Statistics and Its Interface Volume 3 (2010) 523–531

# Predicting kinase functional sites using hierarchical stochastic language modelling

Huan Yu, Guojun Pei, Peng Ge, Xiangzhong Fang, Fengzhu Sun, Luhua Lai, Minping Qian and Minghua Deng*

*Motivation*: Predicting functional sites in kinases is an important problem in biology. Both the functional sites and the relationship among the amino acids within the sites need to be understood. An algorithm is developed for kinase functional site prediction using amino acid sequence data based on hierarchical stochastic language (HSL) modelling.

*Results*: Our method is validated by using two complementary approaches. Firstly, the predicted functional sites using the HSL were compared with experimentally verified functional sites including the patterns in PROSITE, the contacting sites in the Protein Data Bank (PDB), and the domains in Pfam. Compared to the patterns in PROSITE and the contacting sites in PDB, the overall average recall/precision of the HSL model was 83.5% / 23.0% and 66.1% / 79.9%, respectively. Compared to Pfam, 90% of the predicted functional sites were parts of domains with names containing the substring "kinase". Secondly, 10-fold cross-validation was used to study the kinase function prediction accuracy of the HSL. The HSL achieved both high sensitivity (94.7%) and specificity (94.0%) compared to 94.5% and 85.8%, respectively, for MEME. The HSL model automatically detected kinase sub-families. The identified sub-families were consistent with known phylogenetic trees of the kinase sequences. Therefore, the HSL was applicable to kinase sequences with heterogeneous subsets sharing the same catalysis function.

*Availability and Supplementary information*: The software and supplementary materials are available at http://www.math.pku.edu.cn/teachers/dengmh/HSL

Keywords and phrases: Kinase, Functional sites, Hierarchical stochastic language (HSL).

## 1. INTRODUCTION

Kinases are a ubiquitous group of enzymes participating in a variety of cellular pathways. They catalyze the transfer of the terminal phosphate group from adenosine triphosphate (ATP) to an acceptor, which can be a small molecule, lipid, or protein substrate. Because of their universal role in cellular processes, the classification of kinases and the prediction of their functional sites are very important in biology.

Several databases of kinase functional sites based on experimental data are available. PROSITE [1] provides biologically significant sites in regular expression forms (or patterns). The Protein Data Bank (PDB) [2] contains the 3-dimensional coordinates of all amino acid atoms and some binding atoms, which are direct evidence for the presence of functional sites. However, only a small number of kinase functional sites are available in PROSITE and PDB due to difficulties in biological experiments.

Several computational approaches for functional site prediction have been developed. Homology searches have been widely used in motif finding. BLOCKS [3] and MEME [4] generate a position specific scoring matrix by Gibbs sampling and an expectation-maximisation (EM) algorithm, respectively. PFAM [5] associates protein function with protein domains identified by hidden Markov models.

The available methods have two major drawbacks. First, the homology assumption does not always hold. Two types of kinase families, homologies and analogies [6], are present. Kinases of homology families have a common ancestor and similar functional sites, while in analogy families, kinases of similar functions do not inherit their function similarity from common ancestry, but from convergent evolution. Homology-based methods can only identify motifs of the largest sub-family in an analogy family. Second, the available motif finding approaches often neglect potential interactions among the active sites, which are important in performing catalysis functions. Better models for kinase functional sites should be able to distinguish the potential mixed kinase sub-families and to consider the interdependency of the amino acids within each motif.

In this paper, we introduce a hierarchical stochastic language (HSL) model for the identification of functional sites in kinase families. The model integrates the advantages of the $k$-tuple approach for motif finding with the syntax. The HSL first finds keywords by consensus of $k$-tuples that characterize the functional sites, and then finds a stochastic grammar to constitute different types of sentences for each kinase functional family. The model automatically detects kinase sub-families.

*Corresponding author.

We built models for 81 kinase functional families each containing at least 20 sequences from Swiss-Prot. Comparing with patterns in PROSITE, the HSL achieved 83.5% recall and 23.0% precision. Comparing with PDB patterns, the HSL achieved 66.1% recall and 79.9% precision. For sequence classification, 63 families were found to have a unique model, 14 families had two sub-families, and 4 families had three sub-families. These sub-families are consistent with their different branches of phylogenetic trees and significantly increased classification sensitivity. Overall, the HSL model achieved both high sensitivity (94.7%) and specificity (94.0%) in 10-fold cross-validation classification, which outperformed MEME, that with sensitivity 94.5% and specificity 85.8%.

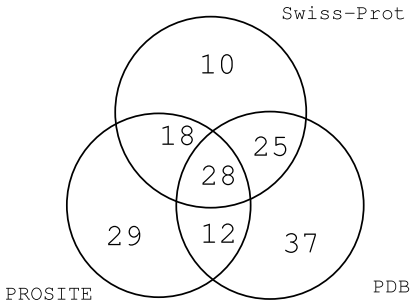


*Figure 1. The numbers of kinase families with overlaps in Swiss-Prot, PDB and PROSITE.*

## 2. MATERIALS AND METHODS

### 2.1 Materials

In this paper, the functional sites of kinase families corresponding to the Swiss-Prot [12] sequences were studied. Kinase sequences were downloaded from Swiss-Prot (UniProtKB/Swiss-Prot Release 50.0 of 30 May 2006) and classified by the ENZYME database [13]. In the ENZYME database, kinases were classified according to their Enzyme Commission (E.C.) numbers [14] based on the chemical reactions they catalyze. In order to study the functional sites in a kinase family using any statistical approach, it is necessary that the number of sequences is relatively large. In this study, only the kinase families containing at least 20 sequences excluding fragments were selected. 81 kinase families containing 11,115 sequences were selected and used for the following analysis.

We compared our results with the structures from PDB (19 January 2005) and the patterns from the PROSITE database (Release 19.34 of 5 September 2006). PDB provides resources for the structures of biological macromolecules and their relationships to sequences and functions. PROSITE consists of biologically significant patterns that help to reliably identify known protein families for new sequences. Figure 1 shows the number of kinase families in each data source and their overlaps. The result were also validated based on the domains from the SwissPfam database (Version 21.0 of November 2006) [5]. SwissPfam contains the domain structure of SWISSPROT and TrEMBL proteins according to Pfam.

### 2.2 HSL modelling

An HSL model for identifying the functional sites in a kinase family was developed. In the general language model, the patterns in a family were viewed as being composed of simple subpatterns [15]. The patterns were viewed as sentences belonging to a language, the subpatterns were viewed as the alphabets of the language, and the sentences were generated from the subpatterns according to a grammar. Thus, a large collection of complex patterns could be described by a small number of subpatterns and grammatical rules. The stochastic language model assigned probabilities to the grammatical rules [16]. The HSL model in this study comprises the keywords' stochastic language models built on amino acids and the sentences' stochastic language model for describing the interrelationships among the keywords. The details of constructing the HSL are as follows.

Taking the ancestor of the family as the starting point of the language, a stochastic grammar was defined for each functional family. A stochastic language model is a quintuplet $\{E, S, R, P, T\}$, where $E$ is a set of symbols (keywords: $W_1, W_2, W_3$, etc.); $S$ is the starting state (the family's starting state consisting of some keywords in $E$); $R$ is the set of grammatical rules (the evolutionary rules, which indicate variants, such as mutations, insertions and deletions, allowed in sequences); $P$ is the probability of different rules in $R$; and $T$ is the maximum number of variants allowed from $S$. For a sequence $s$ consisting of symbols in $E$, $\{E, S, R, P\}$ gives a probability that $s$ can be generated by the stochastic grammar for this family and $T$ is a threshold for family classification. Figure 2 shows the stochastic language model starting with $S$="$W_1W_2W_3W_4W_5W_6$" while "$W_1W_2W_4W_5W_6$", "$W_1W_2W_7W_3W_4W_5W_6$", "$W_1W_2W_3W_8W_5W_6$", etc. can be generated from $S$.

A stochastic language model was also built for each keyword by taking 20 amino acids as symbols and the consensus motif as the starting state. The same quintuplet structure described above for the sentences can be used. The hierarchical stochastic model consists of the keyword's stochastic model and the sentence's stochastic model.

### 2.3 Modelling the keywords

For the HSL model to work well, it is essential to define the keywords correctly. For a given kinase family, the sequences within the family are referred to as positive sequences and all sequences from other families as negative sequences. The keywords of a kinase family should be over-represented among the positive sequences. Three steps were used to identify the keywords for the family. First, the over-represented $k$-tuples within the positive sequences were

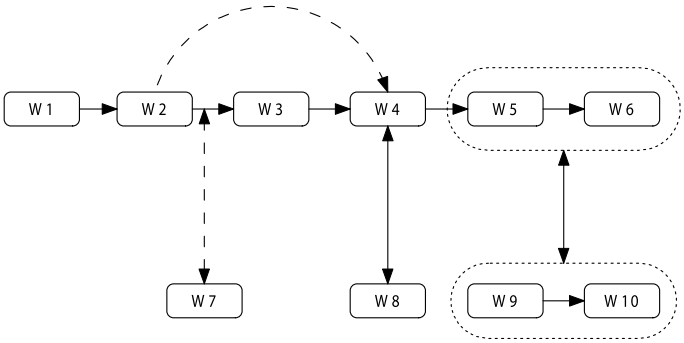


*Figure 2. The stochastic language model.* S=$\{W_1, W_2, W_3, W_4, W_5, W_6, W_7, W_8, W_9, W_{10}\}$, S="$W_1W_2W_3W_4W_5W_6$" *is the standard sentence. Double-headed solid arrows represent mutations, double-headed dashed arrows represent insertions, and single-headed dashed arrows represent deletions (keyword* $W_3$ *can be deleted).*

found. Second, the over-represented $k$-tuples were concatenated to form words. Third, the words were aligned to all the sequences. The keywords were defined as the words whose mean alignment score to the positive sequences is significantly higher than that to the negative sequences.

The 20 amino acids were used as the symbol set $E^w$. To identify the starting keywords $S^w$, the over-represented $k$-tuples were first found. However, $k$ cannot be chosen arbitrarily. If $k$ is too small, say $k = 2$, each 2-tuple is expected to appear in every sequence of 400 residues in a random sequence. When $k$ is too large, say $k \geq 5$, the frequency of the $k$-tuple in the sequences of the family may have large fluctuations. Empirical exploration of the data showed that $k = 4$ is appropriate for the particular problem. There are a total of 160,000 ($20^4$) quadruplets. For each quadruplet $q_i (i = 1, 2, \ldots, 160,000)$, the number of positive sequences containing $q_i$ were counted. The $N_t$ most frequent quadruplets were chosen as candidate quadruplets.

Given a candidate quadruplet, the frequency of its occurrence within the positive samples was compared with that within the negative samples using a chi-square score with Yates correction [17] computed by

$$\chi^2 = \frac{N(|ad - bc| - N/2)^2}{(a+b)(c+d)(a+c)(b+d)},$$

where $N = a + b + c + d$; a and b are the numbers of occurrences of the quadruplet and other quadruplets in the positive sequences, respectively; and c and d are defined as for a and b but for the negative sequences. A candidate quadruplet was referred to as significant if the p-value (with Bonferroni correction) is less than 0.05. The significant candidate quadruplets were considered as parts of the starting words and concatenated to construct the starting words as in algorithm 1.

**Algorithm 1** GREADY CONCATENATE($A$)

**Input:** An array $A$ of significant candidate quadruplets with frequencies within the positive samples.
**Output:** An array of starting keywords.

```
sort A based on frequencies in descending order.
for i = 1 to length(A)-1 do
    for j = i + 1 to length(A) do
        F1 = first three residues of A[i]
        L1 = last three residues of A[i]
        F2 = first three residues of A[j]
        L2 = last three residues of A[j]
        if F1 == L2 then
            Anew = A[j] + (A[i] - F1)
            remove A[i] and A[j] from A
            append Anew to the end of A
            break
        else if F2 == L1 then
            Anew = A[i] + (A[j] - F2)
            remove A[i] and A[j] from A
            append Anew to the end of A
            break
        end if
    end for
end for
return A
```

It was assumed that the positive sequences were generated from the starting words. To derive the generating rules, the starting words were aligned with the positive sequences. The grammatical rules $R^w$ for the keyword model were {insertion, deletion, mutation}. The probability of different rules, $P^w$, was derived by aligning the starting words to the positive sequence using the Smith-Waterman algorithm [18] with the BLOSUM50 matrix [19] and a gap penalty of 8. The effect of the gap penalty on the results was also studied. For each starting word, the highest local alignment score with each sequence in the training sets was obtained. A Wilcoxon rank sum test was performed to test the hypothesis that the alignment score with the positive sequences was higher than the score with the negative sequences. A starting word was referred to as a keyword if the p-value (with Bonferroni correction) was smaller than 0.05.

Fisher's LDA (Linear Discriminate Analysis) [20] was used to discriminate whether a positive sequence contains a keyword $W_i$. For each $W_i$, the maximum number of variants allowed, $T_i^w$, was estimated by comparing the alignment scores of the key word to the positive sequences with that for the negative sequences. Let $\mu_{i1}$ and $\mu_{i2}$ be the mean alignment scores of the keyword to the positive set and negative set, respectively. Let $\sigma_{i1}$ and $\sigma_{i2}$ be the corresponding standard deviations. The classifier's threshold for the $W_i$ was defined as:

$$T_i = \frac{\mu_{i1}\sigma_{i2} + \mu_{i2}\sigma_{i1}}{\sigma_{i2} + \sigma_{i1}} \quad (1)$$

If a sequence's alignment score with $W_i$ was higher than $T_i$, we claimed that the sequence contains $W_i$.

## 2.4 Modelling the sentence grammar

For a given kinase family, the keywords were defined as in the above section. The positive sequences were composed of keywords interspersed with other nonessential amino acids which were ignored in the following analysis. It was assumed that the positive sequences descended from an ancestor sequence composed of all the keywords in a given order and that the positive sequences were generated by deleting some of the keywords. It was further assumed that the keywords in the positive sequences had the same order as in the ancestral sequence. First the keyword order in the ancestor sequence was determined. The deleted keywords and the most likely keyword order in the positive sequences were then determined. Finally a score for a sequence to belong to the kinase family was defined.

To determine the keyword order in the ancestor sequence, the keywords were aligned to each of the positive sequences. For $n$ keywords $W_i (i = 1, 2, \ldots, n)$ and $m$ positive sequences $SEQ_k (k = 1, 2, \ldots, m)$, let $POS_{ik}$ be the starting position of the highest scored segment when $W_i$ was aligned with $SEQ_k$. $W_i$ was referred to as occurring before $W_j$ if $\sum_{k=1}^{m} \text{sign}(POS_{ik} - POS_{jk}) < 0$. The keywords were sorted and the ancestor sentence $S^s = W_{(1)}W_{(2)} \ldots W_{(n)}$ was then defined.

Note that some keywords may be absent in some positive sequences. It is also possible that the order of the keywords in a particular sequence may not be the same as in the ancestor sequence. Therefore, it is not straightforward to find the deleted keywords. The following approach was developed to find the deleted keywords and to give a score for a sequence to be in the family. For keyword $W_{(i)}$, its deletion probability $p_i$ was defined as the fraction of positive sequences with keyword $W_{(i)}$ deleted. Since the deleted keywords are not clear when the keywords are not in order, it is not straightforward to calculate $p_i$. An iterative algorithm was developed to calculate $p_i$ and to determine the deleted keywords in the positive sequences.

1. Initialize $p_i$ by $p_i^{(0)}$.
2. For each positive sequence, determine the deleted keywords using a dynamic programming algorithm (2) to maximize

$$\sum_{i=1}^{n} [u_i + 3 \times \log(1 - p_i)] \times I_i \tag{2}$$

   where $u_i$ is the alignment score between the $i$-th keyword $W_{(i)}$ and the sequence, $I_i = 0$ if the $i$-th keyword is deleted and $I_i = 1$ otherwise.
3. Update $p_i$ by the fraction of sequences with keyword $W_{(i)}$ deleted.
4. Repeat steps 2 and 3 until $p_i$ converges.

Note that the $i$-th keyword $W_{(i)}$ could still be deleted even if $u_i \geq T_i$ (the threshold defined in section 2.3) because of the desired maximisation of equation 2. The second term in the summand is an adjustment for the score of the $i$-th keyword. If $p_i$ is close to 1 (most positive sequences do not contain the keyword), then the adjustment is large. On the other hand, if $p_i$ is small (most positive sequences contain the keyword), then the adjustment is low. The constant 3 was used to be consistent with the definition of the BLOSUM50 matrix [19].

The scores for the negative sequences were also computed using equation 2. By comparing the scores for the positive sequences and those of the negative sequences, Fisher's LDA (linear discriminant analysis) can also be used to define a threshold $T^s$ by equation 1. If the sequence score defined in equation 2 was higher than $T^s$, it was classified as belonging to the function class.

**Algorithm 2** Align Sentences

**Input:**
starting sentence $W_{(1)}W_{(2)} \ldots W_{(n)}$;
keyword models $\{E_i, S_i, R_i, P_i, T_i\}$ $(i = 1, 2, \ldots, n)$;
a sequences $SEQ$ for alignment;
deletion probability $p_i (i = 1, 2, \ldots, n)$.

**Output:** overall matching score and matching flag of each keyword.

let $c_i = 3 \times \log_2 (1 - p_i) (i = 1, 2, \ldots, n)$
let $L_s$ be the length of $SEQ$
**for** $i = 1$ to $n$ **do**
    let $L_i$ be the length of $W_{(i)}$
    let $H$ be the $(L_i + 1) \times (L_s + 1)$ score matrix of local alignment of $W_{(i)}$ and $SEQ$ by the Smith-Waterman algorithm
    **for** $j = 1$ to $L_s$ **do**
        let $u = max\{H_{1j}, H_{2j}, \ldots, H_{L_i j}\} + c_i$
        **if** $u < T_i$ **then**
            $u = 0$
        **end if**
        $U_{i,j} = u$
    **end for**
**end for**
**for** $i = 0$ to $n$ **do**
    $Q_{i,0} = 0$
**end for**
**for** $j = 0$ to $L_s$ **do**
    $Q_{0,j} = 0$
**end for**
**for** $i = 1$ to $n$ **do**
    **for** $j = 1$ to $L_s$ **do**
        $Q_{i,j} = max(Q_{i-1,j}, Q_{i,j-1}, Q_{i-1,j-1} + U_{i,j})$
    **end for**
**end for**
locate the maximum element of $Q$ as $M_Q$, and trace back the path to find out which keywords are aligned.
**return** $M_Q$ as final score and the matching flags of each keywords.

*Table 1. Comparing keywords with the results from PDB structures and PROSITE patterns. HSL model: the matching keywords of the HSL model on the sequence; GLC: the contacting regions for GLC in PDB:1cza; G6P: the contacting regions for G6P in PDB:1cza; PROSITE: the pattern found in PROSITE. The bold capitals in the table are those residues where keywords match exactly with the putative binding sites in PDB. The italic capitals are matching residues with the pattern of PROSITE*

| Source | Hexokinase (EC 2.7.1.1) |
|---|---|
| HSL model | la**LDLGGTNFRV**-*LG**FTFSFP***- ***WTKGF***-**VNDTVGT**- i**NMEWG**- **YEKM**-**SGMY**lgei-**DGSG**-**GA**al |
| GLC (1cza) | **DLGGT**- **TFSFP**c-lit**WTKGF**-**VNDTVGT**-livgtgsn-**NMEWG**afgd-kqr**YEKM**i**SGMY** |
| G6P (1cza) | **LDLGGTNFRV**l- **FTFSF**- **KGF**-**VNDTVGT**mmt-livgtgsn-vdgtlykl- lse**DGSG**k**GA** |
| PROSITE | [***L***ivm]-***G***-***F***-[***T***h]-***F***-***S***-[***F***y]-***P***-x(5)-[livm]-[dnst]-x(3)-[livm]-x(2)-***W***-***T***-***K***-x-[l***F***] |

## 2.5 Sub-families

The family was divided into two or more sub-families if more than 10 positive sequences were incorrectly classified by the model, and the fraction of incorrectly classified sequences of the positive set was higher than 5%. The correctly identified sequences were grouped as a sub-family. The incorrectly classified sequences were further trained using the same procedure as above. The final family model consisted of all the HSL models of the sub-families.

## 2.6 Predictions

Given a new sequence and the HSL model with sub-families, the score of the sequence for each sub-family was calculated as in equation 2. The sequence was predicted to be generated from a sub-family model if the final score was higher than the threshold in the model. A sequence was predicted to be generated from an HSL model with sub-families if it was generated from at least one of the sub-family models.

# 3. RESULTS

The program was applied to the 81 families containing at least 20 sequences and the corresponding HSL models were obtained. The results were validated by comparing the identified keywords with the patterns in PROSITE and PDB. To evaluate the accuracy of the functional classification with the HSL models, 10-fold cross-validation was performed. Comparisons were also made with MEME.

## 3.1 Comparisons with PROSITE, PDB and Pfam

The HSL models were validated by comparing the keywords with patterns in PROSITE and protein structures in PDB. The patterns in PROSITE are functional sites of regular expression. Protein structure data in PDB contain the 3-dimensional coordinates of all amino acid atoms of proteins and the binding atoms. Based on annotation in the Macromolecular Structure Database (MSD)[7], the ligands labeled as base, simple, ion or unknown were removed. A PDB contacting region was then defined as a maximum continuous segment in which all residues were close to the binding ligands with a Euclidean distance less than $10\mathring{A}$, and at least one distance was less than $5\mathring{A}$.

The prediction accuracy was measured by recall and precision. For PROSITE patterns and the positive training sequences, all pattern-matching regions were found on the sequences. The *recall* was defined as the ratio of the number of overlaps between the predicted keywords in the positive sequences and the PROSITE pattern-matching regions over the total number of PROSITE pattern-matching regions. The *precision* was defined as the ratio of the number of overlaps between the predicted keywords in the positive sequences and the PROSITE pattern-matching regions over the total number of predicted keywords. Compared with PROSITE patterns, the HSL achieved an average *recall* of 83.5% and an average *precision* of 23.0% compared to 37.3% and 29.1% for MEME, respectively. Compared with PDB contacting regions, the HSL achieved an average *recall* of 66.1% and an average *precision* of 79.9% compared to 44.7% and 74.7%, respectively, for MEME. The relative low *precision* for PROSITE comparison was due to the incompleteness of PROSITE patterns which were accurate but very short. The detailed results for the PDB and PROSITE comparisons are given in the supplementary materials.

For example, the crystal structure of a mutant monomer of recombinant human brain hexokinase type 1 (PDB code: 1cza) from the hexokinase family (E.C. 2.7.1.1) has two similar domains which form complex with glucose (GLC) and glucose-6-phosphates (G6P). Table 1 shows the comparison results with PDB and PROSITE for one domain. The bold italic capitals are those residues where the keywords match exactly the PDB contacting regions. The italic capitals are matching residues with the PROSITE patterns. Figure 3 shows the locations of keywords in the 3D structure of one domain and their putative functions. The keywords of the HSL model are shown as colour ribbons. These ribbons are close to the ligands and form a pocket to bind with them. Residue Asp657(N) of the purple ribbon, "VNDTVGT", is the putative catalytic base and the conformation of green ribbon "LDLGGTNFRV" is identical with that observed in the G6P/GLC complex of the wild-type enzyme [8].

The predicted keywords using HSL were also compared with the domain structures in the Pfam database. 90% of the keywords predicted by the HSL were part of Pfam domains with names containing the substring "kinase" compared to 77.2% for MEME.

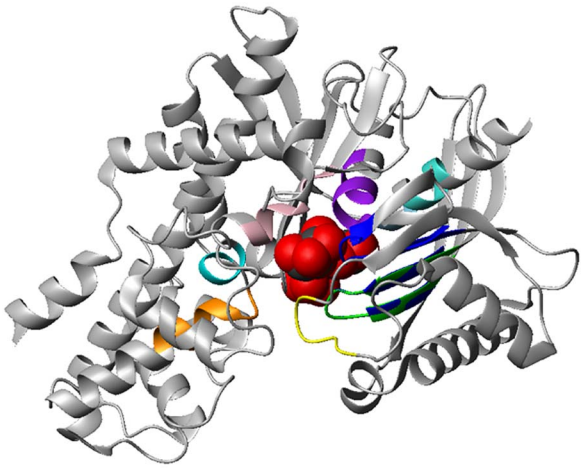

*Figure 3. The ribbon structure of one domain in hexokinase from human brain (PDB code: 1cza) complex with GLC and G6P. GLC and G6P are shown in a space-filling model. The keywords of the HSL model are shown as colour ribbons. These ribbons are close to the ligands and form a pocket to bind with them. Residue Asp657(N) of the purple ribbon, "VNDTVGT", is the putative catalytic base and the conformation of green ribbon "DLGGTNFRV" is identical with that observed in the G6P/GLC complexes of the wild-type enzyme [8]. This drawing was prepared with the program MOLMOL [21].*

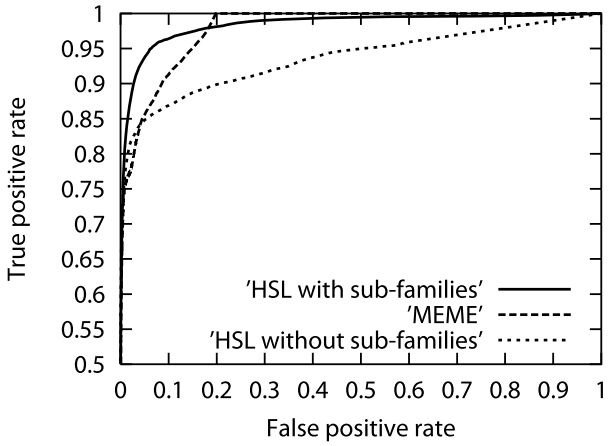


*Figure 4. The ROC curves for HSL with/without sub-families and MEME.*

*Table 2. The average $SN$ and $SP$ of the HSL model with sub-families, the HSL model without sub-families, and MEME*

| Method | $SN$ | $SP$ |
|---|---|---|
| HSL model with sub-families | 94.7% | 94.0% |
| HSL model without sub-families | 83.1% | 96.9% |
| MEME with same $SN$ as HSL model | 94.5% | 85.8% |
| MEME with same $SP$ as HSL model | 86.8% | 93.7% |

## 3.2 Cross-validation and kinase function prediction

The 10-fold cross-validation was also used to evaluate the kinase functional prediction accuracy by the HSL. The sequences from the positive and negative sets were randomly placed into 10 subsets. In each run, 9 positive and 9 negative subsets were chosen for model training, and the remaining positive and negative subsets were held for testing. A score was obtained for each of the testing sequences using the trained HSL model. For a given threshold $T^s$, a test sequence was predicted as positive if its score was greater than $T^s$. The true positives ($TP$), false negatives ($FN$), false positives ($FP$), and true negatives ($TN$) were then obtained. The false positive rate (FPR), true positive rate (TPR), sensitivity ($SN$), and specificity ($SP$) were calculated as follows:

$$SN = TPR = \frac{TP}{TP + TN}, \quad SP = 1 - FPR = \frac{TN}{FP + TN}$$

By adjusting the threshold value $T^s$, the receiver operating characteristic (ROC) curve for the HSL was obtained (Figure (4)). For comparison, the ROC curves were also shown for the HSL model without sub-families and MEME. When the false positive rate is low (e.g. $\leq 0.15$), the true positive rate for the HSL model is always the highest. For another comparison, the relationship between sensitivity and specificity was also studied. Table (2) shows that the HSL significantly outperforms MEME.

## 3.3 Automatic detection of kinase sub-families using the HSL

Another advantage of the HSL model is its ability to detect kinase sub-families with different ancestries automatically. Some proteins evolved from proteins from different ancestries may play the same function, but their active sites may not be the same. For example, Figure (5) shows the unrooted phylogenetic tree for E.C. 2.7.7.12. Classification for E.C. 2.7.7.12 by the HSL model was consistent with the tree. The sequences on the two main branches were classified into two sub-families. The detailed classification results are provided in the supplementary materials.

Adenylate kinase (E.C. 2.7.4.3) is a small monemeric enzyme that catalyzes the reversible transfer of MgATP to AMP (MgATP+AMP⇄MgADP+ADP). The family contains 262 protein sequences in SWISS-PROT. The HSL divided this family into two sub-families, one having 232 sequences and the other 30. Both sub-families have crystal structures in PDB, chain A of PDB:2ECK [9] and chain F of 1NKS [10] were selected for comparison. The global RMSD calculated by SuperPose [11] between the two structures was higher than 10, illustrating they were different. Table (3) shows the keywords of two sub-families and their comparisons with PDB and PROSITE. Most of the keywords in

*Table 3. Comparing keywords with the results from PDB structures and PROSITE patterns. HSL model: the matching keywords of the HSL model on the sequence; AMP: the contacting regions for AMP in chain A of PDB:2eck and chain F of PDB:1nks; ADP: the contacting regions for ADP in chain A of PDB:2eca and chain F of PDB:1nks; PROSITE: the pattern found in PROSITE. The bold capitals in the table are those residues where keywords match exactly with the putative binding sites in PDB. The italic capitals are matching residues with the pattern of PROSITE*

| Source | The first sub-family of E.C. 2.7.4.3. |
|---|---|
| HSL model | **LGAPGAGKGTQA-** **QISTGDMLR-** **LVTDE-** g*FL**LDGFPRTI*** |
| 2eck:AMP | l**LGAPGAGKGTQA-** **QISTGDMLR-**gkqakdimdagk**LVTDE**lvialv-**LDGFPRTI**pga-rkddqeetvrkrlvey |
| 2eck:ADP | l**LGAPGAGKGTQA**qf-**QISTGDMLR-** **DGFPR-**grvyhvkfnpp-dgtkpvaev |
| PROSITE | [livm***F***ywca]-[***L***ivmfyw](2)-***D***-***G***-[***F***yi]-***P***-***R***-x(3)-[nq] |

| Source | The second sub-family of E.C. 2.7.4.3. |
|---|---|
| HSL model | kig**IVTGIPGVGK-** **RDEMR-** **IDTH-** -gy**LPGLP-**vlagstvkv |
| 1nks:AMP | **IVTGIPGVGK**stvl-inygdfm-d**RDEMR**kl-qkklq-**IDTH**avirtp- **LPGLP-**rnr |
| 1nks:ADP | **VTGIPGVGK**stv- **DTH-**srqkrdttr-vivnvegdps |
| PROSITE | [livmfywca]-[livmfyw](2)-d-g-[fyi]-p-r-x(3)-[nq] |
| | this pattern can not be found in the sequences of this sub-family. |

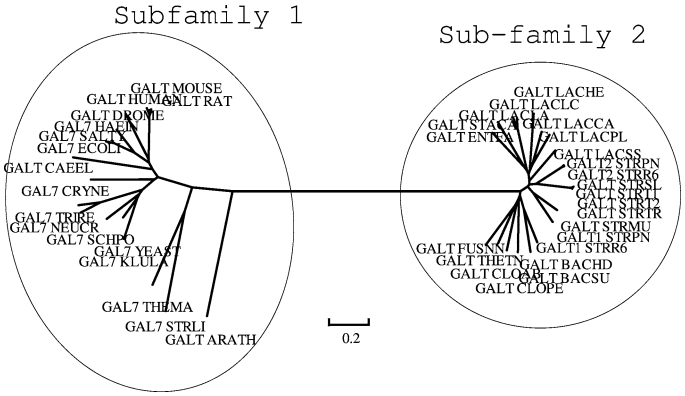


*Figure 5. The unrooted phylogenetic tree for the sequences in E.C. 2.7.7.12 and the sub-families identified by the HSL. The figure was generated using MEGA [22].*

different sub-families match the putative binding sites in PDB and the pattern in PROSITE. For the unique pattern in PROSITE, it fails to characterise the sequences in the second sub-family. Figure (6) shows the different structures of each sub-family displayed as ribbons, and the keywords are coloured in the structures. It indicates that the HSL model is able to classify proteins with different structures from the same functional family.

## 4. DISCUSSION

In this paper, we developed a hierarchical stochastic language (HSL) model for the functional site prediction and classification of kinases. The HSL model takes into account inter-dependencies among the residues and the keywords within the functional sites. In the model, the most conserved residues are first extracted as keywords, then the keywords (with certain variation allowed) are used as basic units to build the high level language model. The sub-families are identified automatically.

The HSL model was applied to 81 kinase families containing at least 20 sequences in Swiss-prot. The predicted sites were validated by protein structures in PDB and patterns in PROSITE. Moreover, most of the predicted keywords were part of kinase domains in Pfam. Some keywords which were not parts of a kinase domain might be part of other functional domains. For example, keyword "ISVK" in the EC 2.7.11.1 family was a part of the additional interactional domains (PH domain) of the Serine/threonine-protein kinase CLA4 (UniProt ID:O14427).

With 10-fold cross-validation for sequence classification, the model achieved both higher sensitivity and specificity than MEME. The model could also deal with kinase families with multiple ancestors. The automatically divided sub-families were consistent with the phylogenetic trees and known protein structures.

The effects of parameter values were also studied, in particular $N_t$ (the number of chosen most common k-tuples in the positive samples) and the gap penalty, on the conclusions. The model was applied with $N_t$=20, 30 and 40, and gap penalties as 6, 8 and 10. The classification results are given in the supplementary materials and the results show that the conclusions are insensitive to the parameter values.

The HSL model has several distinct features. First, the number of parameters needed in the HSL model is much less than other complex models, such as the hidden Markov models used by PFAM. Second, the HSL model captures the most important signals of the family since it starts with the keywords which are short and conserved in a kinase family. Third, the HSL model takes the relationship between keywords into account, which is important for predicting functional sites from the protein primary structures.

In summary, the HSL model can be successfully used to predict kinase functions and kinase functional sites. It can

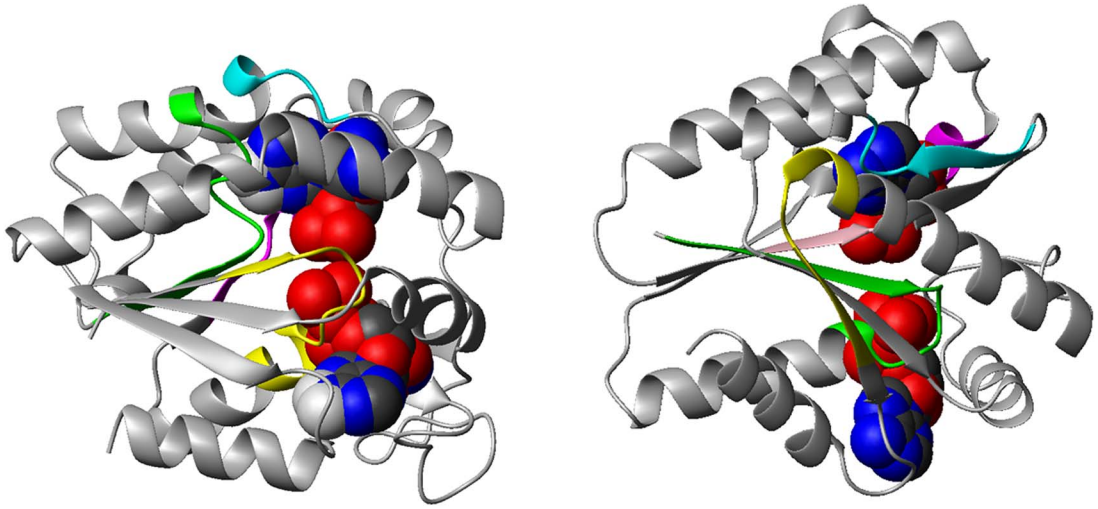

*Figure 6. The ribbon structures of two Adenylate kinases. The left figure shows a domain structure of E. coli. Adenylate kinase with bound AMP and ADP (PDB code:2eck, chain A) [9]. AMP and ADP are shown in the space-filling model. The coloured ribbons are keywords found by the HSL model. The right figure is the domain structure of Adenylate kinase from a trimeric archaeal with bound AMP and ADP (PDB code:1nks, chain F) [10]. AMP and ADP are shown in the space-filling model. The coloured ribbons are keywords found by the HSL model. This drawing was prepared with the program MOLMOL [21].*

also automatically detect kinase sub-families if the sequences within a kinase family came from different ancestors. The kinase functional sites predicted by the HSL may be used as candidates for further experimental validations. The HSL model studied in this paper may also be used to predict other functional protein families.

## ADDITIONAL FILES

### Additional file — supplementary results

The supplementary tables give several results: 1) Comparison of the keywords to PDB and PROSITE. 2) 10-fold cross-validation results of kinase families classification with HSL models. 3) 10-fold cross-validation results of kinase families classification with a different setting of parameters $N_t = 20, 30, 40$ and gap penalties $6, 8, 10$.

## ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China (No.10871009, No.10721403), the National High Technology Research and Development of China (No.2008AA02Z306), the National Key Basic Research Project of China (No.2009CB918503).

*Received 4 August 2010*

## REFERENCES

[1] Hulo N., Bairoch A., Bulliard V., et al. (2006). The Prosite Database. *Nucleic Acids Research* **34 (Database issue)** D227–D230.

[2] Berman H. M., Westbrook J., Feng Z., et al. (2000). The Protein Data Bank. *Nucleic Acids Research* **28** 235–242.

[3] Henikoff S., Henikoff J. G. and Pietrokovski S. (1999). Blocks+: A Non-Redundant Database of Protein Alignment Blocks Derived from Multiple Compilations. *Bioinformatics* **15(6)** 471–479.

[4] Bailey T. L. and Elkan C. (1994). Fitting a Mixture Model by Expectation Maximization to Discover Motifs in Biopolymers. *Proc Int Conf Intell Syst Mol Biol* **2** 28–36.

[5] Finn R. D., Mistry J., Schuster-Bockler B., et al. (2006). Pfam: Clans, Web Tools and Services. *Nucleic Acids Research* **34 (Database issue)** D247–D251.

[6] Fox J. L. (1976). *Protein Structure and Evolution*. Marcel Dekker Inc.

[7] Golovin A., Oldfield T. J., Tate J. G., et al. (2004). E-Msd: An Integrated Data Resource for Bioinformatics. *Nucleic Acids Research* **32 (Database issue)** D211–D216.

[8] Aleshin A. E., Kirby C., Liu X., et al. (2000). Crystal Structures of Mutant Monomeric Hexokinase I Reveal Multiple Adp Binding Sites and Conformational Changes Relevant to Allosteric Regulation. *Journal of Molecular Biology*, **296(4)** 1001–1015.

[9] Berry M. B., Meador B., Bilderback T., et al. (1994). The Closed Conformation of a Highly Flexible Protein: The Structure of E. Coli Adenylate Kinase with Bound Amp and Ampppnp. *Proteins* **19 (3)** 183–198. MR1358433

[10] Vonrhein C., Bonisch H., Schafer G., et al. (1998). The Structure of a Trimeric Archaeal Adenylate Kinase. *Journal of Molecular Biology* **282** 167–179.

[11] Maiti R., Van Domselaar G. H., Zhang H., et al. (2004). Superpose: A Simple Server for Sophisticated Structural Superposition. *Nucleic Acids Research* **32 (Web Server issue)** W590–W594.

[12] Wu C. H., Apweiler R., Bairoch A., et al. (2006). The Universal Protein Resource (Uniprot): An Expanding Universe of Protein Information. *Nucleic Acids Research* **34 (Database issue)** D187–D191.

[13] Bairoch A. (2000). The Enzyme Database in 2000. *Nucleic Acids Research* **28** 304–305.

[14] Webb E. (1992). *Enzyme Nomenclature, 1992: Recommendations of the Nomenclature Committee of the International Union of*

*Biochemistry and Molecular Biology on the Nomenclature and Classification of Enzymes*. Academic Press: San Diego, CA.
[15] Fu K. S. (1992). *Syntactic Pattern Recognition and Applications*. Prentice-Hall.
[16] Durbin R., Eddy S., Krogh A., et al. (1998). *Biological Sequence Analysis, Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press.
[17] Yates F. (1934). Contingency Tables Involving Small Numbers and the Chi-square Test. *Supplement to the Journal of the Royal Statistical Society* **1 (2)** 217–235.
[18] Smith T. F. and Waterman M. S. (1981). Identification of Common Molecular Subsequences. *Journal of Molecular Biology* **147** 195–197.
[19] Henikoff S. and Henikoff J. G. (1992). Amino Acid Substitution Matrices from Protein Blocks. *Proc Natl Acad Sci USA* **89 (22)** 10915–10919.
[20] Duda R. O., Hart P. E., Stork D. G. (2001). *Pattern Classification, Second Edition*. John Wiley & Sons, Inc. MR1802993
[21] Koradi R., Billeter M., Wuthrich K. (1996). Molmol: A Program for Display and Analysis of Macromolecular Structures. *Journal of Molecular Graph* **14** 51–55, 29–32.
[22] Kumar S., Tamura K., Nei M. (2004). Mega3: Integrated Software for Molecular Evolutionary Genetics Analysis and Sequence Alignment. *Brief in Bioinform* **5 (2)** 150–163.

Huan Yu
LMAM, School of Mathematical Sciences
Peking University, Beijing 100871
China
E-mail address: yuhuan@math.pku.edu.cn

Guojun Pei
LMAM, School of Mathematical Sciences
Peking University, Beijing 100871
China
E-mail address: peiguojun@math.pku.edu.cn

Peng Ge
Center for Theoretical Biology
Peking University, Beijing 100871
China
E-mail address: gepeng@ctb.pku.edu.cn

Xiangzhong Fang
LMAM, School of Mathematical Sciences
Peking University, Beijing 100871
China
E-mail address: fangxz@math.pku.edu.cn

Fengzhu Sun
Molecular and Computational Biology Program
University of Southern California
1050 Childs Way, Los Angeles, CA 90089-2910
USA
E-mail address: fsun@usc.edu

Luhua Lai
Center for Theoretical Biology
Peking University, Beijing 100871
China
E-mail address: lhlai@pku.edu.cn

Minping Qian
LMAM, School of Mathematical Sciences
Peking University, Beijing 100871
China

Center for Theoretical Biology
Peking University, Beijing 100871
China
E-mail address: qianmp@math.pku.edu.cn

Minghua Deng
LMAM, School of Mathematical Sciences
Peking University, Beijing 100871
China

Center for Theoretical Biology
Peking University, Beijing 100871
China

Center for Statistical Science
Peking University, Beijing 100871
China
E-mail address: dengmh@pku.edu.cn