Design and implementation of novel applications for KBase

Boinzemwende Jarmila Roxane Ouango,
Engineering program, Hostos Community College, Bronx, NY, 10451
David Dakota Blair,
Computational Science Initiative, Brookhaven National Laboratory, Upton, NY, 11793

***Abstract*—The Department of Energy Systems Biology Knowledgebase (KBase) is an open-source project enabling collaborative bioinformatics data processing and supporting reproducible science. KBase is designed to be an extensible community resource whose growth is supported by the KBase Software Development Kit (SDK), which enables any developer to build, test, register, and deploy new or existing software as KBase apps. The SDK enables developers to extend the platform's scientific capabilities. KBase supports data provenance and analysis reproducibility and has a flexible system for sharing data and workflows. Our project at Brookhaven National Laboratory aims to contribute novel applications, modules, and methods to KBase by wrapping an existing tool called JGI-MiniScrub in a KBase app. MiniScrub is a novel Convolutional Neural Network (CNN) based method for de novo identification and subsequent scrubbing of low-quality Nanopore read segments. The technologies required for this include Docker, POSIX shell, python as well as internal KBase APIs. Our approach consists of learning biological topics such as genomes and model organisms, and software engineering concepts such as HTTP, the OSI model, Object-Oriented programming, design patterns, Model-View-Controller (MVC), and public-key cryptography. Additionally, our approach requires effort to achieve familiarity with development tools such as Git, secure shell, and the KBase SDK tool kb_sdk. By learning about KBase and its SDK, and implementing MiniScrub in a KBase app, we can promote user-friendly interfaces, and contribute new features to KBase to enable researchers to pursue new avenues of research made possible by these new features.**

I. Introduction

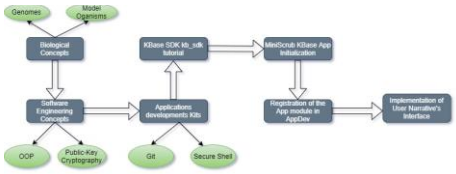
Long read sequencing has become increasingly important in recent years, with sequencing technologies from companies such as Pacific Biosciences and Oxford Nanopore seeing wide use in a variety of applications including genome assembly, detection of antimicrobial resistance genes, sequencing personal transcriptomes, and improving draft genomes. Sequence assembly is one of the most promising and explored of these applications. Long repeat sections have been shown to be among the most important factors that affect assembly quality, and long sequencing reads are much more capable of resolving these long repeats [1]. Current single molecule, long sequencing reads also have very high error rates, ranging from 5% to 40% per read and often average about 10% to 20% up to as high as 30-40% depending on variables such as the type and version of the sequencing technology and the experiment being performed. These high error rates make assembly and other applications inefficient or error prone. It is thus critical that methods be developed towards addressing this issue so that the potential of long read sequencing can be fully realized. a novel Convolutional Neural Network (CNN) based method, called MiniScrub, for de novo identification and subsequent scrubbing of low-quality Nanopore read segments.

Our project at Brookhaven National Laboratory aims to contribute novel applications, modules, and methods to the Department of Energy Systems Biology Knowledgebase (KBase) by wrapping MiniScrub in a KBase app.

II. METHOD

First, In preparation of wrapping MiniScrub in a KBase app, we needed familiarity with biological concepts including genomes and model organism and software engineering concepts such as Object-Oriented programming, and public-key cryptography. we learned about applications development kits such as Git and secure shell. Then, we learned about the KBase software development Kit (SDK) by designing an app called ContigFilters through the kb-sdk tutorial available on KBase for internal and external contributions to the platform. In this process, we have installed and gained acquaintance with open-source containerization platform such as Docker. we assimilate how the SDK works and the general concepts involved in making and running apps. In the process of wrapping MiniScrub, we investigate the literature review on the method, its features to understand its aim and how it could potentially support the KBase on its mission of meet the grand challenge of systems biology.

In implementing MiniScrub in the KBase app along with my mentor, we investigate the dependencies required to install the method. These included Docker, and git. Then, we initialized the MiniScrub module containing all the component for the KBase app. This was performed using the KBase SDK commands. Furthermore, we build the Narrative app user interface by creating a directory for the new app through the spec.json file and setting the app's parameters in the display.yaml file. Finally, we registered the MiniScrub module in AppDev, the KBase narrative server that is used by app developers to publish in-progress versions of their apps and test and share them using real data.


Genomes
Model Organisms
Biological Concepts
Software Engineering Concepts
OOP
Public-Key Cryptography
Applications developments Kits
Git
Secure Shell
KBase SDK kb_sdk tutorial
MiniScrub KBase App Initialization
Registration of the App module in AppDev
Implementation of User Narrative's Interface

MiniScrub KBase App Design Flowchart

III. Results

MiniScrub was wrapped in a KBase app called kb-miniscrub. A demonstration narrative explaining the app's features was created. From the KBase's primary user interface, the Narrative Interface built on the Jupyter10,11 platform, users can (Figure 2) upload their private data, search and retrieve extensive public reference data, select and run the kb_miniscrub app on their data, view and analyze the results from the app, and record their interpretations along with the analysis steps in the markdown cell. Kb_miniscrub can be used for de novo genome assembly and large structural variation identification.

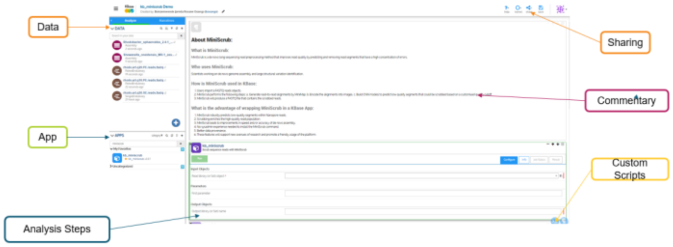


Figure 2: kb_miniscrub user narrative's interface

Kb_miniscrub takes file a FASTQ as input. To use the app in KBase, users import FASTQ reads objects. Then, MiniScrub performs the following steps by generating reads to reads alignments by MiniMap, encode the alignment into images, and build Convolutional Neutral Network (CNN) models to predict low quality segment that could be scrubbed based on a customized quality cutoff. Finally, the app outputs a new FASTQ file with the scrubbed reads. MiniScrub accurately scrubs out low-quality segments within Nanopore raw reads to improve overall read quality, and the scrubbed reads lead to fewer assembly errors. In addition, MiniScrub outperforms alternative preprocessing tools in terms of leading to fewer mis-assemblies and large indels in de novo assembly.

The implementation of the MiniScrub method on kb-miniscrub for KBase can support new avenues of research and promote novel discoveries at Brookhaven National Laboratory as well as anywhere across KBase. MiniScrub robustly predicts low-quality segments within Nanopore reads. It predicts the percent identity of each segment of a read and scrubs out segments below a user-set threshold, splitting the reads at the low-quality regions [1]. Scrubbing enriches the high-quality read population. scrubbing out a small percentage of low-quality regions nevertheless raises average read percent identity by over 3% (from 83.1% to 86.2%). MiniScrub does not seem to perform much false scrubbing, as high-quality raw reads (particularly those at 90% or higher accuracy) remain almost entirely intact (figure 3) [1].

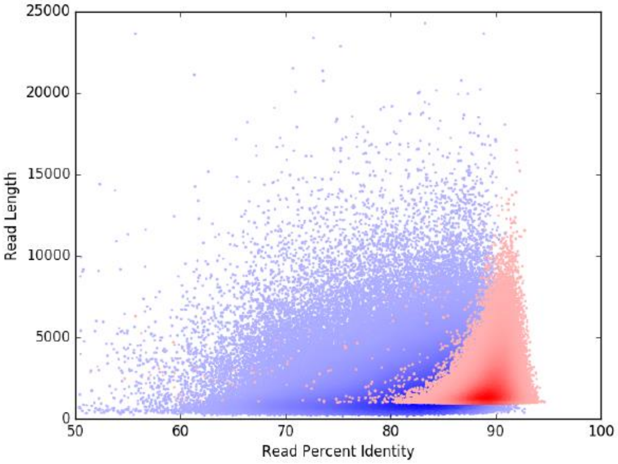


Figure 3: Density scatter plot showing average read quality improvement by MiniScrub versus raw reads [1]

Users are no longer require any sysadmin experience to install the MiniScrub command as it is available on KBase. Data provenance is easily manageable. This aligns with the platform strong commitment to data provenance and its mission of enabling collaborative bioinformatics data processing.

IV. CONCLUSIONS

The KBase app kb-miniscrub's features can support the grand challenges in long reads sequencing using the MiniScrub method. The implementation of the app narrative's user interface is still in progress as we are updating the app's module to enhance users' experience. In terms of the output, when a sequencing reads is imported in the app, the MiniScrub method returns an empty FASTQ file causing the kb-miniscrub app to output the same. We expected the method and thus the app to return a FASTQ file including scrubbed reads with removed low-quality segments. We have already made changes in the MiniScrub's code locally. Further analysis is needed regarding the MiniScrub's output so we can apply these changes to the app.

References

[1] MiniScrub: de novo long read scrubbing using approximate alignment and deep learning Nathan LaPierre, Rob Egan, Wei Wang, Zhong Wang bioRxiv 433573; doi: https://doi.org/10.1101/433573.

Appendix

I would like to thank my mentor Dr. David Dakota Blair for his generous support and dedication to my learning experience. Also, I want to thank Ziming Yang for her thoughtful contributions to our discussions, and project. Additionally, this project was supported in part by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Community College Internships Program (CCI).