Communication Technologies 1 (CT1)

Machine Learning

Viterbi Algorithm

Akancha Choudharry, Philipp Copei, Robert Koczula, Tobias Schellien

Lecture in SS 2019
19.09.2019

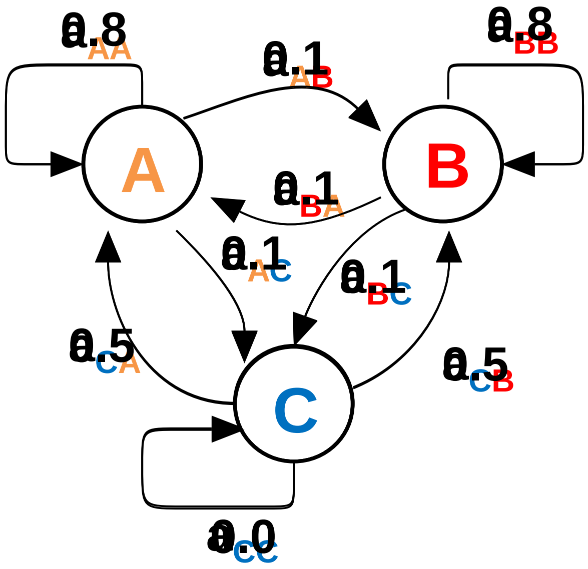
Agenda

- Introduction
- Viterbi Algorithm
- Experiment
- Summary
- Questions

Introduction

- Algorithm to recreate state sequences
- Example student travels between rooms

Marcov chain

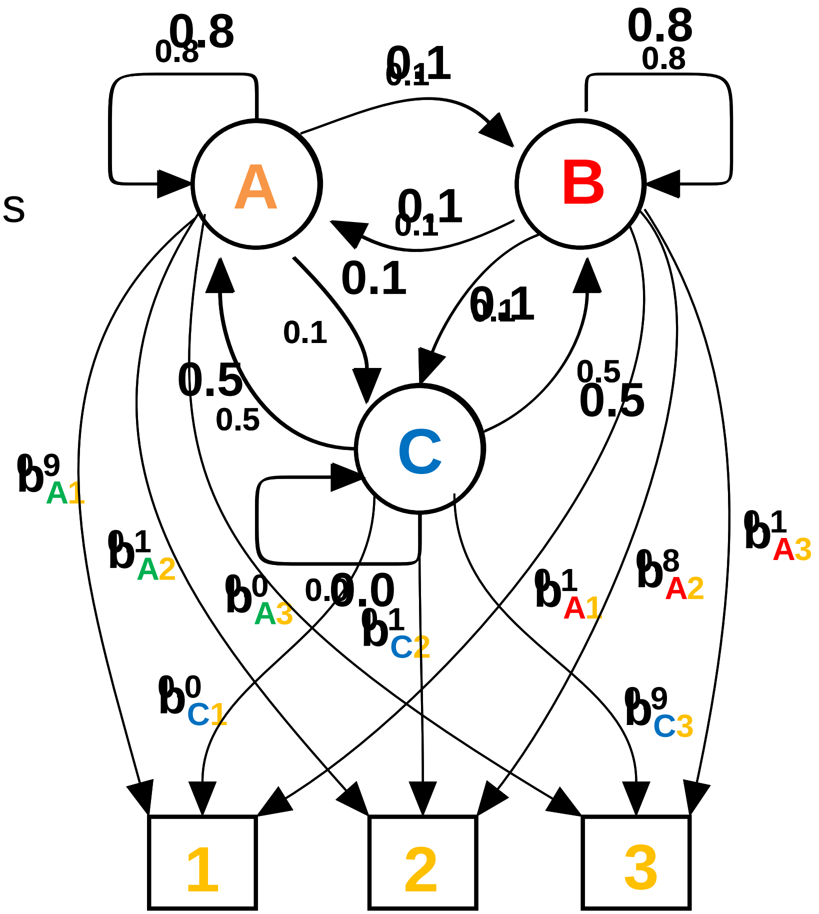


a_{ij} : transition probability from one state to the next

Introduction – Hidden Markov Model (HMM)

- State space S={A, B, C}
- Initial probability of states
- Transition matrix A that store the transition probability between true states

- Observation space O={1, 2, 3}
- Emission matrix B that stores the transition probability from true states to observing states
- Sequence of observations $y_1, y_2, .., y_T$



b_{ij} : probability of an observation for a hidden state

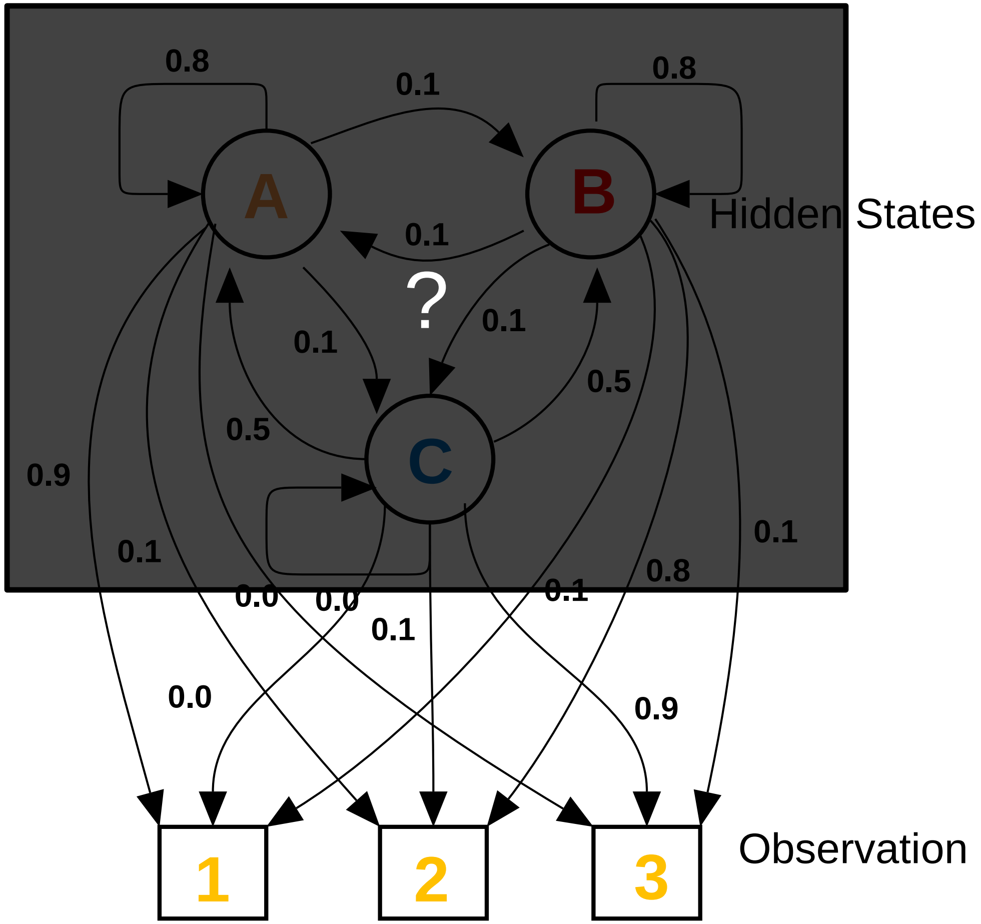
Introduction - Observations

Problem:
We don‘t know the true states of a HMM

Given:
HMM with observed states and transition probabilities

Target:
Estimate the true state (hidden state)

How:
Use the observations of the system
-> emission is visible and gives a stochastically hint



Viterbi Algorithm

- Invented by Andrew J. Viterbi in 1967
- Input via HMM and state sequence
- Calculates path with highest probability
- Used for decodation of convolutial codes
- Extended by G. D. Forney for reduction of communication errors

Viterbi Algorithm - Pseudo Code 1/3

- Given: a sentence of length n
- Matrix ϑ storing highest reachability probabilites
- Matrix ψ storing contributing states for reachability
- Initialization:

```
for all states i do
        $\vartheta_1(i) = \pi_i * B_{i1}$
        $\psi_1(i) = 0$
end
```

Viterbi Algorithm - Pseudo Code 2/3

- Induction:

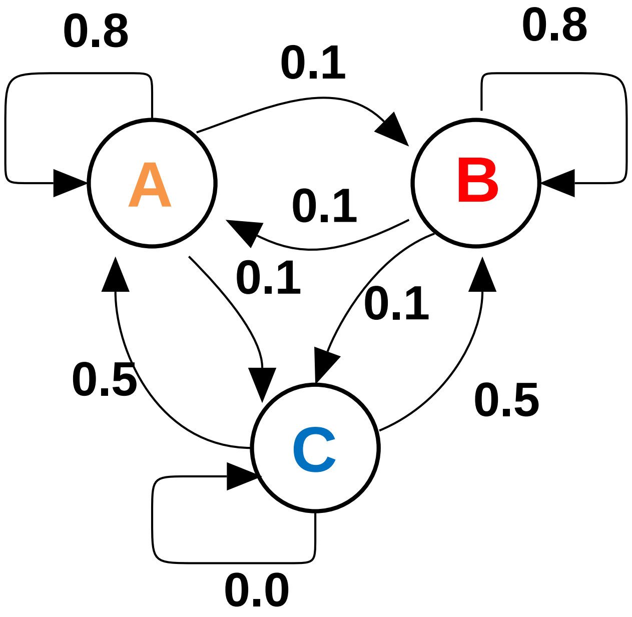
```
for t := 2 to n step 1 do
        for all states i do
                $\vartheta_t(i) = B_{it} * \max_{1 \leq j \leq |S|} \left(A_{ji} * \vartheta_{t-1}(i)\right)$
                $\psi_t(i) = \arg \max_{1 \leq j \leq |S|} \left(A_{ji} * \vartheta_{t-1}(i)\right)$
        end
end
```

Viterbi Algorithm - Pseudo Code 3/3

- Termination and path-readout:

```
$y_T = \arg \max_{1≤j≤|S|} (\vartheta_T(j))$
for j := n to 1 step -1 do
        $y_t = \psi_{t+1}(y_{t+1})$
end
$P(y_1, \dots, y_T) = \max_{1≤j≤|S|} (\vartheta_n(j))$
```

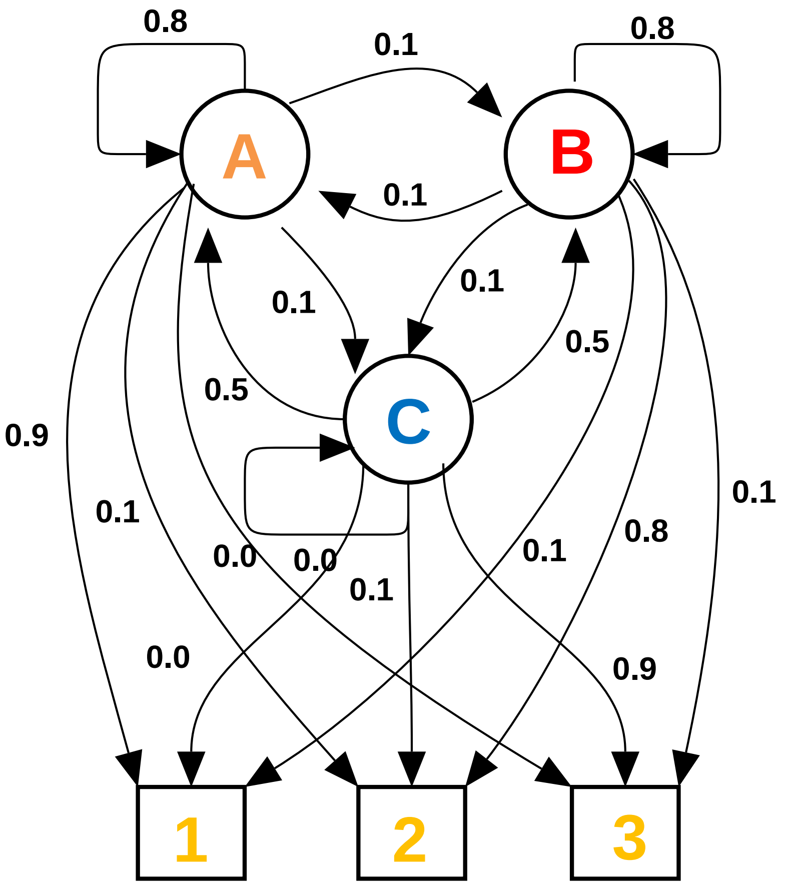
Experiment



N=55

```
AAAAAABBBCBBBBBBCBCAAAAAABBBBBAAABBBBAAAAAAACAAAAAAAAAA
1111111223222222323111112322231112222121111131221211111
```

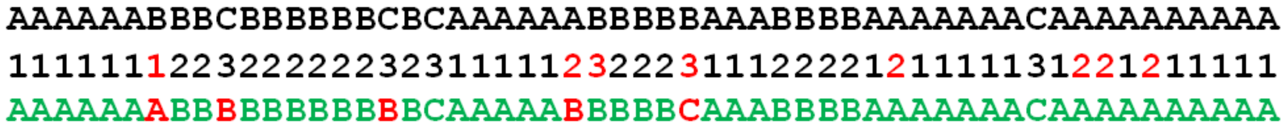
Experiment - Used Sequence



	1	2	3
A	0.9	0.1	0.0
B	0.1	0.8	0.1
C	0.0	0.1	0.9

```
AAAAAABBBCBBBBBBCBCAAAAAABBBBBAAABBBBAAAAAAACAAAAAAAAAA
1111111223222222323111112322231112222121111131221211111
```

Experiment - Recreated Sequence



```
AAAAAABBBCBBBBBBCBCAAAAAABBBBBAAABBBBAAAAAAACAAAAAAAAAA
1111111223222222323111112322231112222121111131221211111
AAAAAAABBBBBBBBBBBCAAAAABBBBBCAAABBBBAAAAAAACAAAAAAAAAA
```

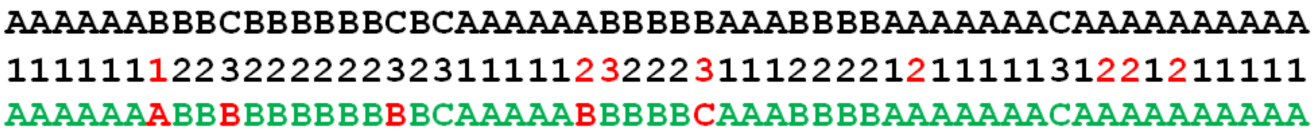
© ComTec 2019

Summary

- Applying viterbi algorithm in real time case.
- We observe the relation between true state and hidden state.
- We also observe the downside of the algorithm.
- Student travelling between rooms.
- Use result for analyzing and reducing error.

Summary

- Applying viterbi algorithm in real time case.
- We observe the relation between true state and hidden state.
- We also observe the downside of the algorithm.
- Student travelling between rooms.
- Use result for analyzing and reducing error.



```
AAAAAABBBCBBBBBBCBCAAAAAABBBBBAAABBBBAAAAAAACAAAAAAAAAA
1111111223222222323111112322231112222121111131221211111
AAAAAAABBBBBBBBBBBCAAAAABBBBBCAAABBBBAAAAAAACAAAAAAAAAA
```

Questions?

Sources

- Pseudo Code: kontext.fraunhofer.de/haenelt/kurs/folien/Haenelt_Viterbi-Algorithmus.pdf
- Used Images: https://forschung.bissantz.de/hidden-markov-modelle-so-bekommt-man-zustaende
- [1] https://de.wikipedia.org/wiki/Viterbi-Algorithmus, 2018.
- [2] S. Alhaidari and M. Zohdy. Network anomaly detection using two-dimensional hidden markov model based viterbi algorithm. April 2019.
- [3] A. R. Arsadjaja and A. I. Kistijantoro. Online speech decoding optimization strategy with viterbi algorithm on gpu. 2018.
- [4] J. Bobbin. An incremental viterbi algorithm. 2017.
- [5] M. S. I. Chen Yuan and A. S. Khalsa. Modified viterbi algorithm based distribution system restoration strategy for grid resiliency. 2017.

Sources

- [6] G. C. D. Hernando, V. Crespi. Efficient computation of the hidden markov model entropy for a given observation sequence. June 2005.
- [7] G. Forney. The viterbi algorithm. 1973.
- [8] B. . C. GmbH. Hidden-markov-modelle: So bekommt man Zustände! December 2014.
- [9] K. M. Kilavo Hassan and S. I. Mrutu. Performance of soft viterbi decoder enhanced with non-transmittable codewords for storage media. ELECTRICAL and ELECTRONIC ENGINEERING, 2018.
- [10] E. G. S.-T. lecture WS 2012/13 Universitity in Jena. Chapter 5 Slide 39 ff. www.minet.uni-jena.de/fakultaet/schukat/MAS/Scriptum/lect05-HMM.pdf, 2019.
- [11] J. H. Ritendra Datta and B. Ray. On efficient Viterbi decoding for hidden semi-markov models. December 2008.
- [12] S. Singhal and M. Gilani. https://www.eetimes.com/document.asp?doc id=1277544#, 2002.

Sources

- [13] R. Yazdani, A. Segura, J.-M. Arnau, and A. Gonz´alez. Low-power automatic speech recognition through a mobile gpu and a viterbi accelerator. 2017.
- [14] G. Yin and D. Bruckner. Data analyzing and daily activity learning with hidden markov model. November 2010.
- [15] https://www.youtube.com/watch?v=6JVqutwtzmo.