Original Paper

A Generalized Hidden Markov Model Approach to Transmembrane Region Prediction with Poisson Distribution as State Duration Probabilities

Takashi Kaburagi[†1] and Takashi Matsumoto[†1]

We present a novel algorithm to predict transmembrane regions from a primary amino acid sequence. Previous studies have shown that the Hidden Markov Model (HMM) is one of the powerful tools known to predict transmembrane regions; however, one of the conceptual drawbacks of the standard HMM is the fact that the state duration, i.e., the duration for which the hidden dynamics remains in a particular state follows the geometric distribution. Real data, however, does not always indicate such a geometric distribution. The proposed algorithm utilizes a Generalized Hidden Markov Model (GHMM), an extension of the HMM, to cope with this problem. In the GHMM, the state duration probability can be any discrete distribution, including a geometric distribution. The proposed algorithm employs a state duration probability based on a Poisson distribution. We consider the two-dimensional vector trajectory consisting of hydropathy index and charge associated with amino acids, instead of the 20 letter symbol sequences. Also a Monte Carlo method (Forward/Backward Sampling method) is adopted for the transmembrane region prediction step. Prediction accuracies using publicly available data sets show that the proposed algorithm yields reasonably good results when compared against some existing algorithms.

1. Introduction

The Hidden Markov Model (HMM) is one of the most successful tools for modeling time-series data sequences. A variety of applications of HMM have been presented, such as speech recognition [1], handwritten character recognition [2),3)], and biological data processing [4),5)]. HMM has also contributed to improved prediction accuracies for applications in transmembrane region prediction [6)–8)]. One reason for its success is the idea of defining the underlying dynamical system of states and regarding the observation as the output of these states with uncertainties. The performance of the model critically depends on the trajectories of the underlying states. Therefore, the transitions among states play an important role in designing a model for an application of interest.

One of the conceptual drawbacks of the standard HMM as applied to several classes of problems, including the problem addressed in this paper, is the fact that the duration d for which the hidden dynamics remains in a particular state s_i follows the geometric distribution

$$P(d \mid s_i) = {a_{ii}}^d, \tag{1}$$

where $a_{ij} := P(s_j \mid s_i)$ denotes the state transition probability from state s_i to s_j.

In several problems, including ours [8)], the duration does not follow such a geometric distribution. **Figure 1**, for instance, shows the histogram of the frequency of transmembrane region length observed in a data set described in Section 3.1, where the dotted line shows a geometric series with parameter $a_{ii} = 0.95$. Note that it is difficult to fit the histogram with a geometric distribution with any parameter [⋆1].

One solution is to consider the particular topology or grammar within an HMM framework to form clusters of states. The topology proposed in Ref. 6), for instance, has 12 submodels (clusters of states) connected with each other in a certain topology. Of the 12 submodels, the helix core consists of 25 states connected together in a feed-forward manner. The inside and outside loop models and helix cap models have their own topologies. Thus, the target duration can be represented by a mixture of geometric distributions and Bernoulli variables serving as their weights. The topology proposed in Ref. 8) has 17 states instead of 25 for the helix core model.

Another solution for the duration probability is to extend the HMM framework itself. The

†1 Waseda University

⋆1 The parameters in this section have been fitted using a maximum likelihood estimation.

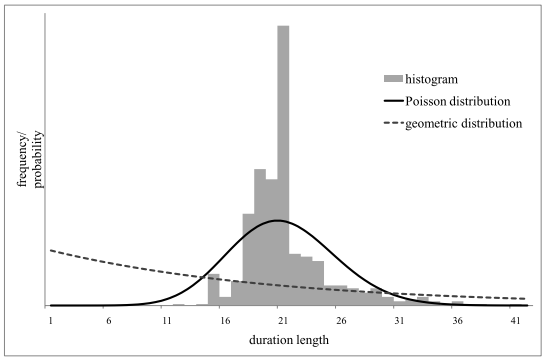

Fig. 1 State duration probability distribution. The horizontal axis denotes the length of the transmembrane region. The bar graph shows a histogram of transmembrane region length of a data set described in Section 3.1. The dotted line is an example of a geometric distribution describing the state duration probability in HMM. The solid line is an example of a Poisson distribution with mean 21.

Generalized Hidden Markov Model (GHMM), an extension of the HMM, has been employed to cope with the problem of duration probability [9]. In the GHMM, the state duration probability can be any discrete distribution, including a geometric distribution. This characteristic allows the GHMM to be used to design the state duration probability for the target application without using clusters of states. The GHMM is reported to yield better performance in speech recognition, handwritten character recognition, and gene searching in DNA sequences [9)–11)].

Here, we follow a method that employs a Poisson distribution as the state duration probability, as proposed in Ref. 10). The solid line in Fig. 1 is the Poisson distribution [⋆1]

$$P(d;l) = \frac{\exp(-l)l^d}{d!}, \qquad (2)$$

with parameter $l = 21$. As one can observe, it seems more natural to fit the target histogram with a Poisson distribution than with a geometric distribution.

In this paper, we present an algorithm utilizing GHMM for predicting transmembrane regions, with a state duration probability distribution based on a Poisson distribution.

1.1 Background of This Study

Transmembrane proteins have long been considered to be critical in understanding biological functions such as cell signaling, ion transport, and intercellular communications [12)–14)]. It has been reported that approximately 45% of the drugs in use today target G protein-coupled receptors (GPCRs) [15),16)], and some 20% to 30% of genes in an average genome are estimated to encode membrane proteins [17)]. Because of their biological and pharmaceutical importance, identification of transmembrane helices in membrane proteins is a priority. Although promising methods in X-ray crystallography and nuclear magnetic resonance (NMR) have begun to open avenues to the determination of these structures [18)–20)], the number of known three-dimensional structures remains small [6),21),22)]. Therefore, reliable algorithms to predict transmembrane protein structures would be very useful.

There are two basic methods of looking at protein structure predictions. One is to use algorithms based solely on the construction principles of proteins associated with the physicochemical properties of amino acids. No training is involved. In this method, windowed averages of physicochemical quantities are taken. There are several successful examples of algorithms of this type [23)–33)]. The other basic method is to collect data sets of known structures, extract their features, and use machine-learning algorithms to make predictions. Some improvements have been made in using this type of algorithm, but further advances are necessary to improve the reliability of predictions [6),7),34)–37)].

We used a novel machine-learning algorithm to predict protein structures and evaluated the reliability of the predictions. A machine-learning algorithm assumes that there are models and associated parameters behind the available data sets. Generally, the degree of success of a machine-learning algorithm depends on two factors: i) how well the model structure characterizes the target molecule from which the data was taken, and ii) how well the learning algorithm incorporates the available data sets. Among protein structure prediction problems, there are, in general, three important aspects in transmembrane structure prediction:

(i) The data is sequential with respect to a one-dimensional space variable, and a particular amino acid is correlated with other amino acids.
(ii) The data sets have uncertainties in that a particular structure may be observed to

⋆1 Since the state duration probability is a discrete probability defined in the range $d \geq 1$, although the probability in Fig. 1 may look like a Gaussian distribution, it cannot be one.

have different amino acid sequences.

(iii) The number of training data sets for learning is severely limited because of the difficulties associated with using X-ray crystallography or NMR for transmembrane proteins.

This paper considers a restricted class of transmembrane protein structure prediction problems instead of a general class of problems. Specifically, we assumed that a particular amino acid sequence is from a transmembrane protein, even though predictions as to whether the sequence is water-soluble or a transmembrane protein could have been attempted instead. The primary reason for this is that there are several very good tools available for such prediction problems [38]. The goal of this paper is, given an amino acid sequence, to predict transmembrane regions, i.e., to predict whether each amino acid belongs to a transmembrane region.

These problems are non-trivial because, as previously mentioned, so few transmembrane protein structures have been fully characterized. A finer model that captures the nature of a set of data would improve prediction accuracy, provided that a sufficient number of training data sets were available. Because there are so few available data sets, serious consideration is essential to make both the model structure and the associated learning algorithm as simple as possible without losing sight of the nature of the problem. Therefore, transmembrane protein structure prediction is a significant challenge for machine-learning approaches.

This paper proposes a novel algorithm for predicting the transmembrane regions in a given test amino acid sequence. Contributions of this paper are listed below.

(i) A generalized finite-state, stochastic dynamical system (GHMM) is utilized for predicting transmembrane regions. The model is applied to our previously proposed scheme reported in Ref. 8) and is evaluated in comparison with other well-known tools for predicting transmembrane regions.

(ii) The transmembrane regions are predicted by predicting the path of the inner stochastic dynamical states. To implement this prediction, a Monte Carlo method (Forward-Backward sampling) is employed.

(iii) The results reported in Section 3.3 suggest that our proposed prediction scheme yields reasonably good results.

1.2 Stochastic Dynamical System Approaches

The application of finite-state stochastic dynamical systems (also known as HMM) is very broad since its techniques are suitable for characterizing the nature of sequential data. These techniques have been used to solve a variety of problems in, for example, speech recognition and handwriting recognition. In the HMM framework, an unobserved state sequence $\{q_t\}_{t=1}^T$ is assumed to exist behind an observed sequence $\{\boldsymbol{o}_t\}_{t=1}^T$.

Approaches for predicting transmembrane regions based on HMMs have been successfully realized in such tools as TMHMM [6),37)] and HMMTOP [7)]. Krogh et al. defined seven types of states in TMHMM: loop cytoplasmic, cap cytoplasmic, helix core, cap non-cytoplasmic, short loop non-cytoplasmic, long loop non-cytoplasmic, and globular domains. A probability distribution of the 20 amino acids, which was learned from the training data set, was defined in each state, taking into account the grammar. Tusnady and Simon also proposed an HMM-based method in HMMTOP [7)]. This model employs five states (inside loop, inside helix tail, helix, outside helix tail, and outside loop). This algorithm focuses on the differences in the amino acid distributions in the structural parts, rather than on the amino acid distribution itself.

The performances of these methods are evaluated in Section 3.

2. Algorithm

In this section, the details of the proposed algorithm are presented. First, the general framework of GHMM is illustrated. Next, a more specific model used in the proposed algorithm is described. Then, the method to learn parameters from a training data set is described. Lastly, the method to predict transmembrane regions is explained.

2.1 The Model

First, we consider the general framework of GHMM employed in our proposed algorithm.

2.1.1 Observation Sequence $\{\boldsymbol{o}_t\}$

In this paper, we consider the two-dimensional vector trajectory $\boldsymbol{o}_t$ of length T associated with amino acids, instead of the 20 letter symbol sequences:

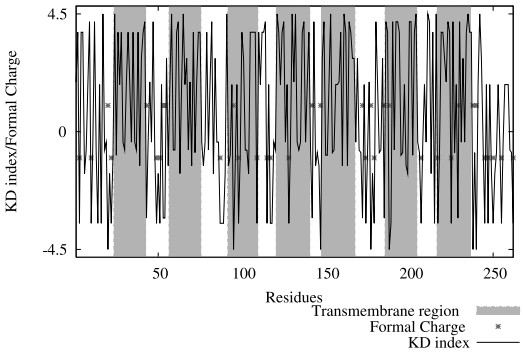

Fig. 2 An example of the two-dimensional observation of a transmembrane protein. The KD index plots are connected with lines in order to show changes with respect to residues. Zero charges are not shown for clarity. The shaded region is a transmembrane region, while the unshaded regions are loop regions.

$$\{\boldsymbol{o}_t := (o_t^1, o_t^2)\}_{t=1}^T$$
$$o_t^1 \in v_{k_1}^1, \quad o_t^2 \in v_{k_2}^2$$
$$k_1 = 1, \cdots, K_1, \quad k_2 = 1, \cdots, K_2.$$

The first component of output o_t^1 is the hydropathy index; the KD index is used in this paper [⋆1]. Even though the hydropathy index is real valued, there are only a finite number of values $v_{k_1}^1$ for the KD index between 4.5 and -4.5, with $K_1 = 17$. The second component o_t^2 is the formal charge associated with an amino acid [⋆2]. Similarly, there is only a finite number of formal charge values, i.e., $v_{k_2=1}^2 = +1$, $v_{k_2=2}^2 = 0$, and $v_{k_2=3}^2 = -1$ ($K_2 = 3$). **Figure 2** shows an example of the two-dimensional observation of a transmembrane protein.

A major consequence of considering these physicochemical indices instead of the 20 letter symbols is the fact that "nearness" among different amino acids can be taken into account. That is, two amino acids with a similar hydropathy index can be considered close to each other based on this particular metric. This allows "smoothing" to avoid overfitting problems.

⋆1 There may be better hydropathy indices than the KD index; as many as 80 different hydropathy indices have been proposed.

⋆2 Histidine can be assumed to have two possible formal charge values, depending on pH. The histidine formal charge will be assumed to be +1 in the experiment reported in Section 3. Since the number of histidines appears to be small in the data sets used in our experiment, our tentative assumption did not appear to have a significant effect on prediction performance.

2.1.2 Unobserved Sequence $\{q_t\}$

One way of taking into account the sequential nature of the problem, i.e., the fact that each amino acid is correlated with other amino acids, is to consider an unobserved auxiliary sequence $\{q_t\}$ of length T and to treat $\boldsymbol{o}_t$ as an *output* with uncertainty. This sequence $\{q_t\}_{t=1}^T$ is a trajectory of a finite-state inner stochastic dynamical system indexed by a one-dimensional parameter t. Here $q_t \in \{s_1, \cdots, s_N\}$, where s_i is the i-th state within an inner stochastic dynamical system, and N denotes the number of states.

2.1.3 Segment Length $\{L_i\}$

In GHMM, the unobservable state sequence $\{q_t\}_{t=1}^T$ can be segmented by its value [39]. Since the proposed algorithm utilizes a simple left-to-right topology, although generalization is possible, the following argument is described in the case of the employed topology. This topological constraint gives the number of segments to be N, which is the number of states. Note that there will be a one-to-one relation between the state s_i and the segmented state sequence Q_i. Using the length of the i-th segment, L_i, called duration, the state segment Q_i can be expressed as:

$$q_{\tau_i+1} = q_{\tau_i+2} = \cdots = q_{\tau_i+L_i} = s_i$$
$$Q_i := \{q_{\tau_i+1}, q_{\tau_i+2}, \cdots, q_{\tau_i+L_i}\}$$

where

$$\tau_i := \sum_{j=1}^{i-1} L_j.$$

Here, τ_i denotes the last position of the $(i-1)$-th state segment sequence. In the following description, $Q_i = s_i$ is equivalent to $q_{\tau_i+1} = q_{\tau_i+2} = \cdots = q_{\tau_i+L_i} = s_i$.

The same segmentation can be applied to the observation sequence as well. The segmented observation sequence $\boldsymbol{O}_i$ is a subset of the whole observation sequence $\boldsymbol{O} := \{\boldsymbol{o}_1, \boldsymbol{o}_2, \cdots, \boldsymbol{o}_T\}$, i.e., $\boldsymbol{O}_i := \{\boldsymbol{o}_{\tau_i+1}, \boldsymbol{o}_{\tau_i+2}, \cdots, \boldsymbol{o}_{\tau_i+L_i}\}$.

2.1.4 Likelihood Function

The likelihood can be obtained by marginalizing over all possible q_t:

$$P(\{\boldsymbol{o}_t\}_{t=1}^T \mid w, \mathcal{H}) = \sum_{\substack{\text{for all} \\ \text{possible } \{q_t\}}} P(\{\boldsymbol{o}_t\}_{t=1}^T, \{q_t\}_{t=1}^T \mid w, \mathcal{H}).$$

where w is a parameter set and $\mathcal{H}$ stands for the underlying model structure. To ensure the readability of this paper, we will omit the de-

pendency on w and $\mathcal{H}$.

The *joint probability distribution* of $\{\boldsymbol{o}_t, q_t\}_{t=1}^T$ is described by:

$$P(\{\boldsymbol{o}_t\}_{t=1}^T, \{q_t\}_{t=1}^T) = P(\{\boldsymbol{o}_t\}_{t=1}^T \mid \{q_t\}_{t=1}^T) P(\{q_t\}_{t=1}^T). \quad (3)$$

Equation (3) is in the general form of a stochastic dynamical system.

The first term of Eq. (3) is the emission probability, which can be described as:

$$\begin{aligned} &P(\{\boldsymbol{o}_t\}_{t=1}^T \mid \{q_t\}_{t=1}^T) \\ &= P(\{\boldsymbol{O}_t\}_{i=1}^N \mid \{Q_i\}_{i=1}^N) \\ &= \prod_{i=1}^N P(\boldsymbol{O}_i \mid Q_i) \\ &= \prod_{i=1}^N P(\{\boldsymbol{o}_t\}_{t=\tau_i+1:\tau_i+L_i} \mid Q_i) \\ &= \prod_{i=1}^N P(\{o_t^1\}_{t=\tau_i+1:\tau_i+L_i} \mid Q_i) \\ &\qquad P(\{o_t^2\}_{t=\tau_i+1:\tau_i+L_i} \mid Q_i) \\ &= \prod_{i=1}^N \prod_{t=\tau_i+1}^{\tau_i+L_i} P(o_t^1 \mid Q_i) P(o_t^2 \mid Q_i). \quad (4) \end{aligned}$$

The second term of Eq. (3) is the state transition probability, which can be described in the proposed model as:

$$\begin{aligned} &P(\{q_t\}_{t=1}^T) \\ &= P(Q_1) P(L_1 \mid Q_1) \\ &\quad \prod_{i=2}^N P(Q_i \mid Q_{i-1}) P(L_i \mid Q_i). \quad (5) \end{aligned}$$

Note that the state duration probability $P(L_i \mid Q_i)$ appears only in GHMM. On the other hand, in HMM, the state transition probability is defined as:

$$P(\{q_t\}_{t=1}^T) = P(Q_1) \prod_{i=2}^N P(Q_i \mid Q_{i-1}).$$

The performance of the model in GHMM critically depends on the design of this state duration probability. Therefore, the state duration probability should be carefully designed. The details of emission probabilities and state transition probabilities will be explained later in this section.

Figure 3 summarizes the probablistic relations among unobservable state segment sequence Q_i, segment length L_i, and observation segment sequence $\boldsymbol{O}_i$.

Schemes described by Eq. (3) are sometimes

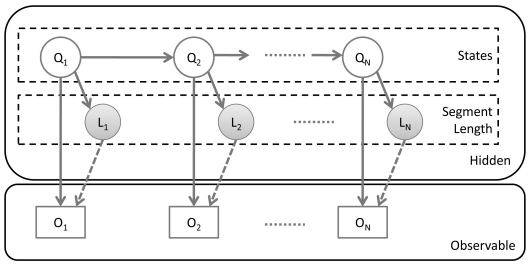

Fig. 3 The probabilistic relations among unobservable state segment sequence Q_i, segment length L_i, and observation segment sequence $\boldsymbol{O}_i$. The circled variables represent the hidden variables. The boxed variables represent the observable variables. The solid arrows show the probabilistic dependencies, while the dotted arrows show the deterministic dependencies between variables.

successful for nonlinear time-series prediction problems in which the inner dynamical system has an infinite number of states [40], handwriting recognition problems [2),3)], and online signature verification problems [41] in which the inner dynamical system has a finite number of states. In these three problem classes, index parameter t is *time*, whereas in a protein primary sequence, t stands for spatial *position* from the N-terminus.

2.2 The Proposed Model

Successful applications of machine learning algorithms crucially depend on the particular model structure chosen. The model structure must be carefully designed taking into account the specific purpose(s) of the prediction problem, as well as the available data sets. A researcher may wish to design a model structure that is as detailed as possible and takes into account many aspects of transmembrane proteins. However, because so few transmembrane protein structures are known, it would not be feasible to tune such detailed models with many delicate parameters. This is one aspect of the data fitting *versus* simplicity dilemma (Occam's razor).

Model $\mathcal{H}$ used in this paper consists of the following.

(i) The model $\mathcal{H}$ carries a fixed number of transmembrane regions $\mathcal{M}$ as a meta-parameter.

(ii) The number of states N is set in a deterministic manner by the value of $\mathcal{M}$.

(iii) Each state s_i carries a meta-parameter Z_i indicating which region the state belongs to. The region μ_v indicates a v-th transmembrane region, while λ_u in-

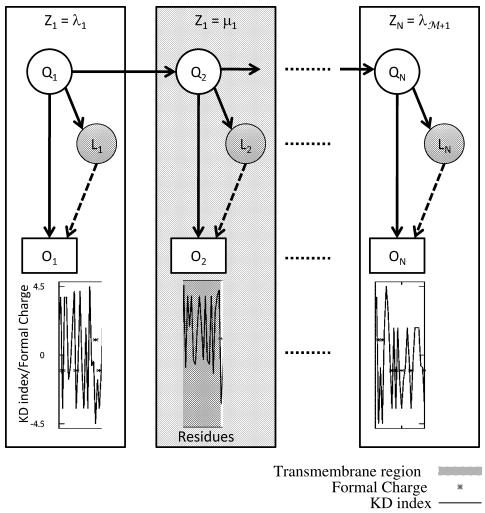

Fig. 4 The overall topology of the proposed model with the two-dimensional observation and the meta-parameter of the states. Each state is connected by a left-to-right topology. The plots show the segmented two-dimensional observation of Fig. 2. The bounding box represents the meta-parameter. The shaded region is a transmembrane region, while the unshaded regions are loop regions.

dicates a u-th loop region, i.e., $Z_i \in \{\lambda_1, \cdots \lambda_{\mathcal{M}+1}, \mu_1, \cdots, \mu_{\mathcal{M}}\}$.

(iv) The model $\mathcal{H}$ has an entirely "open-loop" structure where states are connected in a left-to-right topology. The meta-parameter of the connected states shows alternating connections of loop region and transmembrane region, i.e., $Z_1 = \lambda_1$, $Z_2 = \mu_1$, $Z_3 = \lambda_2$, $\cdots$, $Z_{N-1} = \mu_{\mathcal{M}}$, $Z_N = \lambda_{\mathcal{M}+1}$.

Figure 4 summarizes the model specification described above.

There is a variety of possible topologies for the HMM. The topology should be carefully examined in terms of two aspects: First, the topology needs to capture the structure of a given problem; and second, parameter learning associated with the topology should be feasible.

The most general topology would be "ergodic", i.e., every node is connected with every other node. While this can be flexible, it suffers from a difficulty in the learning phase because the number of parameters, which are associated with the Markov chain transition probabilities, can be extremely large, particularly when the number of training data sets is small. This topology has not been implemented in protein structure prediction problems to the best of our knowledge.

The second possible topology can contain several "closed loops" with specified constraints or grammar. The topologies proposed in Refs. 6) and 7) are among this class. The topology proposed in Ref. 6), for instance, has 12 submodels connected with each other in a certain topology, as mentioned earlier. Of the 12 submodels, the helix core consists of 25 states connected together in a feed-forward manner. The inside and outside loop models and helix cap models have their own topologies. Parameter learning appears to be nontrivial.

The third class of topology, among others, can be "open loop", which has no closed loops except for self loops. While this topology may not decrease the number of overall transition probability parameters, it significantly simplifies transition probability learning because there are only two possibilities at each state: stay in the same state or go to the next (right) state. It is this property that we take advantage of in this study. Note that with this third topology, the initial state and the final state are always distinct. Thus, from a biological point of view, this topology can be interpreted as a model structure in which each trajectory begins with an N-terminal and ends with a C-terminal.

2.2.1 Emission Probabilities

The emission probabilities of the hydropathy index and formal charge are defined as:

$$P(o_t^1 = v_{k_1}^1 \mid q_t = s_i) := b_{i,k_1}^1$$
$$P(o_t^2 = v_{k_2}^2 \mid q_t = s_i) := b_{i,k_2}^2$$
$$i = 1, \cdots, N$$
$$k_1 = 1, \cdots, K_1 \qquad k_2 = 1, \cdots, K_2,$$

where b_{i,k_1}^1 and b_{i,k_2}^2 satisfy the constraints $b_{i,k_1}^1 \in [0,1]$, $b_{i,k_2}^2 \in [0,1]$ and $\sum_{k_1=1}^{K_1} b_{i,k_1}^1 = 1, \sum_{k_2=1}^{K_2} b_{i,k_2}^2 = 1$.

In the formulation Eq. (4), emission probabilities $\{b_{i,k_1}^1\}$ and $\{b_{i,k_2}^2\}$ are assumed to be independent for the sake of simplicity, whereas in reality, they are not.

2.2.2 State Transition Probabilities

Since the proposed algorithm employs the left-to-right topology, state transition probability from state segment $Q_i = s_i$ to $Q_j = s_j$ is defined as:

$$P(Q_j = s_j \mid Q_i = s_i) := \begin{cases} 1 & j = i+1 \\ 0 & \text{otherwise} \end{cases}$$
$$i, j = 1, \cdots, N.$$

2.2.3 Initial State Probability

The probability of initial state segment Q_1 is defined as:

$$P(Q_1 = s_i) := \begin{cases} 1 & \text{if } i = 1 \\ 0 & \text{otherwise} \end{cases}$$
$$i = 1, \cdots, N.$$

2.2.4 State Duration Probability

The probability of state segment length L_i is defined as:

$$P(d = L_i \mid Q_i = s_i) := p_i(d)$$
$$i = 1, \cdots, N,$$

where $p_i(d)$ satisfies the constraints $p_i(d) \in [0, 1]$ and $\sum_{d=1}^{\infty} p_i(d) = 1$.

2.3 Learning

Here, the method to set GHMM parameter vector w using a training data set and hyperparameters is described.

Consider the following available training data sets:

$$\boldsymbol{D}_{train} := \{\boldsymbol{D}_h\}_{h=1}^{H}$$
$$:= \{\{\boldsymbol{o}_{h,t}, z_{h,t}\}_{t=1}^{T_h}, m_h\}_{h=1}^{H},$$

where $z_{h,t} \in \{\lambda_1, \cdots \lambda_{m_h+1}, \mu_1, \cdots, \mu_{m_h}\}$ is an annotation sequence which denotes the region associated with observation $\boldsymbol{o}_{h,t}$; m_h is the number of transmembrane regions; and H is the total number of available sequences for the training data set.

The proposed algorithm attempts to construct one model from one training data set. Thus, if a model $\mathcal{H}_h$ is given, only one training sequence $\boldsymbol{D}_h$ will correspond. This is one of the major differences in the learning scheme compared with the existing algorithms [6),7)]. The existing algorithms create only one model and use all of the available data to train the single model. One of the primary reasons for employing the proposed learning scheme is the availability of the data sets. The training data set used in the experiment described in this paper is well classified. Our preliminary experiments show that the proposed learning method performed better than a learning scheme that trains one model with all of the available data sets. When the training data set is not well classified, this may not be the best way to train the model. Further research into the modification of the learning scheme will be an interesting topic.

We chose not to use the Baum-Welch learning algorithm for two reasons. First, it often suffers from local minima. Second, we wanted to test our first trial parameter values so that our proposed structure would make sense. Of course, the learning scheme must be improved in various ways, including using Monte Carlo methods, which is a subject of ongoing research.

The learned parameter vector $\hat{w}_h$ consists of the following:

(1) The state duration probabilities $p_i(d)$
(2) The KD index emission probabilities $b^1_{ik_1}$
(3) The charge emission probabilities $b^2_{ik_2}$.

Step 1. Setting of meta parameters

Step 1.1. Setting $\hat{\mathcal{M}}_h$

For model $\mathcal{H}_h$, the number of transmembrane regions $\mathcal{M}_h$ is set to be the same as the number of transmembrane regions of the training data m_h:

$$\hat{\mathcal{M}}_h := m_h.$$

Step 1.2. Setting N_h

For model $\mathcal{H}_h$, the number of states N_h is set deterministically depending on the learned number of transmembrane regions $\hat{\mathcal{M}}_h$:

$$N_h := 2\hat{\mathcal{M}}_h + 1.$$

Step 1.3. Setting $\hat{Z}_{h,i}$

Consider a segmentation of $\boldsymbol{o}_{h,t}$ by its region $z_{h,t}$ whose value carries the information which denotes the region. Using the length of each segment L_i and $\tau_i = \sum_{j=1}^{i-1} L_j$,

$$\boldsymbol{O}_{h,i} := \{\boldsymbol{o}_{h,\tau+1}, \boldsymbol{o}_{h,\tau+2}, \cdots, \boldsymbol{o}_{h,\tau+L_i}\}$$
$$z_{h,\tau+1} = z_{h,\tau+2} = \cdots = z_{h,\tau+L_i}$$
$$i = 1, \cdots, N.$$

The meta data $\hat{Z}_{h,i}$ for state s_i is set as:

$$\hat{Z}_{h,i} := z_{h,\tau+1} = z_{h,\tau+2} = \cdots = z_{h,\tau+L_i}$$
$$i = 1, \cdots, N.$$

Step 2. State duration probability

As previously described, the performance critically depends on the design of the state duration probability. There are two reasons why the duration probability proposed in this study is a mixture of a Poisson distribution and a histogram over all available data sets. In order to explain the first reason, observe that the Poisson distribution as defined by (2) appears to be a reasonable model for the duration probability, as is demonstrated in Fig. 1. The mean of the Poisson distribution, $\hat{l}_i$, proposed in this paper

is set as the segment length L_i of a particular state s_i. Note that the segment length L_i is known from the training data set. It should be noted, however, that the mean of the Poisson distribution, L_i, in this case, is also the variance. This may give rise to a problem when L_i is large. Since there are several proteins with more than 200 residues in the loop region, it may not be appropriate to use the Poisson distribution alone for the duration probability. This paper attempts to solve this problem by mixing the Poisson distribution with another distribution. The second reason is to "regularize" the zero frequency problem of the histogram, i.e., when the histogram contains zero values. A mixture with a Poisson distribution could overcome this problem.

The state duration probability is set as:

$$p_i(d) := \begin{cases} (1-\alpha)Poisson(d \mid \hat{l}_i) + \alpha p_\lambda(d) \\ \qquad\qquad \text{if } Q_i \in \lambda \\ (1-\alpha)Poisson(d \mid \hat{l}_i) + \alpha p_\mu(d) \\ \qquad\qquad \text{if } Q_i \in \mu \end{cases}$$

$$Poisson(d \mid \hat{l}_i) := \frac{\exp(-l_i)\hat{l_i}^d}{d!}$$

$$\hat{l}_i := L_i.$$

Here, $p_\lambda(d)$ and $p_\mu(d)$ are histograms of the segment length over all available data sets. $p_\lambda(d)$ is for a loop region, while $p_\mu(d)$ is for a transmembrane region.

α is a hyperparameter indicating a mixture ratio. Our preliminary experiments show that the prediction results moderately depend on the selection of α. In the experiments presented in this paper, α is set empirically. Hierarchal learning of this hyperparameter α from the training data set is an interesting problem, and is one topic of our future research.

Step 3. KD index emission probabilities

Step 3.1. Learning $b^1_{ik_1}$ (Flooring)

For each state s_i, let

$$\tilde{b}^1_{ik_1} := \frac{n(\{KD\}, k_1; Z_{h,i}) + \beta}{\sum_{k_1=1}^{K_1} n(\{KD\}, k_1; Z_{h,i}) + \beta},$$

where
$n(\{KD\}, k_1; Z_{h,i})$:= number of residues with KD index k_1 within $Z_{h,i}$,
and β is a hyperparameter.

Step 3.2. Learning $b^1_{ik_1}$ (Smoothing)

The smoothing operation is performed to avoid overfitting problems:

$$\hat{b}^1_{ik_1} := \frac{1}{\zeta_i} \sum_{j:|k_1-k_j|\le 1} v_j \tilde{b}^1_{ik_1}$$

$$v_j := \int_{x=|k_1-k_j|-\frac{1}{2}}^{x=|k_1-k_j|+\frac{1}{2}} \exp\left(\frac{-x^2}{2\pi\sigma^2}\right) dx,$$

where σ is a hyperparameter and ζ_i is a normalization constant. Our preliminary experiments show that the prediction results slightly depend on the selection of σ, which was set empirically.

Note that Step 3.2 would have been impossible if the nearness between two amino acids were not defined, which is the case when considering the sequence of the 20 letter symbols. Also note that there are four amino acids out of 20 that have the same KD index (-3.5): ASP, ASN, GLU, and GLN.

Step 4. Charge emission probabilities

Learning $b^2_{ik_2}$

For each state s_i, let

$$\hat{b}^2_{ik_2} := \frac{n(\{Charge\}, k_2; Z_{h,i}) + \gamma}{\sum_{k_2=1}^{K_2} n(\{Charge\}, k_2; Z_{h,i}) + \gamma},$$

where
$n(\{Charge\}, k_2; Z_{h,i})$:= number of residues with formal charge k_2 within $Z_{h,i}$,
and γ is a hyperparameter.

Here, all steps of the learning phase are summarized below.

Learning

For $h = 1, \cdots, H$, the parameters $\hat{w_h}$ associated with model $\mathcal{H}_h$ are trained by the training data $\boldsymbol{D}_h$ as follows:

(1) Set meta parameters
  (a) Set the number of transmembrane regions $\mathcal{M}_h$
  (b) Set the number of states N_h
  (c) Set the meta data $Z_{h,i}$ of state s_i
(2) Learn state duration probabilities $p_i(d)$
(3) Learn KD index emission probabilities $b^1_{ik_1}$
(4) Learn charge emission probabilities $b^2_{ik_2}$

2.4 Predictions

Let $\boldsymbol{D}_{test} := \{\boldsymbol{o}^{test}_t\}_{t=1}^{T_{test}}$ be a test sequence.

Note that in the prediction phase, the number of transmembrane regions m and the associated annotation sequence $\{z_t\}$ are *unknown*. As previously mentioned, the state sequence $\{q_t\}$ is also unobservable. The goals of the prediction phase are to 1) predict m, and 2) predict transmembrane regions. To achieve these goals, the proposed algorithm is designed to have two steps in the prediction phase: 1) select the best model, and 2) predict annotation sequence $\{z_t\}$. The details of each step are described here.

Step 1. Selection of the best model

The goal of this step is to predict the number of transmembrane regions m. This goal is achieved by first selecting the best model $\mathcal{H}_{\hat{h}}$ that explains the given test sequence $\boldsymbol{D}_{test}$. $\mathcal{H}_{\hat{h}}$ is the model that gives the largest likelihood in Eq. (6):

$$\begin{aligned}\hat{h} &:= \operatorname*{argmax}_h \left[P\left(\boldsymbol{D}_{test}, q_{T_{test}} = s_N \mid \hat{w}, \mathcal{H}_h\right)\right] \\ &= \operatorname*{argmax}_h \Big[\sum_{\substack{\text{all possible} \\ \text{paths } \{Q_t\}_{t=1}^{T_{test}-1}}} \\ &P\left(\{O_t^{test}\}, \{Q_t\}, Q_{T_{test}} = q_N \mid \hat{w}, \mathcal{H}_h\right)\Big].\end{aligned}$$

After obtaining the best model $\mathcal{H}_{\hat{h}}$, the number of transmembrane regions of the test sequence m is predicted as:

$\hat{m} := \mathcal{M}_{\hat{h}}$.

Step 2. Prediction of transmembrane regions

The prediction of transmembrane regions is achieved by predicting the annotation sequence $\{z_t\}$. As previously described, each annotation z_t indicates which region the associated residue $\boldsymbol{o}_t$ belongs to. The annotation sequence $\{z_t\}$ can be obtained from the state sequence $\{q_t\}$. Several methods are reported for predicting a state sequence $\{q_t\}$. Here, a Monte Carlo method based on Forward-Backward (FB) sampling is employed. Fast mixing has been reported as the main advantage of this approach [42].

FB sampling is a method for sampling a state sequence q_t from a probability distribution $P(\{q_t\}_{t=1}^T \mid \{o_t\}_{t=1}^T)$. Consider a transformation using the state transition of GHMM:

$$\begin{aligned}&P(\{q_t\}_{t=1}^T \mid \{o_t\}_{t=1}^T) \\ &= P(Q_N \mid \{\boldsymbol{o}_t\}_{t=1}^T) P(L_N \mid Q_N, \{\boldsymbol{o}_t\}_{t=1}^T) \\ &\prod_{i=1}^{N-1} P(Q_i \mid Q_{i+1}, \{\boldsymbol{o}_t\}_{t=1}^T) \\ &\cdot P(L_i \mid Q_i, \{\boldsymbol{o}_t\}_{t=1}^T).\end{aligned}$$

Generally the state sequence q_t can be sampled inductively as:

$$q_t = \begin{cases} P(q_t \mid \{\boldsymbol{o}_t\}) & \text{if } t = T \\ P(q_t \mid q_{t+1}, \{\boldsymbol{o}_t\}) & \text{otherwise} \end{cases}$$

The model specification restricts the last state q_T to be s_N.

The next step is to sample the length of the last segment L_N. The sample can be obtained as:

$L_N \sim P(L_N \mid Q_N, \{\boldsymbol{o}_t\})$.

This probability distribution can be defined by a polynomial distribution using $l = 1, 2, \cdots, t$:

$$\begin{aligned}&p_t(l) := P(L_i = l \mid Q_i = s_i, \{\boldsymbol{o}_t\}) \\ &p_t(l) = \frac{f_t(i,l)}{f_t(i)} \\ &f_t(i) = P(o_{1:t}, Q_t = s_i, Q_{t+1} \neq s_i) \\ &f_t(i,l) = P(o_{t-l+1:t} \mid s_i, l) P(l \mid s_i) \\ &\qquad \sum_{j=1}^{N} f_{t-l}(j) a_{ji}.\end{aligned}$$

With the obtained sample of segment length L_N, the state sequence Q_N is set as:

$$\begin{aligned}&Q_N := \{q_{T-L_N+1}, \cdots, q_{T-1}, q_T\} \\ &q_T = q_{T-1} = \cdots = q_{T-L_N+1} = s_N.\end{aligned}$$

The topological constraints deterministically give the state $q_{T-L_N} := s_{N-1}$. Draw a sample of L_{N-1} by:

$L_{N-1} \sim P(L_{N-1} \mid Q_{N-1}, \{\boldsymbol{o}_t\})$.

These steps are inductively repeated until the state sequence reaches Q_1. The FB sampling step can be summarized as follows:

FB Sampling

(1) Set $q_N := s_N$
(2) Sample L_N
(3) Set $q_T = q_{T-1} = \cdots = q_{T-L_N+1} = s_N$
(4) Set $t = T - L_N$; $i = N - 1$
(5) Set $q_t = q_{t+1} - 1$
(6) Sample L_i
(7) Set $q_t = q_{t-1} = \cdots = q_{t-L_i+1}$
(8) Set $t = t - L_i$; $i = i - 1$
(9) Repeat (5) $\sim$ (8) if $t > 0$

After obtaining a sample of state sequence $\{q_t\}$, the annotation sequence is set as:

For each t

$z_t := Z_{q_t}$.

All steps of the prediction phase can be sum-

marized as follows:

Prediction

For test data $\boldsymbol{D}_{test} := \{\boldsymbol{O}_t^{test}\}$:

(1) Select the best model by:

$$\hat{h} := \underset{h}{\operatorname{argmax}} \left[P\left(\boldsymbol{D}_{test}, Q_{T_{test}} = q_N \mid \hat{w}, \mathcal{H}_h\right)\right]$$

(2) Predict transmembrane region by predicting $\{z_t^*\}_{t=1}^{T_{test}}$

3. Evaluation

In this section, we report the evaluation results of the novel algorithm we used. The results are summarized in **Table 1**.

In order to perform experiments, appropriate data sets must be obtained. Currently, one of the most difficult problems in protein structure prediction in general, and in transmembrane protein structure prediction in particular, is the difficulty in obtaining appropriate data sets for experiments. We used two publicly available data sets: one was collected by Möller, et al. [43], and the other by Kernytsky, et al. [44].

The accuracy of the predictions of our algorithm, as to whether particular amino acids were from a transmembrane region, is discussed below.

For comparison, using the same test data sets, we also tested the performance of TMHMM [6] ⋆1, HMMTOP [7] ⋆2, and SOSUI [38] ⋆3, which are three well-known transmembrane structure prediction tools.

3.1 Data Sets

Here, we describe the details of the data sets used in our experiments. One is the data set collected by Möller, et al., which is a well-characterized transmembrane protein data set [43]. This data set will be called Dataset1 in this paper. The other is the data set collected by Kernytsky, et al., which is a benchmarking data set [44]. This data set will be called Dataset2 in this paper. The sequences were downloaded from their websites ⋆4. For training parameters, sequences from Dataset1 were used. For evaluation, sequences from Dataset2 were used.

Annotations for the sequences in Dataset1 have been updated since they were collected from the SwissProt database, which was released in the year 2000. To cope with this change, we updated the annotations and sequences by searching the UniProt database by ID or accession number.

In order to validate the performance of the proposed algorithm, sequences from Dataset2 were used [44]. Dataset2 contains 2247 sequences, but without descriptions of origins or annotations. Since the proposed algorithm targets only transmembrane proteins, we were required to select *only* transmembrane proteins.

In order to select transmembrane proteins out of the 2247 sequences in Dataset2, we ran a FASTA search against the entire UniProt database. We found 128 complete matches that were annotated as transmembrane proteins in UniProt. All 128 sequences were used for testing.

Of the amino acid sequences in Dataset1 and Dataset2, those with the following clear annotations are used for our experiment:
DOMAIN CYTOPLASMIC, DOMAIN MATRIX, DOMAIN EXTRACELLULAR, DOMAIN INTERMEMBRANE, DOMAIN PERIPLASMIC, and TRANSMEM, for which we have interpreted CYTOPLASMIC, MATRIX, EXTRACELLULAR, INTERMEMBRANE, and PERIPLASMIC as loop segments with TRANSMEM as a transmembrane segment.

The number of sequences used for training and testing is shown in **Table 2**. Thus, using the two data sets described above, training data sets and test data sets were selected as follows.

Training data set

244 sequences of Dataset1 were used for training.

Test data set

128 sequences of Dataset2 were used for testing.

Table 1 Accuracies of transmembrane region predictions.

Methods/tools	$n(TP)$	$n(FN)$	$n(FP)$	accuracy
Proposed	591	23	33	90.9%
Ref. 8)	581	33	24	90.7%
TMHMM	563	51	28	87.1%
HMMTOP	580	34	47	86.8%
SOSUI	544	70	29	83.9%

⋆1 http://www.cbs.dtu.dk/services/TMHMM-2.0/
⋆2 http://www.enzim.hu/hmmtop/
⋆3 http://bp.nuap.nagoya-u.ac.jp/sosui/

⋆4 Data set collected by Möller, et al.:
ftp://ftp.ebi.ac.uk/pub/databases/testsets/transmembrane
Data set collected by Kernytsky, et al.:
http://cubic.bioc.columbia.edu/services/tmh_benchmark/

Table 2 Number of sequences of data sets used for evaluation.

m	Training	Testing
1	69	34
2	17	9
3	19	8
4	38	18
5	13	8
6	18	16
7	28	11
8	6	6
9	3	1
10	8	5
11	1	0
12	21	11
13	1	0
14	1	1
15	1	0
Total	244	128

3.2 Evaluation Criteria

Reference 43) points out the existence of ambiguous borders of transmembrane regions in the reference annotation. Therefore, Ref. 45) suggests that some deviation of the prediction from the reference annotation must be tolerated.

The tolerances mentioned in the literature differ. For instance, reference Ref. 37) describes that sharing *five residues* with the reference annotation could be considered correct. Reference 45) states that the predicted region must share at least *nine residues* with the reference annotation, which is a little less than half of the 20 residues expected for a transmembrane region to be considered correct. We will follow Ref. 45) at least in this study since Ref. 45) is one of the earlier attempts at evaluating various transmembrane protein structure prediction tools in equivalent settings.

The evaluation criteria of transmembrane region prediction follows the method described in Ref. 45). In order to define the performance criterion, consider:

True Positive (TP) Segments.

A TP segment must share *at least nine residues* with a transmembrane region of the reference annotation. The following shows this concept schematically, where "T" stands for an amino acid within a transmembrane region, while "-" stands for an amino acid in a loop region.

```
Annotated  -------TTTTTTTTTTTTTTT--------
Predicted  -----TTTTTTTTTTTTTTTT---------
```

False Negative (FN) Segments.

An FN segment is a transmembrane region that is not predicted. This is shown schematically by:

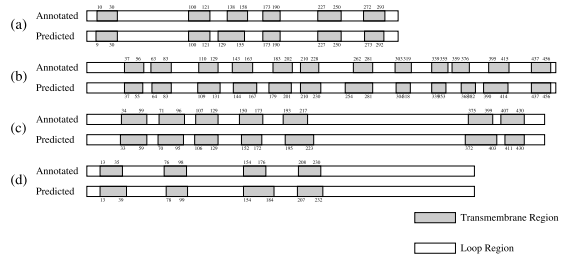

Fig. 5 Typical transmembrane region prediction results. Numbers denote number of residues from the N-terminus.

```
Annotated  -------TTTTTTTTTTTTTTT--------
Predicted  -----------------------------
```

False Positive (FP) Segments.

An FP segment is a predicted transmembrane region that is not a transmembrane region in the reference protein test set. This is shown schematically by:

```
Annotated  -----------------------------
Predicted  -------TTTTTTTTTTTTTTT--------
```

Note:

Each predicted transmembrane region should correspond to only one reference transmembrane region. This excludes the possibility of double counting TP segments. For instance, the following prediction has one TP segment and one FN segment instead of two TP segments:

```
Annotated  -----TTTTTTTTT-TTTTTTTTTT-----
Predicted  -----TTTTTTTTTTTTTTTTTTTT-----
```

Accuracy of transmembrane region prediction is defined by:

$$\textit{Transmembrane region prediction accuracy} := \left(1 - \frac{n(FN) + n(FP)}{n(TP) + n(FN)}\right),$$

where $n(TP)$, $n(FN)$, and $n(FP)$ denote the numbers of True Positive segments, False Negative segments, and False Positive segments, which we presume are the criteria used in Möller, et al. [45]. However, the equation is not explicitly written.

3.3 Results

The results are stated in Table 1. The proposed GHMM method gave:

$n(TP) = 591$, $n(FN) = 23$, $n(FP) = 33$, and Transmembrane region prediction accuracy = 90.9%.

Figure 5 illustrates some of the prediction results using the proposed algorithm. As one can see from Table 1, the proposed algorithm performs reasonably well.

Precise comparisons with other prediction algorithms are difficult because the sequences used for their training could have been different. For comparison purposes, however, we tested the same test sequences against three well-known web-based tools for predicting transmembrane helices.

4. Conclusions

We proposed a scheme to predict transmembrane regions utilizing a Generalized Hidden Markov Model (GHMM). The experimental results reported in Section 3.3 suggest that the GHMM scheme performed reasonably well.

Although it produced favorable results, the algorithm has several drawbacks and a number of aspects that need to be improved, as described below.

(i) The design of the state duration probability critically governs the performance of the algorithm. The experiment reported in this paper utilizes a mixture of a histogram and a Poisson distribution. Selection of another distribution is expected to lead to further improvement.

(ii) When the probability distribution landscape is not simple, a one-time parameter estimation, including the Baum-Welch method, as well as the algorithm reported in this paper, has limited success. The proposed learning algorithm is too simplistic, and a more advanced procedure is called for. Parameters, hyperparameters, and states can be inferred via a Bayesian framework where the Monte Carlo method can be utilized [40]. This is the subject of our ongoing research.

(iii) Sidedness (interior/exterior) can be predicted in situations where formal charge trajectories could be more important than the present problems.

Acknowledgments We would like to express our appreciation to Mr. Takayuki Ohnishi of Tokyo Medical and Dental University, and Mr. Yohei Nakada, Mr. Takahiro Hamada, Mr. Hiroto Sasaki, and Mr. Kazuhiro Ushida of Waseda University for their advice. This research has been partially supported by The Open Research Center Project of the Japanese Ministry of Education, Culture, Sports, Science and Technology. We would also like to thank the reviewers for their comments.

References

1) Rabiner, L.R.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, *Proc. IEEE*, Vol.77, No.2, pp.257–286 (1989).
2) Yasuda, H., Takahashi, K. and Matsumoto, T.: A discrete HMM for online handwriting recognition, *Int. J. Pattern Recognition and Artificial Intelligence*, Vol.14, No.5, pp.675–688 (2000).
3) Sasaki, H., Nakada, Y., Kaburagi, T. and Matsumoto, T.: Bayesian Angle Information HMM with a von Mises Distribution and its Implementation using a Bayesian Monte Carlo Method, *Proc. European Symposium on Time Series Prediction*, Finland, Otaniemi, pp.29–38 (2007).
4) Sonnhammer, E.L.L., Eddy, S.R. and Durbin, R.: Pfam: A Comprehensive Database of Protein Domain Families Based on Seed Alignments, *Proteins*, Vol.28, No.3, pp.405–420 (1997).
5) Krogh, A., Brown, M., Mian, I.S., Sjölander, K. and Haussler, D.: Hidden Markov Models in Computational Biology, Applications to Protein Modeling, *J. Mol. Biol.*, Vol.235, No.5, pp.1501-1531 (1994).
6) Krogh, A., Larsson, B., von Heijne, G. and Sonnhammer, E.L.L.: Predicting transmembrane protein topology with a hidden Markov model: Application to complete genomes, *J. Mol. Biol.*, Vol.305, No.3, pp.567–580 (2001).
7) Tusnady, G.E. and Simon, I.: Principles governing amino acid composition of integral membrane proteins: Application to topology prediction, *J. Mol. Biol.*, Vol.283, No.2, pp.489–506 (1998).
8) Kaburagi, T., Muramatsu, D. and Matsumoto, T.: Transmembrane Structure Predictions with Hydropathy Index/Charge Two-Dimensional Trajectories of Stochastic Dynamical Systems, *J. Bioinformatics and Computational Biology*, Vol.5, No.3, pp.669–692 (2007).
9) Juang, B.H. and Rabiner, L.R.: Hidden Markov Models for Speech Recognition, *Technometrics*, Vol.33, No.3, pp.251–272 (1991).
10) Russell, M. and Moore, R.: Explicit modelling of state occupancy in hidden Markov models for automatic speech recognition, *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '85*, Vol.10, pp.5–8 (1985).
11) Kulp, D., Haussler, D., Reese, M.G. and Eeckman, F.H.: A generalized hidden Markov model for the recognition of human genes in DNA, *Proc. Int. Conf. Intell. Syst. Mol. Biol.*,

St. Louis, Missouri USA, pp.134–42 (1996).
12) Hettema, E.H., Distel, B. and Tabak, H.F.: Import of proteins into peroxisomes, *Biochim. Biophys. Acta*, Vol.1451, No.1, pp.17–34 (1999).
13) Patil, C. and Walter, P.: Intracellular signaling from the endoplasmic reticulum to the nucleus: the unfolded protein response in yeast and mammals, *Curr. Opin. Cell. Biol.*, Vol.13, No.3, pp.349–355 (2001).
14) Le Borgne, R. and Hoflack, B.: Protein transport from the secretory to the endocytic pathway in mammalian cells, *Biochim. Biophys. Acta*, Vol.1404, No.1-2, pp.195–209 (1998).
15) Marchese, A., George, S.R., Kolakowski, L.F., Jr., Lynch, K.R. and O'Dowd, B.F.: Novel GPCRs and their endogenous ligands: Expanding the boundaries of physiology and pharmacology, *Trends Pharmacol. Sci.*, Vol.20, No.9, pp.370–375 (1999).
16) Drews, J.: Drug Discovery: A Historical Perspective, *Science*, Vol.287, No.5460, pp.1960–1964 (2000).
17) Wallin, E. and von Heijne, G.: Genome-wide analysis of integral membrane proteins from eubacterial, archaean, and eukaryotic organisms, *Protein Sci.*, Vol.7, No.4, pp.1029–1038 (1998).
18) Blundell, T.L. and Mizuguchi, K.: Structural genomics: an overview, *Prog. Biophys. Mol. Biol.*, Vol.73, No.5, pp.289–295 (2000).
19) Caffrey, M.: Membrane protein crystallization, *J. Struct. Biol.*, Vol.142, No.1, pp.108-132 (2003).
20) Wüthrich, K.: The way to NMR structures of proteins, *Nature Struct. Biol.*, Vol.8, No.11, pp.923–925 (2001).
21) Chen, C.P. and Rost, B.: State-of-the-art in membrane protein prediction, *Appl. Bioinformatics*, Vol.1, No.1, pp.21–35 (2002).
22) Zhou, C., Zheng, Y. and Zhou, Y.: Structure prediction of membrane proteins, *Genomics Proteomics Bioinformatics*, Vol.2, No.1, pp.1–5 (2004).
23) Kyte, J. and Doolittle, R.F.: A simple method for displaying the hydropathic character of a protein, *J. Mol. Biol.*, Vol.157, No.1, pp.105–132 (1982).
24) Eisenberg, D., Weiss, R.M. and Terwilliger, T.C.: The helical hydrophobic moment: a measure of the amphiphilicity of a helix, *Nature*, Vol.299, No.5881, pp.371–374 (1982).
25) Klein, P., Kanehisa, M. and DeLisi, C.: The detection and classification of membrane-spanning proteins, *Biochim. Biophys. Acta*, Vol.815, No.3, pp.468–476 (1985).
26) von Heijne, G.: Membrane protein structure prediction. Hydrophobicity analysis and the positive-inside rule, *J. Mol. Biol.*, Vol.225, No.2, pp.487–494 (1992).
27) Nakai, K. and Kanehisa, M.: A knowledge base for predicting protein localization sites in eukaryotic cells, *Genomics*, Vol.14, No.4, pp.897–911 (1992).
28) Persson, B. and Argos, P.: Topology prediction of membrane proteins, *Protein Sci.*, Vol.5, No.2, pp.363–371 (1996).
29) Cserzö, M., Wallin, E., Simon, I., von Heijne, G. and Elofsson, A.: Prediction of transmembrane alpha-helices in prokaryotic membrane proteins: the dense alignment surface method, *Protein Eng.*, Vol.10, No.6, pp.673–676 (1997).
30) Juretic, D., Zucic, D., Lucic, B. and Trinajstic, N.: Preference functions for prediction of membrane-buried helices in integral membrane proteins, *Comput. Chem.*, Vol.22, No.4, pp.279–294 (1998).
31) Pasquier, C., Promponas, V.J., Palaios, G.A., Hamodrakas, J.S. and Hamodrakas, S.J.: A novel method for predicting transmembrane segments in proteins based on a statistical analysis of the SwissProt database: the PRED-TMR algorithm, *Protein Eng.*, Vol.12, No.5, pp.381–385 (1999).
32) Jayasinghe, S., Hristova, K. and White, S.H.: Energetics, stability, and prediction of transmembrane helices, *J. Mol. Biol.*, Vol.312, No.5, pp.927–934 (2001).
33) Deber, C.M., Wang, C., Liu, L.P., Prior, A.S., Agrawal, S., Muskat, B.L. and Cuticchia, A.J.: TM Finder: A prediction program for transmembrane protein segments using a combination of hydrophobicity and nonpolar phase helicity scales, *Protein Sci.*, Vol.10, No.1, pp.212–219 (2001).
34) Jones, D.T., Taylor, W.R. and Thornton, J.M.: A model recognition approach to the prediction of all-helical membrane protein structure and topology, *Biochemistry*, Vol.33, No.10, pp.3038–3049 (1994).
35) Rost, B., Fariselli, P. and Casadio, R.: Topology prediction for helical transmembrane proteins at 86% accuracy, *Protein Sci.*, Vol.5, No.8, pp.1704–1718 (1996).
36) Fariselli, P. and Casadio, R.: HTP: a neural network-based method for predicting the topology of helical transmembrane domains in proteins, *Comput. Appl. Biosci.*, Vol.12, No.1, pp.41–48 (1996).
37) Sonnhammer, E.L.L., von Heijne, G. and Krogh, A.: A Hidden Markov Model for Predicting Transmembrane Helices in Protein Sequences, *Proc. 6th Int. Conf. Intell. Syst. Mol. Biol.*, Canada, Montreal, pp.175–182 (1998).
38) Hirokawa, T., Boon-Chieng, S. and Mitaku,

S.: SOSUI: Classification and secondary structure prediction system for membrane proteins, *Bioinformatics*, Vol.14, No.4, pp.378–379 (1998).
39) Murphy, K.P.: Hidden semi-Markov models (HSMMs), unpublished notes available at www.ai.mit.edu/murphyk/ (2002).
40) Matsumoto, T., Nakajima, Y., Saito, M., Sugi, J. and Hamagishi, H.: Reconstructions and predictions of nonlinear dynamical systems: A Hierarchical Bayesian Approach, *IEEE Trans. Signal Processing*, Vol.49, No.9, pp.2138–2155 (2001).
41) Muramatsu, D. and Matsumoto, T.: An HMM On-line Signature Verifier Incorporating Signature Trajectories, *Proc. Int. Conf. on Document Analysis and Recognition*, Scotland, Edinburgh, pp.438–442 (2003).
42) Scott, S.L.: Bayesian Methods for Hidden Markov Models Recursive Computing in the 21st Century, *J. American Stat. Association*, Vol.97, No.457, pp.337–351 (2002).
43) Möller, S., Kriventseva, E.V. and Apweiler, R.: A collection of well characterized integral membrane proteins, *Bioinformatics*, Vol.16, No.12, pp.1159–1160 (2000).
44) Kernytsky, A. and Rost, B.: Static benchmarking of membrane helix predictions, *Nucleic Acids Research*, Vol.31, No.13, pp.3642–3644 (2003).
45) Möller, S., Croning, M.D.R. and Apweiler, R.: Evaluation of methods for the prediction of membrane spanning regions, *Bioinformatics*, Vol.17, No.7, pp.646–653 (2001).

(Received July 2, 2007)
(Accepted November 23, 2007)
(Released March 26, 2008)
(Communicated by *Shigeyuki Oba*)
(Paper version of this article can be found in the IPSJ Transactions on Bioinformatics, Vol. 49, No.SIG5(TBIO4), pp.1–14.)

Takashi Kaburagi was born in 1980. He received his B.S. and M.E. degrees in electrical engineering from Waseda University, Japan in 2003 and 2005, respectively. He is a Ph.D. student in the Department of Electrical Engineering and Bioscience, Waseda University. He is a member of the IPSJ and JSBi.

Takashi Matsumoto was born in 1944. He received his BSEE from Waseda University, Tokyo, Japan, M.S. in Applied Mathematics from Harvard University, Cambridge, Massachusetts, and Ph.D. from Waseda University, Tokyo, Japan. His current interests include sequential Monte Carlo implementations of online Bayesian learning with application to practical problems such as change detection, face detection/tracking/recognition, biometric authentication, event detection, online pattern classification, and reinforcement learning. He also researches machine learning problems for biological data including membrane protein structure predictions, gene regulatory network predictions as well as clustering. He has coauthored several books, including Bifurcations (Springer-Verlag, 1993), and Frontiers of Statistical Science Vol.4 (Iwanami Shoten, 2005). Dr. Matsumoto held visiting positions at UC Berkeley (1977–1979) and Cambridge University, UK (2003–2004). He is a member of the IEEE (fellow), IPSJ, IEICE and the Japanese Society for Bioinformatics.