Published online 00-00-0000

Nucleic Acids Research, 0000, Vol. 00, No. 00 ***1–2***
doi:10.1093/nar/gkn000

Basic implementation of nucleotide-nucleotide BLAST in Seq

Boris Kudryavtsev Jordan Kirchner [1] *

[1]Faculty of Engineering, University of Victoria, 3800 Finnerty Road Victoria BC V8P 5C2 Canada

ABSTRACT

Seq is a programming language specifically designed for applications in computational genomics and bioinformatics. With specialized features and optimizations, C-like performance, and Python-compatible syntax, this language provides a building ground for the rapid development of high-performing solutions in sequence analysis. In this paper, we present our efforts in writing a basic implementation of BLAST using this language for the comparison of nucleotide sequences. BLAST (basic local alignment search tool) is a sequence alignment algorithm and a suite of programs used to search for similarities in genetic sequences, and is considered the de facto standard for sequence alignment and search. Seq allowed us to quickly implement the basic features of the BLAST command-line interface, due to its familiar syntax and specialized features. Our implementation was able to achieve promising results at the initial stage, however, more work needs to be done to improve its alignment accuracy and performance.

INTRODUCTION

The Basic Local Alignment Search Tool (BLAST) is an algorithm, online search tool, and a collection of programs used for sequence similarity analysis (1). It is one of the most widely cited tools in bioinformatics and is considered to be the de facto standard in sequence searching. To implement this classic algorithm, we have chosen Seq (2) to get the job done, as it provides the flexibility and intuitive syntax of Python while achieving performance comparable to C or C++.

Here we will focus on "blastn", the variant of BLAST for comparing nucleotide sequences to other nucleodite sequences. Under normal circumstances, BLAST removes low complexity regions (i.e. extensive regions composed of the same nucleotide that hold less useful information to BLAST than regions of moderate or higher complexity), however, to simplify the process we decided to omit this step as BLAST calls two other programs to filter out these regions - namely DUST for DNA, and SEG for proteins. After this step, BLAST generates all words of length W in the query sequence (k-mers of a specified length - typically 11 for a DNA sequences) to be queried against a target sequence. These words are then scored using a scoring matrix (such as BLOSUM62 for sequences longer than 85), and are assembled into High-scoring Segment Pairs (HSP).

Once the initial "seed" HSPs have been retrieved, they can then be used to determine exact matches when iterating through the target sequence. When an exact match is found, this region is further extended to the left and right until a certain criteria is met, usually either if there is a single mismatch in one direction or if the overall score of the extended seed dips below a certain value. Like the words themselves, these HSPs need to score above a certain threshold in order to be considered meaningful. How this score is weighted is by permuting the HSP and checking whether the original orientation has a better score than a certain (empirically determined) percentage S of its other configurations, in which our case this parameter was assigned a value of 95%. Furthermore, the significance of each HSP is checked to asses whether or not there is indeed a similarity between the HSP and the targeted region – done so through the means of what is known as an E score - as there may just be a mere fluke match and extension. After verifying that the remaining HSPs have meaningful content, they are then stitched together to form the alignment.

RESULTS

We tested our implementation on several Coronavirus nucleotide sequences, as well as an Apple/Citrus junos fruit viroid gene, and Influenza A, B, and D virus sequences. The results were then compared to those given by the blastn command-line interface. The sequences were downloaded from the National Center for Biotechnology Information virus database (3). When comparing Coronavirus sequences, we noticed that our system accurately predicted the bit-score of highly similar sequences, such as the partial sequence MW362224 aligned against the complete genome of SARS-CoV-2, NC_045512, yet struggled to find all similarities in AY429073, a less similar sequence of the SARS virus (see Table 1). However, when aligning the Influenza virus sequences, or the Apple/Citrus virus sequence against NC_045512, our system did not find any significant similarities, matching the blastn results.

*To whom correspondence should be addressed. Email: bkudryavtsev@uvic.ca

```
blastn-seq v0.0.1 alpha

Query = MW362224.1 |Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/RUS/MOS-CRIE-10867366-
D168K0021/2020 nucleocapsid phosphoprotein (N) gene, complete cds
Length = 1260

> NC_045512.2 |Severe acute respiratory syndrome coronavirus 2 isolate Wuhan-Hu-1, complete genome
Length = 29903

Score = 1733 bits (6264)
Identities = 1256/1260 (99%)

Query 0          ATGTCTGATAATGGACCCCAAAATCAGCGAAATGCACCCCGCATTACGTTTGGTGGACCC 60
                 ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Targt 28273      ATGTCTGATAATGGACCCCAAAATCAGCGAAATGCACCCCGCATTACGTTTGGTGGACCC 28333

Query 60         TCAGATTCAACTGGCAGTAACCAGAATGGAGAACGCAGTGGGGCGCGATCAAAACAACGT 120
                 ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Targt 28333      TCAGATTCAACTGGCAGTAACCAGAATGGAGAACGCAGTGGGGCGCGATCAAAACAACGT 28393
```

Figure 1. Sample output of our BLAST implementation (blastn-seq).

Table 1. Comparison of nucleotide alignments produced by our implementation (blastn-seq) and the BLAST command-line tool (blastn). The complete genome NC_045512 (SARS-CoV-2) used as the target genome. Configured with ungapped alignments and score matrix +5/-4.

Method	MW362224		AY429073	
	Bit-score	Identities	Bit-score	Identities
blastn	1733 bits	99%	1592 bits	81%
blastn-seq	1733 bits	99%	833 bits	59%

Table 2. Comparison of time performance of our implementation (blastn-seq) and the BLAST command-line tool (blastn). The complete genome NC_045512 (SARS-CoV-2) used as the target genome. Configured with ungapped alignments and score matrix +5/-4. Machine specifications: MacBook Pro 2017, 2.3 GHz Dual-Core Intel Core i5, 8 GB 2133 MHz LPDDR3

Query	blastn Time (s)	blastn-seq Time (s)
MW362224	1.191	17.075
AY429073	0.781	28.856

The seed and extend process in BLAST can be of O(nm) time complexity, where n is query length, and m is target length. As this process in our implementation is not optimized and does not utilize parallelization, it is not surprising that its performance is marginally slower than that of highly-optimized blastn (see Table 2).

CONCLUSION

Although our implementation of BLAST met many of the necessities outlined in the algorithm, there are still many improvements to this prototype that could be flourished in future work. For instance, searching through the target sequence for matches can be parallelized to see a significant increase in computation speed. This can be easily achieved, thanks to the parallelization features built in to Seq. Another enhancement would be to fine tune the scoring parameters dynamically depending on the sequence length, as we found our bit scores to be off by a small margin. Despite these shortcomings, we feel as though we did a fair job at both translating the idea behind BLAST into code, and have as a result increased our understanding of the lengths researchers have to go in order to produce efficient, albeit sub optimal solutions for some of the gargantuan problems in the field of bioinformatics.

REFERENCES

1. Altschul, S. F., Gish, W., Miller, W., Myers, E. W., & Lipman, D. J. (1990). Basic local alignment search tool. *Journal of molecular biology*, 215(3), 403–410. https://doi.org/10.1016/S0022-2836(05)80360-2
2. Ariya Shajii, Ibrahim Numanagić, Riyadh Baghdadi, Bonnie Berger, and Saman Amarasinghe. 2019. Seq: a high-performance language for bioinformatics. *Proc. ACM Program. Lang. 3, OOPSLA*, Article 125 (October 2019), 29 pages. DOI:https://doi.org/10.1145/3360551
3. Sayers EW, Agarwala R, Bolton EE, Brister JR, Canese K, Clark K, Connor R, Fiorini N, Funk K, Hefferon T, Holmes JB, Kim S, Kimchi A, Kitts PA, Lathrop S, Lu Z, Madden TL, Marchler-Bauer A, Phan L, Schneider VA, Schoch CL, Pruitt KD, Ostell J. Database resources of the National Center for Biotechnology Information. Nucleic Acids Res. 2019 Jan 8;47(D1):D23-D28. doi: 10.1093/nar/gky1069. PubMed PMID: 30395293; PubMed Central PMCID: PMC6323993. 2: Sayers EW, Cavanaugh M, Clark K, Ostell J, Pruitt KD, Karsch-Mizrachi I. GenBank. Nucleic Acids Res. 2019 Jan 8;47(D1):D94-D99. doi: 10.1093/nar/gky989. PubMed PMID: 30365038; PubMed Central PMCID: PMC6323954.