ESANN 2011 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2011, i6doc.com publ., ISBN 978-2-87419-044-5.
Available from http://www.i6doc.com/en/livre/?GCOI=28001100817300.

Iterative multi-task sequence labeling for predicting structural properties of proteins

Francis Maes, Julien Becker and Louis Wehenkel *

University of Liege - Dept of Electrical Engineering and Computer Science
Institut Montefiore, B28, B-4000, Liege - Belgium

Abstract. Developing computational tools for predicting protein structural information given their amino acid sequence is of primary importance in protein science. Problems, such as the prediction of secondary structures, of solvent accessibility, or of disordered regions, can be expressed as sequence labeling problems and could be solved independently by existing machine learning based sequence labeling approaches. But, since these problems are closely related, we propose to rather approach them jointly in a multi-task approach. To this end, we introduce a new generic framework for *iterative multi-task sequence labeling*. We apply this - conceptually simple but quite effective - strategy to jointly solve a set of five protein annotation tasks. Our empirical results with two protein datasets show that the proposed strategy significantly outperforms the single-task approaches.

1 Introduction

Ab initio prediction of the tertiary structure of proteins (*i.e.* computing 3D positions of all their atoms, from their amino-acid sequences) is a very important, extremely difficult, and largely unsolved problem in physical chemistry and biology. To progress towards this goal, many research efforts have already been devoted to address surrogate (and simpler) problems that can be formalized as sequence labeling problems, where the input is a sequence of amino acids and the output is a corresponding sequence of labels describing some property of these amino acids. Well-known examples of such surrogate problems are: (i) *secondary structure prediction*, where labels correspond to local 3D structures such as alpha helices, beta strands or turns; (ii) *solvent accessibility prediction*, where labels are levels of exposition of protein residues to the solvent; and (iii) *disordered regions prediction*, that aims at identifying amino acids belonging to a disordered region of the protein.

In the bioinformatics literature, these problems have mostly been treated independently: *i.e.* one designs (*e.g.* by machine learning) and uses a predictor for inferring secondary structure and separately designs and uses another predictor for inferring solvent accessibility. On the other hand, the field of machine learning has investigated in the recent years so-called *multi-task* approaches, which aim at treating multiple related prediction tasks simultaneously (both at the learning stage and at the prediction stage) with the hope to get an improvement

*This paper presents research results of the Belgian Network BIOMAGNET (Bioinformatics and Modelling: from Genomes to Networks), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, and of the EU FP7 PASCAL2 network of excellence. Julien Becker is recipient of a F.R.I.A. fellowship.

ESANN 2011 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2011, i6doc.com publ., ISBN 978-2-87419-044-5. Available from http://www.i6doc.com/en/livre/?GCOI=28001100817300.

on each one of the addressed tasks with respect to predictors designed and used in a single task fashion. Since the various protein structure prediction tasks are closely related, it is a natural idea to explore such multi-task approaches in this context. Although not formulated explicitly in these terms, one example of such a multi-task approach for protein structure prediction has already been proposed by [1], by combining solvent accessibility prediction and secondary structure prediction within a unified system.

In the machine learning literature, most work on multi-task approaches has focused on the multi-task classification problem, *i.e.* solving multiple related classification problems jointly. In the context of the prediction of protein structural properties, prediction targets are label sequences. To deal with this different kind of data, we introduce in Section 2 a new and conceptually simple, yet quite effective, multi-task framework called *iterative multi-task sequence labeling*. Section 3 provides our experimental protocols and results in the context of five protein annotation tasks, and Section 4 concludes and highlights further research directions.

2 Iterative multi-task sequence labeling

Most multi-task learning approaches rely on the use of an internal representation shared over all considered tasks, such a shared representation being likely to better capture the essence of the input data by exploiting commonalities among the different tasks. We adopt here another approach to multi-task learning, namely *black-box multi-task learning*[1]: we combine the learning of single-task sequence labeling base-models, where base-models are considered as black boxes and may be of any kind, from simple classification-based approaches to modern structured prediction approaches [2, 3].

Notations. We consider the multi-task sequence labeling problem where the aim is to learn a mapping from input sequences $x \in \mathcal{X}$ to target sequences $y_1, \ldots, y_T \in \mathcal{Y}_1, \ldots, \mathcal{Y}_T$ for each task $t \in [1, T]$. To learn these tasks, we have access to a training set composed of pairs of input sequences associated with some or all of the target sequences. The training set is denoted $D = \{(x^{(i)}, y_1^{(i)}, \ldots, y_T^{(i)})\}_{i \in [1,N]}$ where N is the number of training examples. The "state space" $\mathcal{S}$ of the multi-task problem is defined by $\mathcal{S} = (\mathcal{Y}_1 \cup \{\epsilon_1\}) \times \cdots \times (\mathcal{Y}_T \cup \{\epsilon_T\})$, where ϵ_t denotes a special output label used to represent the fact that the target y_t is not (yet) specified.

Principle. The core idea of iterative multi-task learning is to iteratively re-estimate the targets $y_1, \ldots, y_T$, using at each step the global input x and the last predicted targets of each task as input of the base model. The process is initialized with empty predictions for all targets, *i.e.* $s = (\epsilon_1, \ldots, \epsilon_T)$. At the first step, the first target y_1 is predicted with a first sequence labeling model. A second sequence labeling model is then used to predict y_2 given x and the predicted y_1. The third model predicts y_3 given x, and the predictions of y_1 and y_2, and so on. Once all the targets have been estimated once, we have performed one *pass*. The complete model is composed of $P \times T$ models used in this way by performing P passes sequentially (P is a meta-parameter of the algorithm).

[1] See discussion at `http://hunch.net/?p=160`

ESANN 2011 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2011, i6doc.com publ., ISBN 978-2-87419-044-5.
Available from http://www.i6doc.com/en/livre/?GCOI=28001100817300.

Algorithm 1 Iterative multi-task sequence labeling inference

Given an input $x \in \mathcal{X}$ and a model chain $(M_{1,1}, \ldots, M_{1,T}, M_{2,1}, \ldots, M_{P,T})$

1: $s \leftarrow (\epsilon_1, \ldots, \epsilon_T)$ ▷ initial state
2: **for** $p = 1$ to P **do** ▷ for each pass
3: **for** $t = 1$ to T **do** ▷ for each task
4: $\hat{y}_t \leftarrow M_{p,t}(x, s)$ ▷ estimate target t
5: $s \leftarrow (s_1, \ldots, s_{t-1}, \hat{y}_t, s_{t+1}, \ldots, s_T)$ ▷ update targets state
6: **end for**
7: **end for**
8: **return** s ▷ return current state of all targets

Algorithm 2 Iterative multi-task sequence labeling training

Given a training set $D = \{(x^{(i)}, y_1^{(i)}, \ldots, y_T^{(i)})\}_{i \in [1,N]}$,
Given a sequence labeling learning algorithm $\mathcal{A}$,
Given a number of passes P,

1: $S \leftarrow \{s^{(i)} = (\epsilon_1, \ldots, \epsilon_T)\}_{i \in [1,N]}$ ▷ initial state
2: **for** $p = 1$ to P **do** ▷ for each pass
3: **for** $t = 1$ to T **do** ▷ for each task
4: $D_t \leftarrow \{((x^{(i)}, s^{(i)}), y_t^{(i)})\}_{i \in [1,N]}$ ▷ create training set
5: $M_{p,t} \leftarrow \mathcal{A}(D_t)$ ▷ train a model for task t
6: $S \leftarrow$ update S given D_t and $M_{p,t}$ ▷ update current state
7: **end for**
8: **end for**
9: **return** $(M_{1,1}, \ldots, M_{1,T}, M_{2,1}, \ldots, M_{P,T})$ ▷ return model chain

Model chain. At the core of iterative multi-task learning is a chain of sequence labeling models denoted $(M_{1,1}, \ldots, M_{1,T}, M_{2,1}, \ldots, M_{P,T})$, where $M_{p,t}$ is the model of the p-th pass and the t-th task. Distinct models are learned at each pass; this is motivated by the fact that – since targets are re-estimated at each pass – the input-output distribution of the underlying sequence labeling learning problems changes slightly from pass to pass. For example, estimating a target for the first time (given the input only) is not the same problem as estimating it for the second time (given the input and the t initial predictions).

Training and inference. Algorithm 1 and Algorithm 2 respectively describe inference and training in iterative multi-task sequence labeling. Given the model chain, inference simply chains the base inferences iteratively, by maintaining $s \in \mathcal{S}$, the current state of all target, *i.e.* $s = (\hat{y}_1, \ldots, \hat{y}_T)$. It is initialized with *unspecified* targets $(\epsilon_1, \ldots, \epsilon_T)$ (line 1) and each step consists in predicting a target sequence $\hat{y}_t$ and replacing it in the current state (lines 4–5). The final predictions are given by s at the end of inference (line 8).

Training consists in creating the model chain given the training set. Similarly to inference, this is performed iteratively and relies on a set of current states $\{s^{(1)}, \ldots, s^{(N)}\}$. These current states are first initialized to unspecified targets (line 1). Each learning step then adds an element to the model chain. This

ESANN 2011 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2011, i6doc.com publ., ISBN 978-2-87419-044-5. Available from http://www.i6doc.com/en/livre/?GCOI=28001100817300.

involves creating a (single-task) sequence labeling training set (line 4), training a sequence labeling model (line 5) and updating the current state of each example (line 6). Base-model training inputs contain both the global input x and the current state $s = (\hat{y}_1, \ldots, \hat{y}_T)$.

Avoiding over-fitting. Since the chain of models may potentially be long (up to 40 models in our experiments), particular care must be taken to avoid over-fitting. Indeed, training examples may quickly be perfectly learned by the first models in the model chain, hence dangerously biasing the training data for all remaining models of the chain. Since we used very large training sets and simple linear classifiers in our experiments, we did not encounter this problem. However, if necessary, such over-fitting problems could be avoided in at least two ways: either by generating intermediate predictions through the use of cross-validation as exposed for the stacked learning approach [4], or by introducing noise into intermediate predictions as proposed in [5].

3 Multi-task protein annotation

Datasets. We used two datasets extracted from the Protein Data Bank for our experiments. We built the first set, PDB30, by randomly selecting 500 proteins from PDB, with a maximum pairwise identity of 30% to ensure significant differences between training and testing proteins. We also use the standard PSIPRED data set [6], which is composed of 1385 training proteins and 187 testing proteins, all of which have a different fold, *i.e.* significant shape differences. The PSIPRED method is considered as the state-of-the-art in the field of secondary structure prediction. We applied traditional pre-processing to both datasets: secondary structure and solvent accessibility have been determined with the DSSP program, and the input has been enriched with position-specific scoring matrices (PSSMs), computed with three iterations of the PSI-BLAST tool.

Tasks. We consider a set of 5 related tasks: secondary structure prediction (two different versions with 3 and 8 labels), solvent accessibility prediction (2 labels), disordered regions prediction (2 labels) and structural alphabet prediction (27 labels, see [7]). The two versions of secondary structure give two different levels of granularity and seem to be redundant but, in our experiments, we have noted an improvement of both tasks when both are present. The structural alphabet is a discretization of the protein backbone conformation as a series of overlapping fragments of four residues length. This representation, as a prediction problem, is not common in the literature. Here, it is used as a third level of granularity for local 3D structures and seems to also improve predictions of other tasks. Since, the disorder classes are not uniquely defined, we used the definition of the CASP [8] competition, *i.e.* segments longer than three residues but lacking atomic coordinates in the crystal structure were labelled as "disorder" whereas all other residues were labelled as "order". We have used a threshold of 20% to define the two states ("buried" and "exposed") of the solvent accessibility task.

The default scoring measure is label accuracy, *i.e.* the percentage of correctly predicted labels on the test set. Since disordered regions labeling is a strongly unbalanced problem, label accuracy is not appropriate for this task. Instead, we have used a classical evaluation measure for disordered regions prediction: the

ESANN 2011 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2011, i6doc.com publ., ISBN 978-2-87419-044-5. Available from http://www.i6doc.com/en/livre/?GCOI=28001100817300.

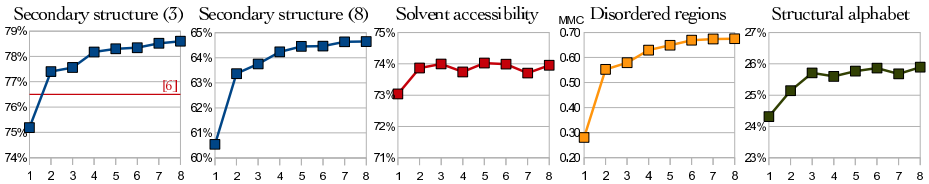

Fig. 1: Evolution of test scores after growing numbers of passes on the PsiPred dataset. State-of-the-art results of [6] are available on the "Secondary Structure (3)" prediction task.

		Pdb30		PsiPred	
Task	Labels	Single-task	Multi-task	Single-task	Multi-task
Secondary structure	3	75.45 %	76.35 %	76.29 %	78.60 %
Secondary structure	8	60.38 %	62.69 %	62.25 %	64.64 %
Solvent accessibility	2	71.56 %	73.52 %	73.51 %	73.95 %
Disordered regions	2	0.4212	0.4983	0.5611	0.6749
Structural alphabet	27	16.81 %	18.14 %	24.88 %	25.89 %

Table 1: Single-task vs multi-task results on Pdb30 and Psipred (8 passes).

Matthews Correlation Coefficient (MCC) [1].

Methods. The base sequence-labeling model uses a simple classification-based approach: each label is predicted independently of the others, on the basis of features describing local and global properties of the protein. The base classifier is a linear SVM trained with stochastic gradient descent (with learning rates tuned on the training set). Our feature set is similar to those proposed by [6, 9]. Global features describe the distribution of amino acids inside the protein and the length of the protein. Local features rely on a sliding window of size 15. For each position in this window, there are features describing the amino acid, the PSSM row and the currently predicted labels of each task.

Results. We have trained iterative multi-task sequence labeling with up to $P = 8$ passes which gives model chains of length $P \times T = 40$. To observe the effect of the iterative re-estimation of targets, we have evaluated each task by "cutting" the model chain after a given number of passes $P_{max} \in [1, 8]$. Figure 1 gives the test scores for each task as a function of the number of passes on the PsiPred dataset. It is clear that all the tasks benefit from iterative re-estimation of targets, especially during the first passes. During the last passes, some scores occasionally degrade, but we do not observe strong over-fitting (see discussion Section 2) in these experiments. Importantly, in all cases, the re-estimated targets after several passes are significantly better than the initially estimated targets.

To measure to what extend our positive results are due to multi-tasking, we have performed one baseline experiment per task by using iterative sequence labeling in a single-task setup. These baselines rely on iterative re-estimation of targets, but do not use predictions from the other tasks. The comparison between our multi-task model and its single-task counterparts is given in Table 1, for models inferred after 5 passes. We observe for these results that on both datasets, the multi-task approach systematically outperforms the single-task approach, *e.g.*: +2.31% for secondary structure prediction and +0.114 MCC

ESANN 2011 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2011, i6doc.com publ., ISBN 978-2-87419-044-5.
Available from http://www.i6doc.com/en/livre/?GCOI=28001100817300.

for disordered regions prediction on the PsiPred testing set. We also observe that only the multi-task approach outperforms the state-of-the-art results on PsiPred with +2.1% improvement.

4 Conclusion

We have introduced a conceptually simple framework for *iterative multi-task sequence labeling*, a new multi-task machine learning approach to jointly solve multiple related sequence labeling tasks, and which can take advantage of any sequence labeling algorithm. We have made experiments with a set of five protein sequence labeling tasks and by using a linear SVM base learner trained by stochastic gradient descent. In this setting, we have shown that our approach systematically outperforms single-task learning on all tasks and on two datasets of medium and large scale. We have also shown that our approach significantly outperforms state-of-the-art (+2.1% improvement) results for secondary structure prediction.

Since our iterative multi-task approach is - as a matter of fact - not restricted to predicting sequence labels, we will proceed by applying it to other protein prediction problems, such as functional predictions, residue-residue contact map predictions, beta-strand alignment predictions, predictions of protein-protein interactions, and tertiary structure predictions. We also believe that the iterative multi-task framework proposed in this paper may be applied in many other complex application domains (text processing, image analysis, network monitoring and control, robotics), where data is available about several related tasks and where synergies could similarly be exploited to enhance machine learning solutions.

References

[1] R Adamczak, A Porollo, and J Meller. Combining prediction of secondary structure and solvent accessibility in proteins. *Proteins*, 2005.

[2] J Lafferty, A McCallum, and F Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning*, 2001.

[3] I Tsochantaridis, T Hofmann, T Joachims, and Y Altun. Support vector machine learning for interdependent and structured output spaces. In *International Conference on Machine Learning*, 2004.

[4] WW Cohen and V R Carvalho. Stacked sequential learning. In *International Joint Conferences on Artificial Intelligence*, 2005.

[5] F Maes, S Peters, L Denoyer, and P Gallinari. Simulated iterative classification: A new learning procedure for graph labeling. In *European Conference on Machine Learning*, 2009.

[6] D Jones. Protein secondary structure prediction based on position-specific scoring matrices. *Journal of molecular biology*, 1999.

[7] AC Camproux, R Gautier, and P Tuffery. A hidden markov model derived structural alphabet for proteins. *Journal of molecular biology*, 2004.

[8] O Noivirt-Brik, J Prilusky, and J L Sussman. Assessment of disorder predictions in casp8. *Proteins*, 2009.

[9] H Zhang, T Zhang, K Chen, S Shen, J Ruan, and L Kurgan. Sequence based residue depth prediction using evolutionary information and predicted secondary structure. *BMC bioinformatics*, 2008.