On the role of local matching for efficient semi-supervised protein sequence classification

Pavel Kuksa, Pai-Hsi Huang, Vladimir Pavlovic

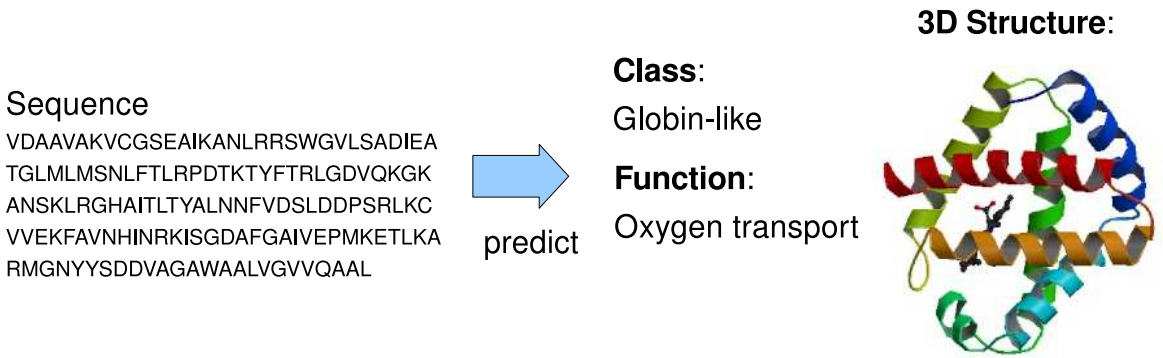
Department of Computer Science
Rutgers University

BIBM, 2008

Outline

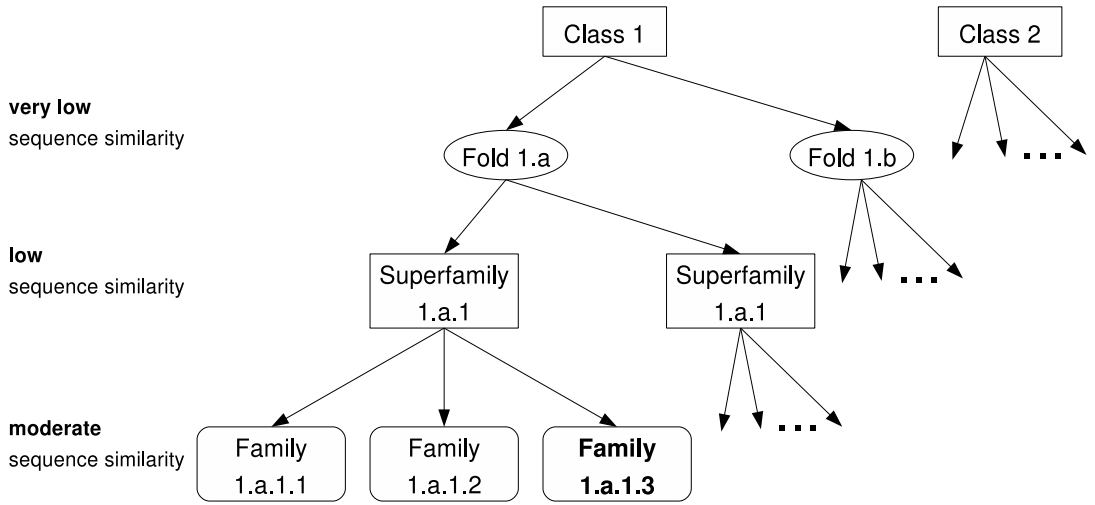
Introduction: problem formulation

- Task: sequence classification in the remote similarity setting
- Goal: classify / group sequences together when basic content of sequences within class very diverse $\Rightarrow$ rely on very *sparse invariant* (preserved) features
- problems occur in many different domains *e.g.* text, music, *etc.*
- focus on biological sequences
  - infer functional properties from *primary sequence only* important (inexpensive)
  - a challenging computational and modeling problem



Introduction: problem formulation (Cont'd)

- Protein sequences: *linear* strings of amino acids ($|\Sigma| = 20$)
- Goal: accurately predict *superfamily* of unknown sequences
- SCOP (Structural Classification of Proteins [6]) hierarchy




```
                10        20        30        40        50        60
Seq1    GMRALEQFANEFKVRRIKLGYTQTNVGEALAAVHGSEFSQTTICRFENLQLSFKNACKLK
                                          ::..  ..: ... :
Seq2    AKRANVSTTTVSHVINKTRFVAEETRNAVWAAIKELHYSPSAVARSLKV
         10        20        30        40        50

                70
Seq1    AILSKWLEEAEQ
```

- remote homology means superfamily
- Challenges:
  - *primary sequence*
  - low similarity
  - variable-length
- focus on methods that are
  - *sparse* (will motivate)
  - *interpretable*

Previous state-of-the-art methods

- mismatch(k,m) kernel [4]: map sequences into k-mer space; similarity defined on *inexact match* of *observed* k-mers for up to m mismatches (induced features have exponential size)
- Sparse spatial sample kernel (SSSK) [3]: map sequences into multi-resolutional sampling space. Inexact matching accomplished using variable-length substrings carrying don't-care characters

Semi-supervised Learning

- Few positive and many negative training sequences: leads to sub-optimal classification performance. Both string kernels overcome such problem using *unlabeled* sequences under the semi-supervised learning framework and show very promising results [2, 7]. The new fixed-length representation of a sequence X takes the following form:

$$\Phi^{new}(X) = \frac{1}{|N(X)|} \sum X' \in N(X) \Phi orig(X'), \quad (1)$$

which implies the following kernel form:

$$K(X, Y) = \sum X' \in N(X) \sum Y' \in N(Y) \frac{K(X', Y')}{|N(X)||N(Y)|}, \quad (2)$$

where $N(X)$, $N(Y)$ the *neighborhood* of sequences X and Y in the unlabeled dataset ($N(X) = \{X' : s(X, X') \leq \delta\}$, $s(.)$ a scoring function, *e.g.* e-Value).

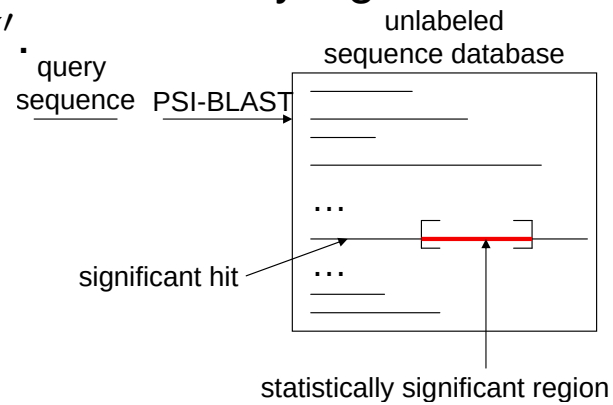
Issues and our proposed methods:

Uncurated unlabeled sequence databases are noisy:

- Abundant long, multi-domain sequences that may contain irrelevant sub-sequences compromising quality of classifiers. (Solution: *Extract* **regions** *from neighboring sequences that are most likely to be biologically relevant.*

$$R(X) \quad = \quad \{x' : s(X, X') \leq \delta\},$$

where $x' \sqsubseteq X'$ the most statistically significant matching region of an unlabeled neighbor X'.



- Abundant (near-)replicated sequences: cause the *averaged* estimate biasing towards over-represented sequences. (Solution: Cluster $R(X)$ to obtain $R^*(X)$ to remove such bias.)

Methods under comparison and evaluation

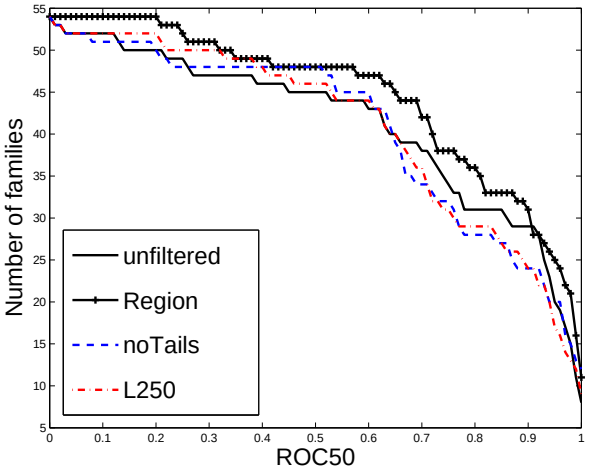
- Methods under comparison:
  1. *Unfiltered*: Use $N(X)$ and $N^*(X)$
  2. *Region*: Use $R(X)$ and $R^*(X)$
  3. *no tails*: Remove sequences that are too long or too short (give mathematical definition here).
  4. *max length*: Remove neighbors whose length is greater than 250 (proposed by Weston *et al*. in [7] for convergence)
- Evaluation method: ROC50 [1] scores, the (normalized) area under the ROC curve computed for *up to* 50 false positives

Settings

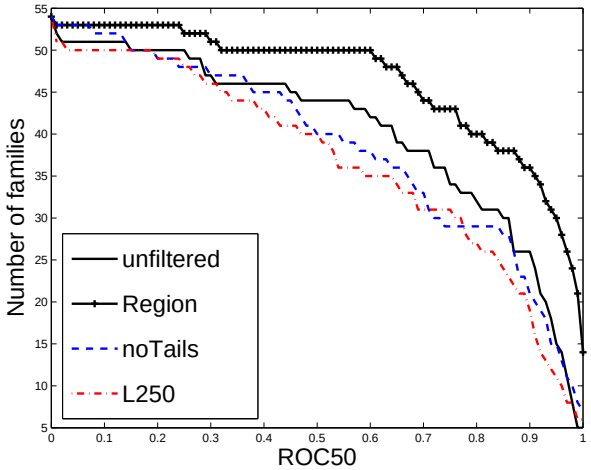
- use $K'(X, Y) = \frac{K(X,Y)}{\sqrt{K(X,X)K(Y,Y)}}$ to remove dependencies between the kernel values and sequence lengths
- use SPIDER[1] for SVMs; linear kernel, default parameters
- use *PDB* (small size) *Swiss-Prot* (moderate size) and *non-redundant* (large size) sets as unlabeled databases
- neighborhood $N(X)$ for each sequence X (train + test) $N(X) = \{X' : eValue(X, X') \leq 0.05\}$ with 2 PSI-BLAST iterations
- Clustering $R(X)$ and $N(X)$ done using the program CD-Hit [5] at 70% similarity level.

[1] http://www.kyb.tuebingen.mpg.de/bs/people/spider

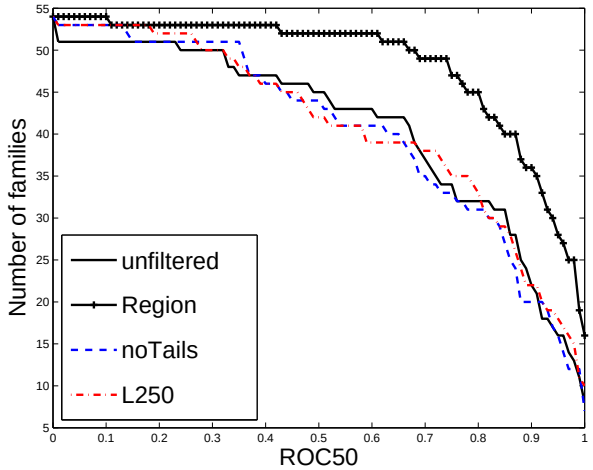
ROC50 curves



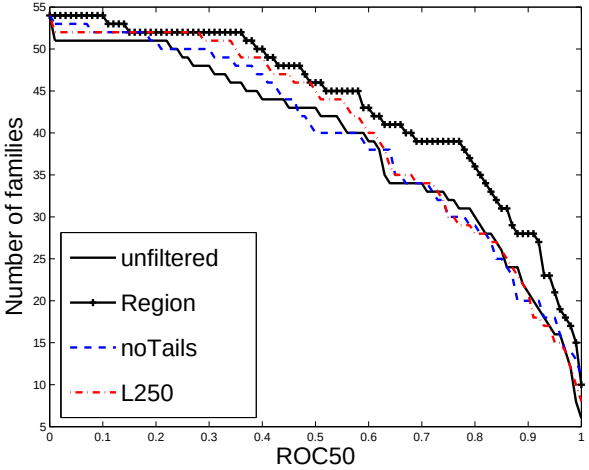
(a) triple(1,3) + PDB



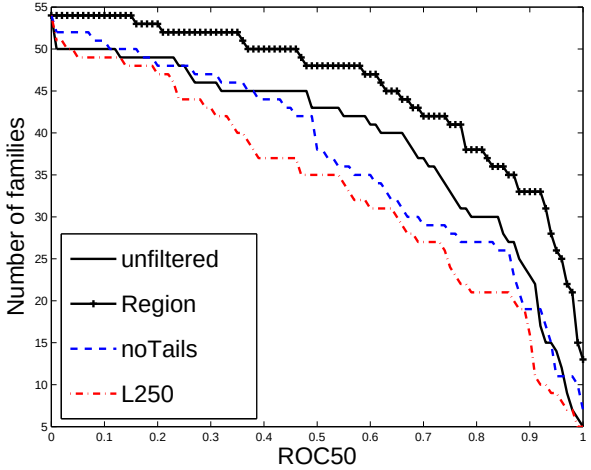
(b) triple(1,3) + Swiss-Prot



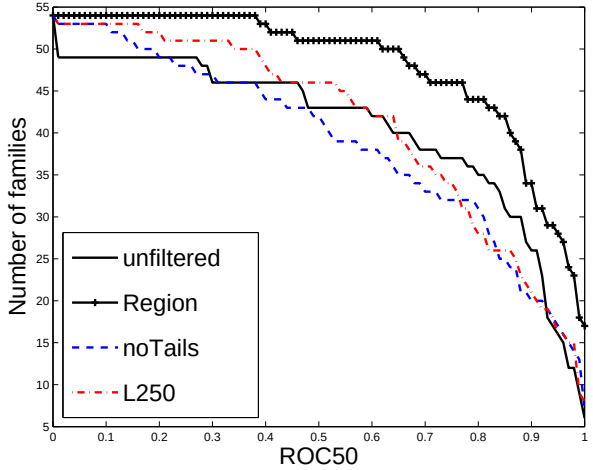
(c) triple(1,3) + NR



(d) mismatch(5,1) + PDB



(e) mismatch(5,1) + Swiss-



(f) mismatch(5,1) + NR

Table: Experimental results for all competing methods using the triple(1,3) kernel.

dataset	neighborhood (no clustering) ROC	ROC50	p-value	clustered neighborhood ROC	ROC50	p-value
PDB						
unfiltered	.9476	.7582	-	.9515	.7633	-
region	.9708	**.8265**	**.0069**	.9716	**.8246**	**.0045**
no tails	.9443	.7522	.5401	.9472	.7559	.5324
max length	.9471	.7497	.4407	.9536	.7584	.5468
Swiss-Prot						
unfiltered	.9245	.6908	-	.9464	.7474	-
region	.9752	**.8556**	**2.46e-04**	.9732	**.8605**	**1.5e-03**
no tails	.9361	.6938	.8621	.9395	.7160	.6259
max length	.9300	.6514	.2589	.9348	.6817	.1369
NR						
unfiltered	.9419	.7328	-	.9556	.7566	-
region	.9824	**.8861**	**1.08e-05**	.9861	**.8944**	**2.2e-05**
no tails	.9575	.7438	.6640	.9602	.7486	.8507
max length	.9513	.7401	.8656	.9528	.7595	.8696

* p-value: signed-rank test on ROC50 scores against *unfiltered* in the corresponding setting

Table: Experimental results on all competing methods using the mismatch(5,1) kernel.

	neighborhood (no clustering)			clustered neighborhood		
dataset	ROC	ROC50	p-value	ROC	ROC50	p-value
PDB						
unfiltered	.9389	.7203	-	.9414	.7230	-
region	.9698	**.8048**	**.0075**	.9705	**.8038**	**.0020**
no tails	.9379	.7287	.9390	.9378	.7301	.7605
max length	.9457	.7359	.4725	.9526	.7491	.3817
Swiss-Prot						
unfiltered	.9253	.6685	-	.9378	.7258	-
region	.9757	**.8280**	**.0060**	.9773	**.8414**	**.0108**
no tails	.9290	.6750	.9813	.9344	.6874	.5600
max length	.9185	.6094	.1436	.9223	.6201	**.0279**
NR						
unfiltered	.9475	.7233	-	.9544	.7510	-
region	.9837	**.8824**	**1.7e-04**	.9874	**.8885**	**1.2e-04**
no tails	.9554	.7083	.7930	.9584	.7211	.7501
max length	.9508	.7421	.7578	.9518	.7613	.9387

*p-value: signed-rank test on ROC50 scores against *unfiltered* in the corresponding setting

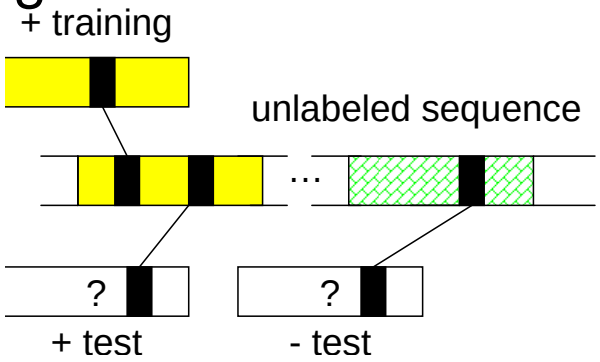
Comparison with other state-of-the-art methods

Table: Comparison of performance (ROC50) against the state-of-the-art methods.

method	PDB	Swiss-Prot	NR
triple(1,3)	.7582	.6908	.7327
triple(1,3), region	**.8265**	.8556	.8861
triple(1,3), region, clustering	.8246	**.8605**	**.8944**
mismatch(5,1)	.7203	.6685	.7233
mismatch(5,1), region	.8048	.8280	.8824
mismatch(5,1), region, clustering	.8038	.8414	.8885
profile(5,7.5)	.7205	.7914	.8151

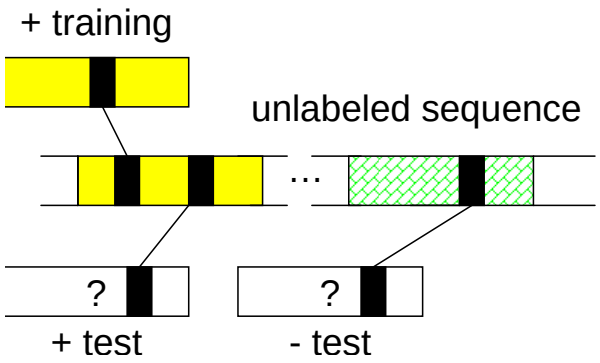
- Number of PSI-BLAST iterations: two
- Region-based method with clustered neighborhood demonstrated the best performance in almost every case.
- ROC50 scores of triple and mismatch kernels *strongly outperform* those of the profile kernel.

The importance of region extraction



- yellow (shaded): positive, green (pattern) negative, arcs: (possibly weak) similarity induced by shared features (black boxes) and absence of arcs indicates no similarity
- Goal: infer membership of the test (unshaded) sequences via the unlabeled sequences (middle).
- Positive training and test sequences share no features and hence no similarity. The unlabeled sequence, which shares features with both sequences, establishes the similarity.
- However, if matching is global, then the unlabeled sequence might also establish the similarity between the positive training and negative test sequences.

The importance of region extraction: an example



- In Swiss-Prot, Sequence Q62059 is multi-domain. The domains belongs to different **folds** (one level higher than superfamily). ROC50 scores without region extraction are .3250 and .3292 for triple and mismatch. ROC50 scores with region extraction improve to .9464 and .9664.

Key Contributions

- motivation: sequences in remotely similar setting; only very few positions *invariant*
- *sparse* profile HMM: recovers critical positions but some not unique to superfamily: need discriminative models
- logistic classifier + *sparsity*-enforcing priors recover unique *critical positions and the preferred residues*; achieve state-of-the-art, but need sub-string comparison and semi-supervised learning for better improvement
- systematic and biologically motivated approach for semi-supervised training + a sparse string kernel: strongly outperforms state-of-the-art methods and also recovers some *critical patterns*
- All presented methods can be applied to a wide range of applications (music, word utterance recognition, text document classification, $\cdots$ *etc.*)

Future work

- still room for improvement; notice some hard to classify superfamilies
- joint training / feature sharing framework to recover (or exploit) tree hierarchy in sequences (have some preliminary results)
- general sequence classification: music, text documents, word utterance, $\cdots$ *etc.*

Publications I

Pai-Hsi Huang, Pavel Kuksa, Vladimir Pavlovic
Blah
conference paper in preparation

References


M. Gribskov and N. L. Robinson.
Use of receiver operating characteristic (roc) analysis to evaluate sequence matching.
Computers & Chemistry, 20(1):25–33, 1996.


P. Kuksa, P.-H. Huang, and V. Pavlovic.
Fast protein homology and fold detection with sparse spatial sample kernels.
In *Proceedings of the Nineteenth International Conference on Pattern Recognition (ICPR 2008)*, 2008.


P. Kuksa, P.-H. Huang, and V. Pavlovic.
Spatially-constrained sample kernel for sequence classification.
In *The Learning Workshop (SNOWBIRD)*, 2008.


C. S. Leslie, E. Eskin, J. Weston, and W. S. Noble.
Mismatch string kernels for svm protein classification.
In *NIPS*, pages 1417–1424, 2002.


W. Li and A. Godzik.