http://www.scirp.org/journal/jbise

J. Biomedical Science and Engineering, 2017, Vol. 10, (No. 8), pp: 390-401

Combined Use of k-Mer Numerical Features and Position-Specific Categorical Features in Fixed-Length DNA Sequence Classification

Dau Phan[1], Ngoc Giang Nguyen[1], Favorisen Rosyking Lumbanraja[1], Mohammad Reza Faisal[1], Bahriddin Abapihi[1], Bedy Purnama[1], Mera Kartika Delimayanti[1], Mamoru Kubo[2], Kenji Satou[2]

[1]Graduate School of Natural Science and Technology, Kanazawa University, Kanazawa, Japan; [2]Institute of Science and Engineering, Kanazawa University, Kanazawa, Japan

Correspondence to: Dau Phan, pdaukg@gmail.com

Keywords: Sequence Classification, Numerical and Categorical Features, Feature Selection

Received: May 27, 2017 **Accepted:** August 27, 2017 **Published:** August 30, 2017

ABSTRACT

To classify DNA sequences, k-mer frequency is widely used since it can convert variable-length sequences into fixed-length and numerical feature vectors. However, in case of fixed-length DNA sequence classification, subsequences starting at a specific position of the given sequence can also be used as categorical features. Through the performance evaluation on six datasets of fixed-length DNA sequences, our algorithm based on the above idea achieved comparable or better performance than other state-of-the art algorithms.

1. INTRODUCTION

In recent years, biological data have been generated at a tremendous rate. According to [1], the number of DNA sequences contained in GenBank repository increased dramatically from 116,461,672 to 181,336,445 between February 2010 and February 2015. The sequences in UniProt doubled during the period of just one year, from 40.4 (June 2013) to 80.7 (August 2014) million [2]. Analysis and interpretation of these data are two of the most crucial tasks in bioinformatics, and classification and prediction methods are key techniques to address such tasks.

As summarized by Xing *et al.* [3], the sequence classification methods can be categorized into three main groups. The first class is distance-based method, which defines distance functions to compute the similarity between two sequences. After that, some of the current classification methods, such as k-nearest neighbor classifier, are applied. The second category is feature-based methods. Before employing conventional algorithms such as decision trees and Support Vector Machine (SVM) to address the problem, sequences are converted into numerical feature vectors. In order to improve the accuracy of prediction, feature selection plays a key role in this type of methods. The last type is model-based classification,

which applies hidden Markov model (HMM) and other statistical models to perform sequence classification.

With regard to the first category, in the research by Borozan *et al.* [4] in 2015, they exploited the complementarity between alignment-free and alignment-based similarity measures to improve biological sequence classification performance. They used five different sequence similarity measures: three of them were alignment-free and two of them were alignment-based, which revealed that their model outperformed previous models. In 2014, Chen *et al.* [5] also tackled the problem of categorical data in a typical distance-based manner. They defined four weighted functions for categorical features, two of them named as simple matching coefficient measures with global weights ($WSMC_{global}$) and the other two named as simple matching coefficient measures with local weights ($WSMC_{local}$), then applied these functions to formulate new nearest neighbor classification algorithms. The classifiers were evaluated by using real datasets and biological datasets. The results showed that their proposed classifiers outperformed the traditional methods.

Moving to the second class, the application of feature selection technique and feature-based method to classify protein sequence data was carried out by Iqbal *et al.* [6] in 2014. The experimental results of their research showed that their model significantly improved in terms of accuracy, sensitivity, specificity, F-measure, and other performance measure metrics. In the study of Weitschek *et al.* [7] in 2015, they used the combination of alignment-free approaches and rule-based classifiers so as to classify biological sequences. At first, the biological sequences were converted into numerical feature vectors with alignment-free techniques, then rule-based classifiers were applied in order to assign them to their taxa.

The study about classifying occupancy, acetylation, and methylation of nucleosomes was carried out by Pham *et al.* [8]. Their method was also a kind of feature-based classification, which converted sequences into numerical feature vectors, then applied a conventional classification method. They adopted SVM with RBF kernel, and feature vectors were k-mer based vectors with a variety of window sizes (k = 3, 4, 5, 6, etc.). Using 10 datasets derived from a research of Pokholok *et al.* [9], they gained a high prediction accuracy. In order to improve prediction accuracy, a technique termed feature selection was used by Higashihara *et al.* [10] to solve this problem. In this research, the importance of features was first measured by MeanDecreaseGini value computed through training and prediction by random forest, then features were ranked as the order from the most to least importance. Exploiting feature selection along feature ranking, they achieved slight improvements in prediction accuracy. What is more, by searching neighbors of the best feature subset, accuracy of prediction improved further.

Although the active researches on sequence classification above, numerical and categorical features were separately studied until now. Since the numerical features like k-mer are typically position-independent and categorical features like nucleotide at a position are position-specific, we can expect that these two types of features could contribute to the classification performance in a complementary manner. In addition, it is still unclear how effective a feature selection algorithm is against the union of numerical and categorical features of sequence. In this study, we propose an effective framework for improving fixed-length DNA sequence classification by using the combination of categorical features (*i.e.* subsequence at a position like "A", "AG", etc.) and numerical features (*i.e.* k-mer frequency). By conducting feature selection on this mixture of features, we could also find which type of features is more effective in each dataset.

The remainder of this paper is organized as follows. In Section 2, we describe the validation datasets and how the model works. We present the experiments and results of evaluating the model in Section 3. Finally, some conclusions and discussions are given in Section 4.

2. MATERIALS AND METHODS

2.1. Datasets

To show the validity of the proposed method in dealing with genetic sequence classification problem,

we applied our approach to six datasets listed in Table 1.

2.1.1. UCI Datasets

We chose the benchmark datasets from UCI machine learning repository, Splice and Promoter datasets, for evaluation of our model. These datasets were used in researches [11, 12]. The Splice dataset is about the splice-junction gene sequences. There are two types of splice junctions. The exon-intron "EI" is the part of DNA sequence ranging from the ending of an exon and the starting of an intron while intron-exon "IE" is a region of DNA between the ending of an intron and beginning of exon. The part of sequence which does not belong to "IE" and "EI" is called no junction "N". This dataset is composed of 3175 labeled samples and each sample has the length of 60 base pair.

During RNA transcription process, transcription factors such as RNA polymerase and accessory proteins bind to the promoter region and carry out the initiation of transcription. Promoter parts are DNA sequences located adjacent to the initial sites of transcription. Promoter dataset consists of 106 labeled promoter sequences, "Positive" and "Negative", with length of 57 base pair. "Positive" sequence contains a DNA region from promoter whereas "Negative" sequence does not include a DNA from promoter.

2.1.2. Nucleosome Benchmark Datasets

The other four datasets are about nucleosome forming and inhibiting sequences of four species (*H. sapiens*, *C. elegans*, *D. melanogaster*, and *S. cerevisiae*). The first three datasets were collected by Guo *et al.* [13]. These datasets were previously used in the research [13-15] and their details are described as follows. Human (*H. sapiens*) involved 2273 nucleosome-forming sequences (positive) and 2300 nucleosome-inhibiting sequences (negative). Worm (*C. elegans*) includes 2567 nucleosome-forming sequences (positive) and 2608 nucleosome-inhibiting sequences (negative). Fly (*D. melanogaster*) contains 2900 nucleosome-forming sequences (positive) and 2850 nucleosome-inhibiting sequences (negative). All the sequences in these three datasets have the same length of 147 base pair. In addition, Yeast (*S. cerevisiae*) consists of 1880 nucleosome-forming sequences (positive) and 1740 nucleosome-inhibiting sequences

Table 1. Description of datasets.

No.	Dataset	Description	# Classes	# Sample	Sequence length (base)
1	Splice	Primate splice-junction gene sequences with associated imperfect domain theory	3	3175 (762 + 765 + 1648)	60
2	Promoter	*E. coli* promoter gene sequences with partial domain theory	2	106 (53 + 53)	57
3	Human	*H. sapiens* nucleosome-forming and nucleosome-inhibiting sequences	2	4573 (2273 + 2300)	147
4	Worm	*C. elegans* nucleosome-forming and nucleosome-inhibiting sequences	2	5175 (2567 + 2608)	147
5	Fly	*D. melanogaster* nucleosome-forming and nucleosome-inhibiting sequences	2	5750 (2900 + 2850)	147
6	Yeast	*S. cerevisiae* nucleosome-forming and nucleosome-inhibiting sequences	2	3620 (1880 + 1740)	150

(negative). Each of these sequences has the length of 150 base pair and this dataset was used in [15, 18, 19].

2.2. Features

In this study, we used the combination of the five different vectors named as 1-categorical vector (1CAT), 2-categorical vector (2CAT), 2-mer vector (2MER), 3-mer vector (3MER), and 4-mer vector (4MER). Given a biological sequence s of length n, S_1, S_2, ⋯, S_n, where $S_i \in$ {A, C, G, T} and $i = 1, 2, \cdots, n$, each of these vectors can be defined as follows:

2.2.1. 1-Categorical Vector (1CAT)

1CAT = (A_1, A_2, ⋯, A_n), where n is the length of sequence s and A_i is a nucleotide at position i^{th}, $i = 1, 2, \cdots, n$. For example, with sequence s, AGGTCCTACT, 1CAT = (A, G, G, T, C, C, T, A, C, T).

2.2.2. 2-Categorical Vector (2CAT)

2CAT = (B_1, B_2, ⋯, B_{n-1}), where n is the length of DNA sequence s and B_i is two consecutive nucleotides from position i^{th} to position at $(i+1)^{th}$, $i = 1, 2, \cdots, n-1$. For instance, with sequence s above, 2CAT = (AG, GG, GT, TC, CC, CT, TA, AC, CT).

2.2.3. 2-Mer Vector (2MER), 3-Mer Vector (3MER), and 4-Mer Vector (4MER)

In term of biological sequence, k-mers can be defined as all possible subsequences of length k within a sequence. A k-mer is a string of k successive nucleotides contained the genetic sequence and there are 4^k possible k-mers: $s_1, s_2, \cdots, s_{4^k}$. The k-mer vector denoted as kMER is defined by $\text{kMER} = \left(c[s_1], c[s_2], \cdots, c[s_{4^k}]\right)$, where $c[s_i]$ is a number of occurrences of the s_i in a sequence s and $i = 1, 2, \cdots, 4^k$. Therefore, using sequence s above, 2MER will be (0, 1, 1, 0, 0, 1, 0, 2, 0, 0, 1, 1, 1, 1, 0, 0).

2.3. Algorithm

The proposed algorithm consists of main four steps. The flowchart of our algorithm is shown in **Figure 1**, and works as below:

1) Block A in **Figure 1** is in charge of converting DNA sequences into feature vectors.

2) At Block B in **Figure 1**, feature ranking is conducted by the randomForest function for R [17].

3) Block C in **Figure 1** is responsible for feature selection by performing learning and predicting with the ksvm function for R in kernlab package [16]. Each feature subset is evaluated by the average of prediction accuracies of 10-fold cross-validation.

4) The prediction performance of the proposed approach is achieved at Block D in **Figure 1** where the best feature subset {f_1, ⋯, f_k} obtained in the previous step are used to evaluate by 10-fold cross-validation ten times. Herein, the best feature subset is the feature subset with the best accuracy.

2.4. Feature Selection

Nowadays, there has been a remarkable increase the number of researches exploiting feature selection techniques. They have been applied to both supervised learning and unsupervised learning. Their aims are threefold: the most important one being to avoid overfitting and improve model performance, the second advantage being to reduce computational time and space required to execute models, and the final goal being to identify which features are relevant to a problem and to gain a deeper insight into the data.

The feature selection approach used in our research is a kind of greedy algorithm, and works as two following steps.

Step 1) With pre-calculated feature set F = {f_1, f_2, ⋯, f_m} being ordered top-to-bottom as highest-to-smallest important values, we evaluate a feature subset {f_1, f_2, ⋯, f_{10}}, a feature subset {f_1, f_2, ⋯, f_{20}}, and so on, until a feature subset {f_1, f_2, ⋯, f_m} by conducting training and predicting with ksvm function [16].

Step 2) Neighbors of the feature subset with the best accuracy of prediction in the preceding step are tested.

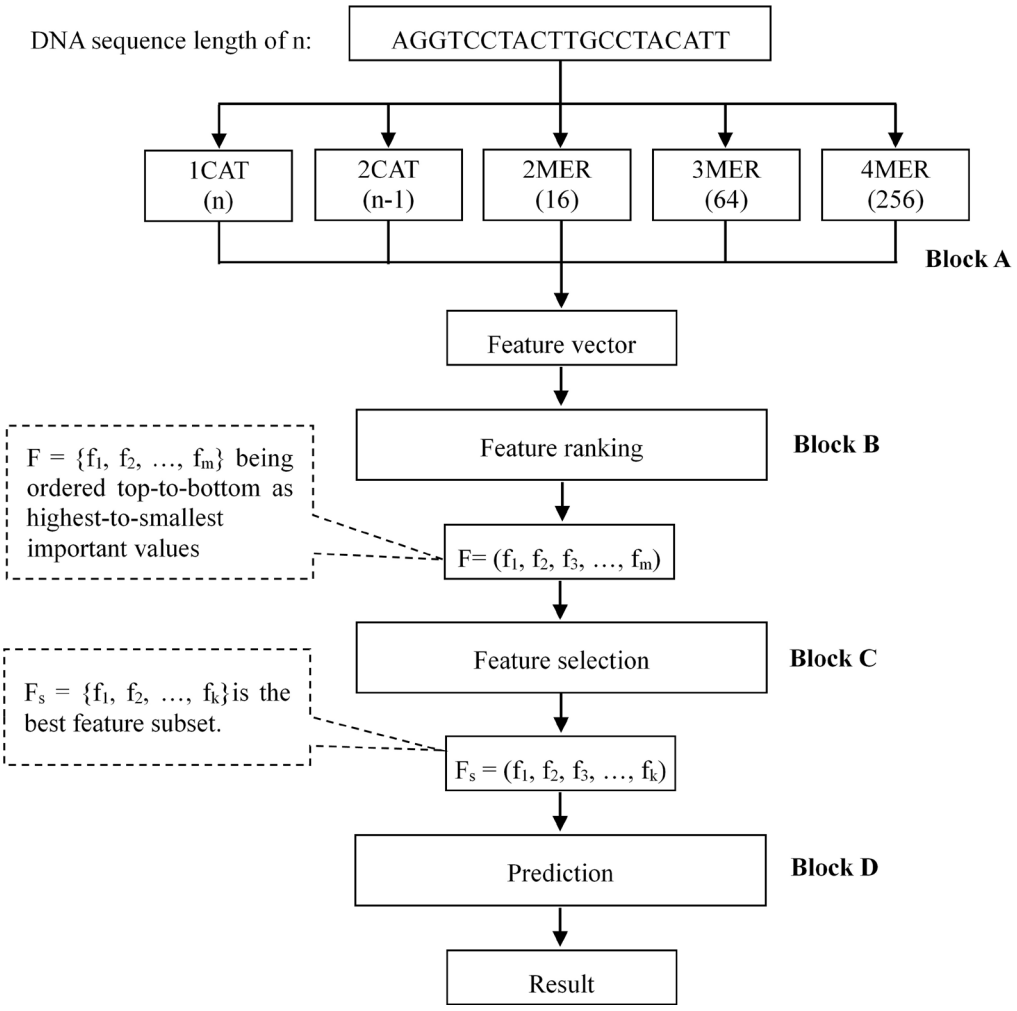


Figure 1. The flowchart of the proposed algorithm.

3. EXPERIMENTS AND RESULTS

3.1. Feature Ranking by Random Forest

Random forests are well-known ensemble learning method which can conduct both classification and regression. Apart from these tasks, the randomForest function for R in randomForest package [17] can measure the importance of all features by Mean Decrease Accuracy or Mean Decrease Gini values. In this research we adopted the Mean Decrease Accuracy value as the importance of features. The connection between rank and Mean Decrease Accuracy normalized into the range between 0 and 1 in each dataset is shown in the Figure 2.

In general, there is a sharp decrease in the importance of features in Promoter and Human datasets in the areas of top 1 - 5. This is then followed by a steady decline trend in the rest region. With Splice, Worm and Fly datasets, the importance of features fall slowly in the region of from top 1 to 16. The importance in the remainder declines gradually.

Features with high importance in validation datasets are listed in Table 2. From this table it is clear that Human, Worm, and Fly datasets have features with high importance containing mainly "A" (adenine) and "T" (thymine). However, the percentage of "C" (cytosine) and "G" (guanine) increase slightly in the Fly dataset. More observations are that TTT, TTTT, AA, AAA, AAAA, AAAT, ATTT features are highly important in Human and Worm datasets. It is similar to the case of Fly dataset where TTT, TT, TTTT,

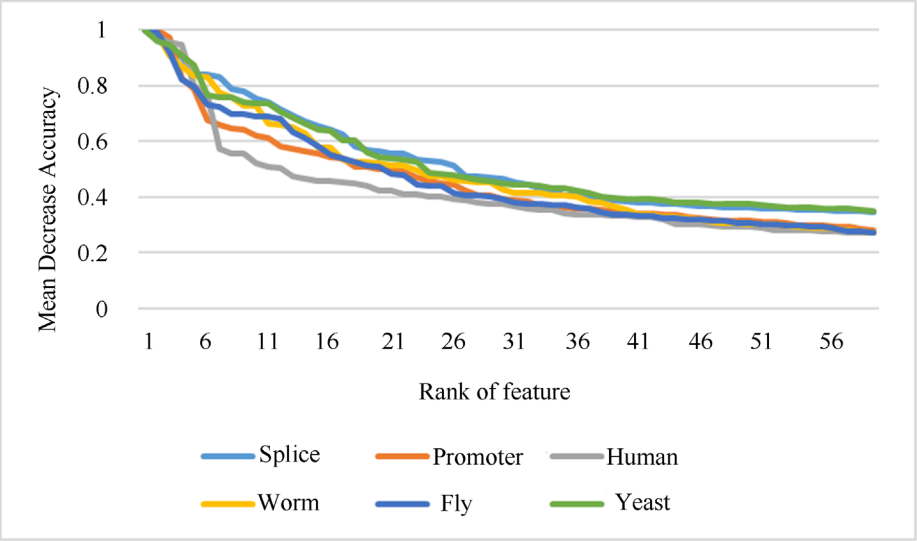


Figure 2. Mean Decrease Accuracy along feature ranking from top 1 - 60.

Table 2. List of important features.

No.	Dataset	List of top 10 features with high importance sorted by descending order of rank
1	Splice	B_{30}, B_{29}, B_{31}, B_{28}, A_{29}, A_{30}, B_{32}, A_{32}, B_{34}, A_{31}
2	Promoter	B_{17}, B_{16}, B_{15}, B_{14}, A_{15}, B_{39}, A_{17}, B_{18}, A_{16}, B_{38}
3	Human	TTTT, AAA, TTT, AAAA, TT, AA, AAAT, ATTT, TG, TAAA
4	Worm	B_1, AAA, AA, A_1, TTT, AAAA, AAAT, TTTT, ATTT, AATT
5	Fly	TA, GC, CG, TTT, TT, TTTT, ATA, CA, AAAA, TAT
6	yeast	AAAA, TTTT, TA, AAA, TTT, TAT, ATA, CGCG, CA, TT

ATA, AAAA, TAT are so important. AAAA, TTTT, TA, AAA, TTT, TAT, ATA are highly important for Yeast dataset. This coincides with the results in the research of Higashihara *et al.* [10] for classification of nucleosome datasets. The research showed that T and A were both highly important. Additionally in Table 2, it was partially demonstrated that the combination of numerical and categorical might be effective. In case of Worm dataset, the first and the fourth important features are categorical (B_1 and A_1), and others are numerical.

For Splice and Promoter datasets, however, features in 2CAT vectors are so important. B_{30}, B_{29}, B_{31}, B_{28} features are highly important in Splice dataset, which means that the nucleotides around the center of splice sequences play a vital role in prediction. This finding agrees with the structure of splice site sequences where splice-junctions are at the middle of sequences. B_{17}, B_{16}, B_{15}, B_{14} are highly important in Promoter dataset. Figure 3 demonstrates the relationship between features in 2CAT vectors of Splice and Promoter datasets and Mean Decrease Accuracy normalized into the interval of [0, 1]. The figure illustrates that the highly important features in 2CAT vector of Splice dataset are located at the region of from 25 to 35. While those of Promoter dataset settled at the area of between 12 to 18.

3.2. Prediction Accuracy of Feature Subsets along Ranking

As described in step 1 in subsection 2.4, feature subsets along the ranking were assessed by SVM. With Human, Worm and Fly datasets, there are 63 different feature subsets at intervals of 10: $\{f_1, f_2, \cdots, f_{10}\}$, $\{f_1, f_2, \cdots, f_{20}\}$, $\{f_1, f_2, \cdots, f_{30}\}$, $\cdots$, $\{f_1, f_2, \cdots, f_m\}$ being tested. The prediction accuracy is based on the average

accuracy of 10-fold cross-validation. However, there are around 45 feature subsets for Promoter dataset and Splice dataset. The results of prediction are demonstrated in Table 3.

3.3. Prediction Accuracy of Neighbors around the Best Feature Subset

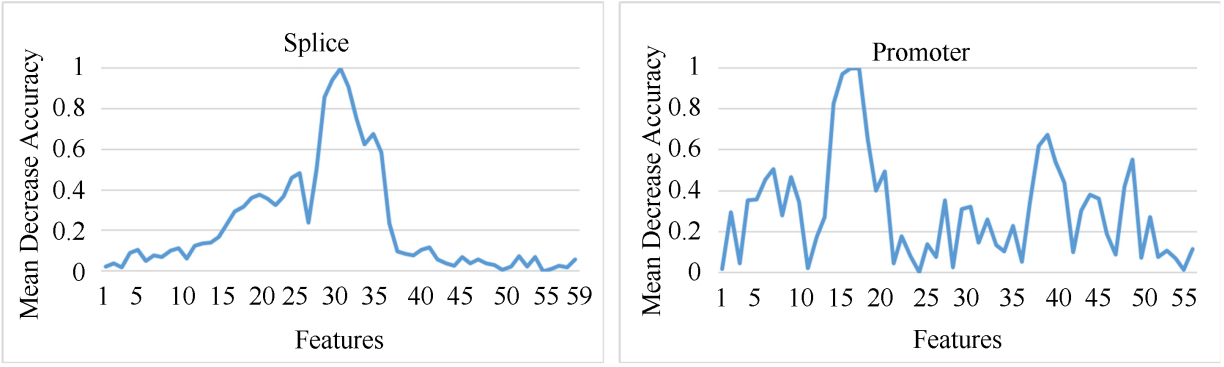
With best feature subset obtained in subsection 3.2, we conducted step 2 in subsection 2.4. The results and the number of features in each feature subset are presented in Table 4.

3.4. Evaluation

3.4.1. Evaluation Metrics

The six key terms are used to computing classification evaluation metrics. Positives (P) is the number of positive samples. Negatives (N) is the number of negative samples. True positives (TP) is the number of the positive samples that were correctly classified by the classifier. True negatives (TN) is the number of the negative samples that were correctly classified by the classifier. False positives (FP) is the number of the negative samples that were incorrectly classified as positive. False negatives (FN) is the number of the positive samples that are misclassified as negative.

$$\text{Accuracy}(\text{Acc}) = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$



(a) (b)

Figure 3. Mean Decrease Accuracy of features in 2CAT vector on (a) Splice and (b) Promoter datasets.

Table 3. Prediction accuracies obtained by using either the whole set of features and the best feature subset in step 1.

No.	Dataset	The whole set of features		The best feature subset in step 1		Improvement (%)
		# Feature	Acc (%)	# Feature	Acc (%)	
1	Splice	455	94.55	40	96.77	2.22
2	Promoter	449	94.34	90	100	5.66
3	Human	629	85.94	420	86.35	0.41
4	Worm	629	89.06	180	89.28	0.22
5	Fly	629	80.16	140	81.79	1.63
6	Yeast	635	100	30	100	0.00

Table 4. Prediction accuracies in step 2 compared with those in step 1.

No.	Dataset	The best feature subset in step 1		The best feature subset in step 2		Improvement (%)
		# Feature	Acc (%)	# Feature	Acc (%)	
1	Splice	40	96.77	48	96.93	0.16
2	Promoter	90	100	90	100	0
3	Human	420	86.35	428	86.49	0.14
4	Worm	180	89.28	177	89.53	0.25
5	Fly	140	81.79	148	81.93	0.14
6	Yeast	30	100	22	100	0.00

$$\text{Sensitivity}(\text{Sen}) = \frac{TP}{TP + FN} \tag{2}$$

$$\text{Specificity}(\text{Sp}) = \frac{TN}{TN + FP} \tag{3}$$

$$\text{Matthews correlation coefficient}(\text{MCC}) = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{4}$$

3.4.2. Performance Evaluation of the Method

Using the best feature subsets obtained in the step 2 in subsection 2.4, we applied our model to classify the DNA sequences in the validation datasets and compared its performance with the previous research. For evaluation, we mainly carried out 10-fold cross-validation ten times, and then computed average prediction results. With Promoter data, however, we employed leave one out ten times due to the fact that the number of its samples is small, 106 samples.

For Splice and Promoter datasets, we compared the performance of proposed model with the performance of previous model conducted by Nguyen *et al.* [12]. The results from this research are known as the best performance prior to our research. The motivation behind this model was the desire to apply a deep learning model for text classification to DNA sequence classification. At first, the researchers translated DNA sequence into sequence of words as a text sentence, then applied the representation technique for text to this produced sequence. Lastly, two-dimensional matrices representing DNA sequences using one hot vectors were directly used as input to the convolutional neural network model.

However, for Human, Worm and Fly datasets, the performance of our model was compared with the results taken from researches [13-15]. The predictors iNuc-PseKNC was proposed by Guo *et al.* in 2014 [13] and two years later, Tahir and Hayat introduced the predictors iNuc-PseSTNC in 2016 [14]. Both predictors achieved better results on predicting nucleosome positioning than previous predictors. In late 2016, Awazu developed two models predicting nucleosome positioning called as 3LS and TNS models [15].

For Yeast dataset, we compared our results with those taken from [15, 18, 19]. Chen *et al.* [18] developed the predictor based on DNA deformation energy in 2015 while Yi *et al.* [19] introduced the predictor based on the nearest neighbor algorithm in 2012.

3.5. Comparison with Other Methods

As can be seen from **Table 5**, the prediction accuracy of our method for Promoter dataset reached

100%. This means that all samples in this dataset were correctly predicted by our proposed model. This result has not obtained by any previous methods. Our method also achieved the high prediction accuracy for Splice dataset with 96.81%.

To confirm whether the average of prediction accuracies of our method and the previous method are significantly different, we performed the two-sample t-test assuming equal variances. The p-values of these t-test comparisons are illustrated in **Table 6**. All the p-values are far smaller than 0.05, which means that our method outperforms the previous research in the term of prediction accuracy on these two datasets.

With Human, Worm and Fly datasets, we compared the performance of the proposed model and the performances of models in [13-15] on four metrics: Accuracy, sensitivity, specificity and Matthews correlation coefficient. **Table 7** indicates the results of all methods in detail. From this table, the first thing to note is that our method outperformed all of competing methods on Worm dataset with Acc of 89.35%, Sen of 92.45% and MCC of 0.79. The second result worth pointing out is that on the Fly dataset our model also achieved better results than those of the other previous models with Acc of 81.75%, Sen of 79.14%, Sp of 84.40% and MCC of 0.64 except 3LS. Moreover, on Human dataset, the prediction Acc of the proposed method (86.33%) was higher than that of iNuc-PseKNC, TNS but lower than iNuc-PseSTNC and 3LS. Although iNuc-PseKNC model achieved the same MCC (0.73) with our model, Acc, Sen of our method were better than those of iNuc-PseKNC except for sensitivity. For Yeast dataset, our method and TNS completely outperformed the previous methods. Our model achieved the Acc of 100%, Sen of 100%, Sp of 100% and MCC of 1.0.

4. DISCUSSION AND CONCLUSIONS

In this research, we proposed a simple but powerful model for solving DNA sequence classification problems. The model was tested on six different datasets: Splice, Promoter, Human, Worm, Fly, and Yeast datasets. On Splice and Promoter datasets, the experimental results show that there was a significant increase in the performance of our model. The improvements were also proved by performing the two-sample t-test assuming equal variances, and all p-values were less than 0.05. Especially, the proposed model reached the accuracy of 100% on Promoter and Yeast datasets.

We also compared our model with the other four models: iNuc-PseKNC [13], iNuc-PseSTNC [14], TNS and 3LS [15]. In terms of accuracy, sensitivity and MCC, our method achieved better performance than any other competing method for predicting nucleosome positioning in worm genome. For fly genome, the proposed method also outperformed the other methods except 3LS model. For predicting nuc-

Table 5. Prediction accuracy comparison of proposed model and accuracy in [12].

No.	Dataset	Accuracy (%) in [12]			Accuracy by our method (%)			Improvement in average (%)
		Minimum	Maximum	Average	Minimum	Maximum	Average	
1	Splice	95.87	96.73	96.18	96.65	96.93	96.81	0.63
2	Promoter	99.06	99.06	99.06	100	100	100	0.94

Table 6. Assessment of our model and model in [12] by two-sample t-test assuming equal variances.

No.	Dataset	degrees of freedom	t-statistic	P(T ≤ t) one-tail	P(T ≤ t) two-tail
1	Splice	11	−4.365114612	0.000563331	0.001126662
2	Promoter	11	∞	0	0

Table 7. **Performance comparison of our model and previous models.**

Dataset	Method	Acc (%)	Sen (%)	Sp(%)	MCC
Human	Our method	86.33	89.77	82.93	0.73
	iNuc-PseKNC [13]	86.27	87.86	84.70	0.73
	iNuc-PseSTNC [14]	87.60	89.31	85.91	0.75
	3LS [15]	**90.01**	**91.69**	**88.35**	**0.80**
	TNS [15]	81.67	-	-	-
Worm	Our method	**89.35**	**92.45**	86.30	**0.79**
	iNuc-PseKNC [13]	86.90	90.30	83.55	0.74
	iNuc-PseSTNC [14]	88.62	91.62	86.66	0.77
	3LS [15]	87.86	86.54	**89.21**	0.76
	TNS [15]	83.94	-	-	-
Fly	Our method	81.75	79.14	**84.40**	0.64
	iNuc-PseKNC [13]	79.97	78.31	81.65	0.60
	iNuc-PseSTNC [14]	81.67	79.76	83.61	0.63
	3LS [15]	**83.41**	**84.07**	82.74	**0.67**
	TNS [15]	70.82	-	-	-
Yeast	Our method	**100**	**100**	**100**	**1.00**
	TNS [15]	**100**	-	-	-
	Chen *et al.* [18]	98.10	98.20	98.00	0.96
	Yi *et al.* [19]	99.06	-	-	-

leosome positioning in human genome, our method performance was higher than iNuc-PseKNC and TNS, but lower than the other two models. Therefore, it can be concluded that our model is effective for DNA sequence classification.

The combination vector can reflect not only the categorical features of DNA sequence, but also the numerical features of sequence. It can characterize a genetic sequence. Moreover, we utilized the ability of executing categorical data and numerical data of random forest and SVM to solve our problem. We also made use of the advantages of random forest in automatically producing variable importance to rank features, then applied the feature ranking to conduct feature selection. The used feature selection technique is a greedy based on technique which does not learning and predicting on all possible feature subsets. This can reduce dramatically computational cost. However, one limitation of this model is that all DNA sequences in one dataset need to be the same length.

Due to the fact that our proposed model was successful in classifying DNA sequence data, in the future, the proposed model can be extended to other areas of sequence recognition like the classification protein sequence data.

ACKNOWLEDGEMENTS

In this research, the super-computing resource was provided by Human Genome Center, the Institute of Medical Science, the University of Tokyo. Additional computation time was provided by the super computer system in Research Organization of Information and Systems (ROIS), National Institute of Genetics (NIG). This work was supported by JSPS KAKENHI Grant Number 26330328.

REFERENCES

1. GenBank and WGS Statistics. https://www.ncbi.nlm.nih.gov/genbank/statistics/
2. UniProt Consortium (2014) UniProt: A Hub for Protein Information. *Nucleic Acids Research*, **43**, D204-D212.
3. Xing, Z., Pei, J. and Keogh, E. (2010) A Brief Survey on Sequence Classification. *ACM SIGKDD Explorations Newsletter*, **12**, 40-80. https://doi.org/10.1145/1882471.1882478
4. Borozan, I., Watt, S. and Ferretti, V. (2015) Integrating Alignment-Based and Alignment-Free Sequence Similarity Measures for Biological Sequence Classification. *Bioinformatics*, **31**, 1396-1404. https://doi.org/10.1093/bioinformatics/btv006
5. Chen, L. and Guo, G. (2014) Nearest Neighbor Classification of Categorical Data by Attributes Weighting. *Expert Systems with Applications*, **42**, 3142-3149. https://doi.org/10.1016/j.eswa.2014.12.002
6. Iqbal, M.J., Faye, I., Samir, B.B. and Said, A.M. (2014) Efficient Feature Selection and Classification of Protein Sequence Data in Bioinformatics. *The Scientific World Journal*, **2014**, Article ID: 173869.
7. Weitschek, E., Cunial, F. and Felici, G. (2015) LAF: Logic Alignment Free and Its Application to Bacterial Genomes Classification. *BioData Mining*, **8**, 2015. https://doi.org/10.1186/s13040-015-0073-1
8. Pham, T.H., Tran, T.B., Ho, T.B., Satou, K. and Valiente, G. (2005) Qualitatively Predicting Acetylation and Methylation Areas in DNA Sequences. *Genome Informatics*, **16**, 3-11.
9. Pokholok, D.K., Harbison, C.T., Levine, S., Cole, M., Hannett, N.M., Lee, T.I., Bell, G.W., Walker, K., Rolfe, P.A., Herbolsheimer, E., Zeitlinger, J., Lewitter, F., Gifford, D.K. and Young, R.A. (2005) Genome-Wide Map of Nucleosome Acetylation and Methylation in Yeast. *Cell*, **122**, 517-527. https://doi.org/10.1016/j.cell.2005.06.026
10. Higashihara, M., Rebolledo-Mendez, J.D., Yamada, Y. and Satou, K. (2008) Application of a Feature Selection Method to Nucleosome Data: Accuracy Improvement and Comparison with Other Methods. *WSEAS Transactions on Biology and Biomedicine*, **5**, 153-162.
11. Li, J. and Wong, L. (2003) Using Rules to Analyse Bio-Medical Data: A Comparison between C4.5 and PCL. *Proceedings of Advances in Web-Age Information Management 4th International Conference*, Chengdu, 17-19 August, 254-265. https://doi.org/10.1007/978-3-540-45160-0_25
12. Nguyen, N.G., Tran, V.A., Ngo, D.L., Phan, D., Lumbanraja, F.R., Faisal, M.R., Abapihi, B., Kubo, M. and Satou, K. (2016) DNA Sequence Classification by Convolutional Neural Network. *Journal of Biomedical Science and Engineering*, **9**, 280-286. https://doi.org/10.4236/jbise.2016.95021
13. Guo, S.H., Deng, E.Z., Xu, L.Q., Ding, H., Lin, H., Chen, W. and Chou, K.C. (2014) iNuc-PseKNC: A Sequence-Based Predictor for Predicting Nucleosome Positioning in Genomes with Pseudo k-Tuple Nucleotide Composition. *Bioinformatics*, **30**, 1522-1529. https://doi.org/10.1093/bioinformatics/btu083
14. Tahir, M. and Hayat, M. (2016) iNuc-STNC: A Sequence-Based Predictor for Identification of Nucleosome Positioning in Genomes by Extending the Concept of SAAC and Chou's PseAAC. *Molecular BioSystems*, **12**, 2587-2593. https://doi.org/10.1039/C6MB00221H
15. Awazu, A. (2016) Prediction of Nucleosome Positioning by the Incorporation of Frequencies and Distributions of Three Different Nucleotide Segment Lengths into a General Pseudo k-Tuple Nucleotide Composition. *Bioinformatics*, **33**, 42-48. https://doi.org/10.1093/bioinformatics/btw562
16. Karatzoglou, A., Smola, A., Hornik, K. and Zeileis, A. (2004) Kernlab—An S4 Package for Kernel Methods in R. *Journal of Statistical Software*, **11**, 1-20. https://doi.org/10.18637/jss.v011.i09
17. Liaw, A. and Wiener, M. (2002) Classification and Regression by Randomforest. *R News*, **2**, 18-22. http://CRAN.R-project.org/doc/Rnews/
18. Chen, W., Feng, P., Ding, H., Lin, H. and Chou, K.C. (2015) Using Deformation Energy to Analyze Nucleosome

Positioning in Genomes. *Genomics*, **107**, 69-75. https://doi.org/10.1016/j.ygeno.2015.12.005

19. Yi, X.F., He, Z.S., Chou, K.C. and Kong, X.Y. (2012) Nucleosome Positioning Based on the Sequence Word Composition. *Protein and Peptide Letters*, **19**, 79-90. https://doi.org/10.2174/092986612798472811