BIOINFORMATICS

Vol. 00 no. 00 2012
Pages 1–2

A Timed String Biological Motif comparison of Biopython, Motility and Tamo

Philip Trosko [1], Eric Macdonald [2] and Titus Brown [2*]

[1]Trosko's Consulting and Programming, 1630 Sylvan Glen, Okemos, MI, 48864.
[2]Department of Molecular Genetics, Michigan State University, 48823.

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

Motivation: When doing a google search of motif searching routines for biological sequences one easily finds the routines: Biopython, Motility, and Tamo. This paper gives a brief comparison of the routines in doing exact matching, IUPAC matching and PAM matrix positional weighting calculations.

Results: All three routines produced consistent matching results. The documentation for all routines allowed for the construction and running of our tests in python. When doing exact matching, IUPAC matching and PAM weighted matches Motility is almost as fast as doing hard coded searches in Python, while Biopython and Tamo run quite a bit slower.

Availability: All three routines are available online, and can be downloaded and compiled by a person with some sophistication using Git, ftp, make and python.

Contact: brown@msu.edu

1 INTRODUCTION

The field of string searching usually falls in the discipline of Computer Science. Usually we refer to a query q of length m and all the offsets of q that lie on a text of length n, where n is usually much longer than m.

$$q[m] \rightarrow t[n] \tag{1}$$

Figure 2 follows.

$$\sum_{i=1}^{n}\sum_{j=1}^{m} W_i(q[j], t[i+j]), O(m*n) \tag{2}$$

2 APPROACH

Our approach was to dowload Biopython, Motility and Tamo all on to the same machine, and run them using the same queries and the same domain data. Once running we furthermore for reproducibility made our data run in scripts and (will) ran these scripts under similar conditions on the Amazon cluster.

*to whom correspondence should be addressed

Table 1. Biopython

Biopython Query	Exact Match Time
TATAA (5 bases)	1:44.02
Small (8 bases)	1:45.27
Medium (12 bases)	1:43.11
Large (20 bases)	1:44.54
VLarge(60 bases)	1:43.97

This is a footnote

Table 2. Motility

Motility Query	Exact Match Time
TATAA (5 bases)	3.13
Small (8 bases)	3.00
Medium (12 bases)	2.99
Large (20 bases)	2.99
VLarge (60 bases)	2.99

This is a footnote

3 METHODS

Our method involved dividing the Motif queries into three groups:,

- Exact matching,
- IUPAC matching,
- PAM weight matrix matching.

All three routines support these matches, the queries and data were from the bacteria data set. Escherichia.,xxxx We feel they are representative of the kinds of queries of strings into genomic information. First the exact queries were broken down by size into four selected five groups: TATAA, Small, Medium, Large and Very Large of randomly selected query data. The reason for an additional small recognized region called the TATAA box was because in our test data, Bacteria, it has a large number of recognition sites and is a functional recognition site. Furthermore forward match, reverse match of each of the queries was performed because in Biology DNA is two stranded with matches possible in either direction.

Table 3. TAMO

TAMO Query	Exact Match Time
TATAA (5 bases)	1:28.88
Small (8 bases)	2:22.33
Medium (12 bases)	3:17.94
Large (20 bases)	4:37.30
VLarge (60 bases)	12:33.25

This is a footnote

Table 4. This is table caption

Routine	WGTATA	PAM
Biopython	97.14	39.44
Motility	3.56	3.56
TAMO	26.19	2:29.99

This is a footnote

4 DISCUSSION

We feel the text and the size of data and queries are very representative of the kinds of ongoing queries in biology. When trying to find epigenetic structure in a single small organism's DNA, small restriction site cuts or operative regions in the DNA all three routines are adequate. If doing a simple exact match a hard coded string search would be fast, however Motility is fastest. Biopython and Tamo are slower.

5 CONCLUSION

Our brief software run time comparison shows Motility to be superior in run times and that all software compared are capable of doing the searches required for simple DNA comparison.

ACKNOWLEDGEMENT

We would like recognise ???

Funding: Funding for Philip Trosko was provided by Kay Trosko, Titus Brown was provided by XXXXXX, and funding for Eric Macdonald was provided by XXXXXX.

REFERENCES

[1]Knuth,D.(1965) Volume 3, Chapter 6, Searching *The Art of Computer Programming*, pp-pp.

[2]Sankoff, X and Kruskal,X.(XXXX), Chapter X, SXXXXXX *Time Warps, String Edits and Macromolecules, The Art and Practice of String Comparison*, pp-pp.

[Escherichia.,xxxx]Name, X. (xxxx) Escherichia coli, compete genome.