Zaid Ur Rehman
HiWi Report

under Dr. Fabrizio Costa
Chair of Bioinformatics

Project Details:
The project involved comparing motif-based sequence analysis tools. The tools under consideration were MEME [1], GLAM2 [2], DREME [3], and, our main tool of interest, the motif discovery feature of EDeN [4].

In the first phase, I developed python based wrappers for all tools. This involved developing an abstract class 'MotifWrapper' with fit(), predict() and transform() methods. The python wrappers for all tools inherited from this class.

Python wrappers for Sequence Logo and Muscle Alignment were also developed in this phase. The former was used to graphically represent motifs while the latter was used for alignment of motif sequences.

The next phase involved performing experiments on artificial data sets, and improving the results for EDeN. Experiments performed were based altering on noise levels for motives, number of motives, length of full sequences, number of motives in a sequence and number of sequences in a data set. A single data set was then split in two halves to serve as training data and test data. The metric used in comparisons was Area under Curve (AUC) of Receiver Operating Characteristic curve. Another experiment involved comparing running time of all tools. A sample result is shown below.

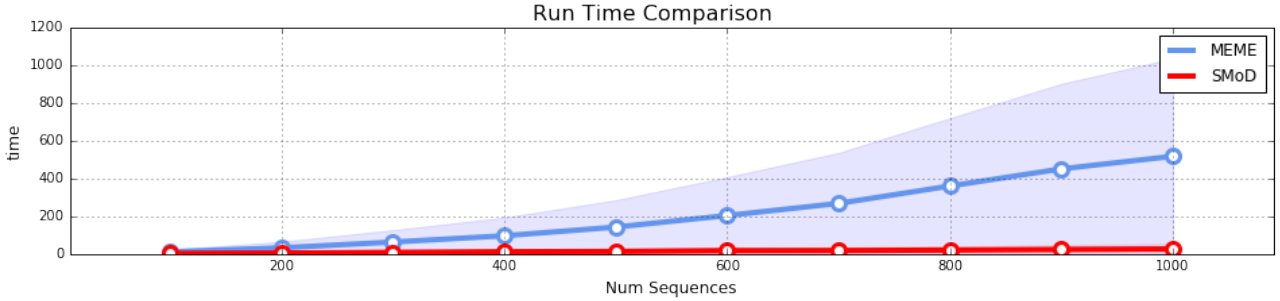

Fig 1. Running time comparison between MEME and SMoD

The last phase involved optimizing parameters of SMoD, sequence motif decomposer feature of EDeN which also serves as a motif discovery tool. Random values were generated from a Gaussian distribution for each parameter and performance for the whole parameter setting was tested for five different data sets. The parameter setting was reported and added to the Gaussian distribution if it achieves and average score higher than 0.6 on all data sets.

All python codes are available in a github repository under the MIT License.

Future Improvements:
Parameter optimization for noise experiments has not yielded robust parameters for SMoD yet. Further investigation into experiment failure for the tool and improving the process can lead to

satisfactory results. Parameter optimization can also be performed for experiments other than varying noise levels.

Github Repository File Reference:

- dreme_wrapper.py – python wrapper for DREME tool
- DREME_example.ipynb – Ipython notebook for DREME wrapper usage
- eden_wrapper.py – python wrapper for EDeN's sequence motif tool
- eden_example.ipynb – Ipython wrapper for eden wrapper usage
- glam2_wrapper.py – python wrapper for GLAM2 tool
- glam2_example.ipynb – Ipython wrapper for GLAM2 wrapper usage
- meme_wrapper.py – python wrapper for MEME tool
- meme_example.ipynb – Ipython notebook for MEME wrapper usage
- script_optimization.py – python script used to finding optimal parameters for SMoD
- smod_wrapper.py – python wrapper for EDeN's sequence motif decomposer tool
- smod_example.ipynb – Ipython notebook for SMoD wrapper usage
- utilities.py – contains the MotifWrapper abstract class, wrapper for Sequence Logo and Muscle Alignment
- utilities_examples.ipynb – Ipython notebook with examples for usage of sequence logo and muscle alignment
- Experiment_NumOfMotives.ipynb – Ipython notebook comparing results of tools on data sets with varying number of motives in sequences
- Experiment_SequenceLength.ipynb – Ipython notebook comparing results of tools on data sets with varying sequence length
- Experiment_Time.ipynb – Ipython notebook comparing running times of tools on different data sets
- Experiment_noise.ipynb – Ipython notebook comparing results of tools on data sets with varying levels of noise for motif sequences. This notebook also contains results for SMoD's optimized parameters.
- Perfomance Comparison.ipynb – Ipython notebook which serves as a general guide to perform different experiments for all tools.
- Motif extraction comparison – EDeN, MEME, SMoD.ipynb – Ipython notebook to visualize score signals for the three tools
- Score Signal Visuals for SMoD.ipynb – Score signals represented in a graphical manner for data sets with increasing noise levels.
- Score_Comparison_SMoD_vs_Wrapper.ipynb – Ipython notebook for comparing the scoring schemes for SMoD wrapper and SMoD's internal score scheme.

References:

[1] Timothy L. Bailey and Charles Elkan, "Fitting a mixture model by expectation maximization to discover motifs in biopolymers", *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pp. 28-36, AAAI Press, Menlo Park, California, 1994.

[2] MC Frith, NFW Saunders, B Kobe, TL Bailey, "Discovering sequence motifs with arbitrary insertions and deletions", PLoS Computational Biology, 4(5), e1000071, 2008.

[3] Timothy L. Bailey, "DREME: Motif discovery in transcription factor ChIP-seq data", *Bioinformatics*, 27(12):1653-1659, 2011.

[4] Costa, Fabrizio, and Kurt De Grave. "Fast neighborhood subgraph pairwise distance kernel."

Proceedings of the 26th International Conference on Machine Learning, 2010.