Amino Acid Sequence Based Method for Prediction of Cell Membrane Protein Types

Seyed Koosha Golmohammadi, Lukasz Kurgan, Brendan Crowley, and Marek Reformat
Department of Electrical and Computer Engineering, University of Alberta, Canada
{koosha, lkurgan, bcrowley, reform}@ece.ualberta.ca

Abstract

Cell membrane proteins play a vitally important role in influencing the behavior of cells. Knowledge of membrane protein type facilitates the determination of its functions, which has implications in numerous applications including drug design. Due to the increasingly large number of uncharacterized proteins in data-banks such as NCBI's RefSeq, there is a high desire to replace time and cost consuming experimental methods for membrane protein type classification with computational methods. This paper introduces a new computational method that accurately predicts the type of unclassified membrane proteins based on their sequence. Our method is based on a novel representation of protein sequences that incorporates seven different feature sets. Empirical comparison, which includes twelve competing methods, shows that the presented method generates predictions that result in 8% and 28% error rate reduction when compared with the best existing computational method and when using the jackknife test and testing on an independent dataset, respectively. We also show that the most influential sources of information for making the predictions include the composition of 2-gram exchange groups and the amino acid composition of the underlying sequence.

1. Introduction

Cells are considered to be the smallest living part of an organism, and are often referred to as building blocks for life. The cytoplasm, or insides of a cell, is contained by a membrane constructed mainly from a lipid bilayer. The membrane also includes numerous embedded cell membrane proteins that carry out a number of essential functions of the membrane. Cell membrane proteins are classified as either *transmembrane* proteins, which span across the cell membrane, or *anchored* proteins, which are attached to only one side of the cell membrane; see Figure 1. This classification is further refined into five sub-types [2]: type I transmembrane, type II transmembrane, multi-pass transmembrane, lipid chain-anchored, and GPI-anchored. The function of a membrane protein usually depends on its relationship with the lipid bilayer [2]. For instance, the transmembrane proteins can transport molecules across the membrane. In addition to molecular transportation, cell membrane proteins are used for sensing signals external to the cell that determine the cell's behavior, among other things.

An important result of the genome project was the rapid increase in the number of known protein sequences. Nowadays, information on millions of proteins is stored in several data banks, such as the SWISS-PROT, PDB (Protein Data Bank), and NCBI's (National Center for Biotechnology Information) RefSeq. SWISS-PROT is a manually curated database that includes partial functional and structural annotation that currently

includes about 330,000 proteins. PDB is another manually curated database of around 48,500 tertiary protein structures. The largest protein data bank is maintained by the NCBI and contains over 4.4 million sequence entries, but without functional and structural information. Due to their importance, cell membrane proteins have become one of the main targets for both research and drug design [1]. Unfortunately, it is extremely time-consuming and costly to experimentally determine the type and function of every new (membrane) protein. Currently, the tertiary structure is known for only about 150 unique membrane proteins. For this reason and due to the large and widening gap between the number of known protein chains and the number of functionally and structurally annotated proteins, it is highly desirable to develop accurate computational methods for high-throughput prediction of functional and structural information of chains that were not yet annotated. Such prediction models could also be used to identify proteins that are potential candidates for future drug design activities.

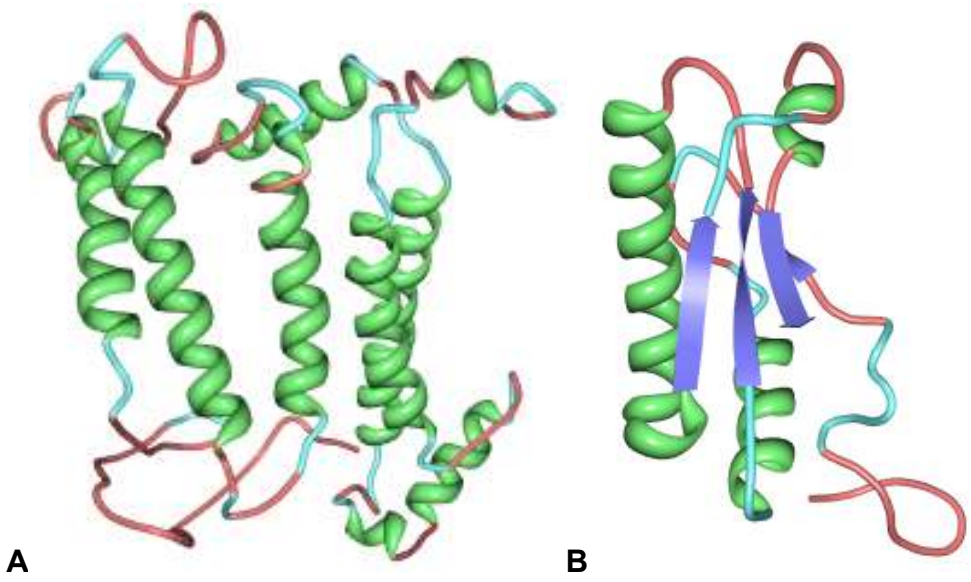

Figure 1. Ribbon structures of two membrane proteins. Panel A shows chain M of 2RCR protein (Rhodobacter sphaeroides) which belongs to multipass transmembrane proteins. Panel B shows chain P of 2AIZ protein (peptidoglycan associated lipoprotein from Haemophilus influenza) which is an anchored membrane protein. The structures were drawn using mbt package [29].

The classification of membrane proteins into their corresponding types is usually a two-part process. The first step is to convert each protein chain (sequences) into a feature-based representation. The second step feeds the feature-based representation into a classification model whose output is the predicted type. Existing computational methods for membrane protein classification can be divided into two categories: (1) those that use amino acid composition to represent the protein sequences [2]; and (2) those that use pseudo-amino acid composition to represent the sequences [3]. The latter feature-based representation incorporates sequence order, while the former is based solely on order-independent counts of amino acids. The methods that use amino acid

composition apply a number of different classification models, including Hamming distance [4], Euclidean distance [5], Protlock [6], and covariant discriminant analysis [2]. The methods that use pseudo-amino acid composition are generally more accurate, and utilize the above classification models [3], as well as support vector machines [7, 8], fuzzy k-nearest neighbor [9], optimized evidence-theoretic k-nearest neighbor [10], supervised locally linear embedding [11], and various ensembles of classifiers [1,12]. The two most recent contributions, which both use pseudo-amino acid composition and an ensemble of classifiers, are:

1. A stacked generalization based method that attempts to maximize the classification accuracy by combining the results of a support vector machine and an instance-based learner through a meta-classifier implemented with the used of 4.5 decision tree [1].

2. An ensemble of classifiers that is formed by merging a set of nearest neighbor classifiers, each of which is defined in a different pseudo-amino acid composition space [12].

Additionally, similar computational approach is used to address prediction of other structural and functional aspects of proteins; for instance, to predict the structural class [13] or subcellular location [14].

This paper introduces a novel, computational method for identifying the types of membrane proteins using their amino acid sequence as the only input. Our main goal is to achieve a classification accuracy higher than that of existing approaches. To design our method, first, each protein sequence was mapped into a novel feature-based vector. Next, the best performing classifier was selected to predict the type based on our feature-based vector. Three conventional tests performed on two large benchmark datasets [2,3] were used to evaluate the performance of the proposed method. The classification accuracy was compared with 12 competing computational methods. The unique characteristic of the proposed method is that the sequences are represented by seven features sets, while the existing methods usually use only one feature set.

Section 2 describes the design of the proposed method. Section 3 presents and discusses our experimental results, and section 4 concludes the paper.

2. Methodology

Preparation of the input for the classifiers is a crucial and time-consuming task since the classification accuracy depends on the features that are selected to represent the protein sequence. Section 2.1 describes the raw data, i.e., protein datasets, which were used to design and test the proposed method. Section 2.2 describes the features that were considered as inputs for the classification model, and section 2.3 describes the methods used to select the best performing classifier.

2.1 Data

Two datasets were used to design and test our prediction method. These datasets (see Table 1) are widely used to evaluate the performance of cell membrane protein classification methods [1-3, 7-14], allowing for fair comparison with models described in the literature.

Table 1. Datasets used to design and test the proposed prediction method.

	ref.	number of proteins of a given type				
		type-I	type II	multipass	lipid	GPI
Dataset 1 2059 proteins	[3]	435	152	1311	51	110
Dataset 2 2625 proteins	[2]	478	180	1867	14	86

2.2 Feature-based Sequence Representation

There are 20 unique amino acids that constitute a protein's building blocks. All amino acids have a common basic chemical structure, but different chemical properties due to differences in their side chains. A protein can be represented by a string (chain) of amino acids. Different proteins have different amino acid chains, in terms of the ordering of the amino acids and their total number (length of the sequence). The first step in classifying proteins is to find a common way to represent the sequences. In this work we developed a new feature vector to represent protein chains. Any protein, regardless of the length or composition of its sequence, can be mapped to our feature vector representation. We use 7 distinct feature sets within our feature vector. These feature sets along with the corresponding number of features are shown in Table 2.

Table 2. Feature based sequence representation.

Feature set	Number of Features
Amino Acid Composition	20
Sequence Length	1
2-Gram Exchange Group Frequency	36
Hydrophobic Group	2
Electronic Group	6
Sum of Hydrophobicity	1
R-Group	5

2.2.1. Amino Acid Composition, CV_i where i=1,2, …, 20, is defined as the normalized frequency of occurrence of each of the twenty amino acids in the given protein's amino acid sequence [1].

2.2.2. Sequence Length, L, is defined as the total number of amino acids in the given protein's amino acid sequence.

2.2.3. 2-Gram Exchange Group Composition, ExG_i where i=1,2, …,36, is defined by converting the sequence into its equivalent 6-letter exchange group representation [15], see Table 3, which was derived from the PAM matrix. The exchange groups are broader classes of amino acids that represent the effects of evolution. For example, all H, R, and K amino acids in the original sequence are replaced by e_1. After the amino acids are replaced, the resulting sequence consists of an alphabet of only 6 different characters. We compute the frequency of occurrence of each possible 2-gram (pair) [16] of the consecutive exchange group amino acids.

Table 3. Property groups of amino acids used to derive features.

Group	Sub-group	Amino Acids
Exchange group	e_1	KHR
	e_2	DENQ
	e_3	C
	e_4	AGPST
	e_5	ILMV
	e_6	FYW
Hydrophobic group	hydrophobic	ACFILMPVWY
	hydrophilic	DEGHKNQRST
Electronic group	electron donor	DEPA
	weak electron donor	VLI
	electron acceptor	KNR
	weak electron acceptor	FYMTQ
	neutral	GHWS
	special AA	C
R group	non-polar aliphatic	AILV
	glycine	G
	non-polar	FMPW
	polar uncharged	CNQSTY
	charged	DEHKR

Table 4. Eisenberg hydrophobicity index values of amino acids.

Amino Acid	Index value	Amino Acid	Index value	Amino Acid	Index value	Amino Acid	Index value
A	0.62	E	-0.74	L	1.06	S	-0.18
R	-2.53	Q	-0.85	K	-1.5	T	-0.05
N	-0.78	G	0.48	M	0.64	W	0.81
D	-0.9	H	-0.4	F	1.19	Y	0.26
C	0.29	I	1.38	P	0.12	V	1.08

2.2.4. Hydrophobic Group, HG_i where i=1,2. The side chains may be polarized. The non-polar side chains are hydrophobic, while polar side chains are hydrophilic; see Table 3 [17][18]. The corresponding two features are based on counts of the hydrophobic and hydrophilic amino acids in the protein sequence.

2.2.5. Electronic Group, EG_i where i=1,2…,6. The electronic group specifies whether a given amino acid is electrically neutral, donates electrons, or accepts electrons. We again compute the frequency (count) of amino acids in each of the electronic groups; see Table 3.

2.2.6. Sum of Hydrophobicity, *Y*. Each amino acid has an associated hydrophobic affinity, which is often measured using a hydrophobic index. The Eisenberg hydrophobic index (see Table 4), which was used to analyze membrane-associated helices [20], is applied in this feature set. This index is normalized and ranges between -2.53 for R (the least hydrophobic) and 1.38 for I (the most hydrophobic). Similarly to [21], we compute the sum of this hydrophobic index over all amino acids in the protein sequence, which gives one feature.

2.2.7. R-Group, RG_i where i=1,2…,5. As discussed above, each amino acid has a different side chain. However, some of these side chains have similar characteristics

and can be clustered into five sub-groups; see Table 3 [21]. The composition (count) of amino acids in each of these groups is computed.

The resulting feature vector, which consists of 71 features grouped into seven feature sets, constitutes the input for our classification model.

2.3 Design of the Proposed Prediction Method

To find the best performing classifier we updated our design iteratively based on a series of tests that were divided into three phases. We designed our method using the Weka environment [22]. The tests were performed utilizing 10 fold cross-validation on Dataset 1.

2.3.1. Phase 1: Phase one was devoted to preparing the input data for classification. We computed 71 features, as described in Section 2.2, for each sequence in Datasets 1 and 2.

2.3.2. Phase 2: We tested all 70 classifiers in Weka (except for models that required discrete input data) to compare their performance for this classification problem. These classifiers include Bayesian methods, regression, support vector machines, neural networks, instance based nearest neighbor methods, decision trees, rule based and cost based methods. Table 5 shows the overall accuracy and the accuracy for each protein type for the top 9 classifiers, with respect to their overall accuracy, based on a 10 fold cross-validation test on Dataset 1.

Table 5. Top nine classifiers with the highest overall accuracy on Dataset 1.

Classifier	Overall accuracy	Accuracy for a given membrane protein type				
		type I	type II	multip ass	lipid chain	GPI
Decision Tree with Naïve Bayes at the leaves	81.29	74.25	36.84	93.06	62.74	38.53
Bagged Decision Tree	81.78	74.25	36.84	93.06	62.74	38.53
Logistic Regression based metaclassifier	81.88	76.32	26.31	94.20	64.70	41.28
Support Vector Machine with polynomial kernel	82.31	71.84	25.66	95.04	50.98	24.77
Decorate based ensemble of Decision Trees	83.04	78.63	40.13	93.74	49.02	47.71
Random Forest	83.04	82.52	32.90	94.13	47.06	38.53
Neural Network with backpropagation training	84.26	80.46	42.76	94.35	52.94	50.46
K-nearest neighbor	85.76	82.30	50.00	94.81	47.06	58.71
K*-nearest neighbor	86.30	82.53	47.37	95.88	45.10	59.63

The overall classification accuracy of a model is an important factor, but not sufficient to select the best classifier for this problem. The accuracies for each membrane protein type should also be considered in choosing the best performing classifier. From the top 9 models we eliminated those that had the worst accuracies for individual protein types, and retained those that had the best accuracies for the different

types. Among all models, K* performed the best considering both overall and majority of per type accuracies. Section 2.4 provides a detailed description of K* model.

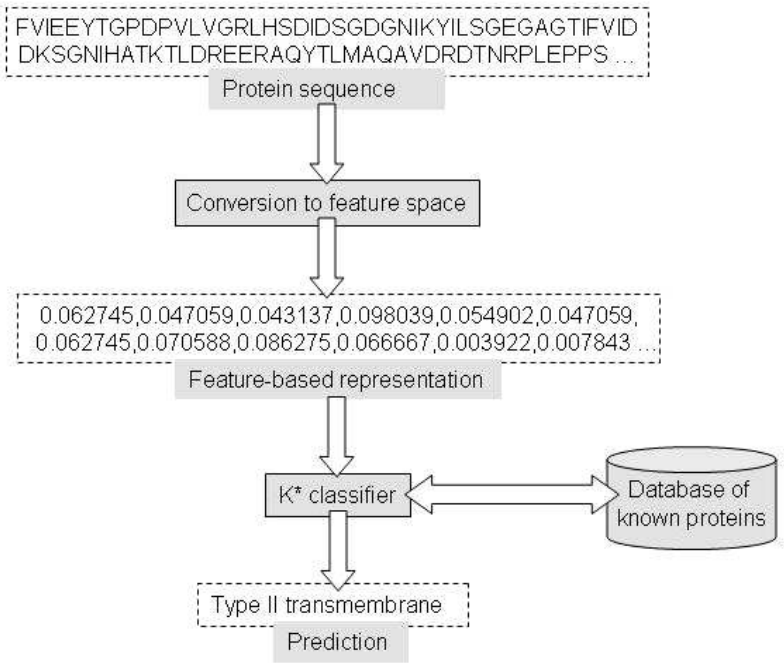

Figure 2. Process flow in the proposed method.

2.3.3. Phase 3: In this phase the K* classifier was tested with different parameters through 10 fold cross-validation to optimize the resulting overall accuracy. K* performed the best when the *globalBlend* parameter was equal to 38%; see Section 2.4 for more details.

The prediction process of the proposed method is depicted in Figure 2.

2.4 K* Classifier

The selected method, K*, is an instance-based classifier [23]. Instance-based classifiers compare an instance to a database of known classified instances. The underlying idea is that similar instances should have similar class labels. Instance-based classification algorithms use a distance function to compare instances and choose which database instance(s) is closest to the test (predicted) instance. They employ a classification function to determine the final prediction of the test instance, based on the classes of the similar database instances.

A k-nearest neighbor algorithm finds the k instances that are the closest to the test instance. The most common class among the k nearest neighbors is chosen as the predicted class.

The defining characteristic of K* is that it uses an entropy-based distance function. The function computes distance as the complexity of transforming one instance into the other. The basic probability function for the algorithm is

$$P^*(b \mid a) = \sum_{\bar{t} \in P : \bar{t}(a) = b} p(\bar{t})$$

where $p(\bar{t})$ is the probability of transformation t, from instance a to instance b.

This is the probability distribution of all paths from instance a to instance b. The K* function is the log domain version of the above distribution

$$K^*(b \mid a) = -\log_2 P^*(b \mid a)$$

The above K* function considers a single attribute of an instance. The union of transformations for the individual attributes takes several attributes into consideration. The result of this is that the probability for the composite transformation is simply the product of probabilities of each individual transformation. The overall distance function is therefore equal to the sum of the distances for each attribute transformation.

The second part of the algorithm involves finding the probability that a given instance belongs to a certain class, C, which is found by taking the sum of probabilities for the test instance a to each database instance b in the given class

$$P^*(C \mid a) = \sum_{b \in C} p^*(b \mid a)$$

where $P(x|y)$ is the conditional probability of x given y.

The predicted class is the class with the highest probability. The K* algorithm includes a *globalBlend* parameter that specifies the number of neighbors that should be considered. Choosing 100% results in all neighbors having an equal weighting. Choosing 0% turns K* into a 1-nearest neighbor algorithm.

3. Results and Discussion

3.1 Experimental Setup

Three test methods were used to evaluate the quality of the proposed prediction model [24]: (1) the re-substitution (self-consistency) test, (2) the jackknife (leave-one-out) test, and (3) the independent dataset test. The self-consistency test involves training the model with Dataset 1, and then testing the model with the same Dataset 1. During the jackknife test, we designed and tested the model through n-fold cross validation on Dataset 1, where n is the size of the dataset. The independent dataset test involves training the model on Dataset 1, and next testing it on Dataset 2. Among the three tests the jackknife test is the most objective [14]. This type of test is widely used to evaluate related prediction methods [2, 24-28].

In addition to reporting overall accuracy, we also report the accuracy, specificity, and Matthew's Correlation Coefficient for each membrane protein type and for each test type. The Matthew's Correlation Coefficient (MCC) ranges between -1 and 1. A value of 1 means the classifier never makes any mistakes. A value of -1 means the classifier always makes mistakes.

3.2 Experimental Evaluation of the Proposed Method

The experimental results for the proposed method are summarized in Table 6.

Table 6. Classification results for the proposed prediction method.

		Test method		
		Self-consistency	**Jackknife**	**Independent**
Accuracy [%]	*Overall*	*99.9*	*86.9*	*97.1*
	Type I	100	83.4	96.4
	Type II	100	52.6	80.6
	Multipass	100	95.8	99.0
	Lipid	100	45.1	78.6
	GPI	99.1	61.5	96.5
Specificity [%]	Type I	100	94.7	99.2
	Type II	100	98.3	99.8
	Multipass	99.9	83.4	93.9
	Lipid	100	99.9	99.9
	GPI	100	98.7	99.8
MCC	Type I	1.00	0.77	0.95
	Type II	1.00	0.59	0.87
	Multipass	1.00	0.81	0.94
	Lipid	1.00	0.64	0.78
	GPI	0.99	0.65	0.96

The worst jackknife test accuracies of about 50% are obtained for type II transmembrane and lipid-chain anchored membrane proteins, while type I and multipass types are predicted with 87% and 96% accuracy, respectively. We emphasize that the proposed method is characterized by high specificity values that range between 95% and 100%, which shows that our method is selective. Although the accuracies obtained for the independent test set are higher, they are consistent with the jackknife based results.

The results show that the weakness of our model is in classifying lipid-chain-anchored membrane proteins, while the model performs relatively well for transmembrane proteins. One of the possible reasons for the favorable performance for multipass transmembrane proteins could be that they constitute the majority of the samples in the two datasets. At the same time, the number of samples for the lipid-chain anchored membrane proteins is the lowest, which could lead to the poorer prediction accuracy for this type.

3.3 Comparison with Competing Methods

The proposed method was also compared with twelve competing methods that were published after 1986; see Table 7.

Table 7. Comparison of the overall accuracy with twelve competing methods.

Prediction method (year published)	Reference	Test method		
		Self-consistency	Jackknife	Independent
*K** (2008)	*this paper*	*99.9*	*86.9*	*97.1*
Ensemble of NNs (2007)	[12]	not available	85.8	96.8
Fuzzy KNN (2006)	[9]	not available	85.6	95.7
Stacking (2006)	[1]	98.7	85.4	94.3
OET-KNN (2005)	[10]	99.5	84.7	94.2
SLLE (2005)	[11]	not available	82.3	95.7
Weighted SVM (2004)	[8]	99.9	82.4	90.3
SVM (2004)	[7]	not available	80.4	85.4
Augmented covariant discriminant (2001)	[3]	90.9	80.9	87.5
Covariant-discriminant (1999)	[2]	85.5	73.0	80.9
ProtLock (1997)	[6]	51.5	48.7	46.7
Least Hamming distance (1989)	[4]	49.3	47.8	47.0
Least Euclidean distance (1986)	[5]	50.8	49.1	48.7

Table 7 shows that prediction methods based on nearest neighbor (NN) and *k*-nearest neighbor (KNN) classifiers, including the proposed method, perform quite well, suggesting that this type of the classifier is the best choice for the membrane type prediction problem. As Table 7 is organized chronologically, we observe that the prediction quality was being improved over the last two decades. The early methods, which were developed in 1980's and 1990's, predicted the types with the jackknife test accuracy of below 75%. Since 2001, the subsequent methods succeeded in improving the accuracy by about 5%, reaching almost 86%. Our method produced the highest accuracies of 86.9% and 97.1% for the jackknife and the independent dataset tests, respectively. It improved the error rate of the jackknife test by 8% (1.1/14.2) and of the independent dataset test by 28% (0.9/3.2), when compared with the second best ensemble classifier [12].

Figures 3 and 4 show a side-by-side comparison of accuracies for prediction of each cell membrane type between the proposed and the second best methods for the jackknife and independent dataset tests, respectively. When compared with the recent ensemble based method, our solution which uses one classification model (and therefore has a simpler architecture) performs better on type-I, type-II, and GPI-anchored proteins, while offering the same quality for multipass proteins. For the lipid-anchored proteins our method performs slightly worse for the jackknife test, but better for the test on the independent dataset. We note that the improvement in the overall accuracy of the proposed method results from the consistent improvement over the majority of the membrane protein types.

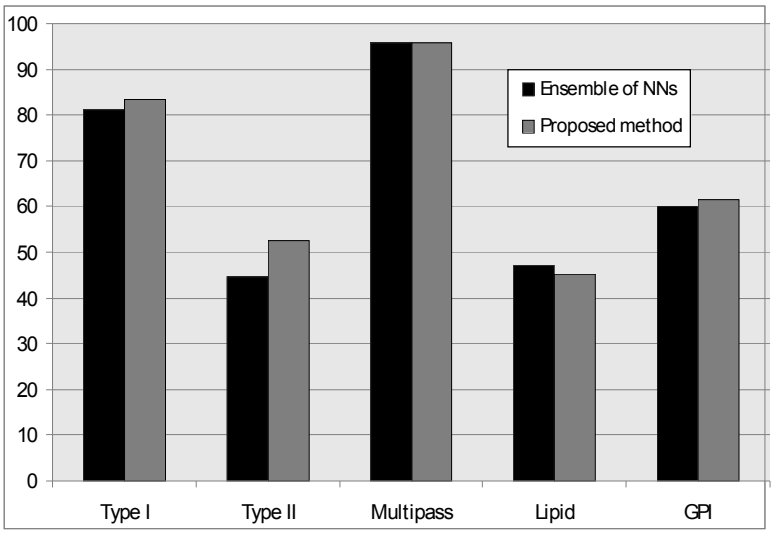

Figure 3. Comparison of results of jackknife test between the proposed and the second best prediction method.

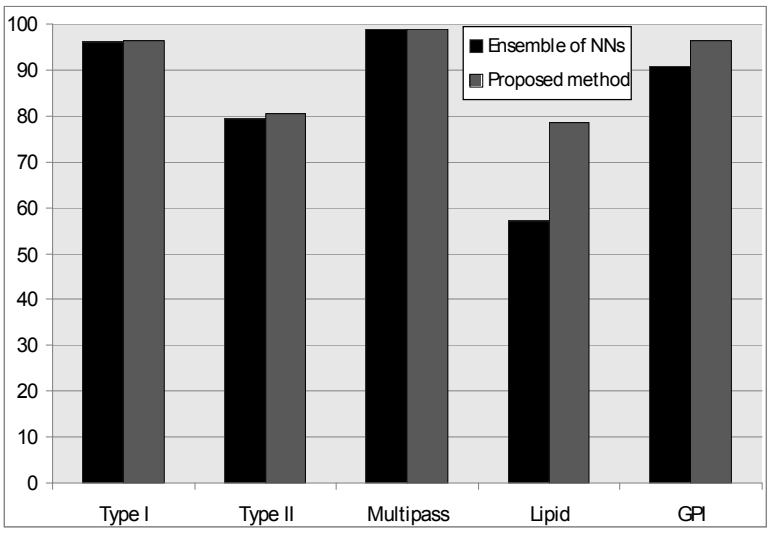

Figure 4. Comparison of results of independent dataset test between the proposed and the second best prediction method.

3.4 Evaluation of Feature Sets

The proposed method includes seven distinct sets of features to encode the protein sequence. Following, we study the impact of each of these sets on the quality of the prediction. Table 8 shows results of experiments in which one of the feature sets was removed and the prediction was performed with the remaining 6 sets. The difference in the prediction accuracy when using all seven sets and when a given set is removed allows for estimating the value added by that feature set. Table 8 sorts the attributes from the best performing (top of

the table) to worst performing (bottom of the table) based on the sum of the differences for the jackknife and independent dataset tests.

Table 8. Comparison of the overall accuracy between using all seven feature sets to encode sequence (the proposed method) and when one of the features sets is eliminated.

Eliminated feature set	Overall accuracy [%] for a given test method	
	Jackknife	Independent
2-Gram Exchange Group Composition	84.2	95.2
Amino Acid Composition	85.0	95.8
Sequence Length	86.6	96.0
R-Group	86.8	96.6
Sum of Hydrophobicity	86.7	96.9
Hydrophobic Group	86.9	96.8
Electronic Group	87.2	97.0
with all feature sets	*86.9*	*97.1*

The results show that the most valuable feature set is the 2-gram exchange group composition, i.e., the corresponding decrease in the overall accuracies for the jackknife and independent test equal 2.7% and 1.9%, respectively. We emphasize that the 2-gram exchange group composition was first proposed to be used to predict the membrane protein type in this contribution. We hypothesize that the presented improvements in accuracy obtained by our prediction method, see Section 3.3, are mostly due to using this feature set. The second best set is the amino acid composition, for which the corresponding decreases equal 1.9% and 1.3%, respectively. This feature set was frequently used to perform prediction of membrane protein types [2,4-6]. Removal of the remaining features sets has only a relatively small impact, i.e., between 0% and 0.5%, on the overall accuracy.

We also note that the feature sets are complementary to each other, i.e., only small differences of about 0% to 2.7% in the overall accuracy were observed when removing the sets. This means that the remaining sets cover the majority of the "functionality" of the removed set, and so they can be also successfully used for the prediction. Finally, we observe that removal of some sets leads to identical or even slightly better accuracy of the jackknife test, e.g. electronic group, but at the same time the corresponding accuracy of the test on the independent dataset is lower, therefore justifying their inclusion.

4. Conclusions

We introduced a novel computational method for classifying cell membrane protein types based solely on their amino acid sequences. The main defining feature of the proposed method is the inclusion of seven feature sets to encode a protein sequence. This is in contrast to existing methods that usually use only one feature set; either amino acid composition or pseudo amino acid composition. The proposed method is characterized by a simple architecture, i.e., is based on a single classification model, while the most recent competing methods are based on complex, ensemble based models.

The empirical evaluation performed with the help of two large, standard benchmark datasets shows that the proposed method provides higher prediction accuracy than methods previously reported in the literature. Our method correctly predicts the membrane protein type 86.9% of the time when evaluated using jackknife test and 97.1% of the time when tested on the independent dataset.

We evaluated the importance of each feature set used by the proposed method. We found that 2-gram exchange group composition is the most important feature set, and that the feature sets exhibit a high degree of overlap with each other. Our future work will include development of additional feature sets that would provide further improvements in the classification accuracy.

References

[1] S. Q. Wang, J. Yang, K.C. Chou, "Using stacked generalization to predict membrane protein types based on pseudo amino acid composition", Journal of Theoretical Biology (2006), vol. 242, pp. 941-46.
[2] K.C. Chou and D.W. Elrod, "Prediction of membrane protein types and subcellular locations", Proteins: Structure, Function, and Genetics (1999), vol. 34, pp. 137-53.
[3] K.C. Chou. "Prediction of protein cellular attributes using pseudo amino acid composition", Proteins: Structure, Function, and Genetics (Erratum: Proteins: Structure, Function, and Genetics, vol. 44, pp. 60) (2001), vol. 43, pp. 246-55.
[4] P.Y. Chou, "Prediction of protein structural classes from amino acid composition". In: GD. Fasman, editor. "Prediction of protein structure and the principles of protein conformation". New York: Plenum Press, (1989), pp. 549–586.
[5] H. Nakashima, K. Nishikawa and T. Ooi. "The folding type of a protein is relevant to the amino acid composition", Journal of Biochemistry (1986), vol. 99, pp.152–162.
[6] J. Cedano, P. Aloy, J.A. Perezpons and E. Querol. "Relation between amino acid composition and cellular location of proteins", Journal of Molecular Biology (1997), vol. 266, pp. 594–600.
[7] Y. D. Cai, P. W. Ricardo, C.H. Jen, K.C. Chaou, "Application of SVM to predict membrane protein types", Journal of Theoretical Biology (2004), vol. 226, pp. 373-6
[8] Wang M., Yang J., Liu G. P., K.C. Chou, "Weighted-support vector machines for predicting membrane protein types based on pseudo-amino acid composition", Protein Engineering Design and Selection (2004), vol. 17, pp. 509-16
[9] H.B. Shen, J. Yang and K.C. Chou, "Fuzzy KNN for predicting membrane protein types from pseudo-amino acid composition", Journal of Theoretical Biology (2006), vol. 240, pp. 9-13
[10] H.B. Shen, K.C. Chou, "Using optimized evidence-theoretic K-nearest neighbor classifier and pseudo-amino acid composition to predict membrane protein types", Biochemical and Biophysical Research Communications (2005), vol. 334, pp. 288-92.
[11] M. Wang, J. Yang, Z.J. Xu, K.C. Chou, "SLLE for predicting membrane protein types", Journal of Theoretical Biology (2005), vol. 232, pp. 7-15.
[12] H.B. Shen and K.C. Chou, "Using ensemble classifier to identify membrane protein types", Amino Acids (2007), vol. 32, pp. 483-8.
[13] K.C. Chou, "Progress in protein structural class prediction and its impact to bioinformatics and proteomics", Current Protein and Peptide Science (2005) vol. 6, pp. 423-436.
[14] K.C. Chou, and H.B. Shen, "Recent progresses in protein subcellular location prediction", Analytical Biochemistry (2007), vol. 370, pp. 1-16.
[15] C.H. Wu, M. Berry, Y.S. Fung, and J. McLarty, "Neural Networks for Full-Scale Protein Sequence Classification: Sequence Encoding with Singular Value Decomposition", Machine Learning (1995), vol. 21, pp 177–93.
[16] C. H. Wu, G. Whitson, J. McLarty, A. Ermongkonchai, and T .C. Chang, "Protein Classification Artificial Neural System", Protein Science, (1992), vol. 1, 667-77.
[17] S. Zumdahl and S. Zumdahl, "Chemistry", Fifth edition, Houghton Mifflin Company, (2000).
[18] D.F. Waugh, "Protein-protein interactions", Advances in Protein Chemistry (1954), vol. 9, 325-437.
[19] M. Ganapathiraju, J. Klein-Seetharaman, N. Balakrishnan, and R. Reddy, "Characterization of protein secondary structure", IEEE Signal Processing Magazine (2004), vol. 21, pp. 78–87.
[20] D. Eisenberg, E. Schwarz, M. Komaromy, and R. Wall, "Analysis of membrane and surface protein sequences with the hydrophobic moment plot", Journal of Molecular Biology (1984), vol. 179, 125-42.

[21] K. Kedarisetti, L. Kurgan, and S. Dick, "Classifier ensembles for protein structural class prediction with varying homology", Biochemical and Biophysical Research Communication (2006), vol. 348, pp. 981-8.
[22] I.H. Witten and E. Frank, "Data Mining: Practical Machine Learning Tools and Techniques", 2nd edition, Morgan Kaufmann, San Francisco, (2005).
[23] J.G. Cleary and L.E. Trigg, "K*: An instance-based learner using an entropic distance measure", Proceedings of the 12th International Conference on Machine Learning (1995), pp. 108–14.
[24] K.C. Chou, C.T. Zhang, "Prediction of protein structural classes", Critical Reviews in Biochemistry and Molecular Biology (1995), vol. 30, pp. 275–349.
[25] Y.D. Cai, K.Y. Feng, W.C. Lu, K.C. Chou, "Using LogitBoost classifier to predict protein structural classes", Journal of Theoretical Biology (2006), vol. 238, pp. 172–6.
[26] G.P. Zhou, K. Doctor, "Subcellular location prediction of apoptosis proteins", Proteins: Structure, Function, and Genetics (2003), vol. 50, pp. 44–8.
[27] G.P. Zhou, N. Assa-Munt, "Some insights into protein structural class prediction", Proteins: Structure, Function, and Genetics (2001), vol. 44, pp. 57–9.
[28] G.P. Zhou, "An intriguing controversy over protein structural class prediction", Journal of Protein Chemistry (1998), vol. 17, pp. 729–38.
[29] J.L. Moreland, A. Gramada, O.V. Buzko, Q. Zhang, and P.E. Bourne, "The molecular biology toolkit (mbt): A modular platform for developing molecular visualization applications", BMC Bioinformatics (2005), 6:21.

Authors

Koosha Golmohammadi is a M.Sc. student at the University of Alberta, Department of Electrical and Computer Engineering in Edmonton - Alberta, Canada since January 2007. His research interests include machine learning, intelligent agents, data mining, and the semantic web.

He has competed in RoboCup international competitions, an international joint project to foster AI and intelligent robotics in the last 3 years and received many awards. He is the member of the IEEE Computational Intelligence Society.

Lukasz Kurgan received his M.Sc. degree (with honors) in automation and robotics from the AGH University of Science and Technology, Krakow, Poland in 1999 and his Ph.D. degree in computer science from the University of Colorado at Boulder, U.S.A. in 2003.

Dr. Kurgan is with the Department of Electrical and Computer Engineering at the University of Alberta in Edmonton, Canada. His research interests include development and application of modern data mining methods in bioinformatics, with focus on analysis of sequence, structure, and function of biologically interesting macromolecules. He currently serves as an associate editor of *Neurocomputing*, *Open Proteomics Journal*, and *Journal of Biomedical Science and Engineering*. Dr. Kurgan is a member of the IEEE, ACM, ISCB, MITACS, and CAN.

Brendan Crowley received his B.Sc. in Electrical Engineering with distinction from the University of Alberta, Edmonton, AB, Canada, in 2006. He is currently pursuing a M.Sc. degree in Electrical and Computer Engineering, also at the University of Alberta. His thesis work is on the low-power design and implementation of low-density parity-check (LDPC) codes.

His research interests include VLSI design for error control coding and bioinformatics applications. He is a member of APEGGA and the IEEE.

Marek Reformat received his M.Sc. (with honors) from Technical University of Poznan, Poland, and his Ph.D. from University of Manitoba, Winnipeg, Manitoba, Canada. Currently, he is with the Department of Electrical and Computer Engineering at the University of Alberta, Edmonton, Alberta, Canada. His research interests are in the areas of application of computational intelligence techniques, such as neurofuzzy systems and evolutionary computing, as well as probabilistic and evidence theories to intelligent data analysis and modeling leading to translating data into knowledge. He applies these methods to conduct research in the areas of software and knowledge engineering. Dr. Reformat has been a member of program committees of several conferences related to computational intelligence and evolutionary computing. He is a member of the IEEE and ACM.