Transport Protein Classification through Structured Prediction and Multiple Kernel Learning

Hongyu Su
Helsinki Institute for Information Technology
Department of Computer Science
Aalto University, Finland
hongyu.su@aalto.fi

Giorgio Valentini
Anacleto Lab
Department of Computer Science
University of Milan, Italy
valentini@di.unimi.it

Sandor Szedmak
Helsinki Institute for Information Technology
Department of Computer Science
Aalto University, Finland
sandor.szedmak@aalto.fi

Juho Rousu
Helsinki Institute for Information Technology
Department of Computer Science
Aalto University, Finland
juho.rousu@aalto.fi

1 Introduction

Membrane transport systems comprise roughly 10% of all proteins in a cell and play a critical role in many biological processes [1]. Improving and expanding their classification is an important goal that can affect studies involving comparative and functional genomics, probing molecular mechanisms of diseases and metabolic processes, and searching new therapeutic targets and pharmacologically relevant transport proteins. In this context, a relevant classification problem is represented by the characterization of transport proteins according to the TC (Transporter Classification) data base (TCDB). Indeed by exploiting this hierarchical taxonomy that includes thousands of families and subfamilies of transporters we implicitly predict the mode of action of the transport activity, the energy coupling mechanism used for the transport, the phylogenetic grouping of the proteins and their substrate specificity [2].

The computational methods proposed so far in literature significantly contributed to enlighten the critical roles played by transporters in the living cells and in several diseases, but suffer of several drawbacks that limit their effectiveness in proteome-wide studies [3]. In particular, most of the proposed computational approaches have been applied to specific categories of transporters (e.g. only to a small subset of the transporter families, or limited to only the most general classes or subclasses of the TCDB), or to specific organisms, and we lack of computational analyses extended to the overall TCDB taxonomy and involving transporters belonging to large sets of organisms. Most methods used only part of the available features that could be helpful to predict transport proteins, but other features could be added in an integrated prediction system to significantly improve performance [3]. Moreover, to our knowledge no methods exploited the hierarchical nature of the TC taxonomy, thus loosing relevant a priori information about the hierarchical relationships between classes.

In this work we introduce an integrative machine learning based approach that tries to consider all the above issues. To this end we propose a novel structured-output method able to explicitly consider the hierarchical relationships between TCDB categories. The proposed classifier exploits state-of-the-art Multiple Kernel Learning (MKL) strategies to integrate a very large set of features extracted from up-to-date databases and it is conceived to be applied virtually to any organism for the TCDB-wide and proteome-wide prediction of the categories of transporters.

2 Methods

We consider a supervised learning setting with an arbitrary input space $\mathcal{X}$ and an output space $\mathcal{Y}$ consisting of the set of ℓ dimensional multilabel vectors $\mathbf{y} = (y_1, \cdots, y_\ell),\ y_i \in \{+1, -1\}$ whose components are called microlabels. In transporter protein classification problem, $\mathbf{x}$ is a protein sequence and $\mathbf{y}$ is a vector of all possible function classes. We assume a collection of p input feature maps $\{\varphi_k(\mathbf{x})\}_{k=1}^p$ in which the kth feature map $\varphi_k(\mathbf{x}) \in \mathbb{R}^{d_k}$ transforms an input $\mathbf{x} \in \mathcal{X}$ into a feature space of d_k dimension. The task is to estimate a function $f : \mathcal{X} \to \mathcal{Y}$ computing the best multilabel $\mathbf{y}$ for an input $\mathbf{x}$.

2.1 Multiple kernel learning

We assume p input kernels $\{K_1, \cdots, K_p\}$ composed from p different input feature maps and an ideal target kernel computed by $K_\mathbf{y} = YY'$ where the rows of Y are formed of the multilabel vectors. We assume that all kernels are centered in the corresponding feature space. As input features generated from transporter protein sequences are heterogeneous, a uniform combination of corresponding kernel matrices will be suboptimal. Therefore, we conside the following two advanced multiple kernel learning (MKL) approaches.

Centered Kernel Alignment (ALIGN). This approach computes a weighted combination of input kernels [4] $K_{\text{ALIGN}} = \sum_{k=1}^p \alpha_k K_k^c$ the corresponding weights, α_k, are computed by $\alpha_k = \frac{\langle K^c, K_\mathbf{y}^c \rangle_F}{||K^c||_F ||K_\mathbf{y}^c||_F}$, where $\langle \cdot, \cdot \rangle_F$ denotes the Frobenius product and $||\cdot||_F$ is the Frobenius norm.

Two-Stage MKL (ALIGNF). This approach improves ALIGN by assuming the correlation between input kernels. ALIGNF computes a convex combination of input kernels [4] $K_{\text{ALIGNF}} = \sum_{k=1}^p \beta_k K_k^c$, where kernel weights β_k are learned such that the alignment between the combined kernel and the target kernel is maximized

$$\max_{\beta} \quad \frac{\langle K_{\text{ALIGNF}}^c, K_\mathbf{y}^c \rangle_F}{||K_{\text{ALIGNF}}||_F \left|\left|K_\mathbf{y}^c\right|\right|_F}, \qquad \text{s.t.} \sum_{k=1}^p \beta_k^2 = 1, \beta_k \geq 0,\ \forall k.$$

Uniform Kernel Combination is used as the baseline. This approach computes an 'average' kernel from input kernel matrices

$$K_{\text{UNIF}} = \frac{1}{p} \sum_{k=1}^p K_k^c,$$

which is equivalent to concatenating original feature vectors.

2.2 Hierarchical multilabel classification approaches

The transport classification (TC) is a four-level hierarchical system defined on transporter protein function classes. In particular, the system is a rooted tree in which nodes correspond classes and directed edges correspond to relationship between classes and subclasses.

We use two different multilabel classification approaches to predict the hierarchy.

Hierarchical Structured Output Prediction (SOP) . The first method is based on the hierarchical structured output prediction framework of [5, 6] that models the hierarchy as an associative Markov network $G = (E, V)$ where nodes corresponds to microlabels (classes). There is an edge $(i, j) \in E$ if two microlabels y_i and y_j are connected in the underlying classification hierarchy.

The proposed model (SOP) is based on embedding the input and output into a joint feature space and learning in that space a linear score function $F(\mathbf{w}, \mathbf{x}, \mathbf{y}) = \langle \mathbf{w}, \boldsymbol{\phi}(\mathbf{x}, \mathbf{y}) \rangle$ given by the inner product of parameters $\mathbf{w}$ and the joint feature $\boldsymbol{\phi}(\mathbf{x}, \mathbf{y})$. The joint feature map $\boldsymbol{\phi}(\mathbf{x}, \mathbf{y})$ is composed by the tensor product of the input feature map $\boldsymbol{\varphi}(\mathbf{x})$ and the output feature map $\boldsymbol{\psi}(\mathbf{y})$. The tensor prodct will then consist all pairs of input and output features $\boldsymbol{\phi}_{i,j}(\mathbf{x}, \mathbf{y}) = \boldsymbol{\varphi}_i(\mathbf{x}) \boldsymbol{\psi}_j(\mathbf{y})$.

The output feature map will encode the multilabel $\mathbf{y}$ according to the structure of the network G defined by

$$\psi(\mathbf{y}) = (\psi_{e,u_e}(\mathbf{y}))_{e,u_e} = (\mathbf{1}_{\{\mathbf{y}_e = u_e\}})_{e,u_e},\ e \in \mathbf{E},\ u_e = \{+1, -1\}^2,$$

where $\mathbf{1}_{\{\cdot\}}$ is the indicator function. In other words, for each edge e and edge label u_e, we define the feature $\psi_{e,u_e}(\mathbf{y})$ to be 1 if edge label $\mathbf{y}_e$ is u_e in the network G (and 0 otherwise).

The feature weight parameters $\mathbf{w}$ of the score function are learnt by optimizing the following regularized structured output learning problem

$$\begin{aligned} \min_{\mathbf{w},\xi_i} \quad & \frac{1}{2}\,||\mathbf{w}||_2^2 + C\sum_{i=1}^{m}\xi_i \\ \text{s.t.} \quad & \langle \mathbf{w}, \boldsymbol{\phi}(x_i,\mathbf{y}_i)\rangle - \max_{\mathbf{y}\in\mathcal{Y}/\mathbf{y}_i} \langle \mathbf{w}, \boldsymbol{\phi}(x_i,\mathbf{y})\rangle \geq \ell(\mathbf{y}_i,\mathbf{y}) - \xi_k, \\ & \xi_i \geq 0\,, \forall\, i \in \{1,\ldots,m\}, \end{aligned} \tag{1}$$

where $||\cdot||_2$ denotes the L_2 norm, and $\ell(\mathbf{y},\mathbf{y}_i)$ is the loss function defined on two multilabels. The impact of the constraints of the above optimization problem is to push the score of input $\mathbf{x}_i$ with correct multilabel $\mathbf{y}_i$ above the scores of all competing multilabels $\mathbf{y} \in \mathcal{Y}/\mathbf{y}_i$. The slack parameters ξ_i is used to relax the constraints so that a feasible solution can always be found. C is the margin slack parameter that controls the amount of regularization in the model. The objective minimizes the L_2-norm of the weights and the slacks allocated to the training data which is equivalent to maximizing the margin subject to allowing some data to be outliers.

The search space appears to be exponential in the number of microlabels $|\mathcal{Y}| = 2^{\ell}$. However, we observe that a valid functional annotation is always a simple path from the root to a leave in the transport classification (TC) system. Therefore, we are able to substantially reduce the search space to a set of valid annotations which is linear in the number of leaves in G. The exponential reduction of the search space dramatically improves the training time and the performance of the model. To allow the use of kernels for high dimensional feature spaces, we use marginalized dual representation [7, 5] of (1) combined with conditional gradient descent algorithm.

Max-margin regression (MMR). To find the most suitable type of kernels the exponentially large search space is a real bottleneck. To alleviate that problem we can apply a compressed approach provided by the Maximum Margin Regression (MMR) whose primal formulation is given as

$$\begin{aligned} \min_{\mathbf{w},\xi_i} \quad & \frac{1}{2}\,||\mathbf{w}||_2^2 + C\sum_{i=1}^{m}\xi_i \\ \text{s.t.} \quad & \langle \psi(\mathbf{y}_i), \mathbf{w}\phi(x_i)\rangle \geq 1-\xi_i,\ \xi_i \geq 0\,, \forall\, i \in \{1,\ldots,m\}. \end{aligned}$$

In the MMR the input and output objects are separated, and the multilabel vectors are handled as complete entities by ignoring the potential interactions between the microlabels, [8]. This approach allows to scan the possible kernels at the cost of a simple SVM type binary classification problem.

3 Data and Experiments

Data. We collect 12546 protein sequences from Transporter Protein Classification Database (TCDB) [2]. After removing duplications, 12515 unique protein sequences are remained. We annotate each protein with 3145 function classes extracted from the classification hierarchy. The structure of the hierarchy is also collected and used as the underlying Markov network.

For each protein sequence, we also generate sequence similarity features by aligning the sequence against the whole TCDB database with BLAST. In addition, we generate 18 different features with InterProScan [9]. The statistics of the features are illustrated in Table 1.

Type	Dim	Type	Dim	Type	Dim
Protein Domain	145	Hapmap	209	SMART	240
Protein Family	512	PRINTS	579	Panther	4070
Gene3D	611	PIRSF	283	PfamA	2025
Prosite Profile	282	TIGRFAM	769	Prosite Patterns	285
Coil	1	TMHMM	1	Phobius	7
SignalP1	2	SignalP2	2	SignalP3	1

Table 1: Statistics of protein features generated by InterProScan.

	F1		F1		F1
Blast	**74.5**	PIRSF	11.2	SignalP2	3.6
Coils	03.6	PRINTS	13.7	SignalP3	4.1
Gene3D	04.0	ProDom	03.5	SMART	7.2
Hamap	05.5	ProSite Patterns	03.0	SUPERFAMILY	14.9
PANTHER	42.4	ProSite Profiles	15.6	GRFAM	22.3
Pfam	38.2	SignalP1	1.1	TMHMM	06.8
Phobius	11.7				

Table 2: Microlabel F1 obtained by SVM on individual features.

		F_1				$0/1$			
		Linear		*Gaussian*		*Linear*		*Gaussian*	
	SVM	MMR	SOP	MMR	SOP	MMR	SOP	MMR	SOP
UNIF	68.3	35.1	71.7	79.9	79.9	06.9	55.1	64.1	64.3
ALIGN	74.6	33.9	76.9	83.0	82.8	05.9	58.4	68.3	68.6
ALIGNF	79.2	50.1	80.0	**85.4**	*85.2*	21.0	62.9	*72.7*	**72.8**

Table 3: Prediction performance of models.

Experimental setup. We perform experiments to evaluate the performances of different kernels, multiple kernel learning approaches, and machine learning models. We compute linear kernels over 19 input feature maps and apply three MKL approaches to combine input kernels including: uniform kernel combination (UNIF), ALIGN, and ALIGNF. In addition, a Gaussian kernel is composed over the kernels computed from MKL. We train a collection of SVMs, one for each microlabel. On the other hand, MMR and SOP are both multilabel learners. Kernel parameter and margin slack parameters of the learning models are tuned using cross-validation. The measures of performance include F_1 scores computed by pooling all microlabel predictions, and multilabel accuracy computed by requiring the whole annotation vector to be predicted correctly. The results are from 5-fold cross validation.

Results. Table 2 reports the predictive potential of different individual features in conjunction of SVM. We can notice that Blast provides the best microlabel F1 with 74.5%, with Pfam and Panther the next best features. Most of the features have poor F1 scores when used individually.

We compare the prediction performance of SVM, MMR, and SOP in Table 3. First, we notice that multiple kernel learning with ALIGNF leads to the best performance. On SVM, it is the only MKL methods that beats using Blast feature as the single input feature. The most accurate models overall are obtained with ALIGNF using the Gaussian kernel over the learned mixture and either of the multilabel learning methods SOP or MMR. Between the two multilabel methods, we notice that SOP is very competitive in all setups, with both linear and Gaussian kernel and with all three methods of combining kernels. With Gaussian kernel MMR closely matches the performance of SOP.

In absolute terms, the best prediction accuracies of 85.4% multilabel F1 score and 72.8% multilabel accuracy are remarkably good, and show that transport functions can be well predicted from sequence derived features to a fine-grained detail.

4 Conclusions

We have presented here a study on predicting transport protein functions as encoded by the Transport Classification Database (TCDB). According to our knowledge, this is the first time prediction of the whole hierarchy has been attempted. Our experiments show that combining state-of-the-art sequence-derived features, multiple kernel learning and structured output learning make detailed functional classification viable.

References

[1] Zachary Chiang, Ake Vastermark, Marco Punta, Penelope Coggill, Jain Mistry, Robert Finn, and Milton Saier. The complexity, challenges and benefits of comparing two transporter classification systems in tcdb and pfam. *Briefings in Bioinformatics*, 2015.

[2] Milton H. Saier, Vamsee S. Reddy, Dorjee G. Tamang, and Ake Vastermark. The transporter classification database. *Nucleic Acids Research*, 42(D1):D251–D258, 2014.

[3] M. Michael Gromiha and Yu-Yen Ou. Bioinformatics approaches for functional annotation of membrane proteins. *Briefings in Bioinformatics*, 15(2):155–168, 2014.

[4] Corinna Cortes, Mehryar Mohri, and Afshin Rostamizadeh. Algorithms for learning kernels based on centered alignment. *The Journal of Machine Learning Research*, 13(1):795–828, 2012.

[5] J. Rousu, C. Saunders, S. Szedmak, and J. Shawe-Taylor. Kernel-based learning of hierarchical multilabel classification models. *The Journal of Machine Learning Research*, 7:1601–1626, 2006.

[6] Katja Astikainen, Liisa Holm, Esa Pitkänen, Sandor Szedmak, and Juho Rousu. Towards structured output prediction of enzyme function. In *BMC proceedings*, volume 2, page S2. BioMed Central Ltd, 2008.

[7] Ben Taskar, Carlos Guestrin, and Daphne Koller. Max-margin markov networks. In S. Thrun, L.K. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, pages 25–32. MIT Press, 2004.

[8] Hanchen Xiong, Sandor Szedmak, and Justus Piater. Scalable, accurate image annotation with joint svms and output kernels. *Neurocomputing*, 169:205–214, 2015.

[9] Philip Jones, David Binns, Hsin-Yu Chang, Matthew Fraser, Weizhong Li, Craig McAnulla, Hamish McWilliam, John Maslen, Alex Mitchell, Gift Nuka, et al. Interproscan 5: genome-scale protein function classification. *Bioinformatics*, 30(9):1236–1240, 2014.