

MISAE: A New Approach for Regulatory Motif Extraction *

Zhaohui Sun, Jingyi Yang, Jitender S. Deogun
Department of Computer Science and Engineering
University of Nebraska – Lincoln
Lincoln, NE 68588-0115, USA
zsun{jyang, deogun}@cse.unl.edu

Abstract

The recognition of regulatory motifs of co-regulated genes is essential for understanding the regulatory mechanisms. However, the automatic extraction of regulatory motifs from a given data set of the upstream non-coding DNA sequences of a family of co-regulated genes is difficult because regulatory motifs are often subtle and inexact. This problem is further complicated by the corruption of the data sets. In this paper, a new approach called Mismatch-allowed Probabilistic Suffix Tree Motif Extraction (MISAE) is proposed. It combines the mismatch-allowed probabilistic suffix tree that is a probabilistic model and local prediction for the extraction of regulatory motifs. The proposed approach is tested on 15 co-regulated gene families and compares favorably with other state-of-the-art approaches. Moreover, MISAE performs well on “corrupted” data sets. It is able to extract the motif from a “corrupted” data set with less than one fourth of the sequences containing the real motif.

1. Introduction

With the accumulation of gene expression data and genomic sequence data, it is possible to discover transcriptional regulatory mechanisms using computational approaches. Regulatory motifs are conserved short subsequences in the upstream non-coding DNA sequences, shared by a family of co-regulated genes, which are bound by the transcription factors to activate or repress gene expressions. Therefore, regulatory motifs are also called transcription factor binding sites. Extracting regulatory motifs

is an important problem in biology because the recognition of regulatory motifs is essential to understand the regulatory mechanisms of gene expression. However, automatically extracting regulatory motifs from a set of sample upstream sequences with computational approaches is by no means a simple, straightforward problem. Regulatory motifs are often very subtle, signals of which are unremarkable compared to the signals of some random motifs presented in the samples [7]. Moreover, a regulatory motif often appears in different sequences with mismatches, insertions, and deletions. Another issue making this problem more complicated is the corruption of the samples. In a corrupted sample data set, only a part of the sequences contain the regulatory motifs, which makes the regulatory motifs harder to be discovered.

Many approaches have been proposed for regulatory motif extraction. Generally, these approaches can be divided into two categories: the approaches based on word-counting [6, 13] and the approaches based on probabilistic models [5, 9, 11, 15, 3, 8, 1]. The approaches based on word-counting calculate the frequency of oligonucleotides to detect over-represented motifs. The approaches in this category use a variety of strategies to speed up counting. For example, Vanet et al. [14] use suffix trees to count and store the occurrences of oligonucleotides and then detect motifs. In the approaches based on probabilistic models, the motifs are often represented by position probability matrices while the remainder of sequences are represented by background models. Maximum likelihood estimation in the forms of Expectation Maximization (EM) and Gibbs sampling is applied in these approaches to estimate the position probability matrices representing motifs. While the approaches based on word-counting find global solutions for the motif extraction problem, the approaches based on probabilistic models lead to local solutions. Dyad [4], Gibbs Motif Sampler [12] and MEME [1] are among the best available approaches for regulatory motif extraction so far. The former is based on word-counting. The latter two are based on probability models.

* This research was supported in part by NSF EPSCOR Grant No. EPS-0091900 and NSF Digital Government Grant No. EIA-0091530, a cooperative agreement with USDA FCIC/RMA (2IE08310228), and a NSF Grant No. 0115626 To D.P.Weeks.

In this paper, we propose a new probabilistic model based approach called Mismatch-allowed Probabilistic Suffix Tree Motif Extraction (MISAE) for regulatory motif extraction, which combines the mismatch-allowed probability suffix tree and local prediction. We test MISAE on the sample data sets of the upstream DNA sequences of 15 yeast co-regulated gene families and different sets of "corrupted" samples, and compare the results with those of other state-of-the-art approaches dyad, Gibbs Motif Sampler and MEME. MISAE compares favorably with these approaches in our experiments. It successfully finds motifs in 14 data sets. Moreover, MISAE performs very well on "corrupted" sample sets. It can discover the regulatory motif in a set that consists of only 8 sequences containing the motif and 25 irrelevant sequences while all the other approaches fail to produce any meaningful results on the same set.

The rest of this paper is organized as follows: Section 2 describes the method used in MISAE. Section 3 presents the experimental results and compares the results of our approach MISAE with those of other approaches. In Section 4, we discuss the related issues of MISAE. The last section concludes this paper.

2. Method

The approach MISAE uses a probabilistic model, the mismatch-allowed probabilistic suffix tree that is enhanced from the probabilistic suffix tree, and local prediction based on the probabilistic model to extract patterns with the highest likelihood scores from input data sets. Those patterns are possible regulatory motifs. The input data of MISAE are sets of upstream non-coding DNA sequences. MISAE finds the subsequences with the highest similarity scores from the sequences and output the subsequences along with the associated scores and position probability matrices.

To illustrate MISAE, we first introduce the concepts of the probabilistic suffix tree (PST), mismatch-allowed probabilistic suffix tree (M-PST), and local prediction; then describe the approach followed in MISAE. Some definitions are given at first:

Definition 1. Given a sequence S of length m , i.e. $S = s_1 \dots s_m$, $S_{i..j}$, $1 \leq i \leq j \leq m$ denotes the subsequence $s_i \dots s_j$ starting at position i and ending at position j .

Definition 2. $P(s_{i+1}|s_1 \dots s_i)$ denotes the conditional probability distribution of observing symbol s_{i+1} given that the preceding segments are: $s_1 \dots s_i$.

2.1. Probabilistic suffix trees

The M-PST used in MISAE is an enhanced form of the PST that is equivalent to the variable order Markov chain with smaller time and space complexity [2]. Therefore, we first introduce the PST.

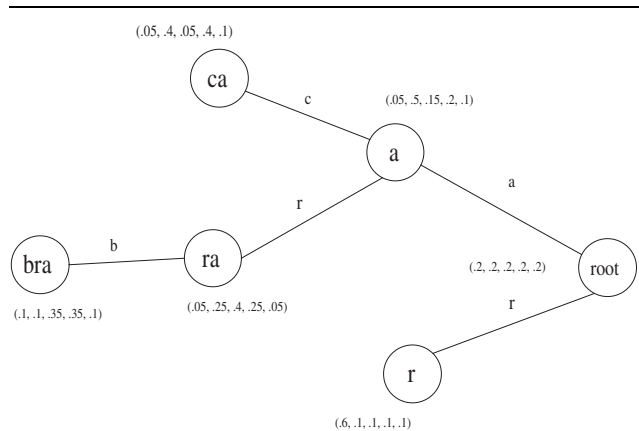


Figure 1. An example of the PST. The vector of each node keeps the probability distribution of next symbols. For instance, the probability distribution of next symbols given the preceding segment as ca is 0.05, 0.4, 0.05, 0.4 and 0.1 for a , b , c , d and r , respectively.

The probabilistic suffix tree, also called prediction suffix tree, is a stochastic model using the suffix tree as the index structure. It is a compact representation of the conditional probability distribution for a set of sequences. The PST is based on the "short memory" feature of biological sequences. That is, in a biological sequence, the empirical probability distribution of the next symbol given the preceding segment can be accurately approximated by observing no more than the last L symbols in that segment, e.g. in a sequence S , $P(s_{i+1}|s_1 \dots s_i) = P(s_{i+1}|s_{i-L+1} \dots s_i)$, $i > L$.

A PST is a rooted tree. A node x of a PST is associated with a probability vector that stores the probability distributions of possible next symbols $\gamma_{label(x)}(\theta)$ given the label of the node $label(x)$ as the preceding segment and θ as the next symbol. A PST has a specified depth bound L . The PST can be built from a set of unaligned sequences in linear time [2]. An example of PST with depth 3 over the alphabet $\Sigma = \{a, b, c, d, r\}$ is shown in Figure 1 (adopted from [2]).

After building a PST from a set of sequences, the probability of a symbol in the sequence can be predicted by searching the PST to find the longest suffix of the preceding subsequence of that symbol. The probability of the entire sequence can be predicted by multiplying the probabilities of each symbol in the sequence. Such prediction is called global prediction. For example, the probability of a sequence *abracadabra* based on the PST in Figure 1 can

be computed as follows:

$$\begin{aligned}
& P(\underline{a}bracadabra) \\
= & P(a)P(b|\underline{a})P(r|ab)P(a|\underline{abr})P(c|\underline{abra}) \cdots \\
& P(a|\underline{abracadabr}) \\
= & \gamma_{root}(a)\gamma_a(b)\gamma_{root}(r)\gamma_r(a)\gamma_{bra}(c) \cdots \gamma_r(a)
\end{aligned}$$

where the underlined subsequences represent the longest suffices that appear in the PST in Figure 1.

To predict the similarity of a sequence to the sequences modelled by a PST, we need to compare the probability of the sequence generated by the PST with the background probability that is the probability of the sequence generated by random distributions. For a sequence $S = s_1 \cdots s_m$, the similarity scores are defined as follows:

$$\begin{aligned}
sim(s_i) &= \frac{P_{PST}(s_i|s_1 \cdots s_{i-1})}{P_{random}(s_i)} \\
sim(S) &= \frac{P_{PST}(S)}{P_{random}(S)} \\
&= \frac{\prod_{1 \leq i \leq m} P_{PST}(s_i|s_1 \cdots s_{i-1})}{\prod_{1 \leq i \leq m} P_{random}(s_i)} \\
&= \prod_{1 \leq i \leq m} sim(s_i|s_1 \cdots s_{i-1})
\end{aligned}$$

where $P_{PST}(\theta)$ denotes the probability derived from a PST while $P_{random}(\theta)$ represents the background probability.

2.2. Mismatch-allowed probabilistic suffix trees

The regulatory motif occurrences in different sequences are often inexact, and some regulatory motifs contain gaps. In a motif containing gaps and mismatches, the probability of observing a symbol at a position may not depend on the immediately preceding symbols. Instead, it may depend on symbols that are several positions away from this position. For example, for a regulatory motif like $AAGCn(2)TCGG$, where $n(2)$ denotes 2 positions that can be any symbol, the probability of observing T in this motif depends on the preceding segments two positions left of it. Based on the above facts, we propose the M-PST, which keeps records of such probabilities, as the probabilistic model used in MISAE.

In the M-PST, the probabilities stored in a node x are $\gamma_{label(x)n(k)}(\theta)$, $0 \leq k \leq Mismatch_{max}$ that are the probabilities of observing θ given $label(x)n(k)$ as the preceding segments, where the preceding segments consist of the label of the node x and a short sequence of length k that can be any symbol. $Mismatch_{max}$ is the predefined maximal number of allowed continuous mismatches. Each node in the M-PST stores $Mismatch_{max} + 1$ vectors of probabilities. When $Mismatch_{max}$ is set to 0, mismatch is not allowed in the construction of the M-PST. In this case, the

M-PST is equivalent to the original PST. The advantage of the M-PST is that it is more sensitive to extract the motifs containing mismatches and gaps.

The algorithm to construct the M-PST represents a modified version of the algorithm to construct PST. It still has linear time complexity. During the construction of an M-PST from a set of sequences, for each node x labelled with $label(x)$ we need to count the occurrences of symbols in each position that is i , $0 \leq i \leq Mismatch_{max}$ positions away to right of the the string $label(x)$ occurring in each sequence, and calculate $Mismatch_{max} + 1$ probability vectors accordingly. Moreover, in the construction a new parameter OCC_{MIN} needs to be specified. For a node x in an M-PST, the string $label(\theta)$ must occur in all sequences not less than OCC_{MIN} times. Otherwise, the node will be pruned during the construction. By specifying this parameter we can avoid searching the subsequences that occur only a small number (relative to the total number of sequences in the set) of times. For motif extraction, only the subsequences occurring frequently in the sample set are considered. Therefore eliminating the nodes that represent the subsequences occurring very few times (e.g. less than OCC_{min} times) would not affect the results of motif extraction.

2.3. Local prediction with M-PST

The motifs in a set of sequences are over-represented patterns, therefore they are likely subsequences with the highest similarity scores that are calculated based on the probabilistic model built from the set of sequences. In a functional region, such as a domain or a motif, the probability of observing a symbol at a certain position in this region depends only on the preceding segments within the region. For example, in a sequence abracadabra, where a motif starts from the fourth position (marked by underline), the probability of observing c in the fifth position is more likely to be $P(c|a)$ rather than $P(c|abra)$ where bra , part of $abra$, is outside the functional region. So in MISAE we use local prediction to predict the similarity scores of subsequences. In local prediction, when we calculate the similarity score of a subsequence, only the symbols within that subsequence are considered. To find the subsequences of a sequence $S = s_1 \cdots s_m$ with highest similarity scores, we consider all the subsequences $S_{i \dots j}$, $1 \leq i \leq j \leq m$ and use local prediction to calculate the similarity scores.

If we do not consider the mismatches, the scores can be

calculated as follows:

$$\begin{aligned}
& \text{sim}(S_{i\dots j}) \\
&= \frac{P_{PST}(s_i \cdots s_{j-1}s_j)}{P_{\text{random}}(s_i \cdots s_{j-1}s_j)} \\
&= \frac{P_{PST}(s_i)P_{PST}(s_{i+1}|s_i) \cdots P_{PST}(s_j|s_i \cdots s_{j-1})}{P_{\text{random}}(s_i)P_{\text{random}}(s_{i+1}) \cdots P_{\text{random}}(s_j)} \\
&= \text{sim}(S_{i\dots j-1}) \times \frac{P_{PST}(s_j|s_i \cdots s_{j-1})}{P_{\text{random}}(s_j)}
\end{aligned}$$

Because we use logarithm of the scores in our implementation, a parameter d ($d > 1$) is introduced to prevent the similarity scores of the subsequences generated by random probability distribution from being much larger than 1. Therefore the actual function to compute the similarity scores is as follows:

$$\text{sim}(S_{i\dots j}) = \text{sim}(S_{i\dots j-1}) \times \frac{P_{PST}(s_j|s_i \cdots s_{j-1})}{P_{\text{random}}(s_j) \times d}$$

To allow mismatches, we can perform local prediction based on the M-PST. The function to compute the similarity scores is changed as follows:

$$\begin{aligned}
& \text{sim}(S_{i\dots j}) \\
&= \max_{1 \leq k \leq \text{Mismatch}_{\text{max}}} \left[\frac{P_{PST}(s_j|s_i \cdots s_{j-k}n(k))}{P_{\text{random}}(s_j) \times d} \right. \\
& \quad \left. \max[\text{sim}(S_{i\dots j-k}) \times \text{penalty}^{-k}, \text{sim}(S_{i\dots j-1})] \right]
\end{aligned}$$

where L is the depth of the M-PST, and penalty is the mismatch penalty. For each mismatch n allowed in the prediction, the score of the subsequence is multiplied by $\frac{1}{\text{penalty}}$ ($\text{penalty} > 1$).

In MISAE, we compute the similarity score $\text{sim}(S_{i\dots j})$, $1 \leq i \leq j \leq m$ for each subsequence of one sequence using the above equation. Although the number of the subsequences of one sequence is exponential to the length of the sequence, the local prediction to compute similarity scores for all the subsequences can still be finished in linear time. At a position i , we retrieve all the probability vectors of the nodes on the path in the M-PST when searching the longest suffix of $s_1 \cdots s_{i-1}$. Thus one traversal of the M-PST at position i provides the necessary probabilities to compute the similarity scores for all the subsequences including the position i . Therefore, to compute the similarity score of all the subsequences of a sequence of length m , we need to scan the sequence only once. And at each position of the sequence we traverse the M-PST only once to retrieve all the necessary probabilities. So the time complexity of the local prediction to compute similarity scores for all the subsequences of a sequence based on the M-PST is $O(Lm)$ where L is the depth of the M-PST and m is the length of the sequence.

2.4. Motif extraction with the M-PST and local prediction

MISAE finds the subsequence with the highest similarity score in each sequence compared with the remaining sequences. For each sequence in the given data set, an M-PST is constructed from the remaining sequences. Then the local prediction is performed to find the most conserved pattern in the sequence which is the subsequence with the highest similarity score based on the M-PST constructed from the remaining sequences. This procedure is repeated for all the sequences. The patterns with the highest scores among all the patterns found are selected and outputted. The length of the patterns can be minimized by adjusting the parameters (penalty , OCC_{MIN} , and/or d).

2.5. Time complexity of MISAE

Let l be the average length of the sample sequences, t be the size of the sample set, Σ be the alphabet, L be the depth of the M-PST, and M be the maximum number of continuous mismatches $\text{Mismatch}_{\text{max}}$. The construction of an M-PST takes $O(|\Sigma|ltM)$ time. To find the subsequence with the highest score from one sequence of length m with local prediction based on the M-PST takes $O(LmM)$ time. Therefore, the time complexity of the overall procedure of extracting motifs from a set of sequences is $O(|\Sigma|t^2lM + LtlM)$.

3. Experimental Evaluation

We implement the proposed algorithm in C++ and test it on 15 sets of upstream DNA sequences of 15 yeast co-regulated gene families with known regulatory motifs to evaluate the performance of our approach for regulatory motif extraction. We also test MISAE on some artificially ‘‘corrupted’’ data sets to evaluate its robustness. Besides, we compare the results of MISAE with those of other approaches including dyad, Gibbs Motif Sampler and MEME.

3.1. Experiments on 15 yeast co-regulated gene families

The 15 data sets used in the experiments are collected from literature [4, 10]. Each of the data sets consists of several upstream DNA sequences of a family of co-regulated genes in yeast. The names of the families are listed in Table 1.

The upstream sequences in yeast often have high AT content. Without pre-processing the data sets, the outputs would include the tandem repeat patterns such as *ATATATA*, *AAAAA*, and *TTTTT* that more likely result from the

Table 1: **List of 15 yeast co-regulated gene families. The number in the parentheses following each family name indicates the number of sequences in each data sets. The third column shows the regions those upstream sequences are selected from.**

Annotation	Families	Regions of upstream sequences
Regulated by Zn cluster factors	GAL4(6),LEU(5), LYS(6), PDR(7), PPR1(3), UGA3(3) UME6(25)	-1 to -800
Regulated by non-Zn cluster factors	NIT(7), MET(11), PHO(5), GCN4(38), INO(7), YAP(16)	-1 to -800
Cytoplasmic ribosomal proteins	CRIBOSOME(122)	-1 to -1000
Nucleosome complex proteins	NUCLEO(8)	-1 to -1000

high AT content rather than true indicators of the family-specific regulations. Therefore in the experiments, before beginning the tests, we filter out these tandem repeat patterns from the data sets.

The test results of MISAE on 15 data sets are shown in Table 2. Our approach MISAE successfully extracts motifs from 14 data sets which cover the known motifs. For 12 among the 15 data sets, the known motifs are identified within the top 2 significant patterns. For 2 data sets, the known motifs are covered by the less significant patterns (ranked 3 to 5). MISAE is not able to discover the motif for the PPR1 family which contains only 3 genes. This illustrates a limit in the sensitivity of our approach: the sample data set must have sufficient sequences for the shared motif to be discovered as a significant pattern. In general, our approach is sensitive to the data sets containing a relatively large number of sequences (i.e. larger than 5). Note that for some of the families, such as UMET6, the reverse complement of the known motif is also identified by MISAE in addition to the motif itself. The results also show that our approach is able to find inexact and mismatch-allowed motifs. The LYS family has an inexact regulatory motif containing one mismatch. MISAE identifies the motif for the family as the top ranked pattern. Likewise, our approach also finds the inexact motifs for the GAL4 and LEU3 families. Our experiments also show that MISAE is very efficient. For a data set with 8 sequences of average length 1000, MISAE takes only 7 seconds to extract the motifs on a SunBlade 1000 workstation.

The results of other approaches, dyad, Gibbs Motif Sampler and MEME, on the same data sets are shown in Table 3. MISAE performs better than the probabilistic model based approaches MEME and Gibbs Motif Sampler and as well as dyad. Gibbs Motif Sampler only finds motifs for 7 families among 15. MEME discovers motifs for 9 families. For both Gibbs Motif Sampler and MEME, the range of lengths of the motifs need to be specified before the experiment. It seems that Gibbs Motif Sampler is affected greatly by the high AT content in the yeast non-coding regions and tends to find patterns with a large number of *As* and *Ts*. Dyad

finds the matching patterns for all the data sets when the frequencies measured in the whole non-coding regions of the yeast genome is used as the expected dyad frequencies in the experiments. Compared to dyad, MISAE only uses the background probabilities derived from the input data set. If the expected frequencies are directly derived from the input sequences, dyad fails to find motifs for several families, such as HAP1, UGA3 and NUCLEO. Besides, for some of the families regulated by non-Zn cluster factors, e.g. YAP and INO, the patterns discovered by dyad are not as good as MISAE.

3.2. Experiments on “corrupted” data sets

In above experiments, all the DNA sequences in a data set contain the same motifs. However, in some situations, the data sets for motif extraction may include some sequences not containing the motifs. Such data sets are called “corrupted” data sets. For example, sometimes the motif extraction are performed on the co-expressed genes extracted from microarray data. Those co-expressed genes may not be co-regulated, and the data sets could include the sequences that do not contain the regulatory motifs. To evaluate the ability of MISAE to handle the “corrupted” data sets, we test it on several artificially “corrupted” data sets. The “corrupted” data sets are made by adding irrelevant sequences into the data set of UME6 family. The ratios of the number of the irrelevant sequences to the number of the sequences containing motifs vary from 8 : 25 to 25 : 6. The results are shown in Table 4. The results of the other approaches, dyad and MEME, on the same “corrupted” data sets are shown in Table 5. Gibbs Motif Sampler is not tested in such experiments because it fails to find motifs for UME6 family.

Although the other two approaches also show some good results in the experiments on the “corrupted” data, MISAE performs better than the other two approaches.

MISAE shows very good robustness in terms of handling “corrupted” data sets. It successfully discovers the correct motif as the top ranked pattern in the first 3 tests which contains 25 sequences containing the motif and up to 25 irrel-

Table 2: Summary of the results of the regulatory motif extraction using MISAE on 15 data sets. The second and third columns of the table show the known motifs and the patterns discovered by MISAE that match the known motifs. If a pattern is followed by (reverse complement), that means the reverse complement of that pattern matches the known motifs. The fourth and fifth columns show the scores of the matched patterns and the ranks of the matched patterns within all the significant patterns discovered by MISAE. For all the above experiments, the depth of the M-PST and the penalty of the mismatches are set as follows: $L = 10$, $penalty = 1.5$. The optimized values of the three parameters used in the tests for each data set are shown in the sixth column. OCC_{MIN} and $Mismatch_{max}$ are denoted by Om and Mis , respectively.

Name	Known motifs	Matching patterns found by MISAE	Score	Rank	Parameters d, Om, Mis
LEU3	RCCG _n CCGGY	TGCGCCGGAACCGCCC	0.748	2	2.3, 4, 9
PPR1	WYCG _n WWYKCCGAW	none			
YAP1	TTACTAA	repeat of TTAGTAA (reverse complement) TTACTAA	0.4027	1	1.9, 10, 0
UGA3	AAARCCGCSGGCGGSAWT	CGCGGGCGGGATTCC	0.3579	2	2.33, 2, 0
GAL4	CGGR _n RCY _n Yn _n Cn _n CCG	CACCGGCGGTCTTCGTCCGTGCG	0.3756	1	2.7, 4, 9
LYS	WWWTCCR _n YGGAWWW	CAAATCCG _n CGGAAT	0.4377	1	2.3, 4, 10
MET	AAAAGTGTGG	AAAGTGTGGCGT	0.9130	1	1.9, 4, 0
PHO	GCACGTTTT	ATTAGCACGTTTTCGCATA	1.065	1	2.2, 4, 4
PDR	TYTCCGCGGARY TCCGTGGA	TTTCCGCGGA TCCGTGGA	1.434 1.445	2 1	1.95, 4, 0
UME6	TAGCCGCCGA TCGGCGGCTA	TAGCCGCCGAAG GGCGGCTAA	0.5312 0.4601	1 3	1.8, 10, 4
GCN4	RRTGACTCTTT	GTGACTCACTT	0.6816	5	1.5, 20, 4
INO	CATGTGAAWT	TTCACATGGA (reverse complement) TCCATGTGAA	0.6713	5	1.9, 4, 0
NIT	GATAAG	TAAGATAAGAAAGATAAGATAAGA	1.07	1	2.32, 5, 4
CRIBSOME	TTTACATCCATACATTTT	TACATCCGTACAT	0.66	2	1.9, 12, 4
NUCLEO	TTACCACCG	CTTTACCACCGTTACCACC	1.35	1	2.6, 6., 4

evant sequences. For the data set consisting of 25 irrelevant sequences and only 18 motif-containing sequences (UME6-18-25), MISAE still finds the motif correctly. For the data set consisting of only 12 or 8 motif-containing sequences and 25 irrelevant sequences (UME6-12-25 and UME6-8-25), MISAE finds the reverse complement of the true motif as the top ranked pattern. MISAE fails to find meaning-

ful patterns when the number of the motif-containing sequences is reduced to 6 against 25 irrelevant sequences in the data set. MISAE is able to find the motif from the data set with less than $\frac{1}{4}$ of the sequences containing that motif, which illustrates its strong ability to handle "corrupted" data sets.

Dyad fails to find matching patterns when the number of

Table 3: Summary of the results of dyad, Gibbs Motif Sampler and MEME on 15 data sets. The second, third and fourth columns of the table show the results of dyad, Gibbs Motif Sampler and MEME, respectively. The number before a pattern denotes the rank of the pattern in all the outputs of the approach for that data set. If a pattern is followed by (reverse complement), that means the reverse complement of the pattern matches the known motifs. dyad is tested with the option "non-coding dyad frequency calibration". Because the outputs of Gibbs Motif Sampler are not consensus patterns but probability matrices and local alignments, we show if the known motifs are covered by the outputs of Gibbs Sampler. Because the web server of MEME refuses to accept data sets containing more than 60,000 symbols, it is not known if MEME can find correct motifs for CRIBSOME family.

Name	Known motifs	Discovered matching patterns by dyad	If discovered by Gibbs Motif Sampler	Discovered matching patterns by MEME
LEU3	RCCGGnnCCGGY	1. ACCGGCGCCGGT	No	CCGGGACCGGC
PPR1	WYCGGnnWWYKCCGAW	2. CGGnnnnnnCCG	Yes	TCGGCATTCTCCGA
YAP1	TTACTAA	2. GCTnnnTAA	Yes	None
UGA3	AAARCCGCSGGCGGSAWT	3. GCCGCCGnCGGC	Yes	1. CAAAAACCGCGGGCGGGA
GAL4	CGGRnnRCYnYnChCCG	1. TCGGAnnnnnnnnnTCCGA	Yes	1. CGGAGGACTTCCCCCG
LYS	WWWTCRRnYGGAWWW	1. AAATTCCG	Yes	1. TTTTCCAGCGGAATTCGC
MET	AAAACGTGG	2. AAATCTGGC	No	1. GAAAACGTGG
PHO	GCACGTTTT	1. GTCGACGTGCAG	No	None
PDR	TYTCCGCGGARY TCCGTGGA	1. TTCCGCGGAA	No	None
UME6	TAGCCGCCGA TCGGCGGCTA	1. TAGCCCGCCA	No	1. CTGGGCGGCTAAAT (reverse complement) 2. TAGCCGCCGAAG
GCN4	RRTGACTCTTT	1. TGAGTCAT (reverse complement)	No	None
INO	CATGTGAAWT	1. CACATGTG	Yes	1. TTTACATGCCCC (reverse complement)
NIT	GATAAG	1. TCTTATC (reverse complement)	Yes	None
CRIBSOME	TTTACATCCATACATTTT	2. GATGTACGGATGT (reverse complement)	No	N/A
NUCLEO	TTACCACCG	1. ACCACC	No	None

sequences containing motifs is reduced to 12 against 25 irrelevant sequences. MEME fails to find matching patterns when the number of sequences containing motifs is reduced to 12 while MISAE can still find the motif at the same time.

4. Discussion

As a probabilistic model based approach, MISAE finds the most probable motifs from the data sets without knowing the length of the motifs. The range of lengths of the mo-

tifs is not needed to be specified before the experiments. However, the parameters of MISAE need to be optimized in the experiments to make MISAE performs best and minimize the length of the output patterns. $Mismatch_{max}$ is the parameter to be specified to construct the M-PST. Based on the value of $Mismatch_{max}$, MISAE works in two modes: non-mismatch allowed mode and mismatch allowed mode. When $Mismatch_{max}$ is set to 0, MISAE works in non-mismatch mode. In this mode, MISAE performs better on the data sets that have short, exact motifs, e.g. MET, YAP1. When $Mismatch_{max}$ is set to larger than 0, MISAE works

Table 4: Summary of the results of MISAE on “corrupted” data sets ($Mismatch_{max} = 4$).

Name of data sets	Number of sequences containing the motif	Number of irrelevant sequences	Rank of the pattern matching the motif	Parameters		
				d	OCC_{MIN}	penalty
UME6-25-8	25	8	1	1.8	12	2.0
UME6-25-16	25	16	1	1.8	12	2.0
UME6-25-25	25	25	1	1.8	12	2.0
UME6-18-25	18	25	1	1.8	12	2.0
UME6-12-25	12	25	1	1.8	8	2.0
UME6-8-25	8	25	(reverse complement) 1	1.8	7	2.0
UME6-6-25	6	25	(reverse complement) not found	1.8	5	2.0

Table 5: Summary of the result of dyad and MEME on “corrupted” data sets ($Mismatch_{max} = 4$).

Name of data sets	Discovered matching patterns by dyad	Discovered matching patterns by MEME
UME6-25-16	TAGCCGCCGA	1. GTCGGCGGCTA reverse complement
UME6-25-25	TAGCCGCCGA	TAGCCGCCGAAG
UME6-18-25	GCCGCCG	1. TTCGGCGGCTAAAT 2. AGCCGCCGGCG
UME6-12-25	None	3. GTG(/C)GG(/A)CGGCT(/A)A
UME6-8-25	None	None
UME6-6-25	None	None

in non-mismatch mode. It performs better on the data sets that have long and/or inexact motifs in this mode. In this mode, $Mismatch_{max}$ is best set to 4 or 10 based on the experiments. The parameter d is usually chosen from the range (1,3] and OCC_{Min} needs to be larger than 0. When larger values of d and OCC_{Min} are chosen, the lengths of the found patterns become shorter until no pattern is found. Therefore, the experiments should begin with small values of d and OCC_{Min} until the patterns with minimal lengths are found. A possible future improvement of our approach is the automatic optimization of the parameters for the experiments.

When the preknowledge of the the frequencies measured in the whole non-coding regions of the genome is incorporated, dyad generates best results in the experiments on 15 yeast gene families. The other approaches do not use such preknowledge in the experiments. It is possible the perfor-

mance of MISAE can be also improved if we incorporate the preknowledge of the data sets. It is another possible future direction for our approach.

5. Conclusion

In this paper, we develop a novel approach for regulatory motifs extraction from upstream non-coding sequences of co-regulated genes based on the local prediction and the M-PST. The M-PST represents an enhanced form of the PST that includes additional distribution probabilities. The experimental results on 15 yeast co-regulated gene families demonstrate the ability of MISAE to identify regulatory motifs for co-regulated gene families. The experiments on “corrupted” data sets further demonstrate the robustness of MISAE. The ability of our approach to handle the “corrupted” samples is a useful feature for regu-

latory motif extraction. Based on the above facts, our approach provides an important alternative for the regulatory motif extraction compared to the other approaches, such as the word-counting based dyad, and the probabilistic model based MEME, and Gibbs Motif Sampler. In the future, we will revise the implementation of MISAE to improve the usability and performance.

6. Acknowledgements

The authors wish to thank Dr. Donald Weeks for his kind support. We also thank Dr. Stephen Scott for his helpful discussion.

References

- [1] T. Bailey and C. Elkan. Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Machine Learning*, 21:51–80, 1995.
- [2] G. Bejerano and G. Yona. Variations on probabilistic suffix trees: statistical modelling and prediction of protein families. *Bioinformatics*, 17(1):23–43, 2001.
- [3] M. Gupta and J. Liu. Discovery of conserved sequence patterns using a stochastic dictionary model. *J. Am. Stat. Assoc.*, 98(461):55–66, 2003.
- [4] J. Helden, A. Rios, and J. Collado-Vides. Discovering regulatory elements in non-coding sequences by analysis of spaced dyads. *Nucleic Acids Research*, 28(8):1808–1818, 2000.
- [5] J. Hughes, P. Estep, s. Tavazoie, and G. Church. Computational identification of cis-regulatory elements associated with groups of functionally related genes in *sacharomyces cerevisiae*. *J. Mol. Biol.*, 296:1205–1214, 2000.
- [6] L. Jensen and S. Knudsen. Automatic discovery of regulatory patterns in promoter regions based on whole cell expression data and functional annotation. *Bioinformatics*, 16(4):326–333, 2000.
- [7] U. Keich and P. Pevzner. Subtle motifs: defining the limits of motif finding algorithms. *Bioinformatics*, 18(10):1382–1390, 2002.
- [8] C. Lawrence, S. Altschul, M. Boguski, J. Liu, A. Neuwald, and J. Wootton. Detecting subtle sequence signals: A gibbs sampling strategy for multiple alignment. *Science*, 262:208–214, 1993.
- [9] J. Liu, A. Neuwald, and C. Lawrence. Bayesian models for multiple local sequence alignment and gibbs sampling strategies. *J. Am. Stat. Assoc.*, 90(462):1156–1170, 1995.
- [10] P. Pavlidis, T. Furey, M. Liberto, D. Haussler, and W. Grundy. Promoter region-based classification of genes. In *Proc. Pacific Symposium on Biocomputing*, pages 151–163, 2001.
- [11] R. Roth, J. Hughes, P. Esterp, and G. Church. Finding dna regulatory motifs within unaligned noncoding sequence clustered by whole genome mrna quantitation. *Nature Biotechnology*, 16:939–945, 1998.
- [12] W. Thompson, R. E.C., and C. Lawrence. Gibbs recursive sampler: finding transcription factor binding sites. *Nucleic Acids Research*, 31(13):3580–3585, 2003.
- [13] M. Tompa and S. Sinha. A statistical method for finding transcription factor binding sites. In *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, pages 37–45, 2000.
- [14] A. Vanet, L. Marsan, A. Labigne, and M. Sagot. Inferring regulatory elements from a whole genome. an analysis of helicobacter pylori σ_{80} family of promoter signals. *J. Mol. Biol.*, 297(2):335–353, 2000.
- [15] C. Workman and G. Stormo. Ann-spec: A method for discovering transcription binding sites with improved specificity. In *Proc. Pacific Symposium on Biocomputing*, pages 464–475, 2000.