공학석사 학위논문

Simultaneous Modeling Hierarchy of GPCR Proteins with Deep Learning in a Single Metric Space

딥러닝 기반 단일 거리 공간 내 GPCR 단백질군 계층 구조의 동시적 모델링 기법

2019 년 8 월

서울대학교 대학원

컴퓨터공학부

이 태 헌

Simultaneous Modeling Hierarchy of GPCR Proteins with Deep Learning in a Single Metric Space

지도 교수 김 선

이 논문을 공학석사 학위논문으로 제출함
2019년 06월

서울대학교 대학원
컴퓨터공학부
이 태 헌

이태헌의 석학학위논문을 인준함
2019년 06월

위 원 장 박 근 수 (인)

부위원장 김 선 (인)

위 원 전 화 숙 (인)

Abstract

Simultaneous Modeling Hierarchy of GPCR Proteins with Deep Learning in a Single Metric Space

Taeheon Lee
Computer Science and Engineering
The Graduate School
Seoul National University

G protein-coupled receptors (GPCRs) belong to diverse families of proteins that can be defined at multiple levels. Computational modeling of GPCR families from the sequences has been performed separately at each level of family, sub-family, and sub-subfamily. However, relationships between classes are ignored in these approaches as they process the information in the sequences with a group of disconnected models.

In this work, we propose a deep learning network to simultaneously learn representations in the GPCR hierarchy with a unified model and a loss term to express hierarchical relations in terms of distances in a single embedding space. The model introduces a method to learn and construct shared representations across hierarchies of the protein family. In extensive experiments, we showed

that hierarchical relations between sequences are successfully captured in our model in both of technical and biological aspect. First, we showed that phylogenetic information in the sequences can be inferred from the vectors by constructing phylogenetic tree using hierarchical clustering algorithm and by quantitatively analyzing the quality of clustering results compared to the real label information. Second, inspection on embedding vectors is demonstrated to be a effective first step toward an analysis of GPCR proteins by showing that biologically significant sequence features can be revealed from multiple sequence alignments on clustering results on embedding vectors. Our work showed that simultaneous modeling of protein families with multiple hierarchies is possible.

Keywords : G protein-coupled receptors, hierarchical structure, embedding space, representation learning, deep learning
Student Number : 2017-23643

Table of Contents

Abstract i

Chapter Ⅰ. Introduction 1

1.1 Background 1

1.2 Motivation 3

Chapter Ⅱ. Methods 7

2.1 Data Preparation 7

2.1.1 Dataset 7

2.1.2 Data representation 7

2.2 Model architecture 8

2.2.1 Feature extractor with CNN 8

2.2.2 Embedding layer 8

2.2.3 Output layer 9

2.3 Loss function 10

2.3.1 Softmax loss 10

2.3.2 Center loss 10

2.3.3 Overall loss 12

2.4 Training procedure 13

2.5 Evaluation metric 14

2.5.1 Silhouette score 14

2.5.2 Adjusted mutual information score 15

Chapter Ⅲ. Results 17

3.1 Evaluation on hierarchical structure 17

3.1.1 Preservation of distances 17
3.1.2 Phylogenetic tree reconstruction 20
3.1.3 Quantitative evaluation on clustering results 21
3.2 Sequence analysis with embedding vectors 26
3.2.1 Technical analysis 26
3.2.2 Biological analysis 28
3.3 Classification accuracy 30

Chapter Ⅳ. Conclusion 32

References 35

초록 39

List of Figures

Figure 1 Overview of the proposed method 6

Figure 2 Illustration of loss terms 12

Figure 3 Distance matrix and hierarchical clustering results overlaid on columns 19

Figure 4 Phylogenetic tree drawn from the embedding vectors... 22

Figure 5 Evaluated scores of embedding vectors 24

Figure 6 t-SNE visualization of embedding vectors and example of inspections on the sequences at varying scale 24

Figure 7 Visualization of discovered motif logos from phylogenetic tree generated from hierarchical clustering results..... 27

List of Tables

Table 1 Selected hyperparameter for model architecture 9

Table 2 Selected hyperparameters for each training phase.. 14

Table 3 Classification accuracy assessed for our model and three models from DeepFam 31

Chapter 1

Introduction

1.1 Background

G protein-coupled receptor (GPCR) is the largest transmembrane protein family[1, 2] and one of the most extensively investigated drug targets[3, 4, 5]. GPCR is of a hierarchical class structure, represented by family, subfamily and sub-subfamily level classes. This structure was constructed following the phylogeny of the proteins[2, 6]. Analyzing the characteristics from the sequences regarding this structure lies at the heart of GPCR studies[1, 7, 8, 9]. Thus, approaches based on machine learning techniques, such as hierarchical classification and clustering, on the class structure have been widely explored[10, 11, 12]. These methods have been successful in modeling GPCR fairly accurately. However, we are yet to know how to model family, subfamily and sub-subfamily simultaneously, thus existing methods had to model GPCR at each of family hierarchies separately. For example, a top-down approach for hierarchical classification presented in 2007 trained multiple classifiers and selected the most suitable classifiers at each hierarchical level[10]. Likewise, PCA-GPCR arranged distinct classifiers dedicated to a specific portion of GPCR classes[11].

Aforementioned approaches inevitably employed series of separate steps to deal with the features in the class structure, since the representations used in the methods cannot reveal distinctive and unified features across the class hierarchies. These methods used representations derived from the low level features such as k-mer frequency vectors[13] or physiochemical properties of amino acid characters at each position[10, 11]. Although these features can be useful in describing the global properties such as hydrophobicity or electric charge, these vectors have limitations in capturing significant sequence patterns, or motifs, for representing the sequences. As a result, processing complex features throughout the hierarchy levels as a whole is nearly impossible with these representations. In the sense that GPCR class hierarchy was constructed using complex features including phylogenetic traits, ligand types and their functions, these approaches may not provide research opportunities to inspect which sequence features determine the relations between the GPCR proteins. Moreover, robust inspections on proteins, for example comparison between sequence clusters at different hierarchy levels, cannot be done with existing methods since comparing models at different hierarchies is not feasible. Since protein families are annotated hierarchically with the help of experiments to reflect the evolutionary history, inspections on relations between sequences and classes are really important in the protein family studies. In this regard, it is important to construct comprehensive representations of GPCR proteins with hierarchical features inclusively incorporated.

1.2 Motivation

Recently, methods for constructing efficient representations of the biological sequences have been widely researched [14], as we cannot inspect the information in the sequences without the help of sophisticatedly designed representations. In particular, with the advance in deep learning techniques, neural networks have been extensively adopted, since neural networks are capable of extracting complex features in the data. Among them, convolutional neural networks (CNN) are widely used to discover motifs in the dataset[15, 16]. Furthermore, representation learning approach to represent the motifs in the protein sequences was introduced[17] and there were studies to embed sequences into more significant vectors in the aspect of metric space. In these deep metric learning based sequence analyses, components of training phases are designed to make distances in embedding space more meaningful. For instances, siamese neural network for biological sequences was introduced, where alignment distances between sequences can be directly inferred from the embedding vectors of the given sequences[18]. However, up to our knowledge, deep learning based approaches for modeling hierarchical relations in the sequences have not been widely investigated yet.

In this work, we present a novel method to simultaneously learn and represent the comprehensive features across the hierar-

chies. First, we propose a deep learning network to process hierarchical features in GPCR proteins at once. Our method adopted a loss function based on metric learning approaches to make distances in embedding space represent hierarchical relations. As a result, the embedding function was devised to incorporate significant features at all hierarchical levels into one vector where further machine learning analysis such as clustering and classification can also be performed and support the analysis of the protein family. In a series of experiments, we showed the efficacies of our approach in two aspects. In terms of hierarchy, phylogenetic trees of the sequences can be inferred from the distances in representations generated from our model. This suggests that distances in the embedding space was shown to be a strong indicator of the relations between the classes. Technically, inspection on the sequences can be designed in multiple-scale using distances in the embedding space, from coarse-grained to fine-grained resolution. Here, inspections at coarse-grained level corresponds to family-level and fine-grained level refers to sub-subfamily level investigations. These inspections can lead to investigation on sequences in biological aspect. We demonstrated examples of these experiments by aligning sequences in the same clusters of the hierarchical clustering results. In such experiments, we showed that biologically significant motifs in the GPCR proteins are well-represented in each clusters. In addition, we investigated how those motifs are generalized or narrowed along clusters at different hierarchical levels. Moreover, in the experiments of GPCR classification, we showed that our method still achieved good

classification power compared to the state-of-the-art GPCR classifiers in all hierarchical levels.

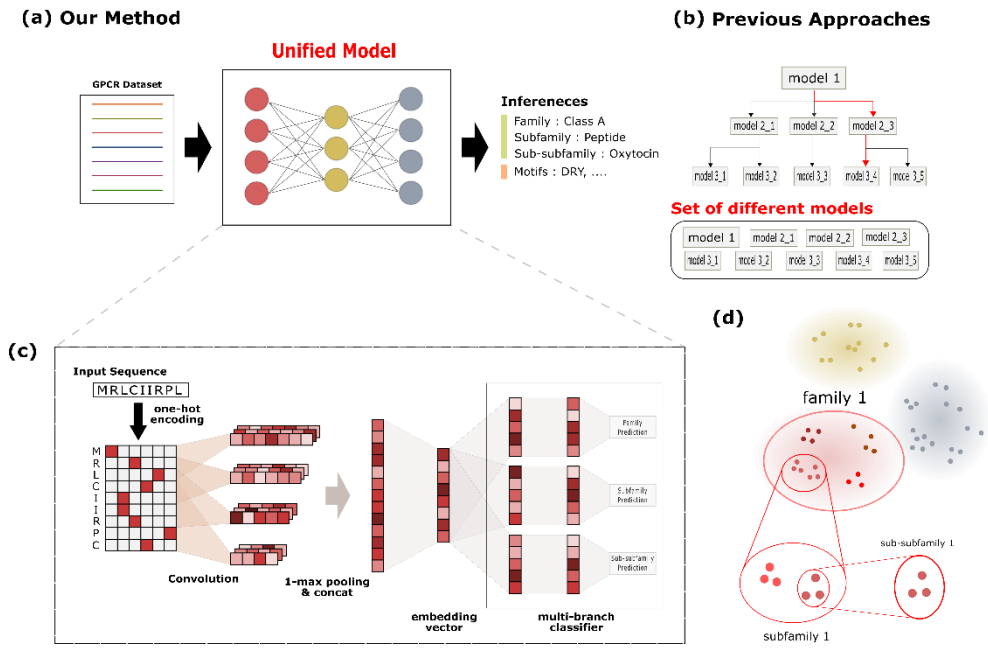

Figure 1 : Overview of the proposed method. **(a)** Our model process hierarchical structure in GPCR with a unified model without introducing additional models. **(b)** Previous approaches arranged multiple set of models at each hierarchical levels. **(c)** Illustration of the proposed method. Input sequence is converted into one-hot encoding vector before fed into neural network. Motif searching convolutional neural network at the first layer builds representations from the presence of motifs. Embedding vectors are generated by dense layer connected from motif features. After that, MLP based three branches classifiers is connected to the embedding vector. **(d)** Schematic illustration of hierarchically embedded vectors. Representations of sequences with the similar hierarchical class information are closely placed in embedding space.

Chapter 2

Methods

2.1 Data Preparation

2.1.1 Dataset

The GPCR sequences were retrieved from BIAS-PROFS GPCR dataset provided by Davies et al[10]. In this dataset, sequence labels are annotated in a hierarchical manner, from family level to sub-subfamily level. For convenience in training and testing, classes with fewer than 10 sequences were removed, resulting in 87 sub-sub-families and 8222 sequences. Train and test dataset were prepared in a 10-fold cross validation manner. Training sequences were used for training the model and the experiment outcomes given in the results section are merely constructed from the test dataset.

2.1.2 Data representation

Inputs to our model are originally in form of amino acid character sequences. To enable computations on such inputs, one-hot encoding scheme is widely adopted in Bioinformatics, where every amino acid position is represented as an one-hot vector [16, 19, 20]. Encoding of amino acid character was conducted following the IUPAC protein codes. In addition, every sequence is padded with zeros to a

fixed length, as CNN requires input vectors of same size.

2.2 Model architecture

The overall figure for the architecture adopted in this work is illustrated in Figure 1. Detailed description of each layer in the architecture is provided in this section.

2.2.1 Feature extractor with CNN

Architectures of the neural network were derived and modified from DeepFam[16]. In DeepFam, variable length convolutional filter was used with 1-max pooling, inspired from DeepBind[15]. Especially, from the experiments in DeepFam and DeepBind, this structure was proven to be successful in finding motifs of variable lengths. Since GPCR protein families are known for their highly conserved structural regions, our model exploits this architecture to effectively extract features from common motifs in the sequences. After that, outputs from the convolutional filters are flattened together and passed to the next layer.

2.2.2 Embedding layer

The next layer after the convolutional layer is embedding layer, fully connected to the previous layer with l2 normalization operator. This layer serves as an embedding function that generates a representative vector of the input sequence in a lower dimensional space. In the

sense that flattened layer from the feature extractor encodes presence of certain sequence patterns, vector representations of input sequences encode the information in biological motifs.

2.2.3 Output layer

Next, branches of Multilayer Perceptron (MLP) classifier is following the embedding layer. A technique of separating branches in a single neural network, with each branch dedicated to a domain-specific task, was proposed in Multi-Domain Network (MDNet)[21] to learn shared features across the multiple domains. In our architecture, three branches are used, each corresponding to family, subfamily and sub-subfamily level classification task. For each branch, MLP classifier with one hidden layer is used and ReLU function is used as an activation function of the hidden layer. Table 1 contains the value of hyperparameters used for the model architecture.

Table 1 : Selected hyperparameter for model architecture

hyperparameter	values
list of convolutional filter lengths	[8, 12, 16, 20, 24, 28, 32, 36]
list of number of filters for each length	[256, 256, 256, 256, 256, 256, 256, 256]
number of nodes in embedding layer	15
number of nodes in hidden layer of MLP	15

2.3 Loss function

2.3.1 Softmax loss

Cross-entropy loss with softmax function is one of the most frequently referenced loss function in machine learning. In general, softmax function is used to generate probability distribution of candidate labels from the output layer of the network. Most cases, softmax function is combined with cross-entropy loss in supervised learning to enforce classifiers to output higher probability on desired labels. This is effective for classifiers in learning separable representations between different classes. Likewise, we also employed this function as a part of the loss to empower neural network to learn separable features in input sequences based on class labels.

2.3.2 Center loss

Center loss has been proposed in Wen et al's work[22] to complement the softmax loss function. Although softmax loss is practical enough to separate features between classes, it lacks ability to learn compact representations of data within a single class. In other words, feature space built from softmax loss is not metrically well-constructed in that distances between feature vectors of the data do not impose any implications. To address more sound feature space with neural network, Wen et al proposed additional loss term to a softmax loss that minimizes the distances between data points within a class.

Center loss can be stated in following form:

$$L_c = \sum_{i=1}^{n} \left\| d(x_i) - \mu_{C_{x_i}} \right\|_2 \tag{2.1}$$

In exploiting the above loss function, mean vectors should be updated simultaneously with parameters being updated. In Wen et al's work, mean vectors are calculated based on the images in mini-batch based fashion[22] as considering vector representations of the whole dataset, generally comprising of 50K to 200M images for computer vision, is computationally exhaustive. However, number of sequences in the training data does not exceed 10K in general for the protein families. Therefore, we updated mean vector of each class based on feature vectors from the whole dataset. Furthermore, we configured a margin[23] for each hierarchy and use it as a class boundary of the class so that clear distinction is made between classes from different hierarchy. In conclusion, our center loss is in following form:

$$L_c = \max\left(\sum_{i=1}^{n} \left\| d(x_i) - \mu_{C_{x_i}} \right\|_2, m \right) \tag{2.2}$$

where $\mathbf{m}$ denotes the margin value of the class. Due to the margin value, center loss will stop updating parameters toward the class center once the deep feature comes inside a class boundary.

Figure 2 illustrates how above loss terms are updating the embedding vectors.

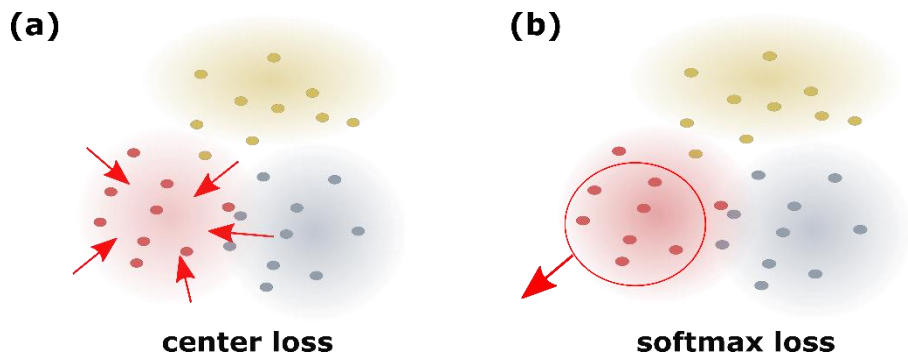

Figure 2 : Illustration of loss terms. (a) Center loss updates weights in the neural network to make representations of data in a same class compactly clustered. (b) Softmax loss makes representations between different classes more separable.

2.3.3 Overall loss

Combining cross-entropy loss L_S from the classifiers and center loss for each hierarchy, overall loss function can be stated in following equation.

$$L_c = \sum_{j \in S} \omega_j L_{S_j} + \lambda_C \left(\sum_{i \in S} \omega_i L_{C_i} \right) \tag{2.3}$$

In the above equation, S denotes the set of hierarchy levels and loss function is stated as a weighted sum of losses from each hierarchy. To balance between center loss and softmax loss, λ_C was introduced as a weight of center loss[22].

2.4 Training procedure

As we are training a network with information from three hierarchical levels, smart way to combine information is required. For that purpose, we compose training procedure with three different phases. These phases are structured with an idea borrowed from transfer learning[19].

In transfer learning, given some tasks, system for solving this problem is trained starting from the knowledge pre-acquired from relevant tasks. Likewise, we devised our network to focus on loss values from one level, from family-level to sub-subfamily level, in each phase. Phases of training are controlled by weight parameters. In selecting the weight for each phase, to avoid catastrophic forgetting, the phenomenon where previously acquired knowledge is abruptly diminished when neural network is facing a new loss function, we empirically searched the weight value on validation dataset. Resulting parameter value is describe in Table 2.

Table 2 : Selected hyperparameters for each training phase. Each phase is focusing on family, subfamily, sub−subfamily level each. About the notation in the table, λ_c denotes weight on center loss, m is for a class boundary for each level and ω_C and ω_S specifies weight for the loss terms from each level in center loss terms and softmax loss respectively.

variable		phase 1	phase 2	phase 3
λ_C (center loss weight)		0.01 (first epoch : 0)	0.1	0.5
m	Family	1.8		
	Subfamily	0.6		
	Sub−subfamily	0.1		
ω_C	Family	0.8	0.1	0.05
	Subfamily	0.15	0.8	0.1
	Sub−subfamily	0.05	0.1	0.85
ω_S	Family	0.8	0.2	0.1
	Subfamily	0.15	0.7	0.15
	Sub−subfamily	0.05	0.1	0.75

2.5 Evaluation metric

To assess the effectiveness of the method, we performed several quantitative evaluations on results from our embedding vectors. In this subsection, explanations on evaluation metrics used in result section is explained.

2.5.1 Silhouette score

Silhouette score measures the average value of the silhouette coef−ficients calculated from each data. Silhouette coefficient is generally

adopted to assess the relative quality of the clusters by evaluating the tightness and separability of data points given clustering labels[24]. For a given sample, this metric compares two distances, distances to data points that lies within a same cluster and distances to data points of nearest cluster. Silhouette coefficient for a data point i is stated as follows:

$$s(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}} \tag{2.4}$$

In the above equation, $a(i)$ denotes the mean distance between the given sample and other samples of the same cluster and $b(i)$ is the mean distance between the sample and the data points in the nearest cluster. Silhouette score can be calculated by taking an average on above silhouette coefficients, resulting in values range between -1 and 1. If intracluster data points are closely placed compared to data points of other clusters, silhouette score will be positive and close to one, indicating that the data points well match the cluster information. Otherwise, silhouette score of negative value implies the poor correspondence between data points and cluster results.

2.5.2 Adjusted mutual information score

Adjusted mutual information (AMI) score is used to evaluate the quality of clustering results compared to the ground truth class labels. This score utilizes information theoretic perspectives to compare the

results from clustering algorithms. AMI score is defined as follows for given two set of clusters, U and V.

$$AMI(U,V) = \frac{I(U,V) - E\{I(U,V)\}}{max\{H(U),H(V)\} - E\{I(U,V)\}} \qquad (2.5)$$

In the above equation, $I(U,V)$ is the mutual information between two results and $H()$ is the information entropy measured for each cluster result. In AMI score, additional term $E\{I(U,V)\}$ is introduced to compensate for agreements between two clusters arouse by chance. This index calculates the proportion of information shared between two clustering results.

For assessing evaluation metrics introduced in this section, we used the function provided by scikit-learn packages in Python[25].

Chapter 3

Results

In this section, experiment results from testing embedding vectors are presented. We first evaluated the quality of embedding vectors in terms of hierarchical structure. Further, we interpreted the results in technical and biological aspects.

3.1 Evaluation on hierarchical structure

Our method produces one embedding space for GPCR families at family, subfamily, and sub-subfamily. Thus, our goal in this experiment was to test how good the embedding space was in terms of three criteria:

(1) Preservation of distances between sequences at multiple levels
(2) Phylogenetic tree reconstruction based on distances in the embedding space
(3) Quantitative evaluation on clustering results

3.1.1 Preservation of distances

We first transformed GPCR sequences into embedding vectors and performed further analyses. Distance matrix was generated using

pairwise euclidean distances of the sequences. On the distance matrix, clustering analyses, such as hierarchical clustering algorithm, Unweighted pair group method with arithmetic mean (UPGMA) [26], were performed on the column vectors. Then, columns were ordered based on the clustering results whereas rows were sorted according to family, subfamily, sub-subfamily class labels. Visualization of distance matrix is given in Figure 3. Three-line color bands on the columns and rows were drawn on the figure to show the relevance between distances and hierarchical class relations.

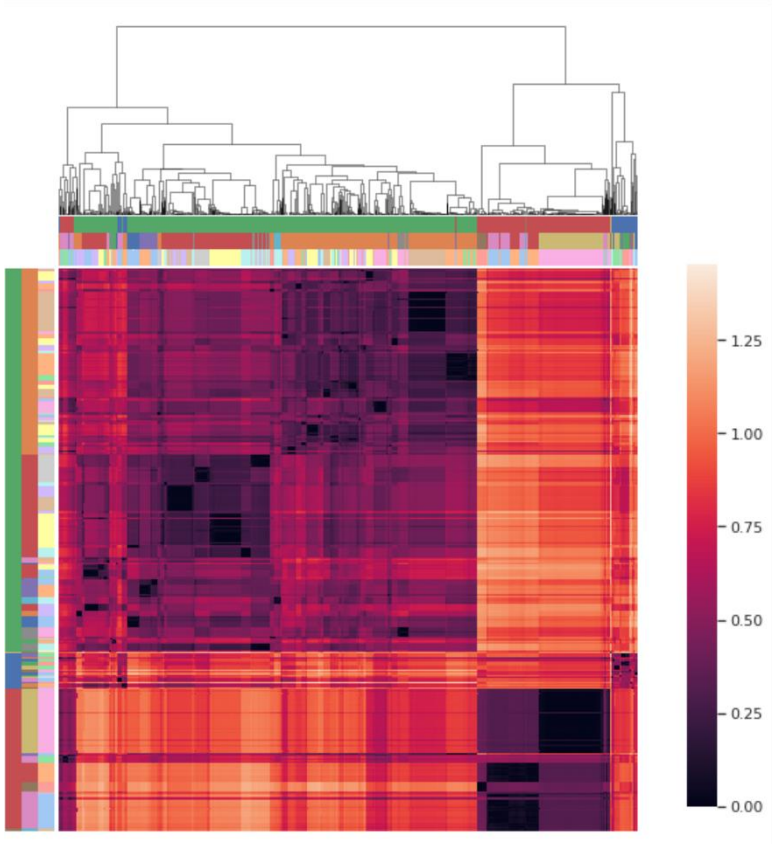

Figure 3 : Distance matrix and hierarchical clustering results overlaid on columns. Colors in the matrix indicate the pairwise distance between the sequences. Color bands on the left side of the matrix and the upper side of the matrix represent the class information of corresponding sequence of that row or column.

Hierarchical relations between sequences are apparent in the distance matrix illustrated in Figure 3. Overall pairwise distances between the sequences represented in different colors as a heat map and the heat map distance matrix shows that there clearly exist three distinctive distance relations among GPCR sequences, the brightest one (0.875 ~ 1.25), the middle-range one (0.375 ~ 0.875) and the darkest one (0.0 ~ 0.375). These distinctions in color level make separation between data points more clearly visible. Comparing boundary regions with true label information represented with color bands on columns and rows, color distinctions in distance matrix correctly correspond to the label information of sequences. Moreover, hierarchical relations between sequences are successfully demonstrated in terms of euclidean distance as three color levels well represent the family, subfamily and sub-subfamily label relation between sequences.

3.1.2 Phylogenetic tree reconstruction

Based on the distances in the embedding vectors, a phylogenetic tree was constructed using the neighbor-joining clustering method [27]. In the phylogenetic tree, every family-level label is represented by a distinct color. A branch in the tree was assigned a color corresponding to the family when more than 70% of the leaves of the branch belong to a specific family. In addition, as we have done in generating the distance matrix, additional figure of overlaying three-

layer color rings with colors corresponds to each layer in the hier-archy. The colored phylogenetic tree is shown in Figure 4. In the phylogenetic trees, sequences belonging to the same family labels are clustered in close positions. In the second and the third rings that represent subfamily and sub-subfamily respectively, sequences be-longing to subfamily and sub-subfamily were also positioned in close locations in the tree. In summary, we created a single embedding space and we showed that a phylogenetic tree based on the single embedding space grouped GPCR sequences closely at family, sub-family and sub-subfamily levels.

3.1.3 Quantitative evaluation on clustering results

To evaluate how good and effective the embedding space is, we com-pared our embedding space with the vector space that is generated by competing methods such as :

(1) Model with same architecture with ours but without center loss
(2) Model with same loss function and feature extractor with ours but without multiple branches output layer
(3) DeepFam
(4) Simple MLP classifier
(5) K-mer frequency vector (3-mer, 4-mer)

For models that do not employ multiple branches in the architecture, we trained the model based on supervisions from subfamily labels. Since it is not possible to compare feature spaces from different methods directly, we performed quantitative evaluation of clustering

analysis based on distances in the feature space. The evaluation was done in terms of *silhouette score* and *adjusted mutual information score*.

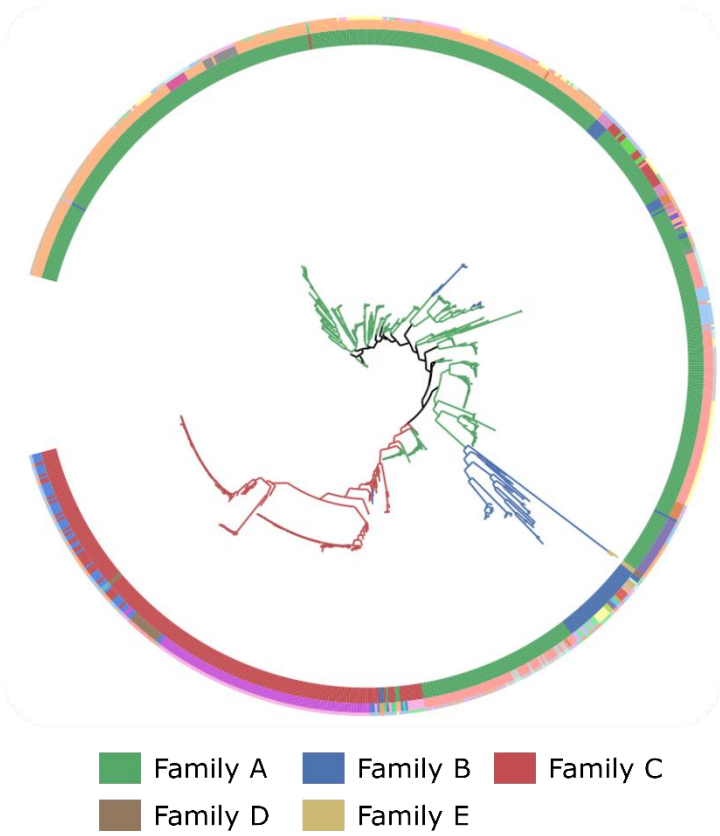

Figure 4 : Phylogenetic tree drawn from the embedding vectors. Branches that reached consensus on family-level classes, with more than 70% of leaves assigned to a majority class, are colored with a family-specific color. Colors of the rigns aroung the phylogenetic tree indicate the label information of each leaf.

We first performed a comparison on silhouette score to measures the consistency of clusters by comparing intracluster and intercluster distances. Detailed description of silhouette score is given in the method section. We first calculated the silhouette score of embedding vector where real class labels have been used for assessing the score. As three label information is presented for each sequence, score was evaluated for family, subfamily and sub-subfamily labels using euclidean distances. Under our scheme, silhouette scores will represent the consistency of distances between embedding vectors to the real class labels. From here, correspondence between representations given in the embedding vectors and the hierarchical class labels can be inferred.

Figure 5 contains the calculated silhouette score coming from each sequence analysis techniques. In the Figure, only *our model* and *our model w/o center loss* are the model equipped with multiple branches and use all three labels information during training. The distinctive result from these models is that they show non-negative silhouette scores in the three class levels, whereas vectors from other models present non-negative values in one level and negative or near-zero scores in other two class levels. Vectors from AutoEncoder, which does not use any of the class information in training, show negative scores in all levels. This result shows that label information can be a powerful supervisor in training a model. Competitive silhouette scores in all three levels show that the neural network architecture adopted in our model successfully incorporated the information from

three labels. For the two methods that use multiple branch networks, although the model without center loss shows best score in sub-subfamily level, our model shows comparable performance to the best score. On the other hand, for the family and subfamily level, our model shows significantly better score. This result indicates that center loss in our model effectively supports neural network to learn compact representations of the sequences. In conclusion, these results demonstrate that components in our model succeed in generating vectors adequate for representing hierarchical class information.

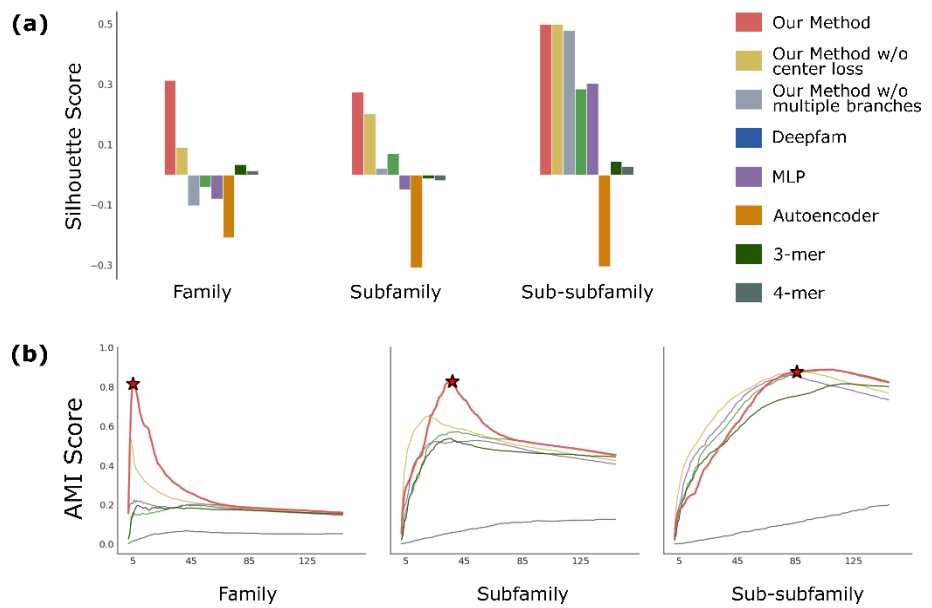

Figure 5 : Evaluated scores of embedding vectors. (a) Silhouette score calculated for vectors from each method. Scores evaluated for class labels from each hierarchical level is illustrated. **(b)** Adjusted mutual information(AMI) score evaluated for each level versus k clusters yielded from agglomerative clustering results. Plot shows the change in AMI scores as the number of clusters increments.

Correspondence between the hierarchical clustering results and the real GPCR family labels was estimated using adjusted mutual information (AMI) score. Detailed description on the index is given in the method part. In estimating the mutual information score, we regard real label information as ground-truth label information to measure the quality of clustering results coming from the embedding vectors. We performed agglomerative clustering, a hierarchical clustering algorithm where pair of clusters with closest distances are merged together, on embedding vector based on average linkage of euclidean distances. In agglomerative clustering, merge between two groups of data points takes place until the number of clusters in the dataset reaches the target number of clusters. AMI score was calculated with target number of clusters incrementally changing. AMI scores were measured for each class level and overall fluctuations following the number of clusters are illustrated in Figure 5.

In Figure 5, AMI score shows the maximum value when the number of resulting clusters from the algorithm gets closer to the real number of labels in that class level, 5 for family, 38 for subfamily and 87 for sub-subfamily level. This indicates that hierarchical clustering results comply with the real hierarchical label structure in the data. Similar to the results from silhouette scores, there are some algorithms that shows comparable scores in sub-subfamily level clustering evaluation. However, our model shows notably higher score in family and subfamily levels. In fact, our model is the only one with

maximum AMI score higher than 0.7 for all class levels. This again supports that our model succeed in incorporating information from all three class levels into an unified embedding vector. In addition, comparison between results from our model and our model without center loss demonstrates that center loss in our approach makes our embedding more compactly represented.

3.2 Sequence analysis with embedding vectors

3.2.1 Technical analysis

In Bioinformatics, the characteristics of the target sequences are often induced from other sequences with high similarity, since experimental identification of every individual sequence is infeasible. In this manner, we demonstrated how distances in embedding space can be utilized in selecting the reference proteins. Given the query sequence, we can infer the characteristics of the sequence by selecting the nearby sequences in the feature space. As the relations between sequences are represented as distances in the embedding space, we can adjust the resolution of inspection by varying the distance boundary of search range. For instances, we can infer the coarse-grained characteristics, or family level features, by selecting the reference protein from loosely bounded search range. On the other hand, fine-grained characteristics, sub-subfamily level features, can be

extracted by selecting the reference proteins from tight ranges.

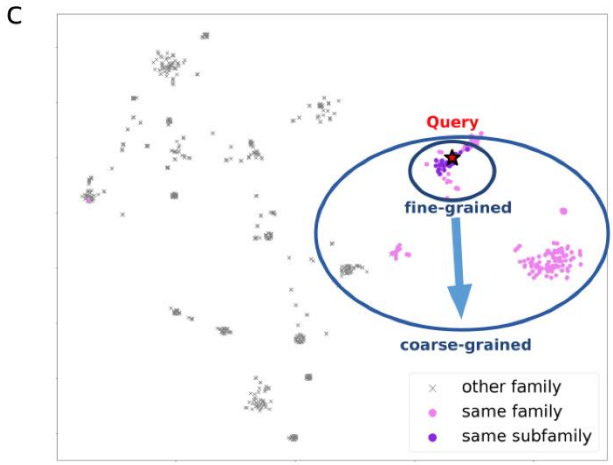

Figure 6 : t-SNE visualization of embedding vectors and example of inspections on the sequences at varying scale. Given a query sequence, which is represented as a star in the plot, color of the points in the plot indicates the relation between the query sequence and the corresponding sequences. In inferring the characteristics of the sequence from the nearby sequences, resolution of the investigation can be adjusted by varying the distance boundary of the reference sequences.

Overall process of above inspection is visualized in Figure 6. The t-Distributed Stochastic Neighbor Embedding (tSNE) was used to map the embedding vectors on two-dimensional space to reveal the overall distribution of embedding vectors. In Figure 6, query sequence is marked as a star and other points are colored according to the relation to the query sequence. Dark violet color represents the sequences from the same sub-subfamily and light violet color denotes the sequences from the same family but different in sub-sub-family level. Gray points are sequences from different families. In the

figure, query protein was selected from the Putpher sub-subfamily in the class C family. For a fine-grained reference proteins, with distance boundary of 0.01 from the query sequence, we were able to select 11 sequences where eight of them belong to Putpher sub-subfamily. By enlarging the search range to distance of 0.1, we selected 76 sequences, with 74 sequences belong to Class C family. This demonstrates the possibility of selecting the reference sequence in a desired resolution from the distances in the embedding vector.

3.2.2 Biological analysis

As a demonstration of biological analysis with embedding vectors, motif discovery experiments were performed on the sequences. In these experiments, Muscle, a multiple sequence alignment tool, was mainly used to detect conserved sequence patterns in the reference proteins set. Sets of reference proteins were selected from hierarchical clustering result. Especially, by changing the target number of clusters in hierarchical clustering algorithm and by selecting one of those clusters, it is possible to construct reference protein set at multiple scale. Afterwards, preserved regions in the alignments were compared to known motifs in the literatures to check if the results coincide with our biological knowledge on GPCR proteins. The results from motif discovery is given in the Figure 7.

From the clusters resulting of the hierarchical clustering with target cluster number of five, conserved sequences of *DRY* and *NSxxNPxxY* were found to be distinctive features in the first cluster. This cluster comprises of 566 sequences, of which 559 sequences belong to Family A of GPCR protein family. In fact, *NSxxNPxxY*(*NSxxY*) and *D*(*E*)*RY*(*F*) motif is the most characterizable sequence features shown in family A or Rhodopsin GPCR family [1, 28, 29]. Unlike the first cluster, however, alignments on the other clusters does not reveal the conserved sequence of *DRY*. This corresponds to our knowledge that *DRY* and *NSxxNPxxY* are distinctive features for Rhodopsin-like (Family A) GPCR proteins. Likewise, other significant motifs were found from other clusters too. On the second cluster among five clusters in the results, conserved sequences of *LIGWG*, *GPVLASLL* and *CFLxxEVQ* were discovered. These sequences belong to the conserved regions in the transmembrane structures of family B or Secretin receptor family of GPCR family [30]. Indeed, this cluster consists of 46 sequences where 44 of them belong to family B GPCR proteins. On deeper hierarchical levels, *RKAAKTLG* and *FKQLHXPTN* were found to be conserved in the 22nd cluster among 50 clusters. These features are known to be the representative motifs in the Traceamine sub-subfamily that belongs to family A of GPCR proteins. This coincide with the fact that all the sequences in the cluster are from Traceamine sub-subfamily.

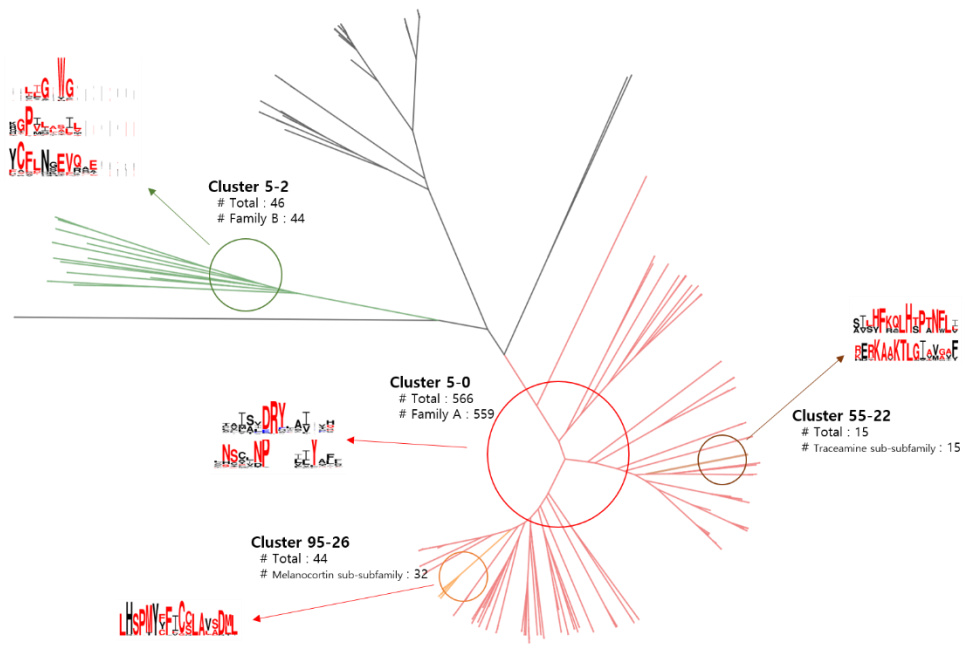

Figure 7 : Visualization of discovered motif logos from phylogenetic tree generated from hierarchical clustering results. Discovered motifs is shown next to the clusters and information on that cluster is provided with texts. Cluster 55-22 denotes that this cluster is 22nd clusters from the hierarchical clustering when the target number of cluster is 55.

3.3 Classification accuracy

Classification accuracy of the model was evaluated to show that our model does not lose the classification power even with the other factors in training phase. Up to our knowledge, DeepFam shows the state-of-the-art performance on the BIAS-PROF GPCR dataset[10]. Therefore, comparison was performed only between DeepFam and our model as classification power is not the primary objective in our work. Accuracies of our model were evaluated for

the prediction yielded from each branch of the multi-branch classifiers. Evaluation result is listed in Table 2. Clearly, our model still shows competitive performance in all hierarchical levels. Unlike DeepFam, which requires construction of separate neural networks for each class level, our model achieved notable performance with a single network.

Table 3 : Classification accuracy assessed for our model and three models from DeepFam. For DeepFam, three models were trained, specialized for family, subfamily and sub-subfamily level classifications and compared to our model.

Methods	Family	Subfamily	Sub-subfamily
Our method	0.984	0.896	0.821
DeepFam (Family)	0.984	–	–
DeepFam (Subfamily)	–	0.894	–
DeepFam (Sub-subfamily)	–	–	0.824

Chapter 4

Conclusion

We proposed a deep learning based approach to embed sequences in the GPCR protein family. In this study, deep neural network was used as a core component to process features from the sequences with a guidance of additional loss terms. These loss terms were designed to make feature vectors comply with background knowledge on the sequences, which is given in form of class labels. Moreover, our work presented a way to integrate multi-level information into a single vector space. As a result, embedding vectors from the model succeed in representing hierarchical features of the sequences in a feature space compactly. Indeed, inspections in Bioinformatics heavily depends on prior knowledge acquired from previous biological studies. Hence, one of the most important issues in Bioinformatics is to transform such knowledge into a computational form. In this aspect, our work proposes a simple yet powerful approach to incorporate understandings on the subject into a computational form. First, based on the fact that class labels of GPCR proteins are annotated in a way that phylogeny of the proteins are reflected, the proposed method defines a feature space and distances to represent the background knowledge during the learning phases. In addition to that, loss function was designed in a way that feature learning can take place with

a supervision from such prior knowledge. As a result, embedding function is trained to put sequences into a space with a guidance of different levels of clusters defined from class labels. In a series of experiments, compliance between real hierarchical label structure and hierarchical clustering results was demonstrated. In addition to that, we demonstrated that further inspections such as phylogeny reconstruction, motif analysis on clusters can be performed on embedding vectors and results correspond to the biological knowledge on the subjects.

Although the proposed method successfully embedded GPCR sequences into a euclidean space with phylogenetic relations conserved, there is still a room for improvement. First, imbalance in class labels is a common problem in real world machine learning tasks. This issue becomes more apparent when the task is dealing with biological data[31]. GPCR protein dataset is also exposed to imbalance in the class distribution, as Family A GPCR proteins account for more than 60% of known GPCR sequences[10]. To address this problem, additional techniques based on data sampling approaches could be considered to learn representations in data more effectively[31, 32]. Second, distances between sequences could be more rigorously defined in loss function. For instance, Zheng et al utilized the distances, inferred from pairwise alignments, in training neural networks for sequence embedding[18]. In this way, previously calculated phylogenetic distances might become more biologically sound when designing a loss function. As a future study, we are opting to address

above issues and bring more capability in deep metric learning approaches in Bioinformatics.

References

[1] R. Fredriksson, M. C. Lagerström, L.-G. Lundin and H. B. Schiöth, "The G-protein-coupled receptors in the human genome form five main families. Phylogenetic analysis, paralogon groups, and fingerprints," *Molecular pharmacology,* vol 63, pp. 1256-1272, 2003.

[2] T. K. Bjarnadóttir, D. E. Gloriam, S. H. Hellstrand, H. Kristiansson, R. Fredriksson and H. B. Schiöth, "Comprehensive repertoire and phylogenetic analysis of the G protein-coupled receptors in human and mouse," *Genomics,* vol 88, pp. 263-273, 2006.

[3] A. S. Hauser, M. M. Attwood, M. Rask-Andersen, H. B. Schiöth and D. E. Gloriam, "Trends in GPCR drug discovery: new agents, targets and indications," *Nature reviews Drug discovery,* vol 16, p. 829, 2017.

[4] S. R. George, B. F. O'Dowd and S. P. Lee, "G-protein-coupled receptor oligomerization and its potential for drug discovery," *Nature Reviews Drug Discovery,* vol 1, p. 808, 2002.

[5] R. M. Cooke, A. J. H. Brown, F. H. Marshall and J. S. Mason, "Structures of G protein-coupled receptors reveal new opportunities for drug discovery," *Drug discovery today,* vol 20, pp. 1355-1364, 2015.

[6] K. J. V. Nordström, M. C. Lagerström, L. M. J. Wallér, R. Fredriksson and H. B. Schiöth, "The Secretin GPCRs descended from the family of Adhesion GPCRs," *Molecular biology and evolution,* vol 26, pp. 71-84, 2008.

[7] Z. Zhang, J. Wu, J. Yu and J. Xiao, "A brief review on the evolution of GPCR: conservation and diversification," *Open J Genet,* vol 2, pp. 11-7, 2012.

[8] E. W. Taylor and A. Agarwal, "Sequence homology between bacteriorhodopsin and G-protein coupled receptors: exon shuffling or evolution by duplication?," *FEBS letters,* vol 325, pp. 161-166, 1993.

[9] K. J. V. Nordström, M. Sällman Almén, M. M. Edstam, R. Fredriksson and H. B. Schiöth, "Independent HHsearch, Needleman--Wunsch-based, and motif analyses reveal the overall hierarchy for most of the G protein-coupled receptor families," *Molecular biology and evolution,* vol 28, pp. 2471-2480, 2011.

[10] M. N. Davies, A. Secker, A. A. Freitas, M. Mendao, J. Timmis and D. R. Flower, "On the hierarchical classification of G protein-coupled receptors," *Bioinformatics,* vol 23, pp. 3113-3118, 2007.

[11] Z.-L. Peng, J.-Y. Yang and X. Chen, "An improved classification of G-protein-coupled receptors using sequence-derived features," *BMC bioinformatics,* vol 11, p. 420, 2010.

[12] G.-M. Hu, T.-L. Mai and C.-M. Chen, "Visualizing the GPCR network: Classification and evolution," *Scientific reports,* vol 7, p. 15495, 2017.

[13] Y. Yang, B.-l. Lu and W.-y. Yang, "Classification of protein sequences based on word segmentation methods," In *Proceedings of the 6th Asia-Pacific Bioinformatics Conference*, 2008.

[14] K. K. Yang, Z. Wu, C. N. Bedbrook and F. H. Arnold, "Learned protein embeddings for machine learning," *Bioinformatics,* vol 34, pp. 2642-2648, 2018.

[15] J. Lanchantin, R. Singh, Z. Lin and Y. Qi, "Deep motif: Visualizing genomic sequence classifications," *arXiv preprint arXiv:1605.01133,* 2016.

[16] S. Seo, M. Oh, Y. Park and S. Kim, "DeepFam: deep learning based alignment-free method for protein family modeling and prediction," *Bioinformatics,* vol 34, pp. i254--i262, 2018.

[17] T. Karydis, "Learning hierarchical motif embeddings for protein engineering," 2017.

[18] W. Zheng, L. Yang, R. J. Genco, J. Wactawski-Wende, M. Buck and Y. Sun, "SENSE: Siamese neural network for sequence embedding and alignment-free comparison," *Bioinformatics,* 2018.

[19] X. Pan, P. Rijnbeek, J. Yan and H.-B. Shen, "Prediction of RNA-protein sequence and structure binding preferences using deep convolutional and recurrent neural networks," *BMC genomics,* vol 19, p. 511, 2018.

[20] D. Quang and X. Xie, "DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences," *Nucleic acids research,* vol 44, pp. e107--e107, 2016.

[21] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[22] Y. Wen, K. Zhang, Z. Li and Y. Qiao, "A discriminative feature learning approach for deep face recognition," In *European conference on computer vision*, 2016.

[23] X. He, Y. Zhou, Z. Zhou, S. Bai and X. Bai, "Triplet-center loss for multi-view 3D object retrieval," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[24] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics,* vol 20, pp. 53-65, 1987.

[25] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research,* vol 12, pp. 2825-2830, 2011.

[26] D. Müllner, "Modern hierarchical, agglomerative clustering algorithms," *arXiv preprint arXiv:1109.2378,* 2011.

[27] N. Saitou and M. Nei, "The neighbor-joining method: a new method for reconstructing phylogenetic trees.," *Molecular biology and evolution,* vol 4, pp. 406-425, 1987.

[28] D. M. Rosenbaum, S. G. F. Rasmussen and B. K. Kobilka, "The structure and function of G-protein-coupled receptors," *Nature,* vol 459, p. 356, 2009.

[29] G. E. Rovati, V. Capra and R. R. Neubig, "The highly conserved DRY motif of class AG protein-coupled receptors: beyond the ground state," *Molecular pharmacology,* vol 71, pp. 959-964, 2007.

[30] A. J. Harmar, "Family-B G-protein-coupled receptors," *Genome biology,* vol 2, pp. reviews3013--1, 2001.

[31] D. J. Dittman, T. M. Khoshgoftaar and A. Napolitano, "Selecting the appropriate data sampling approach for imbalanced and high-dimensional bioinformatics datasets," In *2014 IEEE International Conference on Bioinformatics and Bioengineering*, 2014.

[32] D. J. Dittman, T. M. Khoshgoftaar, R. Wald and A. Napolitano, "Comparison of data sampling approaches for imbalanced bioinformatics data," In *The Twenty-Seventh International Flairs Conference*, 2014.

초록

딥러닝 기반 단일 거리 공간 내 GPCR 단백질군 계층 구조의 동시적 모델링 기법

이태헌
컴퓨터공학부
서울대학교 대학원

G 단밸질 연결 수용체(GPCR)은 계층 구조로 형성된 다양한 단백질군으로 구성된다. 단백질 서열을 통한 GPCR에 대한 계산적인 모델링은 군(family), 아군(subfamily), 준아군(sub-subfamily)의 각 계층에서 독립적으로 실행되는 방식으로 이루어져왔다. 하지만 이러한 접근 방식들은 단절된 모델들을 통하여 단백질 내의 정보를 처리하기 때문에 GPCR 종류 사이의 관계는 고려하지 못한다는 한계를 가지고 있다.

본 연구에서는 딥러닝을 이용하여 GPCR의 계층 구조에서 나타나는 특징들을 단일한 모델로 동시적으로 학습하는 방법을 제시한다. 또한 계층적인 관계들을 하나의 벡터 공간에 거리를 통해 표현할 수 있도록 하기 위한 손실함수도 제시한다. 이 연구는 GPCR 수용체들의 여러 계층에서 공통적으로 나타나는 특징들을 학습하고 표현할 수 있도록 하는 방법을 다루고 있다. 여러 심화적인 실험들을 통하여 우리는 기술적인 측면과 생물학적인 측면에서 단백질 간 계층적인 관계가 성공적으로 학습이 되었다는 것을 보였다. 첫번째로, 우리는 임베딩 벡터에 계층적 군집화(hierarchical clustering) 알고리즘을 적용함으로써 계통수

(phylogenetic tree)를 만들었고, 군집 알고리즘과 실제 계층 구조와의 수치적인 비교를 통하여 임베딩 벡터를 통해 계통학적 특징에 대한 유추가 가능하다는 것을 보였다. 두번째로, 임베딩 벡터의 군집화 결과에 다중 서열 정렬(multiple sequence alignment)를 적용시킴으로써 생물학적으로 유의미한 서열적 특성들을 찾아낼 수 있다는 것을 보였다. 이는 임베딩 벡터 분석이 GPCR 단백질 연구에 있어 효율적인 첫걸음이 될 수 있다는 것을 보여준다. 이러한 결과는 여러 계층으로 이루어진 단백질군에 대한 동시적인 모델링이 가능하다는 것을 말하고 있다.

Keywords : G 단백질 연결 수용체, 계층 구조, 임베딩 공간, 표현 학습, 딥러닝
Student Number : 2017-23643

감사의 글

좋은 연구 환경을 조성해주시고 학위 과정동안 성장할 수 있도록 저를 이끌어주신 김 선 교수님께 감사하다는 말을 전하고 싶습니다. 제가 부족함이 많았음에도 교수님의 도움으로 많은 것을 배울 수 있었습니다. 2년동안 함께 연구실 생활을 하면서 연구와 학문에 대하여 함께 고민하고 토론해준 BHI 연구실 구성원께도 감사하다는 말을 전하고 싶습니다. 이외에도 저를 응원해주시고 지원해주신 가족과 친구들에게도 감사를 표합니다.