A toolkit for reporting on metagenome assembly quality

James Edward Euesden

Department of Computer Science, Aberystwyth University, Aberystwyth, Ceredigion e-mail: jee22@aber.ac.uk

Project Information

The study of metagenomics is taking samples from environmental areas rich in species/sub-species and trying to find interesting genomes within them.

These samples may contain hundreds to thousands of different species, and conventional genome sequencing and assembly tools for single species analysis do not necessarily work for metagenome samples. Considering we often do not know what is within the sample taken to begin with, how then do we determine if our sequence assembly is of good quality?

What is quality in metagenomics?

If we don't know which species are present in our samples, understanding what quality is can be a challenging task. We know that we want to find genomes that exist in nature, and that when we get an assembly from our assembler, we would like to know it is not a chimera, formed of many different species that would never actually exist.

It is also relevant to look at the size of the assemblies produced, are they too small or too big compared to the reads that were used to construct them?

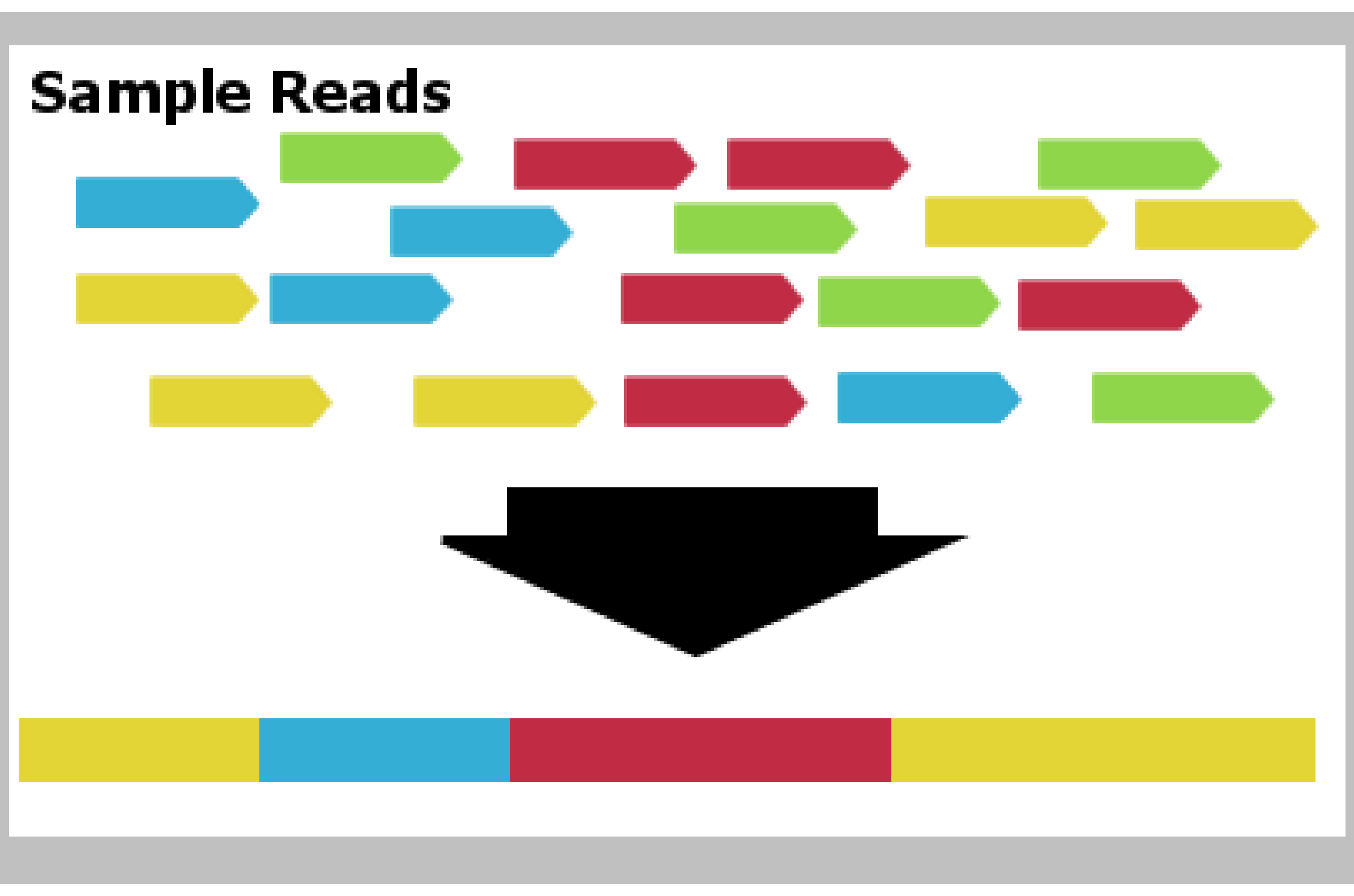

Figure 1. Demonstrating how multiple species in a sample could result in a very mixed assembly that may not actually exist in nature.

Note how the 'green' has not been included. We may not know if this section of the genome was important, and that if used would have made the contig 'good' or 'bad' quality.

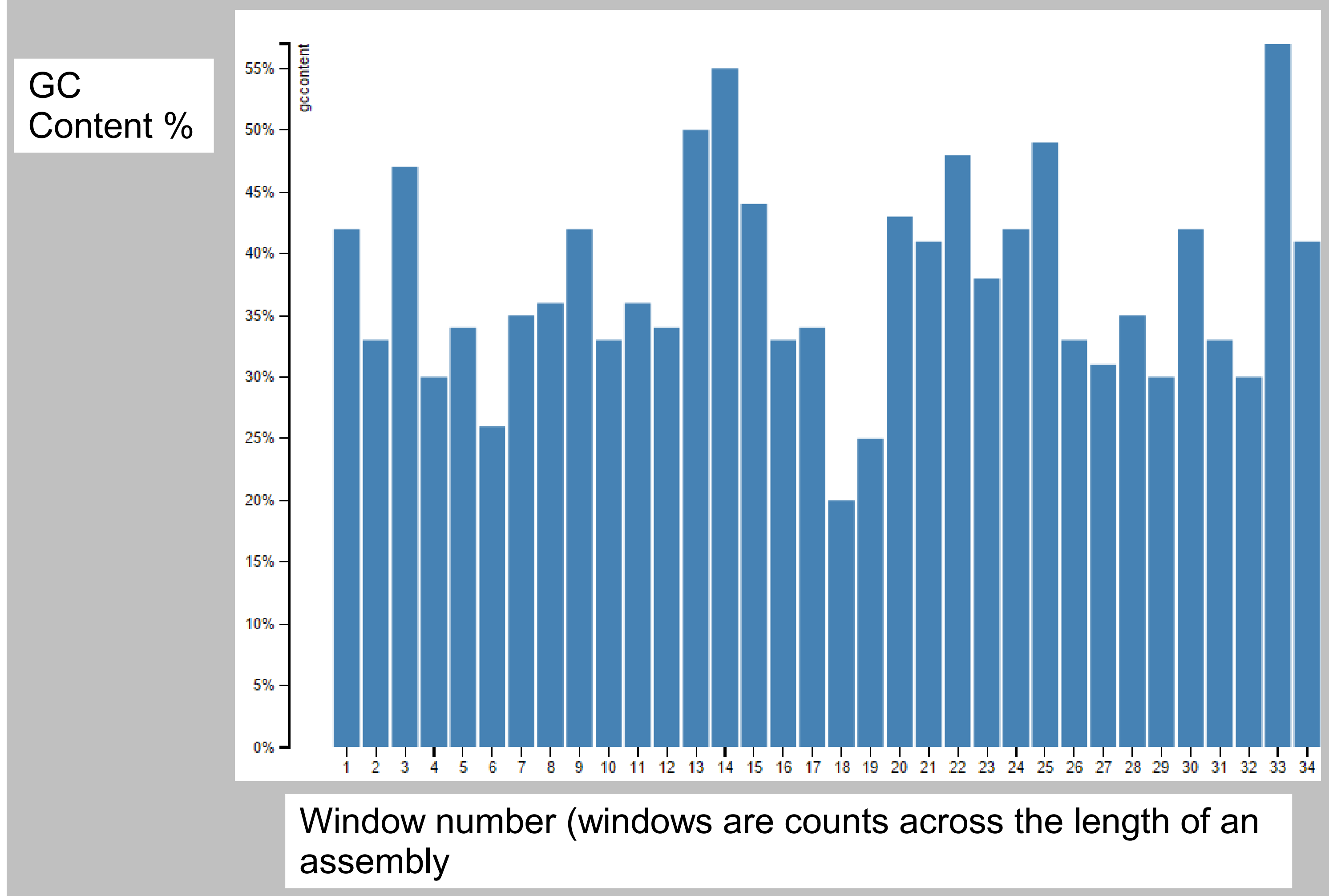

Figure 2. An example of what the GC content chart looks like from the prototype with Plotly. In future revisions, the chart will display potential protein encoding regions to match up, or not, with the GC content displayed.

Technical Information

The software application is being built in Java, using Spring Boot to display the report outputs in a browser with Javascript and Plotly. For sample test data, I have been using an assembly taken from a limpet gut (provided by Sam Nicholls of Aberystwyth University) and creating my own metagenome assemblies by combining single species assemblies from those in the NCBI RefSeq database [1].

For reporting on the quality, the application will mostly look at GC content within the assembly and notifying the user where there might be unnatural changes, k-mer counting for looking at sub sequences in the assembly and n50 (sizing of the assemblies provided). These are well documented and used techniques for helping determine quality of assemblies, although often for single species and not often brought together in one place.

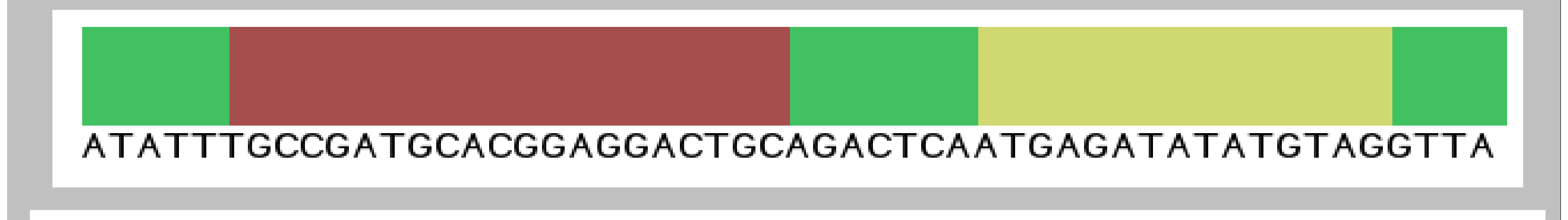

Figure 3. The red section highlights where there are more G's and C's than anywhere else in the assembly (the green). The yellow section indicates where there might be a protein encoding region (starts with ATG, and in this instance ends with TAG). This is an example of GC content and an ORF location to try and highlight potential bad quality areas

Project aims and progress

The goal of this project is to produce a software application to be used as a toolkit for helping researchers determine whether their metagenome assemblies are of good or bad quality, and attempt to indicate to them where there may be problems in the assemblies in a textual and graphical report.

My progress on the application has taken me through learning about metagenomics, what is considered as quality and how is it found, and what technologies I should use to build it. The application itself is in a prototype stage, where it will consider the GC content (the amount of G's and C's in the assembly versus A's and T's) of any number of assemblies provided in a FASTA file type and output the percentages based on window sizing, specified by the user. The FASTA format is a way of representing the assemblies that gives them a header and breaks down the lines of the assembly to be easier to work with when reading the file in to an application.

After GC content, the application then looks at Open Reading Frames to find protein encoding regions, which naturally have a higher GC content than most other areas in the genome. The aim is to match up any potential protein encoding regions with changes in GC content to be able to say they could be occurring naturally, and then highlight any GC content changes where it may be a sign of a chimera.

I have also made another prototype that allows me to display the outputs of the processing in a browser using Plotly, a Javascript based chart and graphing script for displaying the results of the GC content processing.

Remaining Work

Since the application is currently only a prototype and dealing with GC content, the next stage is to first tie this in with ORF locations to detect where there might be protein encoding regions causing discrepancies in the GC content that are not due to the presence of a chimera, and then to start work on the full application connecting the report and the processing.

Once this basic beginning is done, this will then allow me to begin adding the k-mer counting and processing of the size of the assemblies with the read sizes.

Tying in with these, there is a lot of work to be done on displaying a useful report to the user to indicate the results of these techniques and whether it indicates that the provided assembly is of reasonable quality or not and why. Future work may involve checking to see if any genome in the NCBI RefSeq database matches the assembly provided.

Further Reading & References

The NCBI handbook [Internet]. Bethesda (MD): National Library of Medicine (US), National Center for Biotechnology Information; 2002 Oct. Chapter 18, The Reference Sequence (RefSeq) Project. Available from http://www.ncbi.nlm.nih.gov/books/NBK21091/

Spring Boot - http://spring.io/