Conclusion

In this work, I presented two ways of calculating the Jaccard similarity between two strings using max- and minwise sketches. One had been previously described and the other was invented by the author. These ways of calculating the Jaccard index were then used to create two greedy clustering algorithms: **MM** and $\mathbf{MM}\frac{1}{2}$. I described the background material necessary to understand how MinHash works, and then went on to describe how strings should be processed so that their Jaccard similarity could be found using k-mers.

MM and $\mathbf{MM}\frac{1}{2}$ were developed using greedy algorithms which were described using pseudo-code. Their extension using the Hadoop MapReduce framework was described by examining their Pig scripts. Using MapReduce allowed for distribution of the workload, and greatly reduced the memory needed to run the algorithms.

Using real metagenome data for the tests, we saw clearly that there was a big difference between the results of **MM** and $\mathbf{MM}\frac{1}{2}$. $\mathbf{MM}\frac{1}{2}$ did not live up to the expectations set for precision; contrarily **MM** was at times even more precise than `Uclust`. In addition to this, **MM** was faster than `Uclust` in some cases, which made **MM** a potential alternative to `Uclust`.

As we mentioned in the Reflections section, the sparse number of tests that were performed in this paper were not sufficient to confidently determine the exact precision of **MM** or $\mathbf{MM}\frac{1}{2}$. These tests will be left for future works.

The source code for single-threaded and MapReduce frameworks of both **MM** and $\mathbf{MM}\frac{1}{2}$ are attached in the zip file this paper came in.