DSP HW3

By Fan-Yun Sun(b04902045)

Environment

- System: Ubuntu 16.04
- Compiler: g++ (Ubuntu 5.4.0-6ubuntu1~16.04.4) 5.4.0

How to execute

- `make`: compile mydisambig.cpp and link srlim library
- `make map`: execute mapping.py, generate ZhuYin-Big5.map from Big5-ZhuYin.map
- `make run`: run mydisambig on all testdata

Implementation

After reading useful srlim library files in `srlim/include`, I had a grasp on how to utilize their api and codes.

The structure of my program can be divided into three parts.

- Parsing arguments and reading in language model and mapping model
- Run DP(Viterbi) on a single sentence
- Backtrack to retrieve results

Discussion

- There are many useful constant in `Prob.h`, `Vocab.h` that can be easily utilized.
- For the DP table, use vector instead of fixed size array is better. There are two resons for this:
    - Most rows will have only one column. Using fixed size 2d array will be wasting a lot of space.
    - We don't have to know how many characters are there in the longest sentence and what is the maximum number of possible candidates.

Bug Lists

- Handling Vocab*SentStart and Vocab* SentEnd uncorrectly. Forget to add +1 at `mydisambig.cpp:33`.
- Beware of whether the array size is big enough if we use fixed-size 2d array for the dp table.