target protein using the generated classification rules. When the final Deductive inference outputs incorrect function classes, it requires to select another candidate hypotheses generated in the former inference processes.

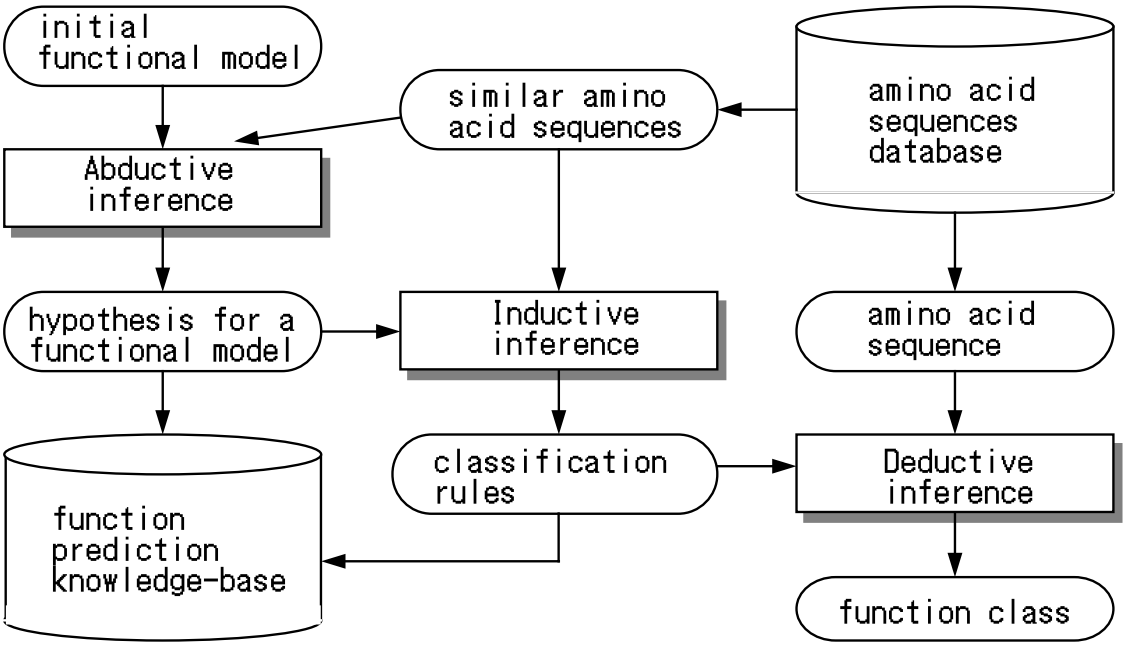

Figure 3: Multistrategy learning architecture

5 Knowledge Representation and Inference Algorithms

The knowledge-base for protein function prediction consists of the following knowledge sources (Figure 4):

(1) *functional models* - relationships between protein functions and the constituent functional elements using functional parameters

(2) *feature rules* - relationships between functional elements and its functional features using feature parameters

(3) *properties of functional parameters* - relationships among functional parameters described in concept hierarchies

(4) *properties of feature parameters* - relationships among feature parameters described in concept hierarchies.

These knowledge sources are inter-relatedly used in *Abductive inference*, *Inductive inference*, and *Deductive inference*.

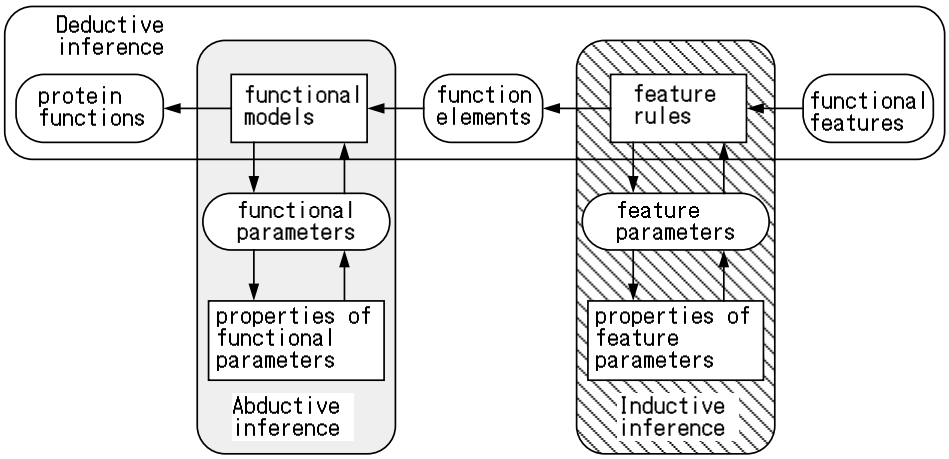

Figure 4: Knowledge structure of the function prediction knowledge-base

ABA (Analogy by Abstraction) [3] generates hypotheses for a functional model of a target protein by transforming a given initial functional model of a base protein. Since the algorithm ABA used in Abductive inference of the learning system is based on Horn clause, the learning system and the knowledge-base are described in Prolog. The algorithm ABA for hypothesis generation is as follows:

Step 1. Define a target literal

Describe a target literal as a goal clause to explain using the knowledge-base and hypotheses.

Step 2. Retrieve a base fact similar to the target literal

Find a fact clause whose literal is similar to the target literal. The similarity between laterals is defined as follows. Two literals L_1 and L_2 are said to be similar when the following conditions are satisfied: 1) predicates are the same, 2) arities of the predicates are equal, 3) all corresponding terms in the literals have each common superior concepts. A common superior concept of two terms, t_1 and t_2, is a common ancestor term `s`, which is defined as facts, $s(t_1)$ and $s(t_2)$, in

the knowledge-base. For two similar literals, an *analogy* is the correspondence of their terms, $\{\langle t_{11}, t_{21}\rangle, \langle t_{12}, t_{22}\rangle, \ldots\}$, where $t_{11}, t_{12}, \ldots$ are terms in L_1 and $t_{21}, t_{22}, \ldots$ are terms in L_2.

Step 3. Search for base rules to explain the base fact

Search for rule clauses which are able to prove a literal representing the base fact, and to explain the base fact using the rules found. The explanation of the base fact is a proof tree for the base literal using the given knowledge-base.

Step 4. Transform the base rules

First, transform the explanation of the base fact using the analogy obtained in *Step 2*. To do so, replace each term in the base explanation with the corresponding term in the analogy. Second, variablize all terms in the transformed explanation except for replaced terms, and instantiate the variablized explanation in order to get a detailed analogy between the target and the base. Finally, transform the base rules using the detailed analogy to get target rules.

Step 5. Verify the target rules

Add the target rules to the knowledge-base and prove the target literal as a goal. If the goal is proved, then determine the target rules as a hypothesis. If not, retract the target rules from the knowledge-base, then backtrack to *Step 4* in order to find another transformation, backtrack to *Step 3* to select another base rules, and backtrack to *Step 2* to select another base fact until the target rules are able to prove the target literal.

The algorithm of *Inductive inference* to refine classification rules is based on the following incremental refinement method:

Step 1. Select a certain class to refine classification rules.

Step 2. Select facts which contains positive examples and negative examples for the class as a training data set.

Step 3. If all the positive examples and the negative examples can be classified correctly, then stop. Otherwise, goto *Step 4*.

Step 4. Find the most specific generalization of the classification rules so that all the positive example can be classified correctly.

Step 5. Find the most general specialization of the classification rules so that all the negative examples can be classified correctly. If such a description cannot be found, then backtrack to *Step 4*.

Step 6. Let the generated rules be classification rules for the given class.

Applying the above procedure of Inductive inference for all classes to learn, we obtain a set of classification rules. The Steps *4* and *5* use the same specialization and generalization procedures using the concept hierarchies of constant terms as in ABA. The concept hierarchies must be given before or during the learning.

We use the resolution mechanism built in Prolog as the algorithm for Deductive inference.

Now, we explain the learning procedure for building a knowledge-base in Figure 3.

(1) *Initialization: describing initial functional models*

First we describe known functional models of proteins in the knowledge-base. The functional model for a protein consists of functional elements and corresponding functional features of its amino acid sequence.

(2) *Abductive inference: generate a target functional model*

The learning system generates a hypothesis for a functional model of a target protein from the

initial functional models. The hypothesis is generated by transforming base rules representing the functional model of the base protein using analogy.

(3) *Inductive inference: generate classification rules*
The system generates classification rules using the obtained hypothesis for the target functional model so as to correctly classify all positive and negative examples.

(4) *Deductive inference: test the classification rules*
Using the obtained classification rules, the system tests to classify proteins in amino acid sequences database with Deductive inference.

6 An Example

We have implemented a Prolog program based on the proposed method using algorithms described in Section 5 except for Inductive inference. In order to show the effectiveness of the method, we have applied the program to function prediction of proteins having amino acid sequences similar to *bacteriorhodopsin* (abbreviated as bR). bR is one of a few proteins whose structures and functions are well studied [1]. bR is a trans-membrane protein and has a function of a proton pump. It is known that the structure of bR has a hydrophilic center structure comprising seven alpha-helices and retinal as working material in it. Figure 5 shows its abstract functional model. One of the other proteins with sequences similar to bR is *halorhodopsin* (abbreviated as hR). hR transports chloride ion (Cl^-) instead of proton (H^+) in bR as an ion pump. In the following, we will explain the learning process of building a knowledge-base for the protein function prediction. A functional model of hR is generated by analogical reasoning from the functional model of bR and function classification rules to discriminate bR and hR using these functional models.

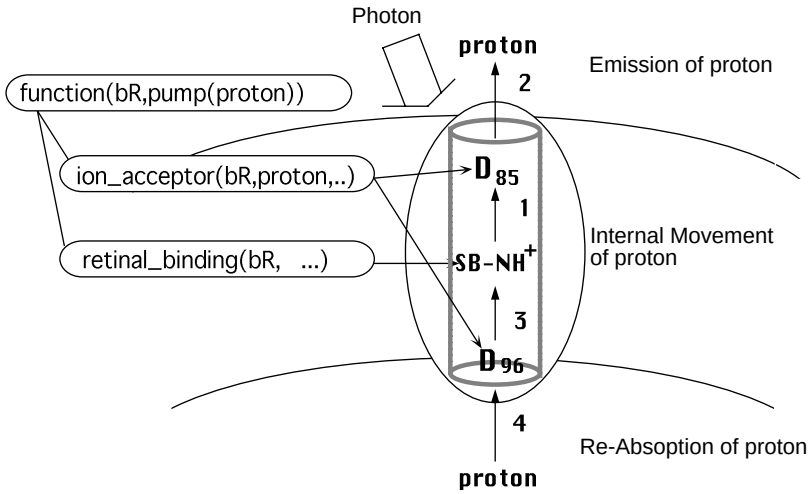

Figure 5: A functional model of bR

(1) *Initialization: describe the initial functional model of bR*
First we describe the initial functional model of bR as a proton pump, which is inferred from

its known structure (Figure 5) [1]. In the model, the function of proton pump is decomposed into three functional elements; two ion acceptors and a retinal binding. These ion acceptors are corresponded to a functional feature that two amino acid residues `D` with negative charge exist in `helix(3)`. The retinal binding is corresponded to a functional feature that an amino acid residue `K` for retinal binding exists in `helix(7)`. The helices from `helix(1)` to `helix(7)` are recognized by secondary structure prediction [8]. A part of Prolog description for the functional model of bR is shown below.

```
function(X, pump(proton)) :-
        ion_acceptor(X, proton, helix(3), Position1),
        ion_acceptor(X, proton, helix(3), Position2),
        Position1<Position2,retinal_binding(X, helix(7)).

ion_acceptor(X, Ion, helix(I), Position) :-
        trans_mem_sequences(X, SQ),member(NSQ, SQ, I),
        charge(Ion, C),opposite(C, AC),charge(Res, AC),
        string_member(Res, NSQ, P),in_membrane(P),length(NSQ, L),
        (member(I, [1,3,5,7]) -> Position=P ; Position is L-P+1).

in_membrane(Pos) :- Pos>3,Pos<20.

retinal_binding(X, helix(I)) :-
        trans_mem_sequences(X, SQ),member(NSQ, SQ, I),
        string_member('K', NSQ, P),in_membrane(P).
```

(2) *Abductive inference: generate a functional model of hR*

The function of hR is a `pump(chloride)`, while the function of bR is a `pump(proton)`. Since the amino acid sequence of hR is similar to that of bR, the structure of hR is inferred to be similar to bR [1]. The difference in the functions of hR and bR is caused by different transporting ions. If we assume that they have the same mechanism except for the difference of transporting ions, then we can infer the functional model of hR from that of bR using analogical reasoning. To execute the algorithm ABA, it only requires to describe the concept hierarchies for the correspondence of terms in the base and target of analogical reasoning. In this example, we define superior concepts; `protein` for `bR` and `hR`, and `ion` for `proton` and `chloride`. To obtain a valid functional model for hR, `helix(3)` of bR should be replaced with `helix(6)` using concept hierarchies for the terms `3` and `6`. We also define a superior concept `helix` for them. ABA program generates the following result for a hypothesis for the functional model of hR:

```
Goal = function(hR, pump(chloride))

Analogue = function(bR, pump(proton))

Base_rule = function(X, pump(proton)) :-
                ion_acceptor(X, proton, helix(3),Position1),
                ion_acceptor(X, proton, helix(3), Position2),
                Position1<Position2,retinal_binding(X, helix(7)).


Target_rule = function(X, pump(chloride)) :-
```

```
ion_acceptor(X, chloride, helix(3), Position1),
ion_acceptor(X, chloride, helix(6), Position2),
Position1<Position2,retinal_binding(X, helix(7)).
```

In the above execution of ABA program, the following concept hierarchies are used to correspond constant terms in the target literal and the base literal.

```
protein(bR). protein(hR). ion(proton).  ion(chloride).
helix(1).  helix(2).  helix(3).  helix(4).  helix(5).  helix(6).  helix(7).
```

(3) *Inductive inference: generate classification rules*
Next, the hypothesis generated for the functional model of hR is refined so that the functions of proteins having similar amino acid sequences can be correctly classified. The refinement is executed by the specialization and generalization of constant terms using the same concept hierarchies as in ABA. In the example here, Inductive inference is simulated by hand.

(4) *Deductive inference: apply classification rules*
By applying the obtained classification rules, we have succeeded to classify protein functions of hR as a chloride pump. Table 1 summarizes the functional features for the bacteriorhodopsin-like proteins, bR and hR.

Table 1: Functional features of bacteriorhodopsin-like proteins

protein	protein function	functional features
bR	proton pump	having *two* residues with *negative* charge in helix-3
hR	chloride pump	having *one* residue with *positive* charge in helix-3 and having *one* residue with *positive* charge in helix-6

7 Discussion

(1) The proposed method is effective for generating classification rules from a very few number of training examples. Instead of applying the proposed method using analogical reasoning to generate hypotheses of classification rules, it is difficult to generate the same classification rules by the direct applications of *Inductive inference* to amino acid sequences. Because the proposed method generates functional models for protein functions in top down manner, so obtained rules have abstract structures easy to understand for domain experts.
(2) The use of *Analogical reasoning* prunes meaningless generations of hypotheses in *Abductive inference*. Therefore, the proposed method improves the efficiency of learning. ABA generates a valid hypothesis for a target literal simply by variablizing constant terms in the explanation of the target literal and by instantiating the explanation without searching for generalizations and specializations of the terms.
(3) We have suggested that a knowledge-base building tool for the same kind of problems can

be constructed from the fundamental inferences in the proposed method. Since the proposed method consists of the fundamental inference methods, we can implement the proposed method with any combinations of previously developed algorithms.
(4) The algorithm ABA is closely related to EBG (Explanation based Generalization) and also to PDA (Purpose Directed Analogy). However, a major distinction of ABA from these methods is the utilization of concept hierarchies for constant terms as similarities between the base and the target of analogy.

8 Conclusion

The paper has described the method of multistrategy learning approach to build a knowledge-base for protein function prediction using functional feature analysis and an application of the method. In functional feature analysis, it is important to recognize functional features of amino acid sequences to characterize the functions of proteins. We have also implemented an multistrategy learning method to find functional features by analogical reasoning about the functional models. An example in which we build a knowledge-base to classify functions of proteins similar to bR shows the effectiveness of the proposed method. We have a plan to implement the algorithm of *Inductive inference*, and to improve the method to extend the knowledge-base for predicting protein functions of the other classes of proteins.

References

[1] Futai, M.(ed.) *Bio-membrane engineering* (in Japanese), Maruzen (1991)

[2] Hunter, L.(ed.) *Artificial Intelligence and Molecular Biology*, AAAI Press (1993)

[3] Ishikawa, T. and Terano, T. Analogy by Abstraction: Theory of Case Retrieval and Adaptation in Inventive Design Problems, *AAAI-93 CBR workshop* (1993)

[4] Ishikawa, T. and Terano, T. Using Analogical Reasoning to Predict a Protein Structure., *Proc. of Genome Informatics Workshop IV* (1993)

[5] Ishikawa, T., Mitaku, S., Terano, T., Hirokawa, T., Suwa, M., and Seah, B-C. Finding Functional Features of Proteins using Machine Learning Techniques., *Proc. of Genome Informatics Workshop '94*, pp.168-169 (1994)

[6] Ishikawa, T. and Terano, T. Generation of protein function prediction rules using multistrategy learning (in Japanese), *proceedings of IPSJ workshop*, 95-FI-36, pp.23-30 (1995)

[7] Michalski, R.S. and Tecuci, G.(eds.) *Machine Learning: A Multistrategy Approach Vol. IV*, pp.3-62, Morgan Kaufmann (1994)

[8] Mitaku, S. et al. Prediction methods for structures of trans-membrane proteins like Bacteriorhodopsin (in Japanese), Proteins-Nucleic acids-Enzymes, Vol.34, No.5, pp.518-527 (1989)

[9] Mitaku, S. It is hard to understand lives without recognition of hierarchical structures (in Japanese), J. of Physics Society Japan, Vol.50, No.4, pp.255-2 62 (1995)

[10] Nakamura, H and Nakai, K. *Introduction to computer for biotechnology* (in Japanese), Corona-sha (1995)

[11] Schulz, G. E. and Schirmer, R. H. (translated by Ohi, T.) *Proteins - Structures, Functions, and Evolution* (in Japanese), Kagaku-doujin (1980)