Modifying BLASTer to help make observing protein and nucleotide sequences easier.

By: Paul Steele

Introduction

What is BLAST?
- Stands for Basic Local Alignment Search Tool. It runs through a command line.
- Compares unknown protein or nucleotide sequences to a database of known sequences to find similarities and differences.

Why is it useful?
- Results can help researchers: cure diseases, improve crop resilience, or create commercial products.

Is BLAST easy to use?
- No graphical user interface.
- Consumes large amounts of processing power, so running searches on a local machine is not always practical.

Are there any tools that make it easier to use?
- The National Center for Biotechnology Information (NCBI) hosts an online search wrapper for BLAST.
- Search is performed on a faster machine.
- User doesn’t have to worry about syntax.
- Nice output options.
- Large searches are not allowed.
- Purdue has their own wrapper for BLAST called BLASTer.
- Same benefits of the NCBI’s site.
- Search distributed to cluster nodes, increasing performance.
- Large searches allowed.
- Only standard output options.

What needed modified on BLASTer?
- User needed ability to choose output options after a search finished instead of before it started to not waste time, as some searches can take days to complete.
- Extra output options would help enrich user experience.

Modification Process

Step 1: Choose an initial output format that can be reformatted later
- BLAST supports 15 different output formats.
- BLASTer needs to support them all.
- Initially thought all of the options would have to be manually recreated from one format. Candidates were likely either XML or JSON as they contained the most information.
- Found that BLAST has a built-in reformatting tool that uses the ASN format.
- BLASTer just needed to make a call to the reformatter instead of converting manually. This makes it easy to maintain.

Step 2: Combine results obtained from the clusters
- BLAST databases can be large (> 50GB), but they are split up into 1GB chunks.
- BLASTer also splits up input sequences into subsets if there are a large number.
- A BLAST job is sent to the cluster for every entry in the cross product of the database set and the input set. (# Database chunks x # input subsets)
- BLASTer used to simply concatenate the results from each job together. That doesn’t work well with the ASN format as results for a singular query would be split amongst the entirety of the file making it hard to decipher.
- The merger works by creating an intermediate representation tree of the data it is fed, only interpreting components it needs to. With that it then writes information to 3 distinct sets of temporary files.

Step 3: Update UI to allow new functionality
- New options were added to the workflow to support new features.
- Users at anytime after a search is finished can create a new output option.

Step 4: Add additional output formats
- To enhance usability, two new output options were added unique to BLASTer.

Outcome & Acknowledgments

- After modification, BLASTer is now more user friendly.
- With the more advanced result merging method, researchers can find results quicker as everything pertaining to one query is in the same spot in the file.
- The new visualization tool allows seeing matches in a way other than text.

Acknowledgments
Christopher Thompson
Carol Song
Purdue RCAC
Discovery Park Undergraduate Research Internship (OUR) program
Jeff Wintersinger (Kablammo tool - http://kablammo.wasmathlab.org/)