

# Identification of Phosphorylation Sites Using A Hybrid Classifier Ensemble Approach

Zhiwen Yu, Zhongkai Deng, Hau-San Wong  
Department of Computer Science, City University of Hong Kong  
kevinyuzhiwen@hotmail.com, cshswong@cityu.edu.hk

## Abstract

*Protein phosphorylation is an important step in many biological processes, such as cell cycles, membrane transport, apoptosis, and so on. We design a new classifier ensemble approach called Bagging-Adaboost Ensemble (BAE) for the prediction of eukaryotic protein phosphorylation sites, which incorporates the bagging technique and the adaboost technique into the classifier framework to improve the accuracy, stability and robustness of the final result. To our knowledge, this is the first time in which the ensemble approach is applied to predict phosphorylation sites. Our prediction system based on BAE focuses on five kinase families: CDK, CK2, MAPK, PKA, and PKC. BAE achieves good performance in six families, and the accuracies of the prediction system for these families are 84.7%, 87.4%, 85.5%, 85.2%, and 82.3% respectively.*

## 1. Introduction

Phosphorylation can be defined as the introduction of a phosphate group into a protein molecule, and is a widespread reversible post-translational modification of proteins in both prokaryotes and eukaryotes. Phosphorylation is catalyzed by a type of enzyme called kinase, which is also known as phosphotransferase. The kinases identified largely in eukaryotic organisms (ePKs) constitute one of the largest known protein superfamilies. Phosphorylation sites and the relevant kinases can be identified in vivo or in vitro.

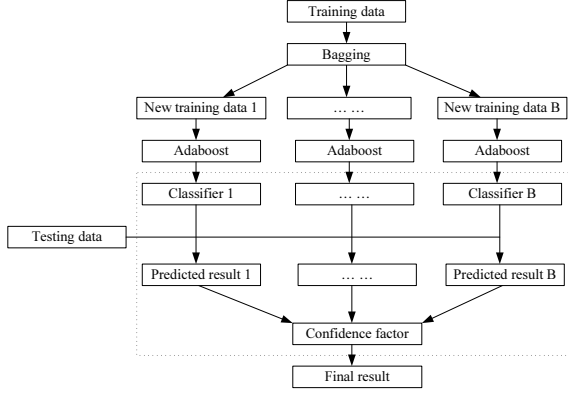
In recent years, a number of approaches for predicting protein phosphorylation sites have been developed. Blom et al. [1] designed a prediction system called Netphos, which adopted artificial neural network (the standard feed-forward type) to predict phosphorylation sites in independent sequences. It focuses on distinguishing phosphorylation sites from non-phosphorylation sites. They [2] also extended Netphos to NetPhosk, which is

a kinase-specific prediction system. Yaffe et al.[3] developed Scansite which contains the profiles of 63 kinases. Kim et al. [4] designed a prediction system Pred-Phospho for phosphorylation sites. The prediction system belongs to the class of kinase-specific systems and adopts SVM (support vector machine) as the prediction algorithm. Xue et al. [5] proposed a group-based phosphorylation scoring (GPS) method for predicting kinase-specific phosphorylation sites. They [6] also applied the approach of Bayesian decision theory, which they refer to as PPSP (Prediction of PK-specific Phosphorylation site), to predict the potential phosphorylation sites. Huang et al.[7] developed a new tool called KinasePhos 1.0 based on the profile Hidden Markov Model to identify catalytic kinase-specific phosphorylation sites. Wong et al.[8] extended KinasePhos 1.0 to KinasePhos 2.0, which adopted SVM, together with the protein sequence profile and protein coupling pattern, to predict phosphorylation sites.

Although there exist a number of approaches to predict phosphorylation sites, none of them consider the ensemble approach which integrates multiple classifiers to obtain more robust, stable and accurate results. We design a new classifier ensemble approach called Bagging-Adaboost Ensemble (BAE) for prediction of eukaryotic protein phosphorylation sites, which incorporates the bagging technique [9] and the adaboost technique [10] into the classifier framework to improve the final prediction result.

## 2. Method

In view of the advantages of the bagging technique and the adaboost technique, we propose a new classifier ensemble approach called Bagging-Adaboost Ensemble (BAE) for prediction of eukaryotic protein phosphorylation sites, which incorporates both the bagging technique and the adaboost technique into the classifier framework to improve the accuracy, stability and robustness of the final result.



**Figure 1. The framework of the Bagging-Adaboost ensemble approach**

Figure 1 illustrates the framework of the BAE approach. Specifically, BAE first generates a set of datasets  $\{D^1, D^2, \dots, D^B\}$  (where  $B$  is the total number of datasets) based on the original training dataset by the bagging technique. Then, the adaboost algorithm is applied to these new training datasets to obtain a set of classifiers  $\{C^1, C^2, \dots, C^B\}$ . Next, the set of classifiers are evaluated through a test set, and the corresponding set of prediction results of different classifiers is denoted as  $\{L^1, L^2, \dots, L^B\}$ . Finally, BAE combines prediction results based on a set of confidence factors to obtain the final result.

## 2.1 Bagging

BAE first generates a set of new training datasets  $\{D^1, D^2, \dots, D^B\}$  from the original training dataset  $D$ . Assume that the original training set  $D$  contains  $n$  sample pairs  $D = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$  (where  $\mathbf{x}_i$  is the sample,  $y_i$  is the label of the sample  $\mathbf{x}_i$ ), the  $i'$ -th sample pair ( $1 \leq i' \leq n'$ ,  $n'$  is the size of the new training dataset, and  $n' \leq n$ ) in the new training dataset  $D^b$  ( $1 \leq b \leq B$ ) is selected as follows:

$$(\mathbf{x}_{i'}, y_{i'}) = (\mathbf{x}_r, y_r), (\mathbf{x}_{i'}, y_{i'}) \in D^b, (\mathbf{x}_r, y_r) \in D \quad (1)$$

where  $\gamma$  is a uniform random index over the set  $\{1, \dots, n\}$ , and  $(\mathbf{x}_{i'}, y_{i'})$  and  $(\mathbf{x}_r, y_r)$  are sample pairs which belong to the new training dataset  $D^b$  and the original training dataset  $D$  respectively. BAE repeats the above process  $n'$  times and selects  $n'$  sample pairs for the new training dataset  $D^b$  (where  $n' = \rho n$  and  $\rho$  is a sub-sampling rate which is pre-specified by the user).

## 2.2 Adaboost

In the second step, BAE trains a set of classifiers  $\{C^1, C^2, \dots, C^B\}$  based on the adaboost algorithm. Specifically, each of the classifier is trained using one of the training datasets generated through the previous bagging step. Given the new training set  $D^b$  with  $n'$  sample pairs ( $D^b = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_{n'}, y_{n'})\}$ ) (where  $y_{i'} \in \{0, 1\}$  for two-class classification problem), the goal of adaboost is to find a classification rule  $R$  from the training data, so that when given a new input testing sample  $\mathbf{x}$ , we can assign to it a class label  $R(\mathbf{x})$ .

Algorithm **AdaBoost**(Training dataset  $D^b$ )

1. Initialize the weight vector  $\omega_{i'}^1 = \frac{1}{n'}$  for  $i' \in \{1, n'\}$ ;
2. **For**  $t = 1$  to  $T$   
(where  $T$  is the maximum number of weak classifiers)
3. build a weak classifier  $\xi^t$  based on the training data weighted using  $\omega_{i'}^t$ ;
4. Compute  $ER^t = \sum_{i'=1}^{n'} \omega_{i'}^t \cdot \delta(\xi^t(\mathbf{x}_{i'}), y_{i'})$ ;
5. **If** ( $ER^t > \frac{1}{2}$ )
6. set  $m = T - 1$  and abort loop;
7. **Else**
8.  $\alpha^t = \log \frac{1-ER^t}{ER^t}$ ;
9.  $\omega_{i'}^{t+1} = \omega_{i'}^t \cdot \exp(\alpha^t \cdot \delta(\xi^t(\mathbf{x}_{i'}), y_{i'}))$ ;
10. Re-normalize  $\omega_{i'}^{t+1}$ ;

**Figure 2. The AdaBoost algorithm**

The AdaBoost algorithm is an iterative procedure that combines a number of weak classifiers together to form a single strong classifier with better accuracy. Figure 2 provides an overview of the Adaboost algorithm. The function  $\delta(\xi^t(\mathbf{x}_{i'}), y_{i'})$  is defined as follows:

$$\delta(\xi^t(\mathbf{x}_{i'}), y_{i'}) = \begin{cases} 1 & \text{if } \xi^t(\mathbf{x}_{i'}) \neq y_{i'} \\ 0 & \text{if } \xi^t(\mathbf{x}_{i'}) = y_{i'} \end{cases} \quad (2)$$

The algorithm first assigns the same weights to all the training data. Then, it trains the first weak classifier  $\xi^1$  based on the training data using the set of initial weights. If the training sample  $\mathbf{x}_{i'}$  is misclassified, the weight of that training sample  $\mathbf{x}_{i'}$  will be increased. Next, the second weak classifier  $\xi^2$  is constructed based on the re-weighted set of training data, and the process is repeated. The final classification result  $\xi_{final}$  is expressed as follows:

$$\xi_{final}(\mathbf{x}) = \arg \max_{y \in \{0,1\}} \sum_{t=1}^T \alpha^t \cdot \delta(\xi^t(\mathbf{x}), y) \quad (3)$$

where  $\mathbf{x}$  is the sample, and  $\alpha^t$  is the importance weighting associated with the  $t$ -th weak classifier ( $t \in \{1, \dots, T\}$ ).

**Table 1. Comparison of the performance of different approaches in the CDK family**

Approach	mean(AC)	std(AC)	mean(SE)	std(SE)	mean(SP)	std(SP)	mean(CC)	std(CC)
BAE	0.847	0.0344	0.7879	0.0728	0.8863	0.0321	0.6805	0.074
SVM	0.7671	0.0435	0.7011	0.0768	0.811	0.0492	0.5151	0.091
ADA	0.8352	0.0382	0.7937	0.0601	0.8730	0.0572	0.6804	0.0766
k-nn	0.7004	0.0468	0.6655	0.0413	0.7241	0.0779	0.3885	0.088

**Table 2. Comparison of the performance of different approaches in the CK2 family**

Approach	mean(AC)	std(AC)	mean(SE)	std(SE)	mean(SP)	std(SP)	mean(CC)	std(CC)
BAE	0.8736	0.0299	0.8217	0.0783	0.9093	0.0356	0.7371	0.0616
SVM	0.7948	0.0491	0.7144	0.1086	0.8491	0.0381	0.5709	0.1061
ADA	0.8469	0.0373	0.8040	0.0886	0.8876	0.0356	0.7234	0.0788
k-nn	0.6834	0.0485	0.8006	0.0628	0.6051	0.0729	0.4018	0.0898

The weak classifier we use is the Fisher Linear Discriminant classifier (FLD) which finds the linear combination of features which best separate two or more classes of samples. In the final stage, these  $T$  weak classifiers are combined to form a strong classifier to achieve a lower classification error.

### 2.3 Confidence factor

After obtaining the set of classifiers  $\{C^1, C^2, \dots, C^B\}$ , BAE uses these classifiers to predict the labels of the samples in the test dataset  $D^{test} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_l\}$  (where  $l$  is the number of samples in the dataset). The inputs of the classifiers are the samples in the test dataset, while the outputs of the classifiers are a set of predicted labels corresponding to the samples. The predicted results of the  $b$ -th classifier  $C^b$  is denoted as  $L^b = \{y_1^b, y_2^b, \dots, y_l^b\}$ . The confidence factor  $\beta_j(\mathbf{x}_i)$  (where  $j \in \{0, 1\}$  for a binary classification problem) of the  $i$ -th sample  $\mathbf{x}_i$  is calculated as follows:

$$\beta_j(\mathbf{x}_i) = \sum_{b=1}^B 1\{y_i^b = j\} \quad (4)$$

$$y_i = \operatorname{argmax}_j \beta_j(\mathbf{x}_i) \quad (5)$$

where  $y_i$  is the final predicted label of the  $i$ -th sample  $\mathbf{x}_i$ , and  $1\{\bullet\}$  is an indicator function. The sample  $\mathbf{x}_i$  will be assigned to the class with the largest confidence factor.

### 3. Dataset and experiment setting

Raw protein sequence and phosphorylation sites are obtained from the Phospho.ELM dataset [11], which

contains 16471 experimentally verified serine(S), threonine(T) and tyrosine(Y) phosphorylation sites in 5507 eukaryotic proteins. Only 3417 sites (21%) among these 16471 sites are annotated with the list of kinases that modify them, which are candidates of positive samples in our experiments. The distribution of these 3471 kinase-annotated sites among the kinase families is very uneven. Only those kinase groups or families having more than 200 sites are chosen here, i.e., CDK (311 sites), CK2 (291 sites), MAPK (257 sites), PKA (303 sites), and PKC (354 sites)). We adopt a sequence window size of 11 amino acid residues with the phosphorylated residue at the central position (position 6) to extract feature vectors.

Our approach belongs to the class of kinase-specific approaches. The sites in one kinase family will serve as the positive samples to train a classifier, while the sites in other kinase families and the sites which belong to non-phosphorylated S, T and Y residues will be the negative samples. A 10-fold cross validation test was performed by randomly partitioning the positive and negative samples into a training set and a test set with a ratio of 9 : 1. We repeat the cross validation 10 times for each kinase family.

The performance of the approaches is measured by the accuracy (AC), the sensitivity (SE), the specificity (SP) and the correlation coefficient (CC) which are defined as follows:

$$AC = \frac{tp + tn}{tp + fp + tn + fn} \quad (6)$$

$$SE = \frac{tp}{tp + fn} \quad (7)$$

$$SP = \frac{tn}{tn + fp} \quad (8)$$

**Table 3. Comparison of the performance of different approaches in the MAPK family**

Approach	mean(AC)	std(AC)	mean(SE)	std(SE)	mean(SP)	std(SP)	mean(CC)	std(CC)
BAE	0.8549	0.0407	0.8244	0.0685	0.8752	0.0542	0.7025	0.0823
SVM	0.8057	0.039	0.7315	0.0669	0.8545	0.0414	0.593	0.0837
ADA	0.8202	0.0213	0.7893	0.0686	0.8588	0.0538	0.6660	0.0428
k-nn	0.7498	0.0382	0.795	0.0747	0.7198	0.0575	0.507	0.0783

**Table 4. Comparison of the performance of different approaches in the PKA family**

Approach	mean(AC)	std(AC)	mean(SE)	std(SE)	mean(SP)	std(SP)	mean(CC)	std(CC)
BAE	0.8523	0.0495	0.8021	0.0599	0.8857	0.0544	0.6926	0.1036
SVM	0.8318	0.043	0.7785	0.0976	0.8674	0.0521	0.6514	0.0906
ADA	0.8201	0.0344	0.7888	0.0588	0.8642	0.0473	0.6679	0.0711
k-nn	0.8207	0.0375	0.76	0.0699	0.8613	0.0375	0.626	0.08

$$CC = \frac{(tp \cdot tn) - (fn \cdot fp)}{\sqrt{(tp + fn) \cdot (tn + fp) \cdot (tp + fp) \cdot (tn + fn)}}$$

where tp denotes the true positive, tn denotes the true negative, fp denotes the false positive, and fn denotes the false negative.

## 4. Experimental results

BAE will be compared with two popular prediction systems for phosphorylation sites which are implemented based on the support vector machine (SVM) with RBF (radial basis function) kernel function [4], single adaboost classifier (ADA) and K-nearest neighbor classifier (k-nn) respectively. In order to provide a fair comparison, all the approaches (BAE, ADA, SVM, k-nn) are applied to the same training sets and test sets in the same kinase families. Table 1, Table 2, Table 3 and Table 4 (where mean, std, max and min denote the mean, the standard deviation, the maximum, and the minimum of the AC, SE, SP, CC values respectively. Due to the restriction of the space, the table for PKC is not listed here.) illustrate the performance of BAE, ADA, SVM, and k-nn in the CDK, CK2, MAPK, PKA, and PKC families. BAE outperforms the other algorithms in the five kinase families due to its capability to combine multiple classifiers to provide more robust, stable and accurate results.

## 5. Conclusion

In this paper, we investigate the problem of predicting phosphorylation sites. Our major contribution is a new classifier ensemble approach known as Bagging-Adaboost Ensemble (BAE) for phosphorylation site prediction, which incorporates both the bagging technique and the adaboost technique into the classifier ensemble framework. The results of our experiments indicate that our new approach achieves good performance in the CDK, CK2, MAPK, PKC and PKA families, with corresponding accuracies of 84.7%, 87.4%, 85.5%, 85.2% and 82.3% respectively.

## (9) Acknowledgment

The work described in this paper was partially supported by a grant from the City University of Hong Kong [Project No. 7002314].

## References

- [1] Blom, N., Gammeltoft, S., Brunak, S. (1999) Sequence and structure-based prediction of eukaryotic protein phosphorylation sites. *J. Mol. Biol.*, **294**, 1351-1362.
- [2] Blom, N., Sicheritz, T., Gupta, R., Gammeltoft, S., Brunak, S. (2004) Prediction of post-translational glycosylation and phosphorylation of proteins from the amino acid sequence. *Proteomics*, **4**, 1633-1649.
- [3] Yaffe, M.B., Leparo, G.G., Lai, J., Obata, T., Volinia, S. and Cantley, L.C. (2001) A motif-based profile scanning approach for genome-wide prediction of signaling pathways. *Nat. Biotechnol.*, **19**, 348-353.
- [4] Kim, J.H., Lee, J., Oh, B., Kimm, K., Koh, I. (2004) Prediction of phosphorylation sites using SVMs. *Bioinformatics*, **20:17**, 3179-3184.
- [5] Xue, Y., Zhou, F., Zhu, M., Ahmed, K., Chen, G., Yao, X. (2005) GPS: a comprehensive www server for phosphorylation sites prediction. *Nucleic Acids Res.*, **33**, W184-W187.
- [6] Xue, Y., Li, A., Wang, L., Feng, H., Yao, X. (2006) PPSP: prediction of PK-specific phosphorylation site with Bayesian decision theory. *BMC Bioinformatics*, **7:163**.
- [7] Huang, H.D., Lee, T.Y., Tzeng, S.W., Horng, J.T. (2005) KinasePhos: a web tool for identifying protein kinasespecific phosphorylation sites. *Nucleic Acids Res.*, **33**, W226-W229.
- [8] Wong, Y.H., Lee, T.Y., Liang, H.K., Huang, C.M., Wang, T.Y., Yang, Y.H., Chu, C.H., Huang, H.D., Ko, M.T., Hwang, J.K., (2007) KinasePhos 2.0: a web server for identifying protein kinase-specific phosphorylation sites based on sequences and coupling patterns. *Nucleic Acids Res.*, **35**, W588-W594.
- [9] Breiman, L. (1996) Bagging Predictors. *Machine Learning*, **24:2**, 123-140.
- [10] Margineantu, D.D. & Dietterich, T.G. (1997) Pruning Adaptive Boosting. *Proc. 14th Int'l Conf. Machine Learning*, 211-218.
- [11] Diella, F., Cameron, S., Gemünd, C., Linding, R., Via, A., Kuster, B., Sicheritz, T., Blom, N., Gibson, T.J., (2004) Phospho.ELM: a database of experimentally verified phosphorylation sites in eukaryotic proteins. *BMC Bioinformatics*, **5(1):79**.