International Journal of Nanomedicine

Dovepress
open access to scientific and medical research

Open Access Full Text Article

ORIGINAL RESEARCH

An ensemble method approach to investigate kinase-specific phosphorylation sites

This article was published in the following Dove Press journal:
International Journal of Nanomedicine
10 May 2014
Number of times this article has been viewed

Sutapa Datta
Subhasis Mukhopadhyay

Department of Biophysics, Molecular Biology and Bioinformatics and Distributed Information Centre for Bioinformatics, University of Calcutta, West Bengal, India

Abstract: Protein phosphorylation is one of the most significant and well-studied post-translational modifications, and it plays an important role in various cellular processes. It has made a considerable impact in understanding the protein functions which are involved in revealing signal transductions and various diseases. The identification of kinase-specific phosphorylation sites has an important role in elucidating the mechanism of phosphorylation; however, experimental techniques for identifying phosphorylation sites are labor intensive and expensive. An exponentially increasing number of protein sequences generated by various laboratories across the globe require computer-aided procedures for reliably and quickly identifying the phosphorylation sites, opening a new horizon for in silico analysis. In this regard, we have introduced a novel ensemble method where we have selected three classifiers (least square support vector machine, multilayer perceptron, and k-Nearest Neighbor) and three different feature encoding parameters (dipeptide composition, physicochemical properties of amino acids, and protein–protein similarity score). Each of these classifiers is trained on each of the three different parameter systems. The final results of the ensemble method are obtained by fusing the results of all the classifiers by a weighted voting algorithm. Extensive experiments reveal that our proposed method can successfully predict phosphorylation sites in a kinase-specific manner and performs significantly better when compared with other existing phosphorylation site prediction methods.

Keywords: post-translational modification, cell signaling, phosphate

Introduction

Protein phosphorylation, an important reversible post-translational modification, occurs due to the addition of a covalently bound phosphate group into certain acceptor residues (serine, threonine, and tyrosine) in the substrate sequence by a group of enzymes called kinases. It is one of the most ubiquitous post-transitional modifications, found across the phylogeny from prokaryotes to eukaryotes, and plays a major role in the broad range of critical cellular phenomenon such as metabolism,[1] cell signalling,[2,3] apoptosis,[4] and cellular proliferation.[3] It has been shown that almost 30%–50% of the eukaryotic proteins undergo phosphorylation.[5]

Kinases, by which the phosphorylation takes place, constitute one of the largest known protein superfamilies. About 1.7% of all the human genes encode as many as 518 different types of kinases, and they are classified into a hierarchical fashion with ten groups, 134 families, and 201 subfamilies primarily based on the homology of their catalytic domains.[6] Therefore, the accurate recognition of phosphorylation sites along with the relevant kinase of a eukaryotic protein of interest is necessary to decipher the intracellular phenomenon.

Correspondence: Sutapa Datta
Department of Biophysics, Molecular Biology and Bioinformatics, University of Calcutta, 92 APC Road, Kolkata-700009, India
Email sutapadatta2005@gmail.com

Although mass spectrometry techniques are being used to detect the phosphorylation sites in a high-throughput manner, the ever-increasing number of protein sequences has rendered these methods to be prohibitively labor and cost intensive. As a result, the development of an accurate and automated in silico method for predicting phosphorylation sites based on the protein primary sequence information is desirable. A number of in silico methods have been proposed to predict phosphorylation sites; these methods can be roughly categorized into two groups. Some methods can only predict the phosphorylation sites, without providing any information about the specific kinase most likely to interact with the protein of interest; these methods include DISPHOS,[7] CKSAAP_PhSite,[8] PPRED,[9] Netphos,[10–12] PHOSIDA,[13] and AMS.[14] Other methods predict phosphorylation sites in a kinase-specific manner; in addition to predicting whether a candidate site is a phosphorylation site or not, they provide the kinase information of the target proteins. Such methods include Scansite,[15] KinasePhos,[16] NetphosK,[17] PPSP,[18] GPS,[19,20] Postmod,[21] BAE,[22] AMS 4.0 Server,[23] and Metapred.[24] The details of these methods are discussed in a recent review by Trost et al.[25] Most of the phosphorylation prediction methods use a single classifier; these methods sometimes do not produce satisfactory results due to their inherent limitations. In recent years, ensemble methods have received considerable attention in the machine learning community for increasing the effectiveness of a single classifier. A number of ensemble methods have been successfully applied in studying several biological problems such as predicting subcellular locations in proteins,[26,27] gene expression analysis,[28] protein–protein interaction site prediction,[29] and prediction of siRNA efficacy.[30] Some recent methods, such as Bagging-AdaBoost ensemble (BAE) proposed by Yu et al,[22] PHOsphorylation Site FindER (PHOSFER) by Trost et al,[31] and recently a method proposed by Gao et al,[32] have also used ensemble methods to predict phosphorylation sites. BAE is based on the Bagging and Adaboost ensemble approach, whereas PHOSFER uses random forest algorithms to predict phosphorylation sites in an organism-specific way. The method proposed by Gao et al uses a multitask learning framework to predict the phosphorylation site of four kinase groups. Although much progress has been made in the prediction of phosphorylation sites using various ensemble methods, there still exists a wide scope for improvement. None of these methods uses more than one feature scheme to train the classifiers, but a specific type of feature encoding scheme for precisely predicting phosphorylation sites of a protein sequence is not fully exploited. Therefore, no single feature encoding scheme can absolutely differentiate the phosphorylation sites from nonphosphorylation sites for all the kinases. In our proposed method, we have used three types of features: physiochemical features of amino acids, dipeptide composition, and protein–protein scoring for encoding a protein sequence. In the next step, we have used a feature selection method to remove the redundant features without deteriorating the performance of the method. Also, in order to construct a good ensemble method, the ensemble needs to construct accurate and diverse classifiers and to combine outputs from the classifiers effectively. Selecting a set of diverse classifiers that perform individually well is a nontrivial problem. In our proposed algorithm, we have adopted k-Nearest Neighbor (K-NN), multilayer perceptron (MLP), and least square support vector machine (LSSVM) as three diverse classifiers. The cardinal problem in predicting phosphorylation sites is the very small size of the dataset, because the number of known phosphorylation sites is very limited as compared to the nonphosphorylation sites. To overcome this problem, we have incorporated bootstrap resampling techniques to construct a number of datasets. Each of the resampled datasets generated from three feature encoding parameters is used as input in each of these three classifiers separately. The final result is obtained by fusing all the results obtained from the three classifiers through weighted voting. It is empirically demonstrated that an ensemble method can improve the prediction accuracy of the single classifiers performing independently.

Materials and method

Data preparation

In order to evaluate the performance of our proposed method and to compare it with other existing methods, we have extracted the phosphorylation sites from Phospho.ELM database (version 9.0 [available from http://phospho.elm.eu.org]).[33] Experimentally validated phosphorylation sites of eukaryotic cells for 299 types of different kinases are crated in Phospho.ELM database. Version 9.0 of this database contains 8,718 proteins from different vertebrate species covering 31,754 serine, 7,449 threonine, and 3,370 tyrosine instances. Each entry in the database provides information about the substrate proteins along with the exact positions of the residues phosphorylated by a given kinase. In our study, we have considered the kinase families having at least 100 known and experimentally validated phosphorylation sites. Nine kinase families, including eight serine/threonine kinase families (protein kinase A [PKA], protein kinase B [PKB], protein kinase C [PKC], casein kinase 2 [CK2],

cyclin-dependent kinase 1 [CDK1], mitogen-activated protein kinase [MAPK], calmodulin-dependent protein kinase II [CAM KII], glycogen synthase kinase 3 [GSK-3]) and one tyrosine kinase family (SRC) were chosen. In this study, we have considered the phosphorylation sites for the vertebrates only. We extracted the 21-mer sequence including 10 bp upstream to 10 bp downstream from the phosphorylation site with the phosphorylated residue at the central position; ie, at position 11. If the phosphorylated residue (serine, S; threonine, T; tyrosine, Y) appears in the first ten residues or last ten residues, that is, if the distance between the phosphorylated residue and N or C terminal is less than 10-mer, an extra residue X (where X denotes any residue) is added before the first residue or after the last residue of the sequence in order to keep the protein sequence size at 21-mer and phosphorylated residue centered.

A negative dataset is prepared by taking the 21-mer peptides and centering all the nonphosphorylated S/T/Y residues of the substrate proteins corresponding to the nine kinase families mentioned above. We have taken nonphosphorylated S/T residues as negative control for the serine/threonine kinases in the PKA, PKB, PKC, CK2, CDK1, MAPK, CAM KII, GSK-3 kinase families and nonphosphorylated Y residues for tyrosine kinase family in SRC. In order to avoid the overestimation on accuracy while cross-validating, we have discarded the highly homologous sequences (ie, sequences having more than 60% similarity) from positive and negative datasets by using the CD-HIT clustering program.[34]

Furthermore, in context of the multiple kinase families the positive phosphorylation sites are outnumbered by the negative sites, resulting in an imbalanced dataset. As there are far more nonphosphorylated S/T/Y residues than phosphorylated residues, it is impractical to adopt whole nonphosphorylated sites as negative instances; it is preferable to select almost similar numbers of positive instances and negative instances. So in our study, the ratio of positive samples to negative samples is kept at 1:1.5 to avoid biased prediction. The number of positive phosphorylation sites and corresponding negative sites for the nine protein kinases are given in Table 1.

Table 1 Number of positive and negative phosphorylation sites for nine protein kinases after removing the redundant data obtained from Phospho.ELM database version 9.0

Kinase	Positive sites	Negative sites
PKA	288	440
PKB	70	105
PKC	279	420
MAPK	247	375
GSK-3	53	80
CDK1	133	200
CK2	187	281
CAM KII	67	105
SRC	120	180

Note: Phospho.ELM database (version 9.0) available from http://phospho.elm.eu.org.
Abbreviations: CAM KII, calmodulin-dependent protein kinase II; CDK1, cyclin-dependent kinase 1; CK2, casein kinase 2; GSK-3, glycogen synthase kinase 3; MAPK, mitogen-activated protein kinase; PKA, protein kinase A; PKB, protein kinase B; PKC, protein kinase C.

Method

This section describes the general framework of the method proposed in this article. It first introduces the data encoding methods followed by feature selection, data resampling, classification, and weighted voting methods. Figure 1 shows the block diagram of the proposed method.

Data encoding

Data encoding is one of the most crucial factors affecting the performance of the classifier as well as the ensemble architecture. This is the process through which a sequence is converted to its numerical form and is presented to the classifier. Hence, it is necessary to choose a high-quality data encoding method that possesses the salient features of the amino acid sequences, keeping the generated code compact in dimensionality. Instead of using a simple binary representation, we have adopted a variety of data encoding schemes that include three types of amino acids features: physicochemical features, dipeptide composition, and protein–protein similarity score.

Physicochemical features

The Amino Acid Index[35] (AAindex) is a database composed of numerical indices of various physicochemical and biochemical properties of amino acids and amino acid pairs. The latest version (version 9.1) of the AAindex database is divided into three sections: AAindex1, AAindex2 and AAindex3. AAindex1 contains 544 amino acid indices, AAindex2 contains 94 amino acid mutation matrices, and AAindex3 contains 47 statistical protein contact potential matrices. Here we have considered only the 544 indices of AAindex1 for the protein representation; AAindex2 and AAindex3 are matrices, and are not suitable for the protein sequence representation. Among the 544 amino acid indices, 13 had incomplete data or an overrepresentation of zeroes and were removed, leaving 531 properties as potential features for protein sequence representation.

Each amino acid residue in a sequence of length L is represented by the corresponding numerical value of the 531 amino acid indices in the AAindex1 database. Hence,

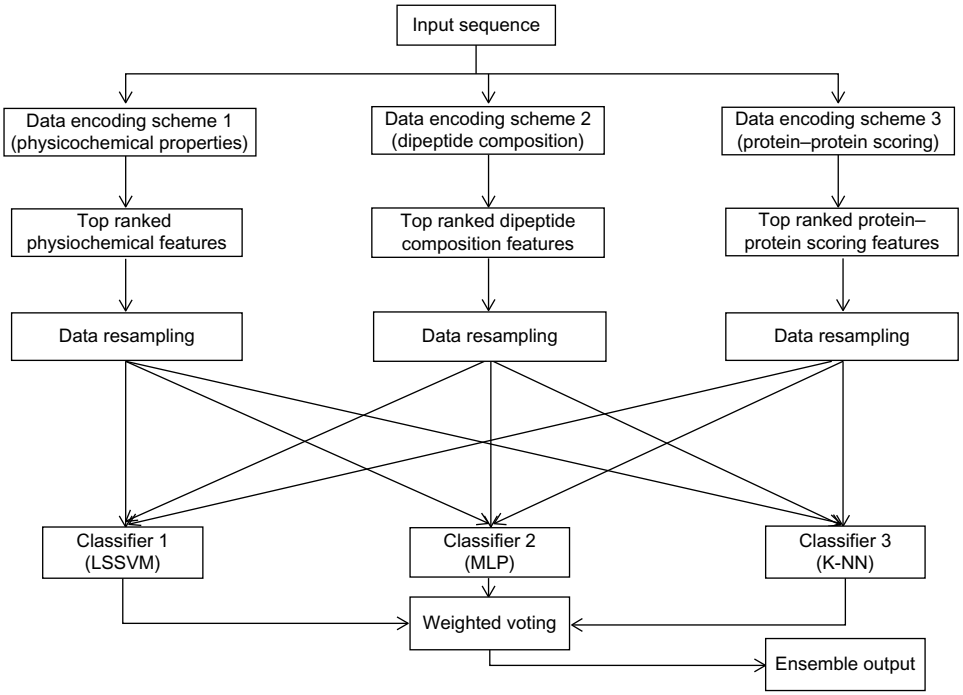

Figure 1 Block diagram of the proposed ensemble method that uses protein sequences as input.
Abbreviations: K-NN, K-nearest neighbor; LSSVM, least square support vector machine; MLP, multilayer perceptron.

a single sequence is represented by a 531*L dimensional matrix. Although only a moderate correlation exists among various AAindex properties, a considerable correlation exists between the physicochemical property values corresponding to different amino acids.

In order to correlate the physicochemical property values of amino acids along a sequence, physicochemical distance transformation is carried out to represent the sequence order information as proposed by Liu et al[36] in 2012. For a given protein sequence of length L, $A_1A_2A_3A_4$ ..... A_L, A_1 is the amino acid at position 1, A_2 at position 2, and so on. For an amino acid index j in AAindex1, the sequence order information associated with the jth physicochemical property can be given by the following equation:

$$\delta_\lambda^j = \frac{\sum_{i=1}^{L-\lambda} D_j(A_i, A_{i+\lambda})}{L-\lambda} \quad (1)$$

where λ is the distance between two amino acids along a sequence. $D_j(A_i, A_{i+\lambda})$ is given by the following sequence:

$$D_j(A_i, A_{i+\lambda}) = (I_j(A_i) - I_j(A_{i+\lambda}))^2 \quad (2)$$

where $I_j(A_i)$ and $I_j(A_{i+\lambda})$ are the normalized physicochemical property value of the ith and $(i + \lambda)$th amino acid in jth index. $I_j(A_i)$ and $I_j(A_{i+\lambda})$ are calculated using the following equation:

$$I_j(A_i) = \frac{\hat{I}_j(A_i) - \bar{I}_j}{\sigma} \quad (3)$$

where $\hat{I}_j(A_i)$ represents the raw physiochemical property value of the amino acid A_i for index j, $\bar{I}_j$ is the mean of the jth index of 20 amino acids, and σ is the variance. $\bar{I}_j$ and σ are given by:

$$\bar{I}_j = \frac{\sum_{i=1}^{20} \hat{I}_j(A_i)}{20} \quad (4)$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{20} (\hat{I}_j(A_i) - \bar{I}_j)^2}{20}}. \quad (5)$$

Using the above information, we have calculated the sequence order information δ for each physicochemical property in AAindex, taking λ=1, 2, and 3. We then averaged the three values of δ corresponding to λ=1, λ=2, and λ=3 to obtain δ_{avg}, which is then used as a feature vector for classification.

Dipeptide composition

Dipeptide composition is a simplistic descriptor of protein sequence features and a well-known technique for amino acid feature encoding. There are 441 combinations of dipeptides considering 21 amino acids (including one dummy amino acid). Dipeptide composition is defined as $fr(r,s) = N_{rs}/(N-1)$,

where r,s =1,2,3, … 20 and N_{rs} is the number of dipeptides of amino acid type r and s.[37] The value of each component of the dipeptide composition gives the fraction of the corresponding amino acid pairs in the sequence fragment. Here we have used dipeptide composition to represent the protein sequences in numerical form and have used as an input feature vector for further classification.

Sequence similarity scoring scheme

Another very popular sequence encoding method used in this study is the protein–protein sequence similarity scores. This scoring scheme is based on the hypothesis that the peptides having high similarity provide high scores; ie, if a peptide sequence receives a high score with another peptide, they are likely to be phosphorylated by the same kinase. It is assumed that the sequences having sequence similarity also bear similar structural and physicochemical properties. In our proposed method, we have used BLOSUM62 substitution matrix to calculate the similarity between two peptides.

The similarity score between two peptides A and B of length L is defined as

$$S(A,B)=\sum_{i=1}^{L} Score(A_i,B_i) \quad (6)$$

where Score(A_i,B_i) is the substitution score of amino acids A_i and B_i in the BLOSUM62 matrix. If $S(A, B) <0$, we have redefined it as $S(A, B) =0$. In this study, we have taken 10 bp upstream to 10 bp downstream of the phosphorylation site; ie, the length of each peptide sequence is L=21.

Feature selection

After encoding the protein sequences into its corresponding numerical form using three feature encoding schemes, we have applied a feature selection method in order to improve the performance of the classifiers because there may be some redundant and irrelevant features that may reduce the performance of the classifiers and increase the complexity. A feature is proved to be high quality if it not only can differentiate classes by itself or in combination with other features but if there remains no redundancy among the features. Here, we have adopted a method proposed by Mitra et al[38] called Feature Selection using Feature Similarity (FSFS) for feature selection. The method uses similarity or correlation among features to remove the redundant one; it does not require any knowledge of class labels as FSFS is an unsupervised feature selection algorithm. FSFS selects features by clustering them into subsets and then chooses a representative feature from each cluster. Then a maximum information compression index is used for enumerating feature similarity measures. The method significantly reduces the dimensionality of the features. In our proposed method, the numbers of features are varied from 100% to 2% and the number of features that yields the best result is opted. We have shown variation of accuracy with the

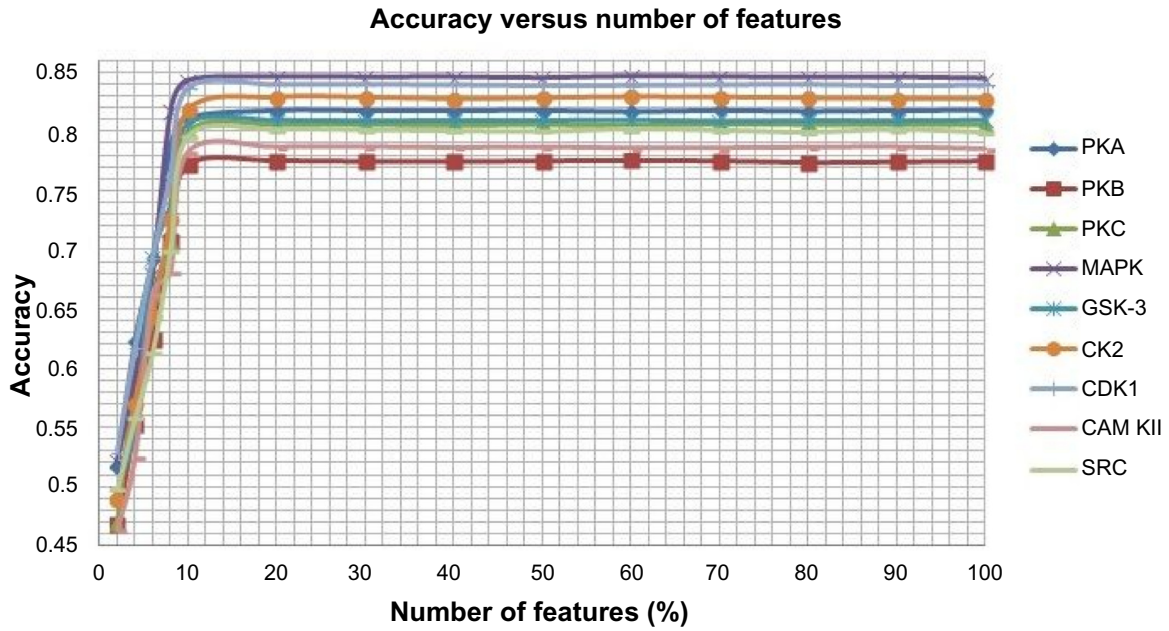

Figure 2 Average accuracy versus number of features for each of the nine kinases families.
Abbreviations: CAM KII, calmodulin-dependent protein kinase II; CDK1, cyclin-dependent kinase 1; CK2, casein kinase 2; GSK-3, glycogen synthase kinase 3; MAPK, mitogen-activated protein kinase; PKA, protein kinase A; PKB, protein kinase B; PKC, protein kinase C; SRC, tyrosin kinase SRC.

number of features in Figure 2. Based on the graph shown in Figure 2, we have selected 20% of features because for 20% and above, there is no significant change in the accuracy for all the kinases. The selected features are used to represent the protein sequences.

Data resampling

The three feature datasets generated by three data encoding methods are then divided into a set of new datasets using a bootstrap resampling method. In bootstrap resampling, a number of resampled subsets of the original dataset are generated by random sampling with replacement (so individual instances may appear in the subsets more than once) such that the size of each resampled subset is equal to the size of the original dataset. From the original dataset D of size n, a set of new datasets $\{D_1, D_2, D_3, D_4 \ldots D_m\}$ is generated each of size n' such that $n' = n$.

In the next step, each of the datasets generated through bootstrap resampling is used as input to a set of classifiers $\{C_1, C_2 \ldots C_1\}$. In this paper, we have used three supervised classifiers, LSSVM, MLP, and K-NN, for classification.

LSSVM

We have used LSSVM as a classifier. To avoid the high computational complexity of support vector machine for high-dimensional data, we have adopted the least square version of SVM. LSSVM simplifies the training procedure by avoiding the solving of quadratic programming problem.[39,40] Let us consider a linearly separable binary classification problem:

$$(x_i, y_i)_{i=1}^{n} \text{ and } y_i = \{+1, -1\} \tag{7}$$

where x_i is an n-dimensional feature vector and y_i is the label of this vector. LSSVM can be formulated as the optimization problem:

$$\min_{w,b,e} \tau(w,b,e) = \frac{1}{2} w'w + \frac{1}{2} C \sum_{i=1}^{n} e_i^2 \tag{8}$$

subject to the quality constraints

$$y_i[w'\varphi(x_i) + b] = 1 - e_i \tag{9}$$

where $C>0$ is a regularization factor, b is a bias term, w is the weight vector, e_i is the difference between the desired output and the actual output, and $\varphi(x_i)$ is a mapping function.

The Lagrangian for problem of Eq 8 is defined as follows:

$$L(w, e_i, b, \alpha_i) = \min_{w,b,e} \tau(w,b,e) - \sum_{i=1}^{n} \alpha_i \{y_i[w'\varphi(x_i) + b] - 1 + e_i\} \tag{10}$$

where α_i are Lagrange multipliers. The Karush–Kuhn–Tucker conditions for optimality,

$$\frac{\partial L}{\partial w} = 0 \rightarrow w = \sum_{i=1}^{i} \alpha_i y_i \varphi(x_i),$$

$$\frac{\partial L}{\partial e_i} = 0 \rightarrow \alpha_i = C e_i,$$

$$\frac{\partial L}{\partial b} = 0 \rightarrow \sum_{i=1}^{n} \alpha_i y_i = 0, \text{ and}$$

$$\frac{\partial L}{\partial \alpha_i} = 0 \rightarrow y_i[w'\varphi(x_i) + b] - 1 + e_i = 0,$$

are the solution to the following linear system:

$$\begin{bmatrix} 0 & -Y \\ Y & \varphi\varphi' + C^{-1} I \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ \overline{1} \end{bmatrix} \tag{11}$$

where $\varphi = [\varphi(x_1)'y_1, ..., \varphi(x_n)'y_n]$, $Y = [y_1, ..., y_n]$, $\overline{1} = [1, ..., 1]$, and $\alpha = [\alpha_1, ..., \alpha_n]$.

For a given RBF kernel function $K(x, x_i)$ and a new test sample point x, the LSSVM classifier is given by

$$f(x) = \text{sgn}\left[\sum_{i=1}^{n} \alpha_i y_i K(x, x_i) + b\right]. \tag{12}$$

Multilayer perceptron

Feedforward MLP has also been used to predict the phosphorylation sites specific to a kinase label. Multilayer perceptron is the most commonly used feedforward neural network. MLP is the extended form of perceptron having one or more hidden layer. The nodes in the input layer are connected to the nodes of the hidden layer, which in turn are connected to the nodes of the output layer. Each connection is associated with a weight. In the input layer, a number of real valued inputs are given and MLP generates a single real values output according to an activation function applied to the weighted sum of the outputs of the units in the preceding layer. The feedforward neural network is trained with a back-propagation learning algorithm to optimize the classification accuracy. The final error E at the output layer is the sum of squared differences of the desired outputs d_i and the actually calculated outputs o_i of each output unit i, and can be expressed as:

$$E = \sum_{i} (d_i - o_i)^2. \tag{13}$$

In this study, we have used Log sigmoid activation function $\log sig(n) = \frac{1}{(1+\exp(-n))}$ in the hidden layer and linear activation function a=purelin(n) in the output layer. The network was initialized with random weights and biases and was trained using the Levenberg–Marquadart algorithm. We have tested the network by varying the number of iterations from 500 to 3,000 and the learning rate 0.1 to 0.5, and averaged all outcomes to obtain the best possible result. A momentum term of 0.95 was added to increase the learning rate with stability. The performance of the network was measured in terms of mean square error. The lower the mean square error, the better the network's performance.

K-NN classifier

K-NN is one of the most fundamental and simple nonparametric methods used for classifying objects based on the closest training examples in the feature space. The K-NN algorithm works in two steps. First, for each query Q, the K-NNs from the training data are identified based on distance (Euclidean, Manhattan, etc). In our work we have used Euclidean distance to find the K-NNs of a given query. If X is a particular training sample with ith feature x_i and Q is a query with ith feature q_i, then the Euclidean distance between the training sample and query is given by

$$dist(X,Q) = \sqrt{\sum_{i=1}^{n}(x_i - q_i)^2} \tag{14}$$

where n is the total number of features.

In the second step, the query is assigned to the class most common among its K-NNs by a majority voting. Once we have obtained K-NNs of query Q using Euclidean distance, it is time for the neighbors to vote in order to predict Q's class. For each class cl, we count how many of the K neighbors have that class cl. If $R(Q)$ denotes the class of the query Q, then

$$R(Q) = \arg\max S(F, cl) \tag{15}$$

where $S(F,cl)$ denotes the number of samples x_h ($x_h \in F$, where F is the set of nearest neighbor of the sample Q) with label $l_h = cl$.

Weighted voting

The set of newly generated m resampled datasets $\{D_1, D_2, D_3, \ldots D_m\}$ for three types of features are given as input into the three individual classifiers. Since there are three different types of feature spaces generated by three sequence encoding methods and each feature space is divided into m resampled datasets, there are 3*m numbers of input datasets given to each of the classifiers, resulting in 9*m numbers of classifiers and 9*m numbers of outputs. The final output of the ensemble method is generated by fusing the outputs produced by 9*m numbers of individual classifiers through weighted voting.

Suppose that the classification results corresponding to the 9*m numbers of classifiers for a query protein sequence P are $R_1, R_2, R_3, \ldots, R_{9m}$ such that

$$R_n \in \{C_1, C_2, C_3, \ldots, C_9\} (n = 1, 2, 3, \ldots, 9m), \tag{16}$$

where $C_1, C_2, C_3, \ldots, C_9$ are the class levels corresponding to the protein sequences.

For a given query sequence P, the weighted sum of the base classifiers for each class is calculated by

$$V_i = \sum_{n=1}^{9m} \partial * w_n, \partial = \begin{cases} 1, & R_n = i \\ 0, & R_n \neq i \end{cases}, \tag{17}$$

where $9m$ is the number of classifiers, $i = C_1, C_2, C_3, \ldots, C_9$ is the class levels, R_n is the predicted class level by the nth classifier, and w_n is the weight associated with the nth classifier. Each classifier has been assigned a weight that would denote the contribution of the classifier to the prediction system. In this study, we have considered the overall accuracy of each classifier as the weight corresponding to that classifier.

The final classification result, the class associated with the sample P, is determined by the following equation:

$$C_{final} = \arg\max \sum_{j=1}^{9} V_j \tag{18}$$

where, V_j denotes the number of occurrence of each class level and is obtained from Eq 17.

Performance assessment

To evaluate the performance of the proposed method, a tenfold cross-validation is done. In tenfold cross-validation, each dataset is divided into ten equal subsets; nine subsets are used for training and one part is used for testing. This process was repeated for all ten parts.

The performance of the proposed method is measured by means of four parameters: recall, precision, F-measure, and accuracy. The measures are given by the following equations:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{19}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{20}$$

$$AC = \frac{TP + TN}{TP + FP + TN + FN} \quad (21)$$

$$\text{F-measure} = 2 * \frac{\text{Precision} * \text{recall}}{\text{Precision} + \text{Recall}} \quad (22)$$

where TP, FP, TN, FN represent the number of true positives, false positives, true negatives and false negatives respectively.

Results and discussion

Positive and negative phosphorylation sites for each kinase family were downloaded from Phospho.ELM database and filtered using the CD-HIT clustering program to get a nonhomologous dataset. The number of positive phosphorylation sites and negative sites corresponding to the nine protein kinases after removing the homologous sequences and data imbalance correction are given in Table 1. As discussed in the previous section, three feature encoding schemes were used to convert the sequences into their numerical form, and each of the encoded datasets was resampled using the bootstrap resampling method. We varied the number of resampled datasets, taking B=10, B=15, and B=20, where B is the number of datasets obtained through bootstrap resampling. The resampled datasets were used as input to the three classifiers. The whole method was repeated by varying the window size from 5 to 21, taking all the odd numbers within the range as the window size with the phosphorylated residue at the center position. The variation of accuracy with various window sizes for all nine kinases is shown in Figure 3A–I, varying the number of resampled datasets (B=10, B=15, and B=20). Figure 3A–I clearly indicates that window size 11 yielded the best result for almost all the kinases. Therefore, an optimal window size of 11 with the phosphorylated residue S/T/Y at center position was taken for further examination. The performance of each individual classifier is annotated in Tables 2–4 in terms of accuracy for three different encoding schemes while varying the number of resampled datasets from 10, 15, and 20. In order to enhance the performance of individual classifiers, we undertook a weighted voting, took the result of each classifier as input, and obtained the final result of our proposed method by using Eq 18. Table 5 shows the result of the ensemble method corresponding to the number of resampled datasets (B=10, B=15, and B=20).

Evaluating different feature schemes on different classifiers

From Tables 2–4, we find that for the three different feature encoding schemes, none of the three classifiers individually performed extensively well in predicting the phosphorylation

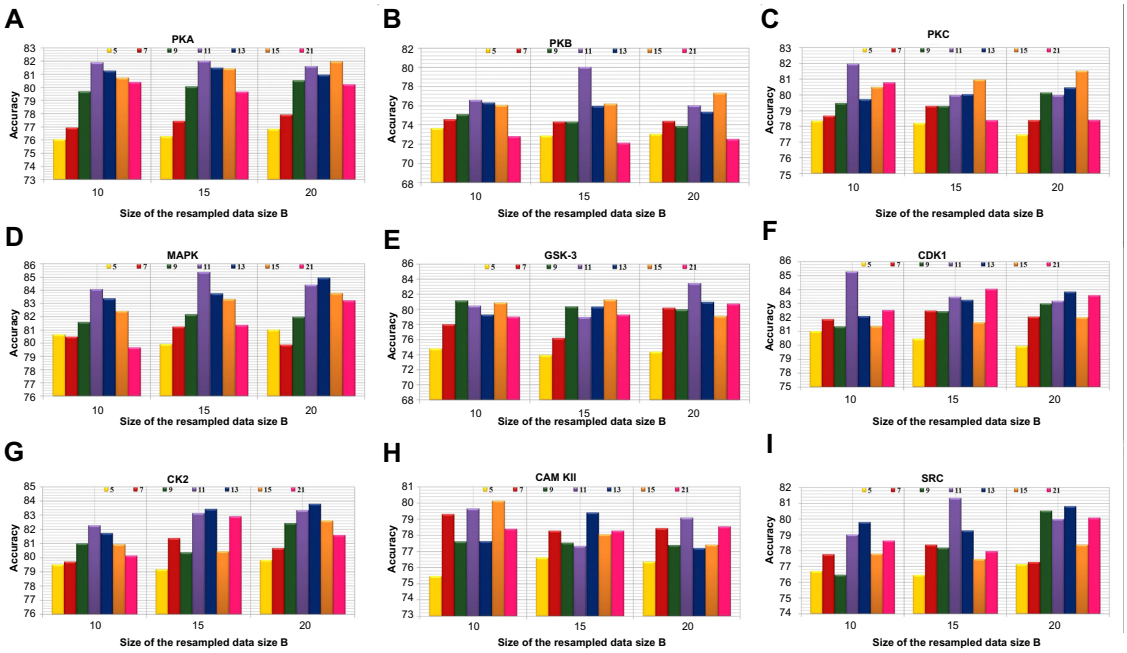

Figure 3 Accuracy with various window sizes for all the nine kinases while varying the number of resample datasets.
Abbreviations: CAM KII, calmodulin-dependent protein kinase II; CDK1, cyclin-dependent kinase 1; CK2, casein kinase 2; GSK-3, glycogen synthase kinase 3; MAPK, mitogen-activated protein kinase; PKA, protein kinase A; PKB, protein kinase B; PKC, protein kinase C; SRC, tyrosin kinase SRC.

Table 2 Performance of individual classifiers with different resampled data sizes in terms of accuracy (%)

Kinase	SVM			MLP			K-NN		
	B=10	B=15	B=20	B=10	B=15	B=20	B=10	B=15	B=20
PKA	65.49	62.81	64.16	60.97	58.64	63.27	67.26	67.25	69.57
PKB	66.06	65.02	62.5	59.42	62.34	60.38	67.52	67.36	66.66
PKC	62.24	64.56	63.61	61.74	58.03	58.64	68.8	67.57	68.6
MAPK	64.36	66.95	63.3	60.13	62.8	59.17	69.29	68.15	67.03
GSK-3	73.01	65.51	77.68	63	60.83	71.25	73	70.16	74.25
CDK1	68.71	69.55	63.58	66.25	64.9	60.77	70.95	71.2	66.17
CK2	68.75	62.41	63.7	64.41	60.61	59.82	69.18	68.58	68.06
CAM KII	66.02	68.93	61.7	61.19	58.81	58.56	70.28	67	69
SRC	67.95	66.28	64.73	64.39	64.59	59.5	71.44	69.62	66.81

Notes: Dipeptide composition is used for sequence encoding.
Abbreviations: CAM KII, calmodulin-dependent protein kinase II; CDK1, cyclin-dependent kinase 1; CK2, casein kinase 2; GSK-3, glycogen synthase kinase 3; K-NN, K-nearest neighbor; MAPK, mitogen-activated protein kinase; MLP, multilayer perceptron; PKA, protein kinase A; PKB, protein kinase B; PKC, protein kinase C; SRC, tyrosin kinase SRC; SVM, support vector machine.

sites. On average, all three features performed similarly for most of the kinases. For some kinases the physicochemical features performed well, whereas for other kinases the dipeptide composition performed better. Performance of Protein–protein scoring was average for all the kinases. Tables 2–4 do not show any significant trend of dominance of any one feature in predicting phosphorylation sites. Although the performance of the individual classifiers varied a lot, none of the single classifiers showed any satisfactory result. LSSVM as an independent classifier yielded the highest accuracy, 83.8%, for kinase family CDK1 for B=10 when physicochemical properties were used for data encoding; for all other kinases, LSSVM gave an accuracy in the range of 65%–80%. K-NN gave an accuracy between 55%–75% for all the kinases. Among the three classifiers, MLP performed worst, with an accuracy ranging between 50%–65%. In order to circumvent this insufficiency and to improve our results, we applied the weighted voting method as discussed in the previous section.

Performance of the ensemble method with weighted voting

In weighted voting, the outputs from various classifiers are fused to obtain a final output for each query sequence. Here, instead of relying on the outputs of the classifier, we assigned a weight to each classifier, which implies that the greater the weight of a classifier; the greater its contribution towards predicting the right sequence.

Table 5 shows the result of the ensemble method, and a remarkable improvement can be seen for all the kinase families. For the kinase family CDK1, the ensemble method yielded 85.3%, 83.4%, and 83.1% accuracy for B=10, B=15, and B=20, respectively. For MAPK kinase, the proposed method yielded a quite impressive accuracy: 84%, 85.3%, and 84.4% for B=10, B=15, and B=20, respectively. Our method gives the lowest accuracy for the kinase family PKB. Table 5 also shows the change in precision, recall, accuracy, and F-measure for the different kinases when the number of resampled datasets was varied from B=10,

Table 3 Performance of individual classifiers with different resampled data sizes in terms of accuracy (%)

Kinase	SVM			MLP			K-NN		
	B=10	B=15	B=20	B=10	B=15	B=20	B=10	B=15	B=20
PKA	66.25	65.43	63.44	55.67	60.06	56.85	59.25	59.72	56.94
PKB	71.69	78.61	66.98	55.42	69.26	59.09	67.61	68.57	63.8
PKC	68.85	61.72	60.76	64.84	57.88	54.7	61.33	61.09	55.36
MAPK	61.24	67.67	62.02	57.52	58.72	58.67	57.41	60.91	61.45
GSK-3	67	73.16	66.5	59.5	56.41	58.12	68.75	66.25	71.25
CDK1	83.8	65.53	65.9	75.5	60.26	58.22	77.5	61	57
CK2	67.35	64.9	67.42	57.36	59.69	56.97	58	64.41	60.49
CAM KII	78.2	71.2	73.6	61.98	61.58	56.788	74.25	60.39	68.31
SRC	66.44	65.48	68.55	58.83	58.18	55.3	63.88	75	60

Note: Physicochemical properties of amino acids are used for sequence encoding.
Abbreviations: CAM KII, calmodulin-dependent protein kinase II; CDK1, cyclin-dependent kinase 1; CK2, casein kinase 2; GSK-3, glycogen synthase kinase 3; K-NN, K-nearest neighbor; MAPK, mitogen-activated protein kinase; MLP, multilayer perceptron; PKA, protein kinase A; PKB, protein kinase B; PKC, protein kinase C; SRC, tyrosin kinase SRC; SVM, support vector machine.

Table 4 Performance of individual classifiers with different resampled data sizes in terms of accuracy (%)

Kinase	SVM			MLP			K-NN		
	B=10	B=15	B=20	B=10	B=15	B=20	B=10	B=15	B=20
PKA	64.32	63.47	63.6	59.01	58.24	60.01	62.26	62.58	61.98
PKB	68.18	67.92	68.04	56.82	56.46	57.21	66.64	65.82	65.57
PKC	63.22	64.38	63.47	62.17	61.84	61.39	65.39	64.22	66.28
MAPK	62.31	63.43	62.16	60.26	59.76	59.79	63.24	62.96	62.01
GSK-3	68.04	68.77	67.58	59.94	60.08	60.17	64.8	64.32	63.87
CDK1	64.71	64.58	63.29	60.72	59.36	61.27	66.48	65.26	68.03
CK2	69.24	66.83	65.42	62.17	63.85	61.9	64.74	62.02	62.72
CAM KII	66.87	64.91	64.7	61.22	59.78	58.23	64.19	63.34	65.81
SRC	67.92	64.28	64.86	61.06	61.58	60.16	61.8	63	64.62

Note: Protein–protein scoring is used for sequence encoding.

Abbreviations: CAM KII, calmodulin-dependent protein kinase II; CDK1, cyclin-dependent kinase 1; CK2, casein kinase 2; GSK-3, glycogen synthase kinase 3; K-NN, K-nearest neighbor; MAPK, mitogen-activated protein kinase; MLP, multilayer perceptron; PKA, protein kinase A; PKB, protein kinase B; PKC, protein kinase C; SRC, tyrosin kinase SRC; SVM, support vector machine.

B=15, and B=20. The result shows that a very small variation in accuracy is found in most of the kinases, including PKA, PKC, MAPK, CDK1, CK2, CAM KII, and SRC, with the change of B. The kinase families PKB and GSK-3 showed a slight variation in accuracy. For PKB, accuracy for B=10 and B=20 was 76.6% and 76%, respectively, which was almost the same; however, we got 80% accuracy for B=15. In the case of GSK-3, accuracy was 80.4%, 78.9%, and 83.5% for B=10, B=15, and B=20, respectively. However, none of these kinase families showed any trend of increase or decrease of accuracy corresponding with an increase or decrease of resampled data size. It might be possible that the presence of some atypical samples in these families increases the difference between resampled datasets and causes the variation of accuracy with the resampled datasets. Thus, the performance of the method is not affected by the number of the resampled datasets. Table 5 shows a balanced precision and recall, which means that the number of true positives is higher relative to the number of false positives and false negatives. As a result, a rather high F-measure is yielded for all the kinases using the proposed algorithm.

Performance comparison with other existing methods

In order to evaluate the performance of our method, we compared our method with five other open access kinase-specific phosphorylation site prediction methods. In most previous studies, these five methods were used because of their high performance. The five methods are PPSP,[18] KinasePhos 2.0,[16] GPS2.0,[19] Scansite,[15] and NetPhosK 1.0.[17] Bayesian decision theory was used to develop PPSP. KinasePhos employs the hidden Markov model to predict kinase-specific phosphorylation sites. Scansite searches for motifs within proteins that are likely to be phosphorylated by specific protein kinases, using the scores calculated from position-specific score matrices. NetPhosK 1.0 uses an artificial neural network to predict 17 kinase-specific phosphorylation sites. The GPS2.0 server uses a modified version of a group-based scoring algorithm[41,42] to predict PK-specific phosphorylation sites in the hierarchy. In this evaluation, we selected all kinase groups and the balance performance option for PPSP. In the case of NetPhosK, prediction without filtering and a threshold value of 0.5 was selected to predict phosphorylation sites. KinasePhos 2.0 was run with the option of default specificity for a specific kinase. In this work, Scansite 2.0 was run by searching all motifs, with the "high stringency level" setting selected. For GPS2.0, a medium threshold was selected for a particular kinase family.

To avoid biased prediction, we have considered a candidate site to be true positive only when the site is predicted correctly. The comparison is done on the basis of the parameter's precision, recall, accuracy, and F-measure. Figures 4–7 show the performance comparison of our proposed method with other methods. Figures 6 and 7 illustrate that the proposed method achieves the best performance in terms of accuracy and F-measure for all nine kinase families. PPSP performance was the second best with respect to accuracy and F-measure. Upon evaluating the results of this comparison, we determined that GPS2.0 performed almost the same as PPSP except in the case of CAM KII, where its performance was the worst among all methods in terms of precision, recall, accuracy, and F-measure. Scansite performed similarly to GPS2.0 for all the kinases except CAM KII. Scansite performance was the second best in terms of precision, accuracy, and F-measure for SRC, while for

Table 5 Results of the ensemble method through weighted voting on various resample data sizes for various kinases

Kinases	B=10				B=15				B=20			
	Precision	Recall	Accuracy (%)	F-measure	Precision	Recall	Accuracy (%)	F-measure	Precision	Recall	Accuracy (%)	F-measure
PKA	0.7746	0.7639	81.87	0.7692	0.7716	0.7743	82.01	0.7730	0.7674	0.7674	81.59	0.7674
PKB	0.6986	0.7286	76.57	0.7133	0.7397	0.7714	80.00	0.7552	0.6892	0.7286	76.00	0.7083
PKC	0.7647	0.7921	81.97	0.7782	0.7356	0.7778	79.97	0.7561	0.7372	0.7742	79.97	0.7552
MAPK	0.7868	0.8219	84.08	0.8040	0.8047	0.8340	85.37	0.8191	0.7907	0.8259	84.41	0.8079
GSK-3	0.7455	0.7736	80.45	0.7593	0.7193	0.7736	78.95	0.7455	0.7818	0.8113	83.46	0.7963
CDK1	0.8088	0.8271	85.29	0.8178	0.7910	0.7970	83.48	0.7940	0.7939	0.7820	83.18	0.7879
CK2	0.7653	0.8021	82.26	0.7833	0.7784	0.8075	83.12	0.7927	0.7685	0.8342	83.33	0.8000
CAM KII	0.7162	0.7910	79.65	0.7518	0.6892	0.7612	77.33	0.7234	0.7123	0.7761	79.07	0.7429
SRC	0.7244	0.7667	79.00	0.7449	0.7540	0.7917	81.33	0.7724	0.7344	0.7833	80.00	0.7581

Note: Precision, recall, accuracy, and F-measure are used for performance evaluation.

Abbreviations: CAM KII, calmodulin-dependent protein kinase II; CDK1, cyclin-dependent kinase 1; CK2, casein kinase 2; GSK-3, glycogen synthase kinase 3; MAPK, mitogen-activated protein kinase; PKA, protein kinase A; PKB, protein kinase B; PKC, protein kinase C; SRC, tyrosin kinase SRC.

other kinase families it gave an average performance. PPSP provided the highest precision value for CAM KII, followed by our method. But for kinases like PKA, PKB, PKC, MAPK, GSK-3, CDK1, CK2, and SRC, our proposed method outperformed the other methods in terms of precision. KinasePhos demonstrated the lowest precision value for all the kinase families except CAM KII. NetPhosK demonstrated better accuracy than KinasePhos, except for PKA and SRC, with an almost equal precision value to that of KinasePhos for PKA and a better precision value for PKB, PKC, MAPK, GSK-3, CDK1, CK2, CAM KII, and SRC. PPSP yielded the best recall values among all the methods, with the exception of CK2 and CAM KII. The proposed method comes next with respect to recall, which is almost equal to the prediction performance of PPSP; NetphosK demonstrated the worst performance.

We also applied our proposed method to predict the phosphorylation sites corresponding to serine, threonine, and tyrosine phospho residues in a nonkinase-specific manner. For this we prepared a new dataset comprising all the annotated S/T/Y sites as positive data in the database Phospho. ELM version 9.0. Negative data was prepared taking all the nonannotated S/T/Y sites from the same database. We took a 21-mer sequence centering at S/T/Y residues for both positive and negative data. In the dataset, positive to negative sequence ratios were kept at 1:1.5 to avoid any unnecessary biased predictions. To evaluate the performance of the proposed method for nonkinase-specific phosphorylation site prediction, we compared our method with three other well-known phosphorylation site prediction methods: DISPHOS,[7] PPRED,[9] and NetPhos.[10] DISPHOS uses a position-specific amino acid composition and structural disorder information to predict phosphorylation sites. NetPhos uses an artificial neural network for prediction, whereas PPRED uses position-specific scoring matrices and support vector machines for distinguishing between phosphorylation and nonphosphorylation sites. The prediction accuracy comparison is shown in Figure 8. This figure shows that our proposed method outperforms the other methods in terms of accuracy to predict S/T/Y phosphorylation sites.

Taking into consideration measures such as accuracy, precision and F-measure, our proposed method yielded the best performance. The reasonably high and balanced values of precision and recall indicate that our method can predict true positives as well as true negatives to a high extent, and yielded the best F-measure among all the methods. Our method outperformed the well-known existing methods

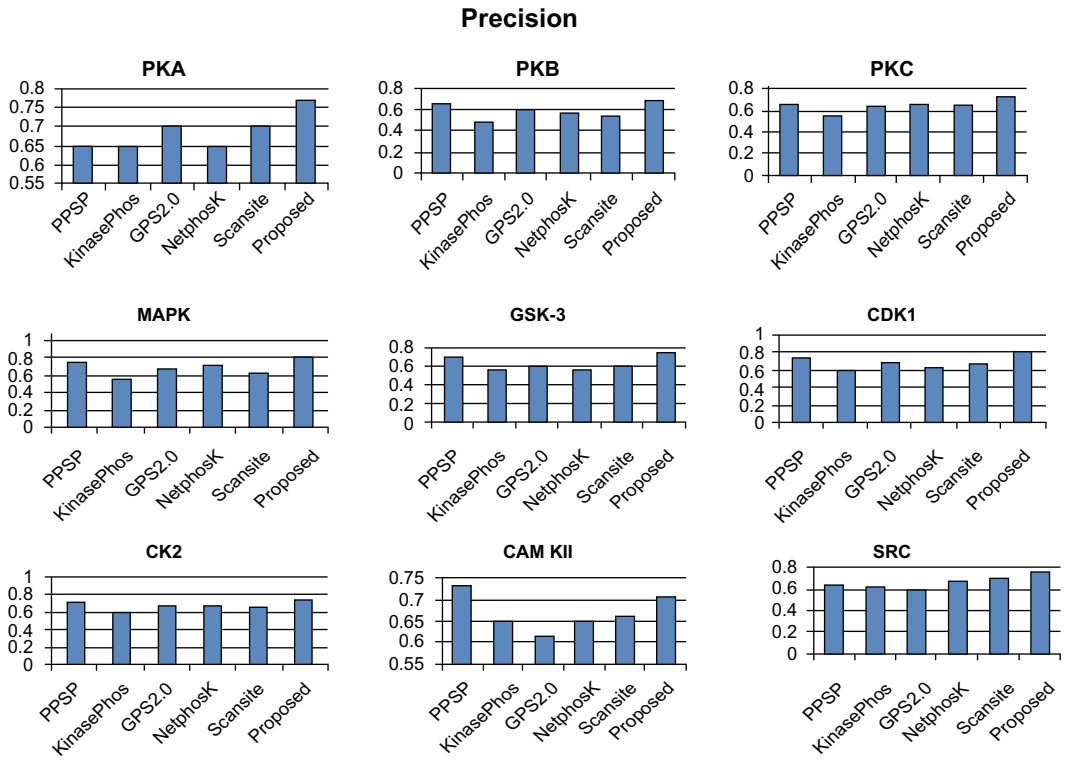

Figure 4 Comparison of various kinase-specific phosphorylation site prediction methods (PPSP, KinasePhos, GPS2.0, Scansite, and NetphosK) with our proposed methods in terms of precision.

Abbreviations: CAM KII, calmodulin-dependent protein kinase II; CDK1, cyclin-dependent kinase 1; CK2, casein kinase 2; GSK-3, glycogen synthase kinase 3; MAPK, mitogen-activated protein kinase; PKA, protein kinase A; PKB, protein kinase B; PKC, protein kinase C; SRC, tyrosin kinase SRC.

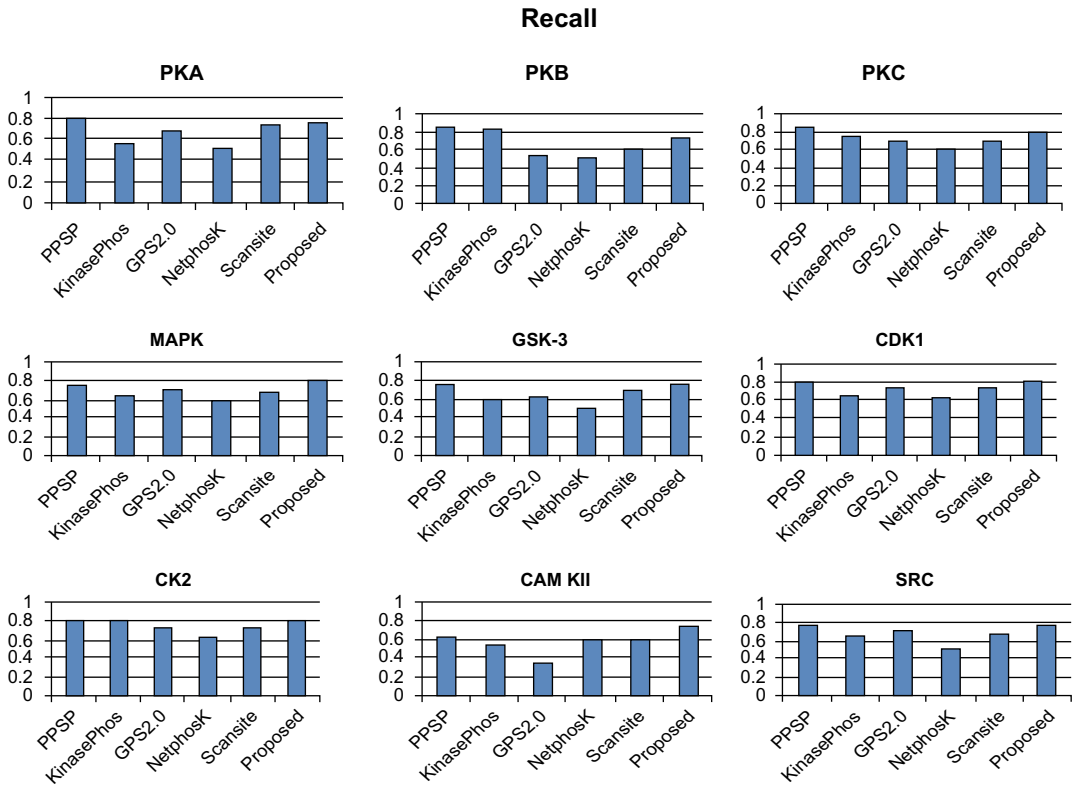

Figure 5 Comparison of various kinase-specific phosphorylation site prediction methods (PPSP, KinasePhos, GPS2.0, Scansite, and NetphosK) with our proposed methods in terms of recall.

Abbreviations: CAM KII, calmodulin-dependent protein kinase II; CDK1, cyclin-dependent kinase 1; CK2, casein kinase 2; GSK-3, glycogen synthase kinase 3; MAPK, mitogen-activated protein kinase; PKA, protein kinase A; PKB, protein kinase B; PKC, protein kinase C; SRC, tyrosin kinase SRC.

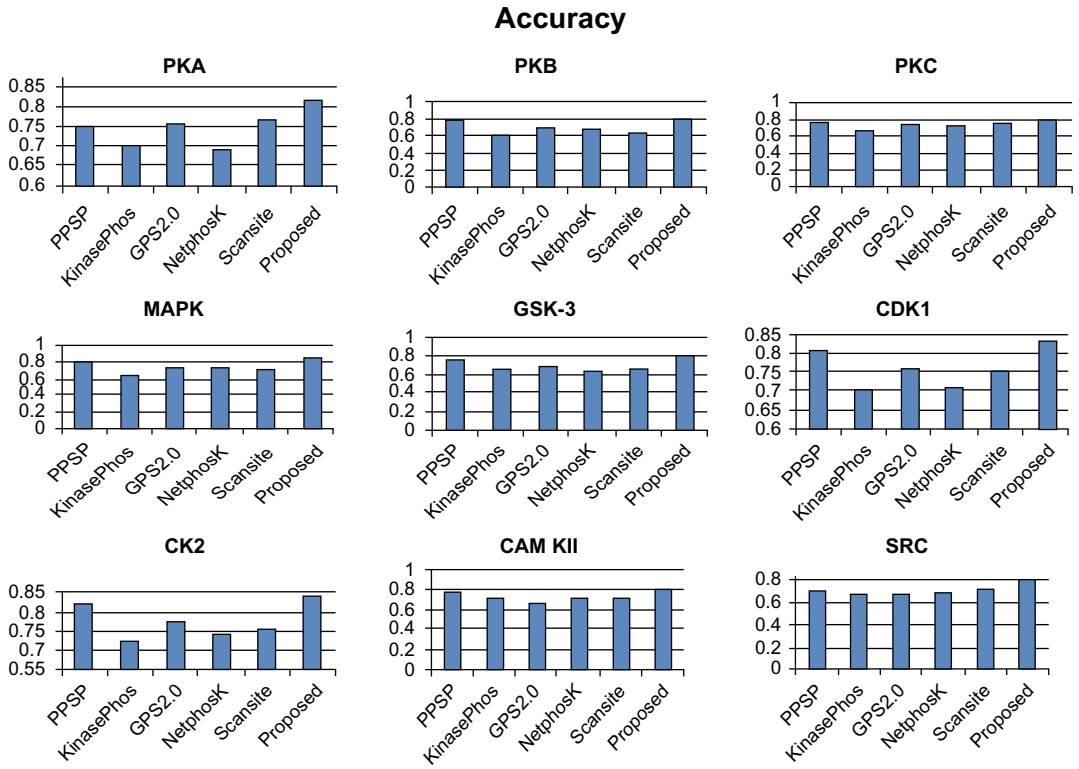

Figure 6 Comparison of various kinase-specific phosphorylation site prediction methods (PPSP, KinasePhos, GPS2.0, Scansite, and NetphosK) with our proposed methods in terms of accuracy.

Abbreviations: CAM KII, calmodulin-dependent protein kinase II; CDK1, cyclin-dependent kinase 1; CK2, casein kinase 2; GSK-3, glycogen synthase kinase 3; MAPK, mitogen-activated protein kinase; PKA, protein kinase A; PKB, protein kinase B; PKC, protein kinase C; SRC, tyrosin kinase SRC.

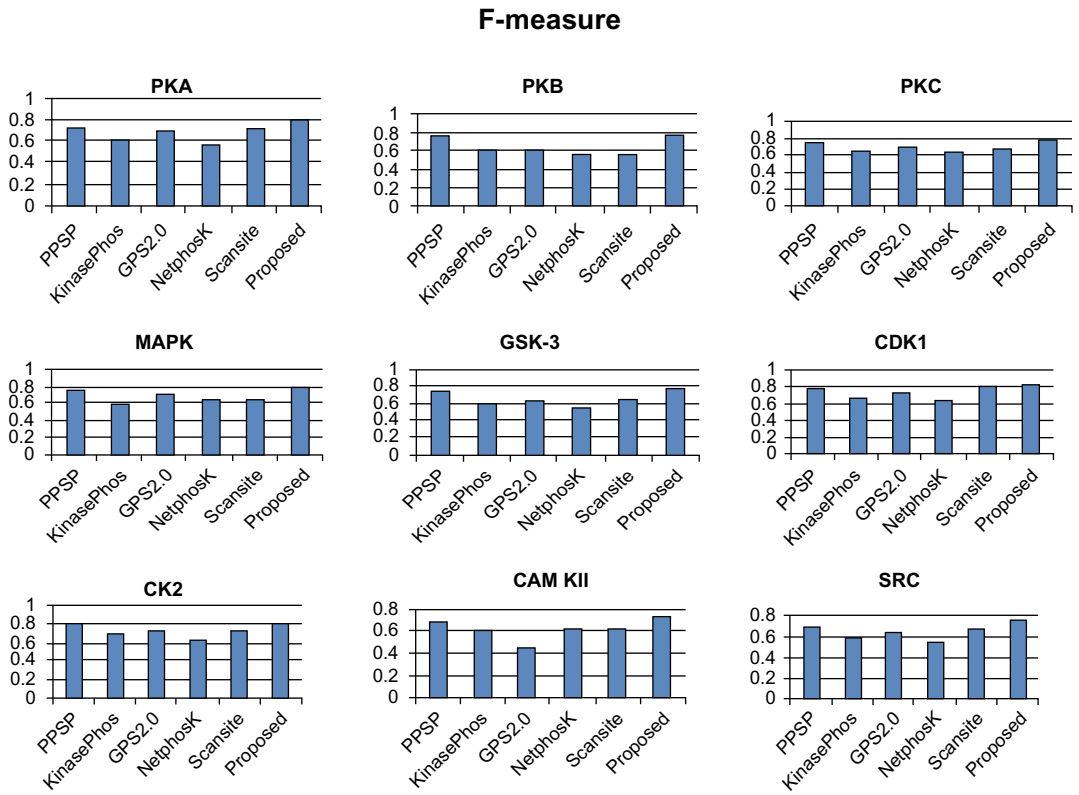

Figure 7 Comparison of various kinase-specific phosphorylation site prediction methods (PPSP, KinasePhos, GPS2.0, Scansite, and NetphosK) with our proposed methods in terms of F-measure.

Abbreviations: CAM KII, calmodulin-dependent protein kinase II; CDK1, cyclin-dependent kinase 1; CK2, casein kinase 2; GSK-3, glycogen synthase kinase 3; MAPK, mitogen-activated protein kinase; PKA, protein kinase A; PKB, protein kinase B; PKC, protein kinase C; SRC, tyrosin kinase SRC.

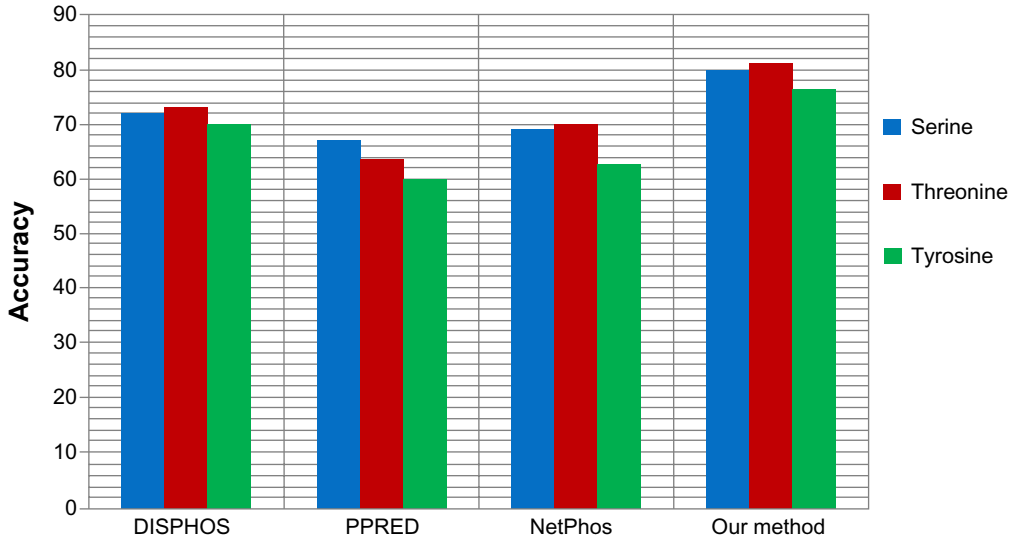

Figure 8 Prediction accuracy comparison of our proposed method with various nonkinase-specific phosphorylation site prediction methods (DISPHOS, PPRED, NetPhos) for serine, threonine, and tyrosine phosphoresidues.

and effectively distinguished the phosphorylation sites from nonphosphorylation sites in a kinase-specific manner compared to existing kinase-specific phosphorylation site prediction methods.

Conclusion

In this study, we used an ensemble method to predict kinase-specific phosphorylation sites in vertebrates only. Three types of strategies, physicochemical features, dipeptide composition, and protein–protein scoring, were taken to represent the protein sequences in their numerical forms, and three popular classifiers, LSSVM, K-NN, and MLP, were used in the proposed ensemble method. The three datasets composed of three different parameter systems were given as input to each classifier, and the final result was obtained by fusing the outputs of the above classifiers through weighted voting. Because of the relatively small data size, we incorporated data resampling so that vigorous experimentation could be performed. The ensemble method yields a better result than individual classifiers. While LSSVM, K-NN, and MLP were used in this work, other classifiers can also be used to form different ensemble methods. In summary, the results of the predictions through the proposed ensemble method indicate that our method is very promising in detecting protein phosphorylation sites and may serve as an important complement to existing methods.

Acknowledgments

The authors wish to thank the anonymous reviewers for their valuable remarks. The authors also acknowledge the computational facility of DIC and Department of Biophysics, Molecular Biology and Bioinformatics, University of Calcutta.

Disclosure

The authors report no conflicts of interest in this work.

References

1. Salway JG. *Metabolism at a Glance*. 2nd edition. Oxford, UK: Blackwell Science Ltd; 1999.
2. Li L, Shakhnovich EI, Mirny LA. Amino acids determining enzyme-substrate specificity in prokaryotic and eukaryotic protein kinases. *Proc Natl Acad Sci U S A*. 2003;100(8):4463–4468.
3. Matthews HR. Protein kinases and phosphatases that act on histidine, lysine, or arginine residues in eukaryotic proteins: a possible regulator of the mitogen-activated protein kinase cascade. *Pharmacol Ther*. 1995;67(3):323–350.
4. Kobayashi K, Nakano H, Hayashi M, et al. Association of phosphorylation site of tau protein with neuronal apoptosis in Alzheimer's disease. *J Neurol Sci*. 2003;208(1–2):17–24.
5. Pinna LA, Ruzzene M. How do protein kinases recognize their substrates? *Biochim Biophys Acta*. 1996;1314(3):191–225.
6. Manning G, Whyte DB, Martinez R, Hunter T, Sudarsanam S. The protein kinase complement of the human genome. *Science*. 2002;298(5600):1912–1934.
7. Iakoucheva LM, Radivojac P, Brown CJ, et al. The importance of intrinsic disorder for protein phosphorylation. *Nucleic Acids Res*. 2004;32(3):1037–1049.
8. Zhao X, Zhang W, Xu X, Ma Z, Yin M. Prediction of protein phosphorylation sites by using the composition of k-spaced amino acid pairs. *PLoS One*. 2012;7(10):e46302.
9. Biswas AK, Noman N, Sikder AR. Machine learning approach to predict protein phosphorylation sites by incorporating evolutionary information. *BMC Bioinformatics*. 2010;11:273.
10. Blom N, Gammeltoft S, Brunak S. Sequence and structure-based prediction of eukaryotic protein phosphorylation sites. *J Mol Biol*. 1999;294(5):1351–1362.
11. MacDonald JA, Mackey AJ, Pearson WR, Haystead TA. A strategy for the rapid identification of phosphorylation sites in the phosphoproteome. *Mol Cell Proteomics*. 2002;1(4):314–322.

12. Mackey AJ, Haystead TA, Pearson WR. CRP: Cleavage of Radiolabeled Phosphoproteins. *Nucleic Acids Res*. 2003;31(13):3859–3861.
13. Gnad F, Ren S, Cox J, et al. PHOSIDA (phosphorylation site database): management, structural and evolutionary investigation, and prediction of phosphosites. *Genome Biol*. 2007;8(11):R250.
14. Plewczynski D, Tkacz A, Wyrwicz LS, Rychlewski L. AutoMotif server: prediction of single residue post-translational modifications in proteins. *Bioinformatics*. 2005;21(10):2525–2527.
15. Obenauer JC, Cantley LC, Yaffe MB. Scansite 2.0: Proteome-wide prediction of cell signaling interactions using short sequence motifs. *Nucleic Acids Res*. 2003;31(13):3635–3641.
16. Huang HD, Lee TY, Tzeng SW, Horng JT. KinasePhos: a web tool for identifying protein kinase-specific phosphorylation sites. *Nucleic Acids Res*. 2005;33(Web Server issue):W226–W229.
17. Blom N, Sicheritz-Pontén T, Gupta R, Gammeltoft S, Brunak S. Prediction of post-translational glycosylation and phosphorylation of proteins from the amino acid sequence. *Proteomics*. 2004;4(6):1633–1649.
18. Xue Y, Li A, Wang L, Feng H, Yao X. PPSP: prediction of PK-specific phosphorylation site with Bayesian decision theory. *BMC Bioinformatics*. 2006;7:163.
19. Xue Y, Ren J, Gao X, Jin C, Wen L, Yao X. GPS 2.0, a tool to predict kinase-specific phosphorylation sites in hierarchy. *Mol Cell Proteomics*. 2008;7(9):1598–1608.
20. Xue Y, Liu Z, Cao J, et al. GPS 2.1: enhanced prediction of kinase-specific phosphorylation sites with an algorithm of motif length selection. *Protein Eng Des Sel*. 2011;24(3):255–260.
21. Jung I, Matsuyama A, Yoshida M, Kim D. PostMod: sequence based prediction of kinase-specific phosphorylation sites with indirect relationship. *BMC Bioinformatics*. 2010;11 Suppl 1:S10.
22. Yu Z, Deng Z, Wong HS, Tan L. Identifying protein-kinase-specific phosphorylation sites based on the Bagging-AdaBoost ensemble approach. *IEEE Trans Nanobioscience*. 2010;9(2):132–143.
23. Plewczynski D, Basu S, Saha I. AMS 4.0: consensus prediction of post-translational modifications in protein sequences. *Amino Acids*. 2012;43(2):573–582.
24. Wan J, Kang S, Tang C, et al. Meta-prediction of phosphorylation sites with weighted voting and restricted grid search parameter selection. *Nucleic Acids Res*. 2008;36(4):e22.
25. Trost B, Kusalik A. Computational prediction of eukaryotic phosphorylation sites. *Bioinformatics*. 2011;27(21):2927–2935.
26. Xu Q, Pan SJ, Xue HH, Yang Q. Multitask learning for protein subcellular location prediction. *IEEE/ACM Trans Comput Biol Bioinform*. 2011;8(3):748–759.
27. Li L, Zhang Y, Zou L, et al. An ensemble classifier for eukaryotic protein subcellular location prediction using gene ontology categories and amino acid hydrophobicity. *PLoS One*. 2012;7(1):e31057.
28. Zhang K, Gray JW, Parvin B. Sparse multitask regression for identifying common mechanism of response to therapeutic targets. *Bioinformatics*. 2010;26(12):i97–i105.
29. Deng L, Guan J, Dong Q, Zhou S. Prediction of protein-protein interaction sites using an ensemble method. *BMC Bioinformatics*. 2009;10:426.
30. Liu Q, Xu Q, Zheng VW, Xue H, Cao Z, Yang Q. Multi-task learning for cross-platform siRNA efficacy prediction: an in-silico study. *BMC Bioinformatics*. 2010;11:181.
31. Trost B, Kusalik A. Computational phosphorylation site prediction in plants using random forests and organism-specific instance weights. *Bioinformatics*. 2013;29(6):686–694.
32. Gao S, Xu S, Fang Y, Fang J. Using multitask classification methods to investigate the kinase-specific phosphorylation sites. *Proteome Sci*. 2012;10 Suppl 1:S7.
33. Dinkel H, Chica C, Via A, et al. Phospho.ELM: a database of phosphorylation sites – update 2011. *Nucleic Acids Res*. 2011;39(Database issue):D261–D267.
34. Li W, Godzik A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*. 2006;22(13):1658–1659.
35. Kawashima S, Pokarowski P, Pokarowska M, Kolinski A, Katayama T, Kanehisa M. AAindex: amino acid index database, progress report 2008. *Nucleic Acids Res*. 2008;36(Database issue):D202–D205.
36. Liu B, Wang X, Chen Q, Dong Q, Lan X. Using amino acid physicochemical distance transformation for fast protein remote homology detection. *PLoS One*. 2012;7(9):e46633.
37. Li ZR, Lin HH, Han LY, Jiang L, Chen X, Chen YZ. PROFEAT: a web server for computing structural and physicochemical features of proteins and peptides from amino acid sequence. *Nucleic Acids Res*. 2006;34(Web Server issue):W32–W37.
38. Mitra P, Murthy CA, Pal SK. Unsupervised feature selection using feature similarity. *IEEE Trans Pattern Anal Mach Intell*. 2002;24(3):301–312.
39. Suykens JAK, Vandewalle J. Least squares support vector machine classifiers. *Neural Process Lett*. 1999;9(3):293–300.
40. Chowdhury M, Das S, Kundu MK. compact image signature generation: an application in image retrieval. *Proc 5th International Conference on Computer Science and Information Technology CSIT 2013*. Amman, Jordan: IEEE Press; 2013:1–7.
41. Zhou FF, Xue Y, Chen GL, Yao X. GPS: a novel group-based phosphorylation predicting and scoring method. *Biochem Biophys Res Commun*. 2004;325(4):1443–1448.
42. Xue Y, Zhou F, Zhu M, Ahmed K, Chen G, Yao X. GPS: a comprehensive www server for phosphorylation sites prediction. *Nucleic Acids Res*. 2005;33(Web Server issue):W184–W187.