

Hw7

Hw7

- Q1: Viterbi algorithm

- `viterbi.sh` `input_hmm` `test_file` `output_file`
- `input_hmm`: the same format as in Hw6
- `test_file`: "`o1 o2 ... ok`"
- `output_file`: "`o1 o2 ... ok => x1 x2 ... xk+1 lgprob`"
- `lgprob` is $\lg P(o_1 o_2 \dots o_k, x_1 x_2 \dots x_{k+1})$
- The output symbols are generated by the **to-states**.
- Do not smooth the probs in `input_hmm`. They might have been smoothed already.
- You can reuse some code from `check_hmm.sh` in Hw6.
- You can assume that the `input_hmm` does NOT allow the emission of epsilon from a state.

Q2: run trigram POS taggers

- The trigram POS tagging models are provided to you as hmm1, ..., hmm5
- decoding: “w1 w2 ... => x1 x2 ... logprob”
 - `viterbi.sh hmm1 test.word sys1`
- convert format: “w1/t1 w2/t2 ...” : [you need to write conv_format.sh](#)
 - `cat sys1 | conv_format.sh > sys1_res`
- evaluation: [calc_tagging_accuracy.pl](#) is provided to you.
 - `calc_tagging_accuracy.pl test.word_pos sys1_res > sys1_res.acc 2>&1`

Table 1: Tagging accuracy

HMM model	tagging accuracy
hmm1	
hmm2	
hmm3	
hmm4	
hmm5	