HMMRWAR

Sys Jonas (Universiteit Gent, Jonas.Sys@UGent.be)

13 mei 2022

Samenvatting

Bij het gebruik van de tool HMMRATAC blijkt de implementatie niet perfect te zijn. HMMRWAR probeert deze fouten te elimineren en een vervanger te zijn voor (een groot deel) van de tool. Een tweede voordeel aan de HMMRWAR implementatie is de snelheid en efficiëntie: C++ is notoir sneller en heeft een veel kleinere geheugenoverhead.

I. HMMRATAC

De tool HMMRATAC werkt in drie grote fasen: eerst wordt een vlugge screening van de data gehouden. Deze fase gebruikt expectimax op de lengte van de reads (deze worden gezien als een discreet signaal). Dit "signaal"wordt gedecomposeerd in vier onderdelen. Hieruit wordt in de tweede fase een HMM met 3 staten gegenereerd. Het grootste deel van fase 2 is het trainen van het model. Dit is nu ook de grootste bottleneck. In de laatste fase wordt de data geannotteerd door middel van het Viterbi-algoritme.

II. Implementatieplan

Aangezien fase 1 nu al volledig en correct geïmplementeerd is, kunnen we de implementatie in de huidige HMMRATAC tool gebruiken. Hiervoor wordt de source code een klein beetje aangepast.

De effectieve focus zal liggen op fasen 2 en 3 (als er nog tijd over is, kan fase 1 ook nog geschreven worden). De programmeertaal voor deze implementatie is C++20 (voornamelijk door dependencies); met de toolstack Conan/CMake. Aangezien het niet nuttig is om het wiel opnieuw uit te vinden, worden libraries gebruikt:

- `cxxopts` is een CLI-argument parsing library;
- `stochHMM` is een C++ implementatie van een HMM (inclusief Baum-Welch training);
- `fpgen` is een C++20 implementatie van generators en manipulators voor generators;
- `seqan3` is een I/O library voor biologische data.

III. Huidige status

Het project is opgezet en de CLI-argumenten kunnen geparst worden. Deze kunnen gespecifieerd worden door de macros; zie het bestand `meta/arguments.meta`.