STATISTICAL STRUCTURED PREDICTION

Question set (Part 1-B)

December 2022

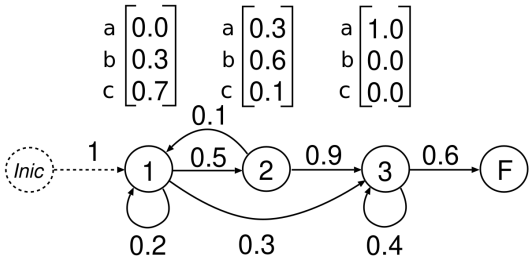
Administrative issues

- This part represents the fourth part of the final score of PEE.
- The score of this part consists of 60% of theoretical questions and 40% of practical assignments.
- The score of each exercise will depend on the depth, quality and justification of the answer both in the theoretical exercises and practical exercises.
- Please present the results in PDF documents and uplodad your document through a poliformat task.
- Deadline for delivering the theoretical questions and the report of practical assignment: **16 Jan 2023**

Theoretical questions

Question 1 (4 points)

Given the following HMM:



and the training sample $\mathcal{D} = \{ccca, ba\}$, apply the expression that appears in page 26 of P-I.4 (adapted to HMM) to estimate the HMM parameters, only one iteration. Note that you have to consider all paths that account for each string that belongs to $\mathcal{D}$. Show the computations only for the probability transition between state 1 and 2. Then, plot the resulting HMM after estimating all parameters.

Question 2 (2 points)

In the previous exercise, some parameters become 0 because the sample does not allow to use all edges and/or emission. Add a new string to the sample $\mathcal{D}$ to avoid this issue. Justify your answer adequately.

Practical questions

This practical part is devoted to develop a simple parser of bibligraphic cites. The parser is based on hidden Markov models (HMM). The parser accepts a bibligraphic cite and generates a bib entry with the tokens distributed among the corresponding fields (author, title, etc.). The parser doest not distiguish among different types of entries (@book, @article, etc.).

The student has a toolkit that has been developed for this purpose. The toolkit includes a program for estimating a HMM with the forward-backward algorithm and with the Viterbi algorithm. The program also performs the parsing by accepting bibliographic cites and generating bib entries. For example, given the following cite:

```
J. Kupiec Hidden Markov Estimation for Unrestricted Stochastic Context-Free
Grammars Proc. of ICASSP'92 Vol. 1 1992 177 180
```

the output should be

```
@inproceedings{Kupiec92,
  author =       {J. Kupiec},
  title =        {Hidden Markov Estimation for Unrestricted Stochastic
                  Context-Free Grammars},
  booktitle =    {Proc. of ICASSP'92 Vol. 1},
  year =         1992,
  pages =        {177 180}
}
```

But in these exercises the string `@inproceedings{Kupiec92,` and the final `}` are not generated.

The following commands and files are included in the toolkit.

```
m0
trainHMM
testHMM
nameStates
hmm2dot.sh
```

The first file is an initial model that is not still trained. The second file is a set of training samples and the third file is a set of test sampples. The second and third files have been simplified by removing some punctuation marks. The fourth file is a mapping between states of the HMM and type of entries in a bibligraphic cite. Have a look to these files. The last one will be describe below.

The program:

```
hmml -h

Usage: hmml [-h] -i mI [-S sF] [-l sZ] [[-v] [-o mO] [-I ite] [-s sm]] | [-D]] [-V n]

-i mI input model file
-S sF sample file (stdin by default)
-l sZ use strings up to length sZ (60 by default)
-v Viterbi learning (Bawm-Welch by default)
-o mO output model file (mI by default)
-I ite number of traing iterations (1 by default)
-s sm smooth value for emission probabilities (float, not by default)
-D decoding (useless in training mode)
-V n verbosity level (0 by default, 3 max)
-h this help

If option -D is used, then options -v, -o, and -I are ignored.
```

is used to train and parse samples. The use for training is as follows:

```
hmml -i m0 -S dataHMM  -o m1 -I 3 -s 0.00001
-1.387600e+02 215
-1.367631e+02 215
-1.366668e+02 215
```

The output shows the normalized loglikelihood of the training sample. The second value is the number of training samples that has been used. The parsing of a file that contains the previous sentence is carried out as follows:

```
hmml -i m0 -S dataTest -D
```

and the output is:

```
J.@0 Kupiec@0 Hidden@1 Markov@1 Estimation@1 for@1 Unrestricted@1 Stochastic@1
Context-Free@1 Grammars@1 Proc.@4 of@4 ICASSP'92@4 Vol.@4 1@8 1992@2 177@8 180@8
```

the token before the symbol `@` corresponds to the input and the integer after the symbol `@` corresponds to the state in which the token has been observed. These integers are mapped into bibligraphic fields as indicated in the `nameStates` file. For example `J.@0 Kupiec@0` means that tokens `J. Kupiec` have been generated in state `author`.

Question 3 (1 points)

File `testHMM` contains three additional cites. You have to parse these cites with `hmml` with model `m0` and write the output in bib format as in the previous examples.

Question 4 (1 points)

Train model `m0` as in the previous example with 10 iterations. Then, parse the same cites included in `testHMM` with model `m1` and write the output in bib format as in the previous examples. Justify the changes with respect to the previous exercise.

Question 5 (2 points)

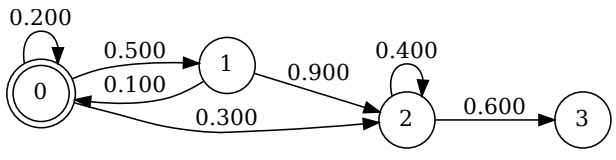
The model in Question 1 can be written in a file as:

```
HMM 4 3
1.0 0.0 0.0
0.2 0.5 0.3 0.0
0.1 0.0 0.9 0.0
0.0 0.0 0.4 0.6
0.0 0.3 1.0 0.0 a
0.3 0.6 0.0 0.0 b
0.7 0.1 0.0 0.0 c
```

Then, the HMM we be can plot (only the transitions) as follows:

```
hmm2dot.sh t0 > t0.dot
dot -Tpdf t0.dot -o t0.pdf
```

and we get



You hav to plot the HMMs obtained in Question 1 and Question 2 after the training process.