MEMOFinder: combining *de novo* motif prediction methods with a database of known motifs

Bartek Wilczyński[*], Miłosz Darżynkiewicz[†] and Jerzy Tiuryn[‡]
Institute of Informatics, University of Warsaw

September 10, 2008

Abstract

Background: Methods for finding overrepresented sequence motifs are useful in several key areas of computational biology. They aim at detecting very weak signals responsible for biological processes requiring robust sequence identification like transcription factor binding to DNA or docking sites in proteins. Currently, general performance of the model based motif finding methods is unsatisfactory, however different methods are succesful in different cases. This leads to the practical problem of combining results of different motif finding tools, taking into account current knowledge collected in motif databases.

Results: We propose a new complete service allowing researchers to submit their sequences for analysis by four different motif finding methods for clustering and comparison with a reference motif database. It is tailored for regulatory motif detection, however it allows for substantial amount of configuration regarding sequence background, motif database and parameters for motif finding methods.

Availability: The method is available online as a webserver at: http://bioputer.mimuw.edu.pl/software/mmf. In addition, the source code is released on a GNU General Public License.

Introduction

One of the key ingredients of regulation of gene expression is the ability of some proteins, known as transcription factors (TFs), to sequence-specifically bind to short contiguous pieces of DNA, called transcription factor binding sites (TFBSs). These sites are usually located upstream of the regulated genes. Finding TFBSs *de novo* is one of the principal challenges of the research area of genomic sequence analysis. This area aims at deciphering the regulatory machinery at the level of a cell, and becomes one of the central topics of Systems Biology.

In principle, finding a TFBS *in silico* amounts to discovering a weak signal which comes from overrepresentation of motif occurrences in promoter regions, and which is very often masked by noise of the background. This signal-to-noise problem is very difficult to solve and it is approached by numerous methods in a number of quite different ways. The algorithmic approaches used for finding motifs range from such techniques as Gibbs sampling [1, 2], through Expectation Maximization [3], to word counting [4, 5]. The interested reader is referred to a good tutorial on discovering DNA sequence motifs and practical aspects of motif discovery by [6]. As a result of the multitude of methods, it usually happens that the outputs produced by different programs for the same input data are quite incongruent to each other, making it very difficult to compare. As a recent study [7] shows, there is no clear winner among the many programs which predict TFBSs. It also follows from this study that the joint wisdom which comes from applying different algorithmic

[*] bartek@mimuw.edu.pl
[†] m.darzynkiewicz@students.mimuw.edu.pl
[‡] tiuryn@mimuw.edu.pl

techniques is an advantage over any single approach. It is therefore quite reasonable to rely on the output of various programs which find TFBSs and produce in the end some kind of a consensus prediction, which can be further used in subsequent analyses. This is the aim of the proposed program called MEMOFinder. Since this is not the first program of this kind, we first briefly discuss other approaches and then explain the essence of the presented program, indicating in which aspects it differs from its predecessors.

Other approaches

The first tools for running multiple motif discovery programs (BEST [8] and TAMO [9]) did not really combine different outcomes, nor did they build a consensus solution. The first approach which used several motif finder programs to discover motifs was, to our knowledge, the MultiFinder suite published in 2006 in [10]. The purpose of MultiFinder was different than that of our program. It was built in order to verify the hypothesis that many orthologous genes in human and mouse which are similarly expressed in various tissue-specific data are co-regulated by orthologous TFs. MultiFinder uses four motif discovery programs: AlignACE [11], Bioprospector [2], MDscan [12], and MEME [3]. After the results of these four programs are collected, MultiFinder uses Pearson correlation coefficient for merging the predicted motifs and further clasterization.

The other two approaches were published in about the same time in 2007. The first, WebMOTIFS [13] is a web-based program which, like MultiFinder, uses four motif finder programs (with Weeder [5] instead of Bioprospector). The scheme of this package looks similar to MultiFinder: it evaluates the significance of each found motif (with hypergeometric enrichment score), and then clusters the significant motifs according to their similarity. For this purpose it uses a suite of tools from TAMO. A novel contribution of WebMOTIFS lies in a construction of a consensus solution.

There is also another approach called STAMP [14] which does not directly employ motif finding programs, so it is only partially relevant to our work. It is a web tool for motif clustering and finding consensus motifs using several well known motif similarity measures. STAMP may be used to analyze results of different motif finding programs in a similar manner as the presented approach.

Presented approach

The overall methodology of our approach is presented in Figure 1. It consists of the following steps:

- Running different motif finding programs and gathering their input,
- Measuring distance matrix between all resulting motifs together with a set of motifs from a reference database,
- Computing motif clusters and calculating consensus motifs for each of them.

Even though the approach is similar to WebMotifs method [13], there are few important improvements. The most important one is the inclusion of the selection of motifs from a reference database (currently, as the default, we use species-specific motifs from JASPAR [15], user specified motifs are also an option). Motifs found by the *de novo* methods are clustered together with the database motifs. This allows the user to separate out the trully novel motifs from those clustered with the known motifs. Also, the cases where a consensus motif of a resulting cluster differs from the known motif assigned to it, may be of special interest since it has been shown that small variations in regulatory motifs can lead to significant changes in function [16].

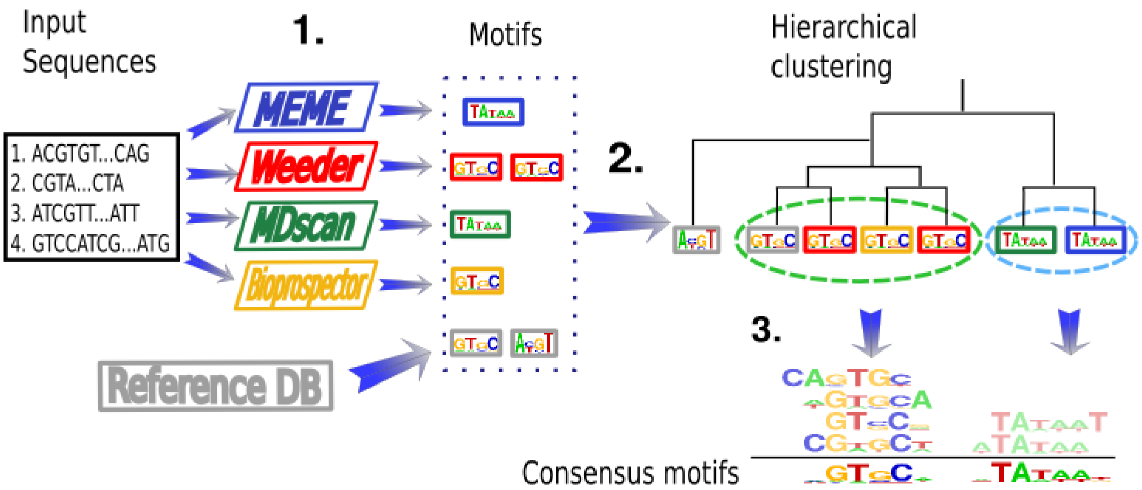

Figure 1: **The overall methodology of MEMOFinder.** Given a set of input sequences, we search for overrepresented motifs using four different methods (1). The motifs are pooled with a selection of motifs from a reference database and then input to a hierarchical clustering (2). Clusters of motifs are chosen based on a threshold depending on distances between motifs in a reference database (i.e. we avoid clusters grouping more than one reference motif). All clusters containing newly found motifs are then aligned and consensus motifs are found (3). Further details may be found in the online documentation at `http://bioputer.mimuw.edu.pl/software/mmf`

Methods

In the following sections, we describe the methodology used by MEMOFinder. Each section is devoted to different part of the overall workflow.

Motif finding

MEMOFinder allows the user to use four different *de novo* motif finding programs:

- BioProspector [2],
- MDscan [12],
- MEME [3],
- Weeder [5].

We have tried to make the sample of the programs representative (i.e. using qualitatively different methodologies), but we allow the user to provide additional motifs obtained using any other method.

There are several parameters which can be set for all methods both in the web version and in the standalone application:

- expected motif length,
- number of motifs returned by each program,
- single or double stranded search,
- sequence backgroud (i.e. one of the model organisms or uniform background).

In addition to these parameters, the users of the standalone program can set any parameters specific to each of the programs in the confog file. The web version uses defaults values for these parameters.

All the programs selected by the user are then subject to pre-filtering procedure which removes multiple instances of very similar motifs returned by any single program. This is to address the fact that some of the programs in some cases return multiple times virtually the same motif which could bias the results of later clustering.

Measuring distances between motifs

After finding the motifs, we need a way to compare them in order to obtain a sensible clustering. The problem of motif comparison is difficult in itself and is currently a field of active research [14,17,18]. The most common way of comparing motifs is to use gapless alignment of the motif Position Specific Score Matrices (PSSM) [13, 14] which is optimal with respect to some natural measure. However, another possibility of obtaining probability distributions from motifs is to calculate the PSSM score distribution over the input sequence [19] and then use one of the methods for comparing probability distributions.

Once the user chooses the probability distributions to compare, MEMOFinder allows him/her to use one of the following measures to perform the actual comparison:

- Relative entropy, or Kullback-Leibler divergence: very common measure for motif comparison, however not satisfying the triangle inequality.
- Pearson correlation based distance: also a common way of comparing motifs. It uses the $1 - P$ value as the distance, where P stands for pearson correlation between the considered distributions.
- D_{PQ} measure [20], a derivative of relative entropy satisfying the triangle inequality.

Clustering motifs

MEMOFinder uses the average linkage hierarchical clustering procedure [21]. In order to obtain proper division into clusters a threshold value needs to be set beforehand. In MEMOFinder, either an absolute value may be specified or the user can choose to base the threshold on a reference database. In this case, the threshold is set as the value relative to the closest pair of motifs in the database. For example, in case of relative value of 1.0, every motif from the reference database is assigned to a different cluster, while some of the newly found motifs might be clustered together with the reference ones.

After computing the clusters, MEMOFinder provides a consensus motif for each of them. This is done in an incremental procedure starting from most informative motif. Every time a new motif is added, the optimal gapless alignment is computed and the new PSSM is computed using all instances of both motifs. After merging all motifs, the flanking columns with information content below a specified threshold (0.4 by default) are removed.

Availability

The method is available online as a webserver at: http://bioputer.mimuw.edu.pl/software/mmf. In addition, the source code is released on a GNU General Public License. In order to run the application from source, a Java compiler (version 1.5 or higher) is required, as well as BioJava library and local installations of all the motif discovery programs.

Authors contributions

BW conceptualized the study and implemented the web interface, MD wrote the MEMOFinder application in Java, BW, MD and JT wrote the manuscript.

Acknowledgements

This work was partially suported by Polish Ministry of Science and Higher Education Grant number PBZ-MNiI-2/1/2005. Computational resources are provided by CoE BioExploratorium project: WKP 1/1.4.3/1/2004/44/44/115.

References

[1] Lawrence CE, Altschul SF, Boguski MS, Liu JS, Neuwald AF, Wootton JC: **Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment.** *Science* 1993, **262**(5131):208–14.

[2] Liu X, Brutlag DL, Liu JS: **BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes.** *Pac Symp Biocomput* 2001, :127–38.

[3] Bailey TL, Elkan C: **Fitting a mixture model by expectation maximization to discover motifs in biopolymers**. *Proc Int Conf Intell Syst Mol Biol* 1994, **2**:28–36.

[4] van Helden J, André B, Collado-Vides J: **Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies.** *J Mol Biol* 1998, **281**(5):827–42.

[5] Pavesi G, Mereghetti P, Mauri G, Pesole G: **Weeder Web: discovery of transcription factor binding sites in a set of sequences from co-regulated genes**. *Nucleic Acids Res* 2004, **32**(Web Server issue):199–203.

[6] MacIsaac KD, Fraenkel E: **Practical strategies for discovering regulatory DNA sequence motifs.** *PLoS Comput Biol* 2006, **2**(4):e36.

[7] Tompa M, Li N, Bailey TL, Church GM, De Moor B, Eskin E, Favorov AV, Frith MC, Fu Y, Kent WJ, Makeev VJ, Mironov AA, Noble WS, Pavesi G, Pesole G, Regnier M, Simonis N, Sinha S, Thijs G, van Helden J, Vandenbogaert M, Weng Z, Workman C, Ye C, Zhu Z: **Assessing computational tools for the discovery of transcription factor binding sites**. *Nat Biotechnol* 2005, **23**:137–144.

[8] Che D, Jensen S, Cai L, Liu JS: **BEST: binding-site estimation suite of tools.** *Bioinformatics* 2005, **21**(12):2909–11.

[9] Gordon DB, Nekludova L, McCallum S, Fraenkel E: **TAMO: a flexible, object-oriented framework for analyzing transcriptional regulation using DNA-sequence motifs.** *Bioinformatics* 2005, **21**(14):3164–5.

[10] Huber BR, Bulyk ML: **Meta-analysis discovery of tissue-specific DNA sequence motifs from mammalian gene expression data.** *BMC Bioinformatics* 2006, **7**:229.

[11] Hughes JD, Estep PW, Tavazoie S, Church GM: **Computational identification of cis-regulatory elements associated with groups of functionally related genes in Saccharomyces cerevisiae**. *J Mol Biol* 2000, **296**(5):1205–1214.

[12] Liu XS, Brutlag DL, Liu JS: **An algorithm for finding protein-DNA binding sites with applications to chromatin-immunoprecipitation microarray experiments.** *Nat Biotechnol* 2002, **20**(8):835–9.

[13] Romer KA, Kayombya GR, Fraenkel E: **WebMOTIFS: automated discovery, filtering and scoring of DNA sequence motifs using multiple programs and Bayesian approaches.** *Nucleic Acids Res* 2007, **35**(Web Server issue):W217–20.

[14] Mahony S, Benos PV: **STAMP: a web tool for exploring DNA-binding motif similarities.** *Nucleic Acids Res* 2007, **35**(Web Server issue):W253–8.

[15] Vlieghe D, Sandelin A, Bleser PJD, Vleminckx K, Wasserman WW, van Roy F, Lenhard B: **A new generation of JASPAR, the open-access repository for transcription factor binding site profiles.** *Nucleic Acids Res* 2006, **34**(Database issue):D95–D97, [http://dx.doi.org/10.1093/nar/gkj115].

[16] Segal L, Lapidot M, Solan Z, Ruppin E, Pilpel Y, Horn D: **Nucleotide variation of regulatory motifs may lead to distinct expression patterns.** *Bioinformatics* 2007, **23**(13):i440–i449, [http://dx.doi.org/10.1093/bioinformatics/btm183].

[17] Roepcke S, Grossmann S, Rahmann S, Vingron M: **T-Reg Comparator: an analysis tool for the comparison of position weight matrices.** *Nucleic Acids Res* 2005, **33**(Web Server issue):W438–W441, [http://dx.doi.org/10.1093/nar/gki590].

[18] Pape UJ, Rahmann S, Vingron M: **Natural similarity measures between position frequency matrices with an application to clustering.** *Bioinformatics* 2008, **24**(3):350–357, [http://dx.doi.org/10.1093/bioinformatics/btm610].

[19] Kowalczyk I: **Analysis of histone protein binding motifs in human promoters**. *Master's thesis*, Institute of Informatics, University of Warsaw 2004. [In polish].

[20] Endres D, Schindelin J: **A new metric for probability distributions**. *IEEE transactions on Information Theory* 2003, **49**(7):1858–1860.

[21] Sokal R, Michener C: **A statistical method for evaluating systematic relationships**. *Univ Kans Sci Bull.* 1958, **38**:1409–1438.