Ročníkový projekt – report – zimný semester

Minimalizácia počtu runov v BWT

Tldr.:

- Naštudoval problematiku
- Otestoval – našiel chyby
- Napísal ďalšie testy
- Reimplementoval do funkčného stavu
- Optimalizoval (DP jednotlivých permutácii)
- Pridal možnosť minimalizovať labels

Zo začiatku som sa oboznámil s programovacím jazykom Rust. Venoval som sa štúdiu Burrows-Wheeler Transform (BWT) a sufixových stromoch. Porozumel som riešeniu problému minimalizácie počtu runov, ako je popísané vo výskumnej práci s názvom "A New Class of Searchable and Provably Highly Compressible String Transformations."

Následne som sa pustil do testovania už hotovej implementácie. Už prvé naivné testy odhalili chyby pri hľadaní počtu minimálneho a zvláštne správanie pri tvorbe klasického bwt.

Napísal som si teda ďalšie teste pre ľahšie debuggovanie a podarilo sa mi opraviť nezvyčajné správanie pri klasickom BWT. Pokúsil som sa opraviť aj rátanie minimálneho počtu runov. Jeden z problémov sa ukázal byť s listami stromu kt. by mali po správnosti byť asociované so symbolom predchádzajúcim suffix kt. prislúchajú dané listy. Ďalej som implementoval IDs pre vrcholy suffixového stromu pre lepšie sledovateľnosť v DP. Do budúcna by sa to mohlo reimplementovať pomocou hashovania... Nakoniec sa mi pôvodný kód nepodarilo sprovozniť. Namiesto toho som na novo reimplemntoval celú časť s dynamickým programovaním.

Ku koncu semestra som ešte trochu zrefaktoroval kód pre lepšiu čitateľnosť a optimalizoval výpočet. Konkrétne zoptimalizovanie sa týkalo najmä dynamického výpočtu jednotlivých permutácii. Po optimalizácii sa spoločné časti rôznych permutácii detí toho istého vrcholu rátajú iba raz. Toto viedlo k cítiteľnému zrýchleniu (~20%).

Zároveň som implementoval možnosť vstupu s labels, pri ktorom sa minimalizuje počet runov lablov pridelených k listom suffixového stromu.