Study on Regulatory Motifs in *Arabidopsis thaliana* *

[Extended Abstract]

Vishal Bahirwani
Computing and Information Sciences
Kansas State University
Manhattan, KS 66506
vishalb@ksu.edu

Doina Caragea
Computing and Information Sciences
Kansas State University
Manhattan, KS 66506
dcaragea@ksu.edu

ABSTRACT

Identification of regulatory motifs in DNA sequences can be seen as a prerequisite for understanding many biological processes including gene transcription and regulation. Regulatory motifs are usually located in gene promoters, which are in turn located in intergenic regions. Thus, the analysis of the intergenic regions can help identify regulatory motifs. However, simply searching for known motifs can result in many hits, which are not necessarily active with respect to transcription regulation. In this paper, we explore a *motif-based machine learning approach* to identify active intergenic regulatory elements. More precisely, we use machine learning algorithms to learn models that can predict the direction of transcription for pairs of consecutive genes in *Arabidopsis thaliana* using motifs from *AthaMap*[1] and *PLACE*[2]. Under the assumption that predictive motifs correspond to active regulatory elements, we identify active motifs by performing *feature selection* and *feature abstraction*. Experimental results show that indeed feature selection and feature abstraction methods are two important means that contribute to good performance for the prediction problem considered in this work [3].

1. INTRODUCTION

Characterization of regulatory mechanisms by which plants sense and respond to abiotic stresses at the molecular level is crucial to understanding the responses of organisms to environmental changes. Towards this goal, researchers study *genes* and *gene regulatory networks* governing plant responses [2]. Identification of regulatory motifs in genomic sequences can lead to better understanding of gene regulation and provide hypotheses about the links in a regulatory network.

*This research was supported in part by an NSF grant (IIS 0711396) and a seed grant from the Ecological Genomics Institute at Kansas State University.

[1]http://www.athamap.de/

[2]http://www.dna.affrc.go.jp/PLACE/

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
ACM-BCB '11, August 1-3, Chicago, IL, USA
Copyright 2011 ACM 978-1-4503-0796-3/11/08 ...$10.00.

However, simply searching the promoter for known motifs in public databases can result in many hits that correspond to inactive binding sites. To identify active binding sites, we use machine learning algorithms to learn models that can predict the direction of transcription for pairs of consecutive genes. In particular, feature selection and feature abstraction methods are used to find the most predictive motifs, under the assumption that they correspond to active binding sies.

Formally, we focus on the following *three-class* prediction problem: Given a data set $\mathcal{D} = \{((g_{i,1}, g_{i,2}), c_i)\}_{i=1,\cdots,n}$ consisting of pairs of consecutive genes $g_{i,1}$ and $g_{i,2}$ over the alphabet Σ of nucleotides $\{A, C, G, T\}$, along with their class labels c_i, which belong to a finite set C, the task is to learn a model that is able to predict the class label $c \in C$ for a novel pair of consecutive genes (g_1, g_2). The class label associated with each pair of consecutive genes represents the direction of transcription for the pair (Figure 1): forward-reverse (FR) if the direction of transcription of g_1 is forward and of g_2 is reverse, reverse-forward (RF) if the direction of transcription of g_1 is reverse and of g_2 is forward, and forward-forward or reverse-reverse ($FFRR$) if the directions of transcription of g_1 and g_2 are either forward-forward or reverse-reverse.

Given the wealth of genomic information available for the model organism *Arabidopsis thaliana*, we will use this plant for our study of regulatory elements. All prediction experiments in this paper are conducted using Weka[3] implementations of machine learning algorithms. We perform *feature selection*, using the information gain criterion along with the ranker's search method also available in Weka, and *feature abstraction* based on families, to identify the most significant regulatory elements.

2. TYPES OF REGULATORY MOTIFS

The motif search programs used by *AthaMap* and *PLACE* identify different types of putative transcription factor binding sites (TFBS) based on the screening parameters supplied. Depending on whether positional weight-matrices or experimentally verified single sites based on consensus sequences are used for screening gene sequences, the resulting motifs are classified as *matrix-based* or *pattern-based* motifs, respectively. The version of *AthaMap* used in this work consisted of 109 motifs (51 matrix-based and 58 pattern-based), while *PLACE* consisted of 73 pattern-based motifs specific to *Arabidopsis* and 469 plant motifs.

[3]http://www.cs.waikato.ac.nz/ml/weka/

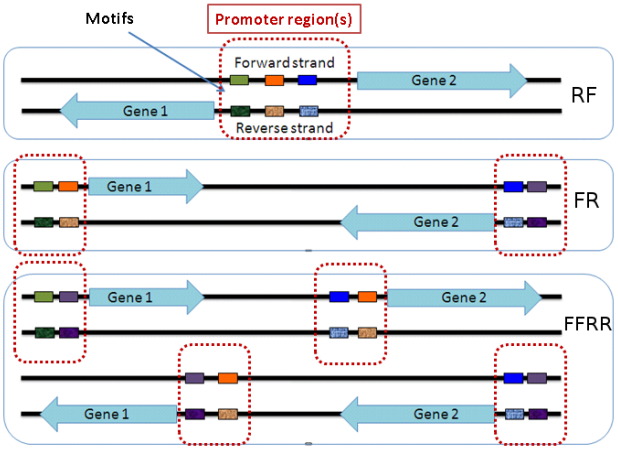

Figure 1: Gene pairs and regulatory elements.

A gene pair can be represented using motif information and general sequence characteristics (e.g., *sequence length* and *GC content*). In this work, we focus on motif information, which refers to the existence of motifs in the region of interest (or equivalently to the existence of the corresponding transcription factors). In addition to simple existence information (0/1 representation), the motif information can be represented using counts or position specific scores. Specifically, we provide learning algorithms with a *feature vector* of the form $\{feature_1, \ldots feature_k, class\_label\}$.

We conduct experiments with motifs from both *AthaMap* and *PLACE*, different types of features (matrix-based or pattern-based) and different types of feature representations (count or score) to find the best way of training classifiers [3].

3. TRANSCRIPTION FACTORS AND TRANSCRIPTION FAMILIES

There is a correspondence between motifs and transcription factors that bind to them. As transcription factors can be grouped in families, we can also group motifs into families. We represent sequences as motifs both at *transcription factor level* (the motifs themselves) and at *transcription family level* (clusters of motifs that belong to the same family).

- *AthaMap*: A simple feature vector will consists of 109 features at the transcription factor level. Matrix-based motifs fall into 21 transcription families, while pattern-based motifs fall into 15 transcription families. When grouped together, we get 24 unique transcription families. Hence, the same gene sequence can be represented using 109 motifs at the transcription factor level, or 24 abstract motifs at the transcription family level.

- *PLACE*: Here, the 73 pattern-based motifs are grouped into 48 abstract features, based upon their respective transcription families and binding sequence similarity.

Factor and family level motifs can be arranged in a hierarchy, as shown in Figure 2. Here, motifs ARF1 and AtLEC2 are grouped in the ABI3/VP1 family. We train classifiers at both transcription factors and transcription family level, to study the effect of the abstraction on the resulting classifiers.

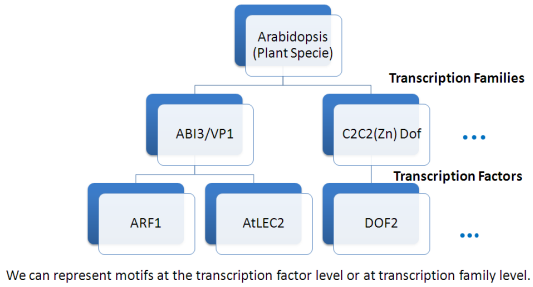

Figure 2: Hierarchical organization *AthaMap* motifs.

4. EXPERIMENTS

We have conducted a series of experiments designed to investigate the performance of several classification algorithms at predicting transcription patterns for pairs of consecutive genes, when presented with different types of feature vectors. In each experiment, we consider the SVMs (with build logistic model option enabled), Random Forests and Logistic Regression (all with default parameters).

1. *AthaMap* factor level motifs: We learn classifiers on 109 *AthaMap* motifs at factor level. Attribute values in the feature vector refer to count representation of respective motifs. This is the simplest feature vector, that is populated with motifs from one of the databases. Initially, we do not perform feature selection and abstraction, so the AUC values for this experiment represent our baseline for motifs derived from *AthaMap*.

2. *AthaMap* family level motifs (feature abstraction): We learn classifiers on the 24 AthaMap motifs at family level (the goal is to capture more general motifs). Attribute values in the feature vector refer to count representation of the respective motif families.

   The aforementioned experiments were also conducted with motifs from *PLACE*.

3. *AthaMap* + *PLACE* factor level motifs: We combine all factor level motifs from *AthaMap* and *PLACE*, to obtain a more comprehensive set of features, and perform feature selection to remove redundant features. AUC values produced are expected to be comparable and possibly better than the baseline values.

5. RESULTS

The AUC values that are highlighted in the following tables show the performance results of classifiers that were *statistically significantly* better than their respective counterparts. Each table presents the AUC values for classifiers when predicting FFRR, FR and RF class labels. In addition, it also presents the overall performance of the classifier calculated as the weighted average of its performances for each class label.

1. *AthaMap* factor level motifs: Results from this experiment (Table 1) show that Simple Logistic classifier has the best performance in predicting FFRR, FR and RF class labels. The overall performance of the classifier is 77%.

Table 1: Cross-validation results with *AthaMap* motifs (factor level) using count representation.

Classifiers Learned	FFRR	FR	RF	Wt. Avg.
Random Forest	0.685	0.9	0.724	0.745
SVM - PolyKernel	0.513	0.76	0.691	0.614
Simple Logistic	**0.703**	**0.906**	**0.776**	**0.769**

2. *AthaMap* family level motifs: Results from this experiment are shown in Table 2. As can be seen from the table, the Random Forest classifier has the best performance, precisely, 85%, 96% and 86% in predicting FFRR, FR and RF class labels, respectively. Besides, the overall performance of the classifier is 88%.

Table 2: Cross-validation results with *AthaMap* motifs (family level) using count representation.

Classifiers Learned	FFRR	FR	RF	Wt. Avg.
Random Forest	**0.849**	**0.961**	**0.864**	**0.879**
SVM - PolyKernel	0.709	0.918	0.784	0.776
Simple Logistic	0.729	0.931	0.787	0.791

By analyzing the AUC values in Tables 1 and 2, we see that the results of the classifiers trained on family level motifs are better than those of the classifiers trained on factor level motifs. Specifically, there is a significant increase in the performance (approximately 10%) because, when we move up in the motif hierarchy, the feature vectors capture more general information ("semantically equivalent" motifs). Experiments were also performed with *PLACE* features (not shown here), endorsing similar conclusions.

3. *AthaMap* + *PLACE* factor level motifs: For this experiment, we grouped 109 *AthaMap* and 469 *PLACE* motifs to get a feature vector of 578 factor level motifs. Its results are shown in Table 3. It is prominent that classifiers perform significantly better (5-10% increase) when provided with features from multiple data sources.

Table 3: Cross-validation results with *AthaMap* and *PLACE* motifs (factor level) using count representation.

Classifiers Learned	FFRR	FR	RF	Wt. Avg.
Random Forest	0.644	0.841	0.692	0.703
SVM - PolyKernel	0.687	0.894	0.752	0.752
Simple Logistic	**0.75**	**0.925**	**0.81**	**0.806**

Figure 3 shows the dependence of the AUC values on the number of features selected. The peaks of the graphs highlight the best performance. As can be seen, less than 100 best features results in best performance. As we increase the number of features, the performance decreases.

Results show that, irrespective of how we deal with motifs found in the region of interest, whether we learn from motifs at factor level or from motifs at family level, counting occurrences (count representation) is a better way of training classifiers as compared to averaging over occurrence scores (score representation) [3].

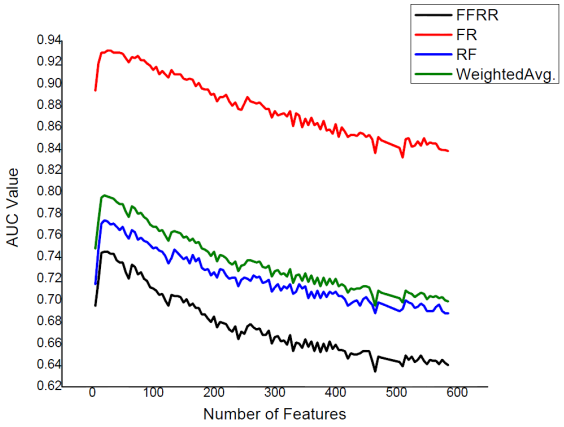

Figure 3: The area under the ROC Curve as a function of number of features selected using *AthaMap* and *PLACE* motifs combined. Using a relatively small number of features (motifs), the classifiers achieve highest performance. As we add more and more features, the performance of classifiers decreases significantly.

6. SUMMARY

Motifs collected from biological databases have different binding sequences, they belong to different transcription families. Some are relevant for transcription prediction problems while others act as noise, as seen from feature selection and abstraction experiments. A careful examination of putative motifs can offer new insights into genomic research. We collected motifs from *AthaMap* and *PLACE*, analyzed them to find out:

- Combined motifs from *AthaMap* and *PLACE* represent better feature vectors as compared to motifs from only one database, *AthaMap* or *PLACE*.
- Classifiers learned from *AthaMap* data, perform better than classifiers learned from *PLACE* data. The former is a more comprehensive database containing regulatory elements found in the region of interest and results in classifier performance which is close to 88%.
- Techniques such as feature selection and abstraction produce better classifiers as compared with those obtained from the original transcription factor motifs.

7. REFERENCES

[1] M. Sarachu and M. Colet. wEMBOSS: a Web interface for EMBOSS. *Bioinformatics*, 21(4):540–541, 2005.

[2] W. Zhang, J. Ruan, T. H. D. Ho, Y. You, T. Yu, and R. S. Quatrano. Cis-regulatory element based targeted gene finding: genome-wide identification of abscisic acid- and abiotic stress-responsive genes in arabidopsis thaliana. *Bioinformatics*, 21(14):3074–3081, 2005.

[3] V. Bahirwani. Exploring Transcription Patterns and Regulatory Motifs in Arabidopsis thaliana. *MS Thesis*, Computing and Information Sciences Department, Kansas State University, USA, 2010.