**Bayesian Lowest Common Ancestor (BLCA) Web-Based Version**

(Web-Based 16s rRNA Classifier)

by

Paul Okoro, Benjamin Lorentz, John Morris

**Project Overview and Summary**

Microorganism are almost ubiquitous in their varying degree. This poses a challenge of being able to catalogue and classify all known and newly identified organism into an organized or taxonomic system for ease of study and use. In times past, microbiologists were able to achieve this classification by merely studying microbial morphological representation and reactions to coloring technique. However, with molecular studies in genetic materials of living organisms, it was observed that the 16s subunit of the ribosomal RNA sequence are unique and conserved across bacteria with little variations enough to distinguish each bacterium from another. Simply put, 16srRNA from bacteria A is compared against other know bacteria 16srRNAs and classified based on percent similarity between the sequences. As such, molecular microbiologists now advanced to using this 16s rRNA to identify and taxonomically classify bacteria. Subsequently, many software tools have been developed that will leverage on the marker sequence (16s rRNA) to classify bacteria. However, most of these tools lack species-level classification power as they are only able to classify to genus level. Even some current tools that classify further to species level suffer limitations; Firstly, by using nucleotide k-mer frequency to measure similarity between query and database sequence thereby wrongly assuming that these k-mer sequence are independent and that their position in the DNA are not important. Secondly by lacking a solid probabilistic-based criterion on which to base the confidence of the taxonomic assignment results.

To this end, the Bayesian Lowest Common Ancestor (BLCA) was developed to tackle these limitations and provide probabilistic taxonomic classification to species level with confidence scores. The BLCA is implemented as a python package and has this simple workflow in fig.1. below. More details on the BLCA algorithm and actual software is found at <https://github.com/qunfengdong/BLCA/tree/cf6790cd111a60966624f5e0759180610431d726> . Although the BLCA implementation makes significant improvement at species level classification as against the other tools, it has some limitation when it comes to public wide usage.

1. It is computationally intensive to run the analysis as it takes approximately 4 human days to complete.
2. It requires the user to have averagely great computer to run and complete at the approximated time.
3. It is very technical for users who do not have programming, and command line skills to implement.
4. The technical requirements a system must meet or dependency software tools a user must setup or specify can be difficult for an average computer user to implement.
5. It requires users to have Linux systems and or be Linux savvy.

Therefore, to make this highly useful software (BLCA) accessible to a wider public, it is paramount for it to be implemented on the web and thereby making the usage seamless for every user. Therefore, the central goal of this project is to develop a simple web version of the BLCA software.

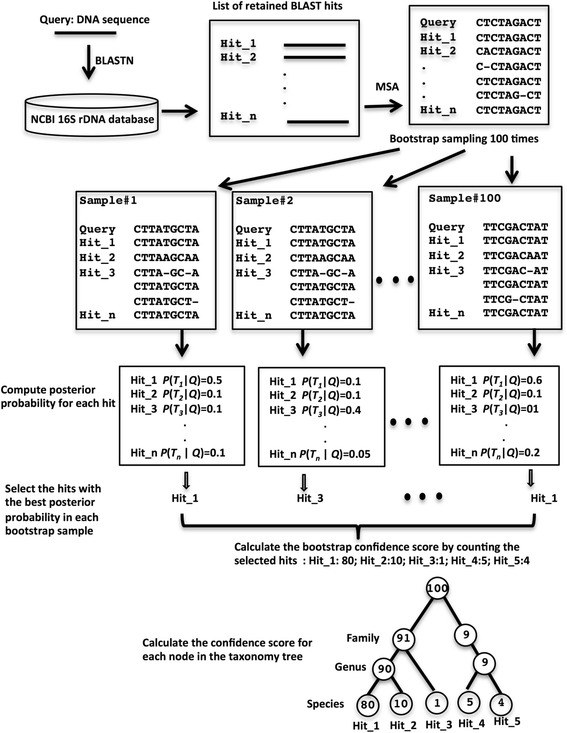


Fig. 1. BLCA Workflow. (Gao et al., (2017). BMC Bioinformatics, 18(1):247)

**Goals and Project Vision**

1. This project is solely to build a web version of the BLCA software with python flask. It will have a permanent web version hosted on the Loyola University Chicago Medical School server. It will also have a docker image with all dependencies built so that users with the requisite technical skills can host in their own servers as well.
2. Implementation of this project will simplify use of BLCA for all non-technical users.

**Non-Goals**

1. The web project will not run further analysis beyond the simple taxonomic classification of input 16srRNA query sequence with BLCA algorithm.
2. The web project will also not create tool for sequence format conversion.

**Web Page**

The baseline web page will look like fig.2. and then grow with more parameters over time

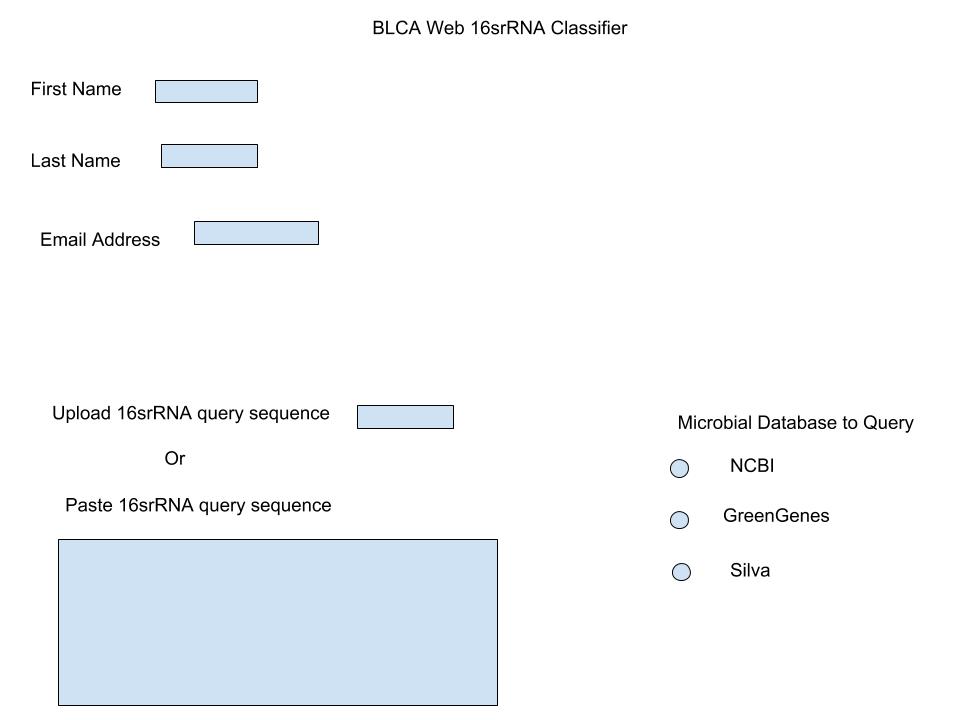


Fig.2. Basic Site Map

**Simple Web Page Operation Workflow**

1. Take user query sequence as well as Names and email contact.
2. Return instant message to the user after submission of the job, stating that the job is running and that the results will be sent to the user after completion via email.
3. Store user name, email and query sequence. Initiate a system call on the BLCA software at the backend server with the query sequence and run the taxonomic classification.
4. At completion of Job, retrieve stored user email, and forward result to it.

**Timeline & Milestones**

March 16, 2019: Run BLCA in command line with test dataset

March 19, 2019: Write a simple python flask web app that make a system call

March 26, 2019: Write python flask web form app that store user information and send email

April 2, 2019: Write python flask web app that use user data to run default BLCA and send result via email

April 9, 2019: Add options in the web app to run BLCA with user specified parameter

April 16, 2019: Complete web app project and deploy.