Lehrstuhl Informatik 6
RWTH Aachen University
Prof. Dr.-Ing. H. Ney

Aachen, 08. 06. 2018

R. Schlüter,
K. Irie

Automatic Speech Recognition

3. Exercise

The objective of this exercise is to get hands-on experience with the word graph. Please use Python to solve the programming tasks and send your source code to `irie@i6.informatik.rwth-aachen.de` on 15. 06. 2018 at the beginning of the exercise session.

Task 3.1 Word graph and forward-backward pruning algorithm

In the files `lattice.*.htk` you find five word graphs (*lattice* is a synonyme for word graph) for English broadcast news data. The word graph file is in the common *HTK* format, whose description can be found under http://www.ee.columbia.edu/ln/rosa/doc/HTKBook21/node257.html

The word graphs were generated using a bigram language model, its arcs are annotated with the corresponding bigram language model and acoustic emission probabilities, see above format definition for details.

In addition, the reference transcription file `transcriptions.txt` in *STM* format and an example of a recognition hypothesis `example.ctm` in *CTM* format are provided.

The scoring toolkit `SCLITE` (part of `SCTK`) needs to be downloaded from:
http://www.nist.gov/itl/iad/mig/tools.cfm

Please run a test:
`./sclite` -e utf-8 -r `transcriptions.stm` stm -h `example.ctm` ctm -o all -O YourOutputDir
The word error rate can be found in the `*.sys` file (for this example, you should get 7.6%).

(a) The word graph density is defined as the total number of arcs divided by the number of actually spoken words. Write a simple script to compute the word graph density (WGD) from the word graph, using the reference transcription. Apply it to the provided word graphs. (1)

(b) Implement the forward-backward algorithm to compute the probability of the best path through each edge of the word graph. Please do not forget to consider the language model scaling factor (LM-scale). (10)

(c) Extract the best scoring hypotheses from the provided word graphs. Compute the word error rate using the scoring script. (3)

(d) Apply forward-backward pruning to prune the word graphs to about half of their original word graph density. (3)

(e) The LM-scale written in the lattices is not optimal. Modify (if necessary) the implementation of (b) and implement a simple grid search algorithm to find the optimal LM-scale. Report the optimal word error rate you observe using this grid search. (3)