Data Mining and Machine Learning

Speech Recognition using HMMs – Viterbi Decoding

Peter Jančovič

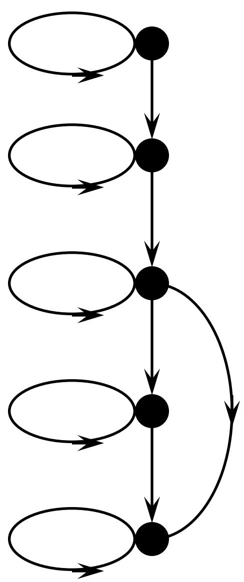
UNIVERSITY OF BIRMINGHAM

Viterbi Decoding

- Viterbi Decoding is the algorithm which is used to find the sequence of HMM states (or HMMs) which is most likely to have generated a given observation sequence
- Similar to the Forward Probability calculation

Viterbi Decoding (1)

y_1 y_2 y_3 y_4 y_5 y_{t-1} y_t y_T

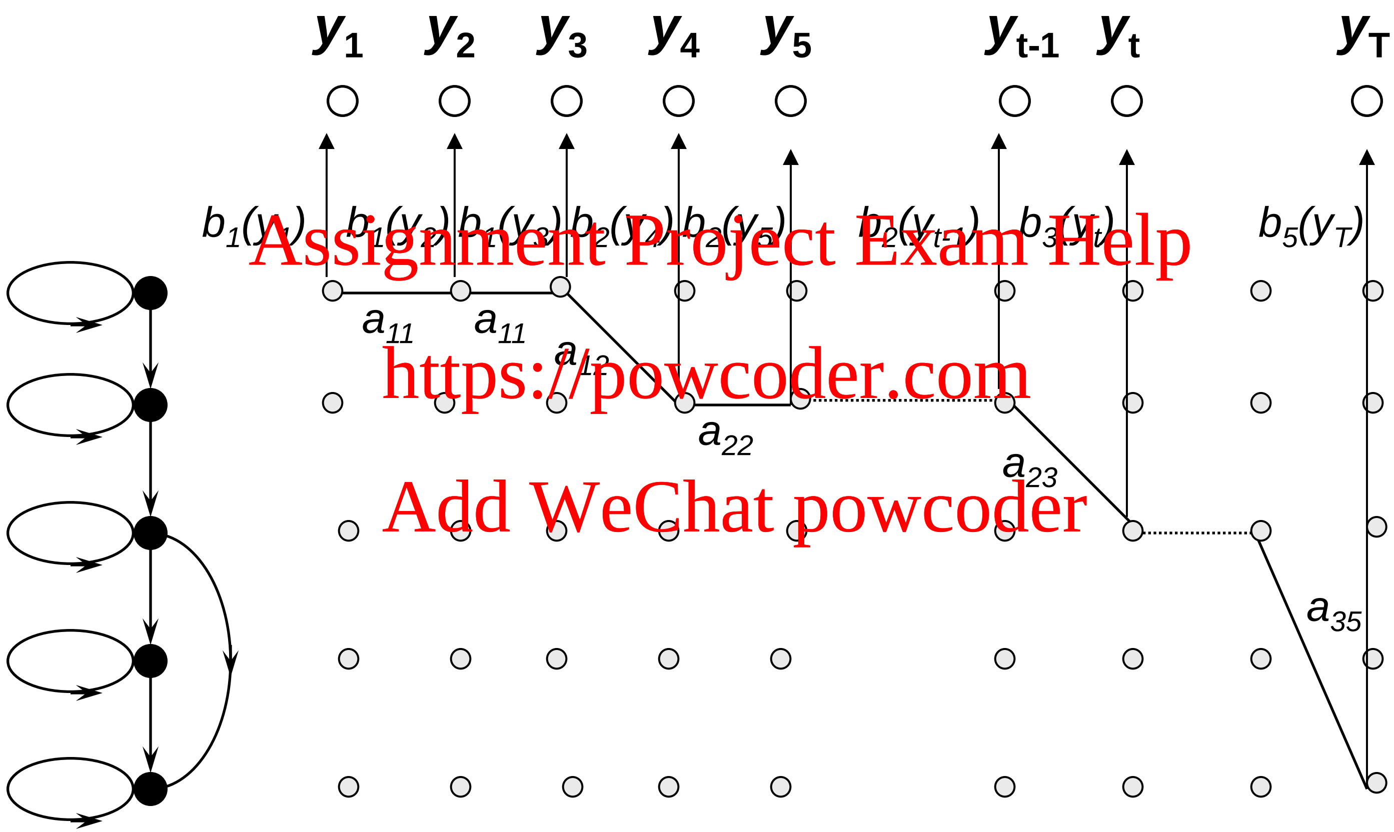


Q: How can *M* have generated *Y*?

A: Via a state sequence of length T

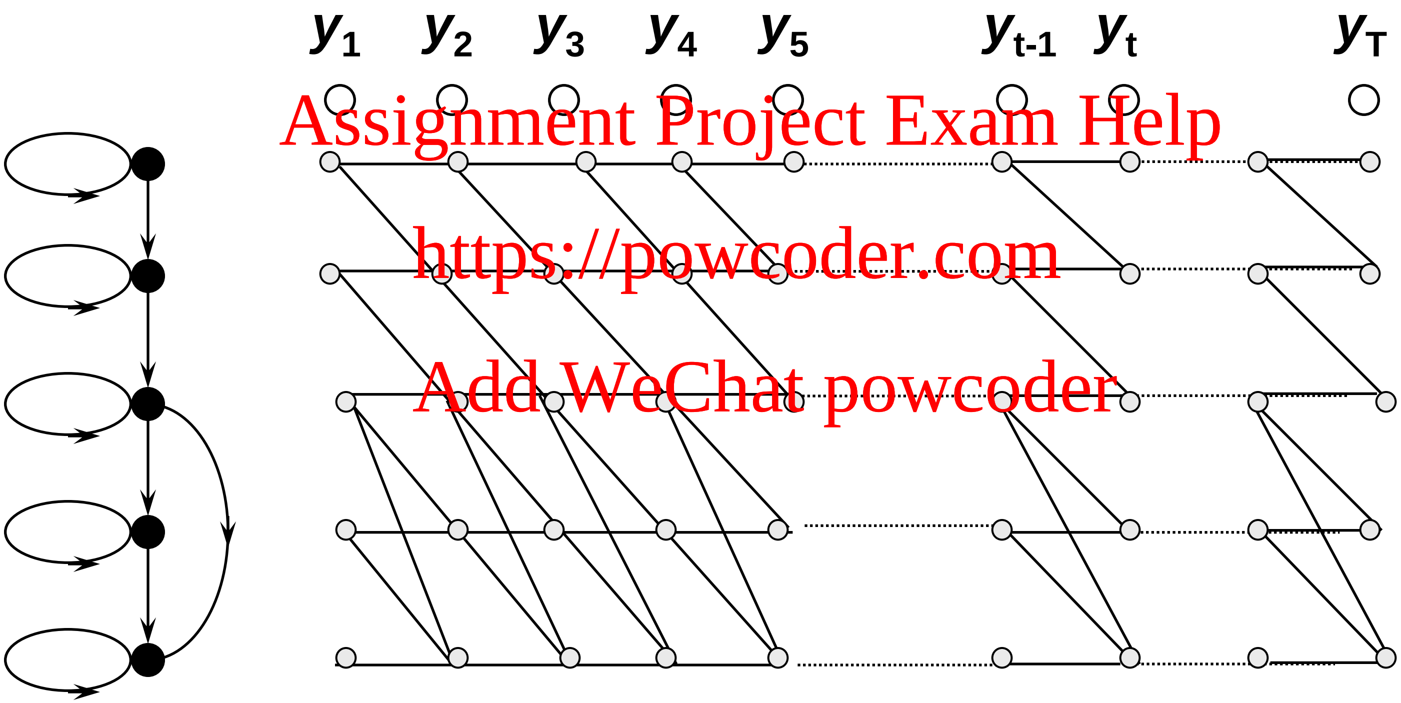
UNIVERSITY OF BIRMINGHAM

Basic Probability Calculation



Viterbi Decoding (2)

- **Construction of ‘state-time trellis’**



Viterbi Decoding (3)

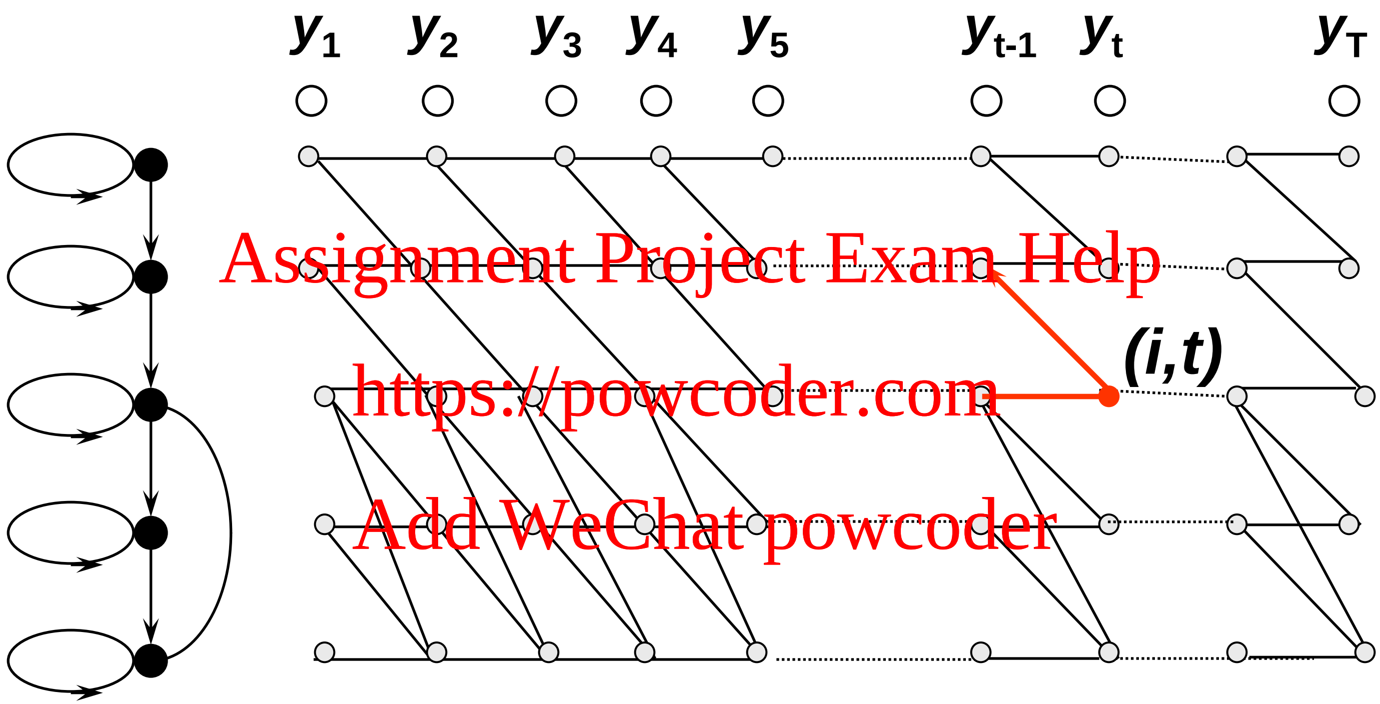
- Let $X = \{x_1, \ldots, x_T\}$ be a state sequence of length T
- The joint probability of Y and X is given by:

$$p(Y, X) = b_{x_1}(y_1) \prod_{t=2}^{T} a_{x_{t-1} x_t} b_{x_t}(y_t)$$

- i.e. the product of the state-output and state transition probabilities along the state sequence

- The <u>optimal</u> state sequence is the sequence X such that $p(Y,X)$ is maximized
- $p(Y)$ is the sum of $p(Y,X)$ over all sequences X

Viterbi Decoding (4)

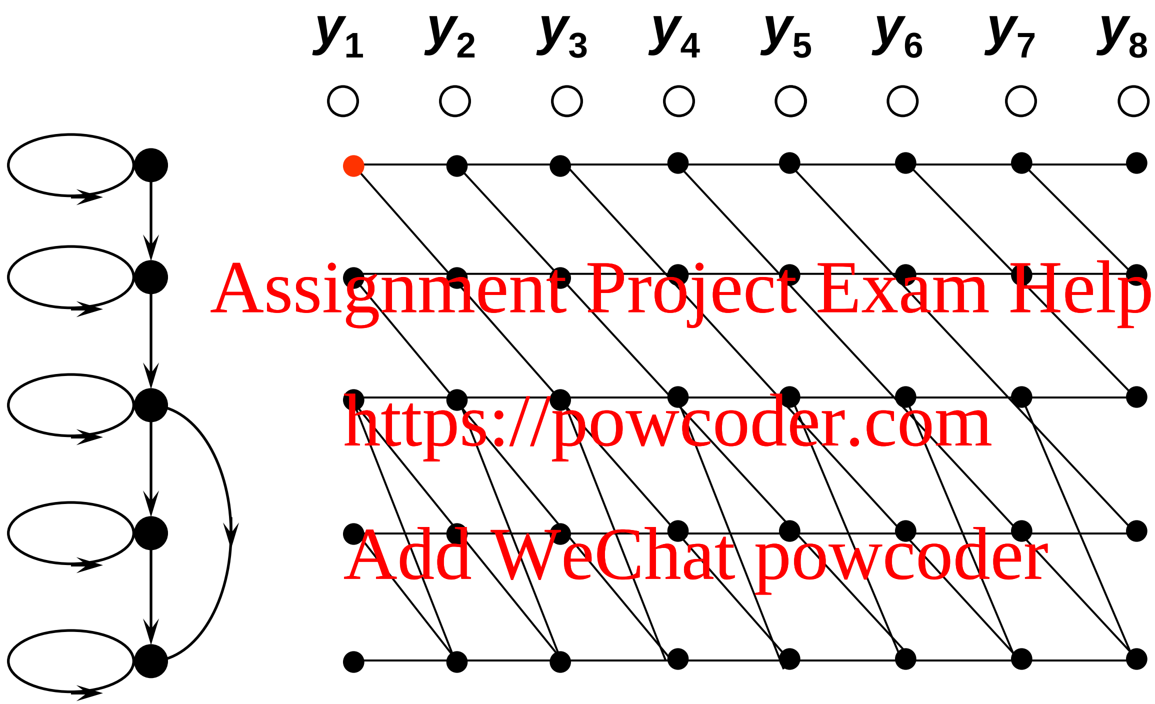


$$\alpha_t(i) = \text{Prob}(y_1,\ldots,y_t \text{ , opt sequence to } (i,t))$$

$$\alpha_t(i) = \max \{\alpha_{t-1}(i-1)a_{i-1,i} \text{ , } \alpha_{t-1}(i)a_{i,i} \}\, b_i(y_t)$$

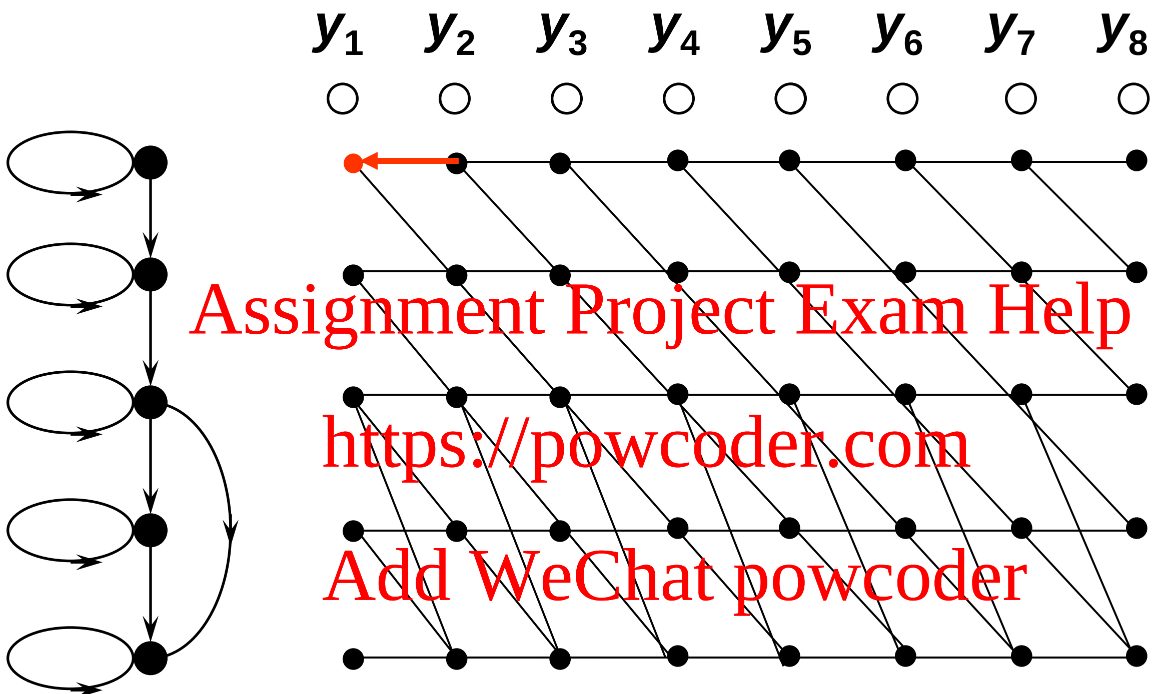
UNIVERSITY OF BIRMINGHAM

Viterbi Decoding – example



$$\alpha_1(1) = b_1(y_1)$$

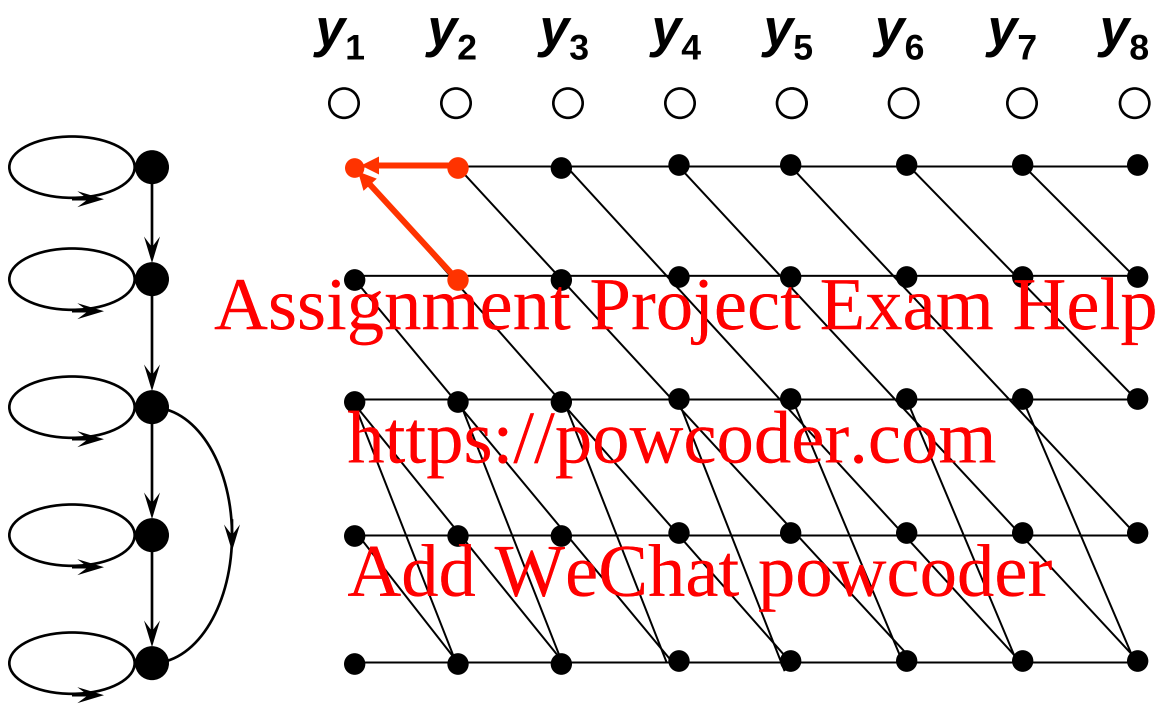
Viterbi Decoding – example



$$\alpha_2(1) = \alpha_1(1) a_{11} b_1(y_2)$$

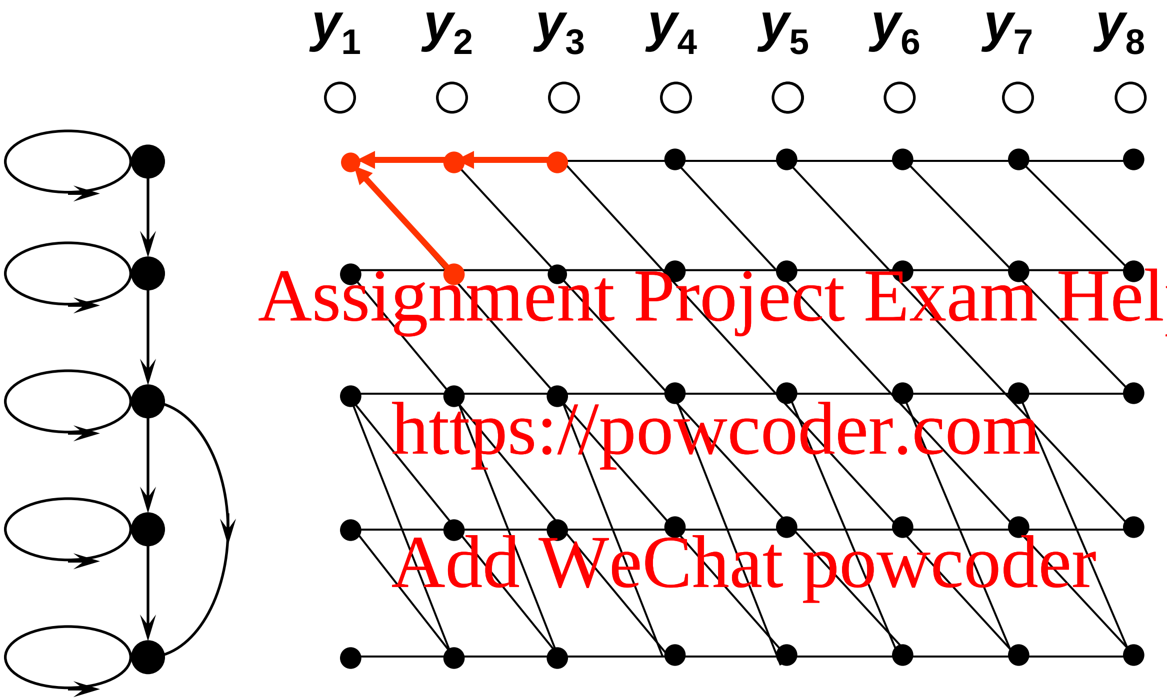
UNIVERSITY OF BIRMINGHAM

Viterbi Decoding – example



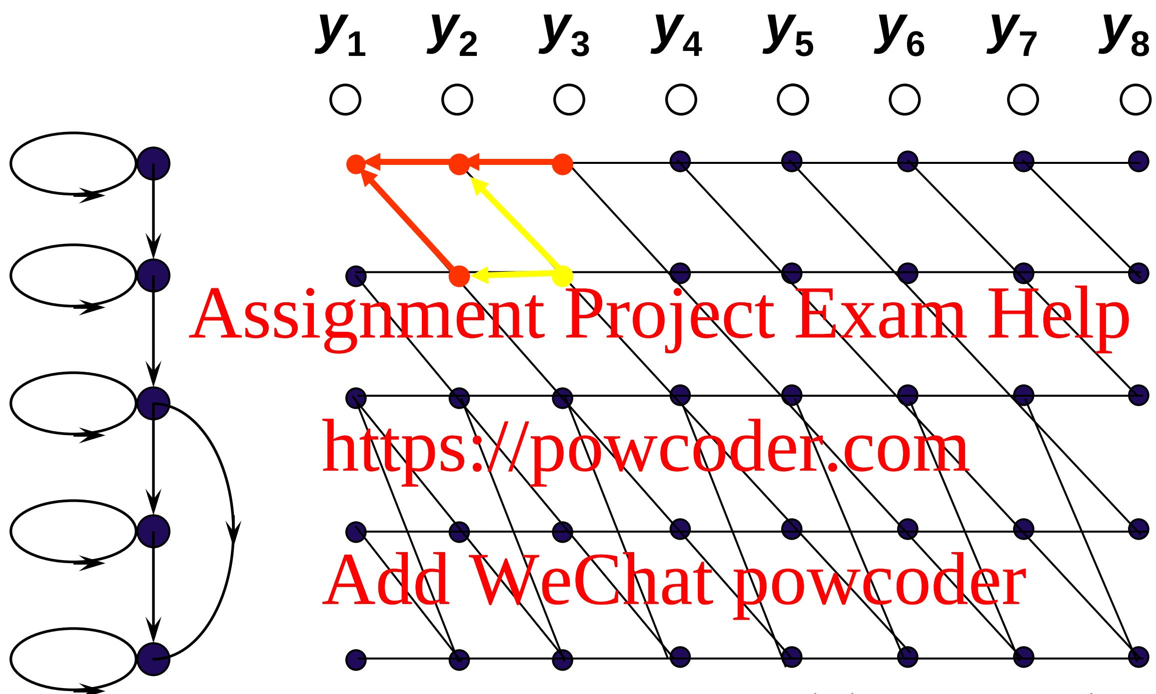
$$\alpha_2(2) = \alpha_1(1) a_{12} b_2(y_2)$$

Viterbi Decoding – example



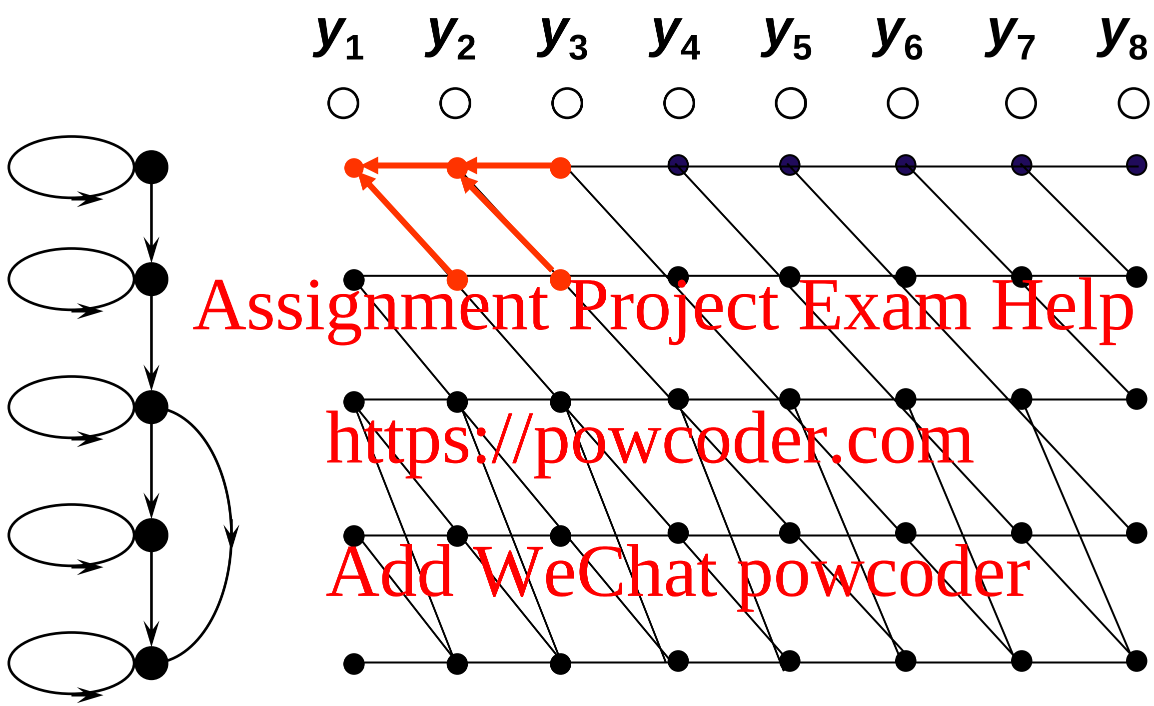
$$\alpha_3(1) = \alpha_2(1) a_{11} b_1(y_3)$$

Viterbi Decoding – example



$$\alpha_3(2) = \max \begin{cases} \alpha_2(1) a_{12} b_2(y_3) \\ \alpha_2(2) a_{22} b_2(y_3) \end{cases}$$

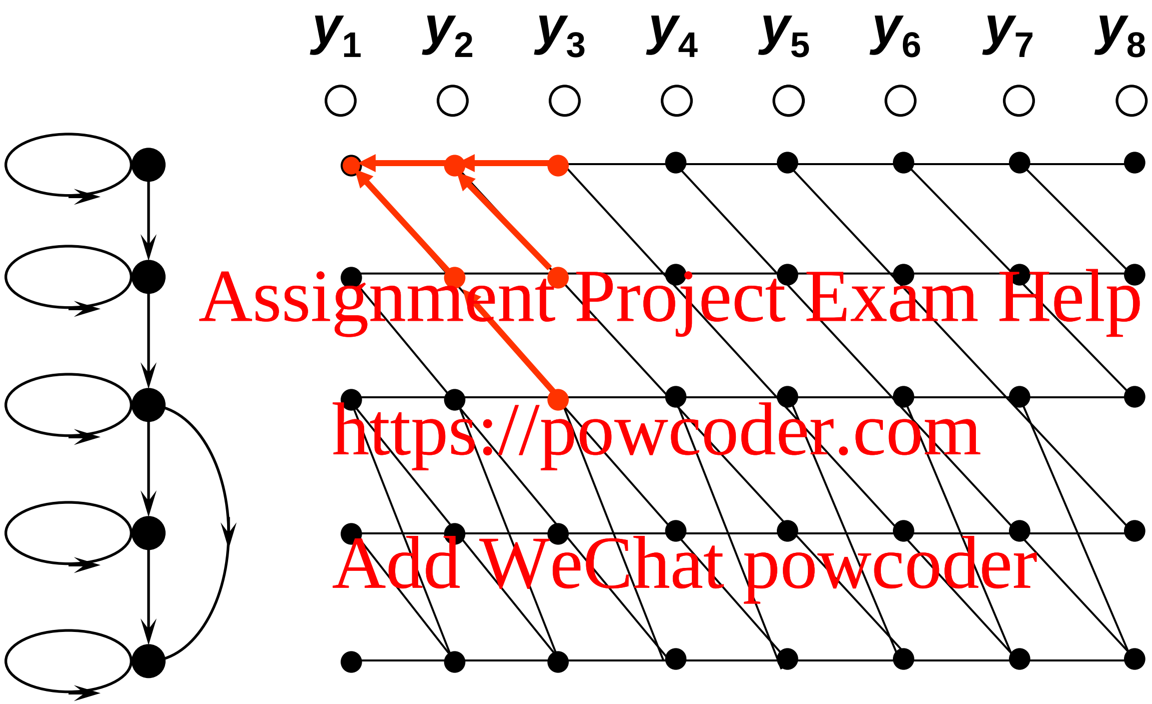
Viterbi Decoding – example



$$\alpha_3(2) = \max \begin{cases} \alpha_2(1) a_{12} b_2(y_3) \\ \cancel{\alpha_2(2) a_{22} b_2(y_3)} \end{cases}$$

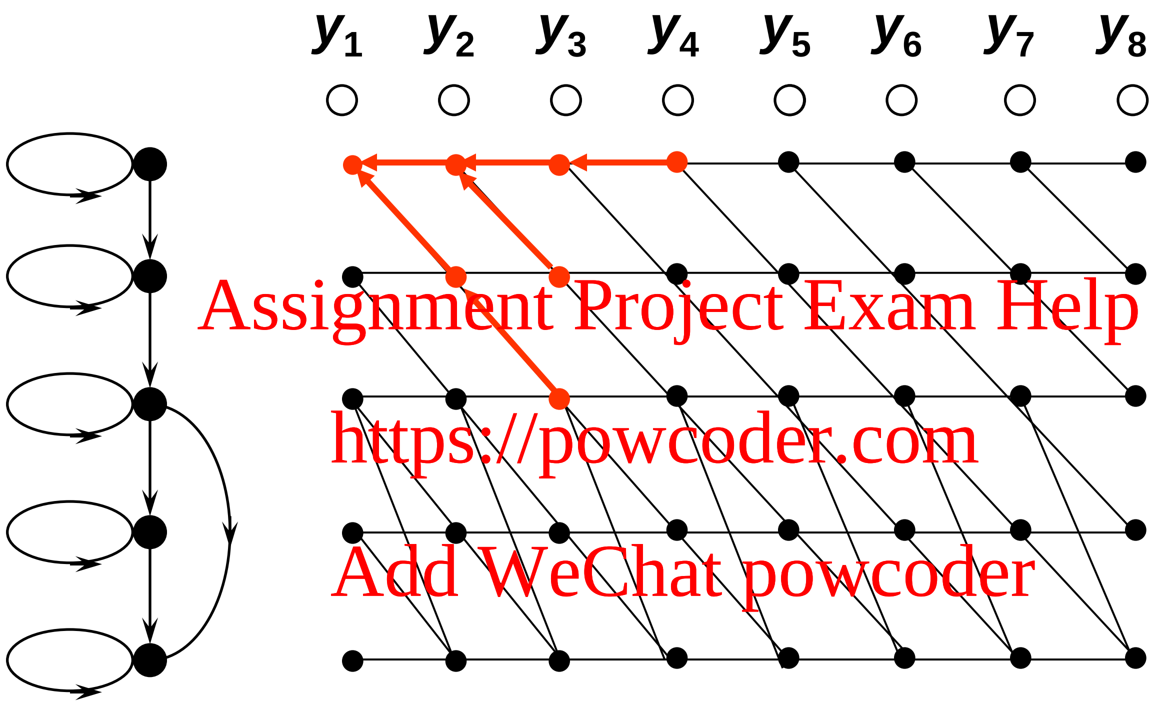
UNIVERSITY OF BIRMINGHAM

Viterbi Decoding – example

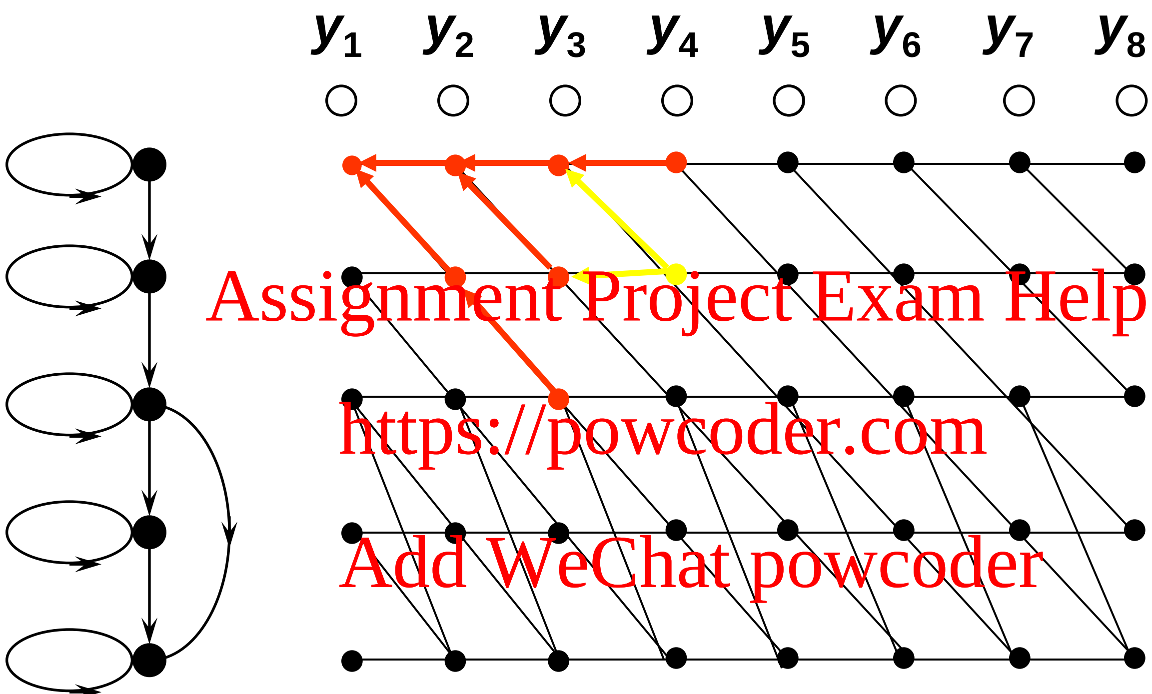


$$\alpha_3(3) = \alpha_2(2)a_{23}b_3(y_3)$$

Viterbi Decoding – example

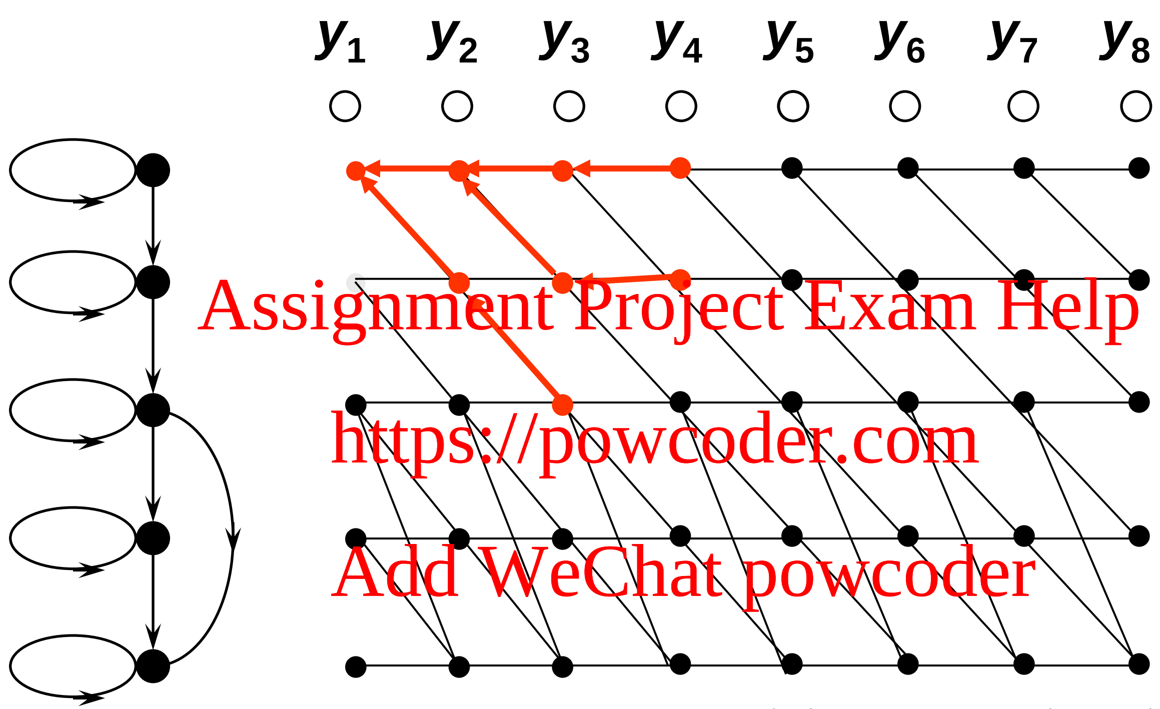


Viterbi Decoding – example



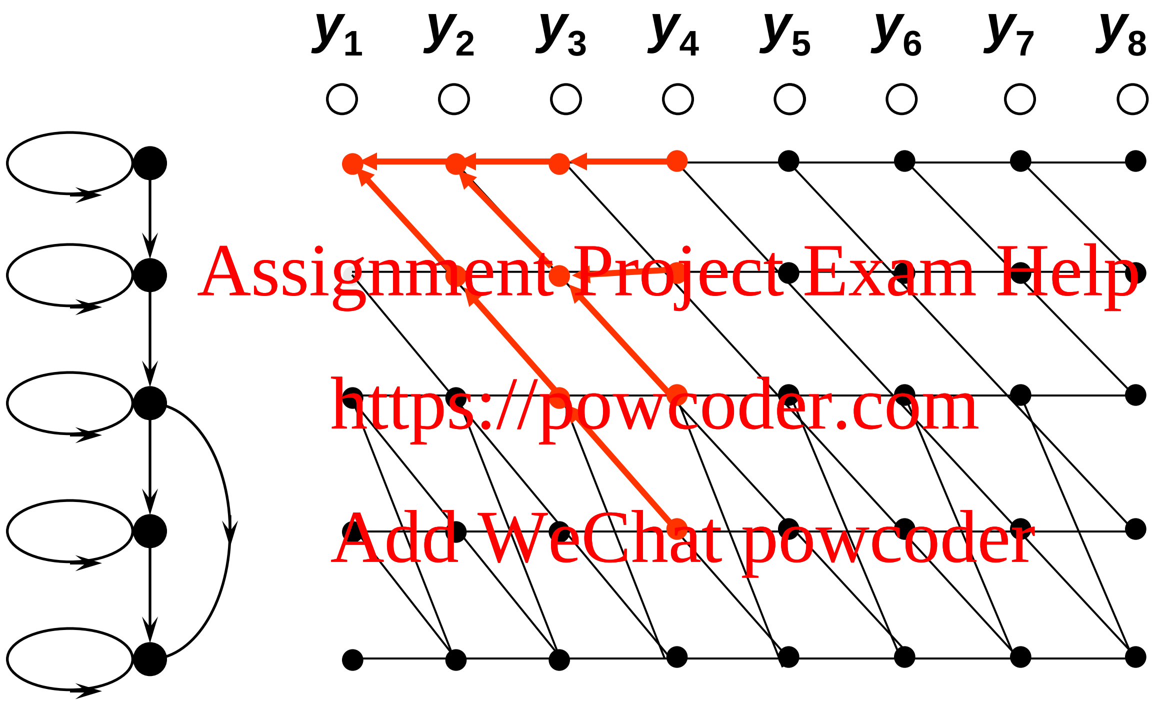
$$\alpha_4(2) = \max \begin{cases} \alpha_3(1) a_{12} b_2(y_4) \\ \alpha_3(2) a_{22} b_2(y_4) \end{cases}$$

Viterbi Decoding – example

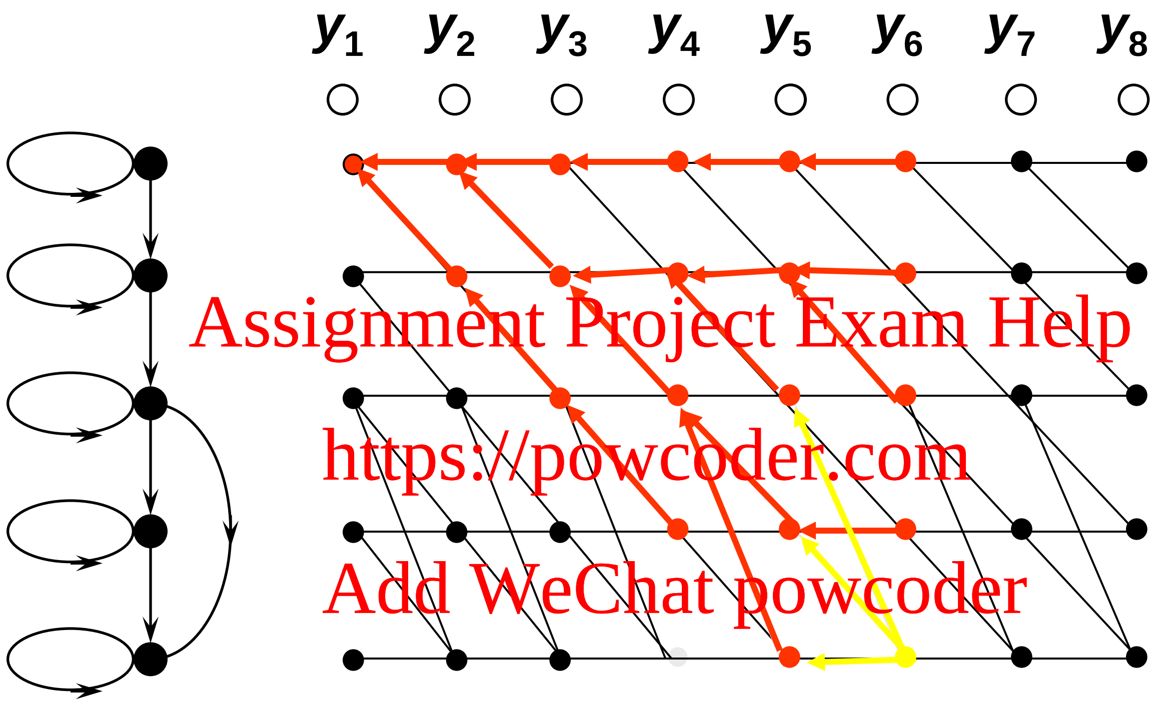


$$\alpha_4(2) = \max \begin{cases} \cancel{\alpha_3(1) a_{12} b_2(y_4)} \\ \alpha_3(2) a_{22} b_2(y_4) \end{cases}$$

Viterbi Decoding – example

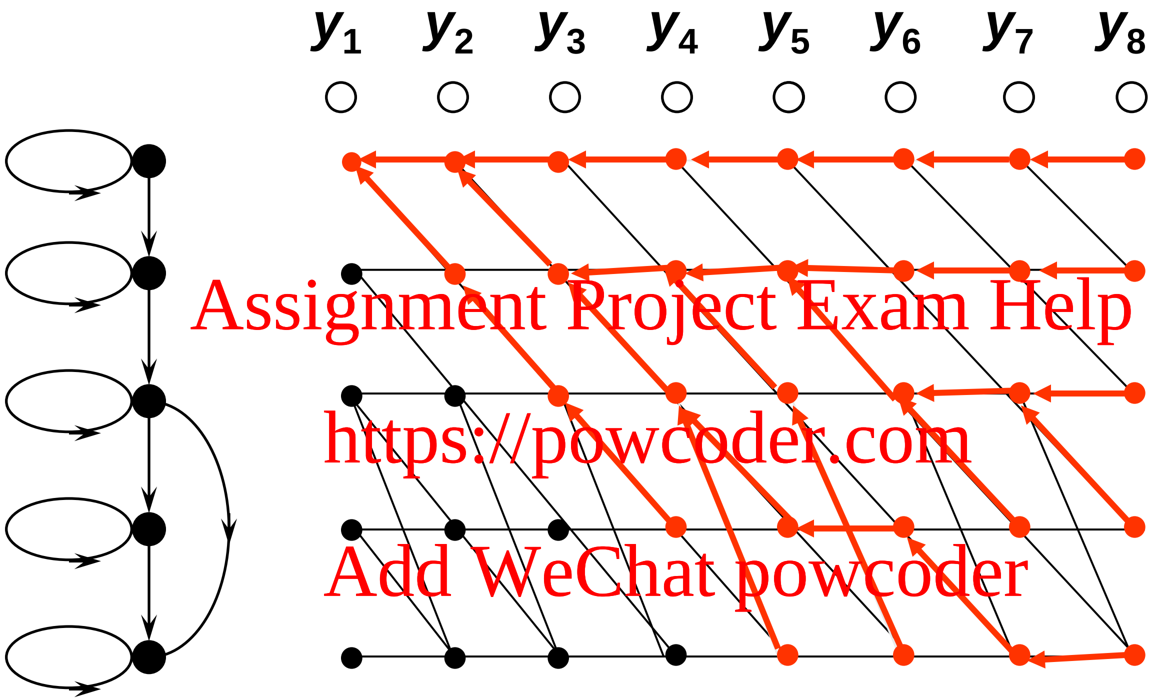


Viterbi Decoding – example

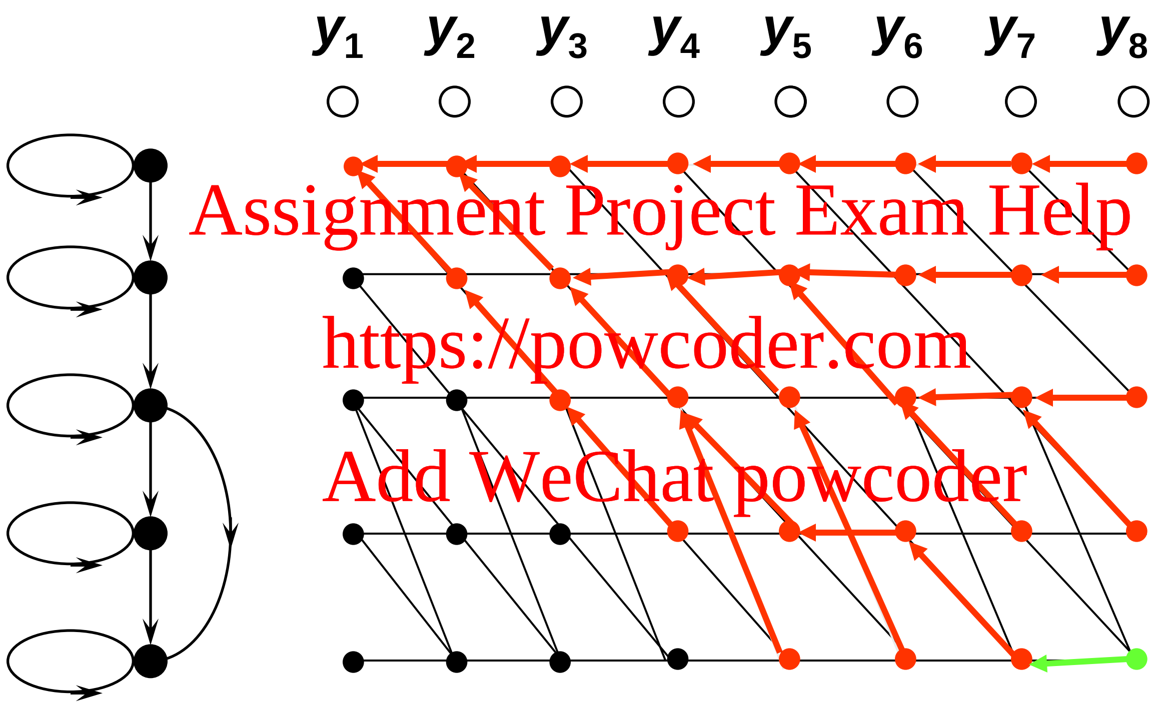


$$\alpha_6(5)=\max\begin{cases}\alpha_5(5)a_{55}b_6(y_6)\\ \alpha_5(4)a_{45}b_6(y_6)\\ \alpha_5(3)a_{35}b_6(y_6)\end{cases}$$

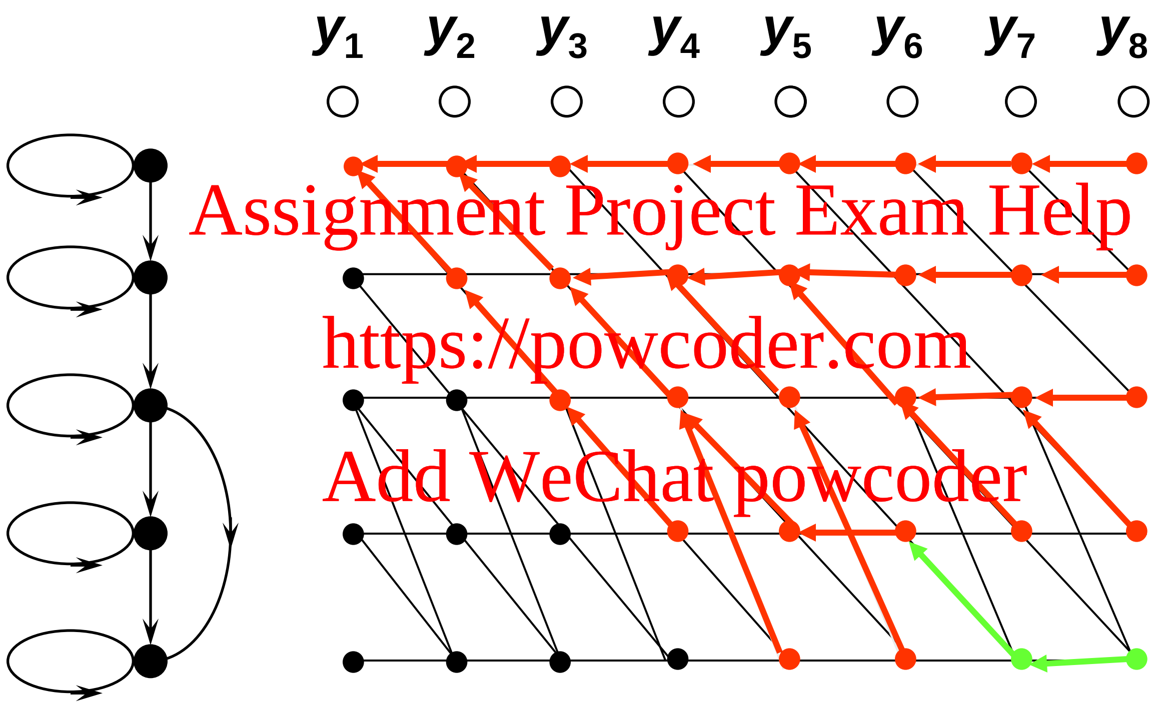
Viterbi Decoding – example



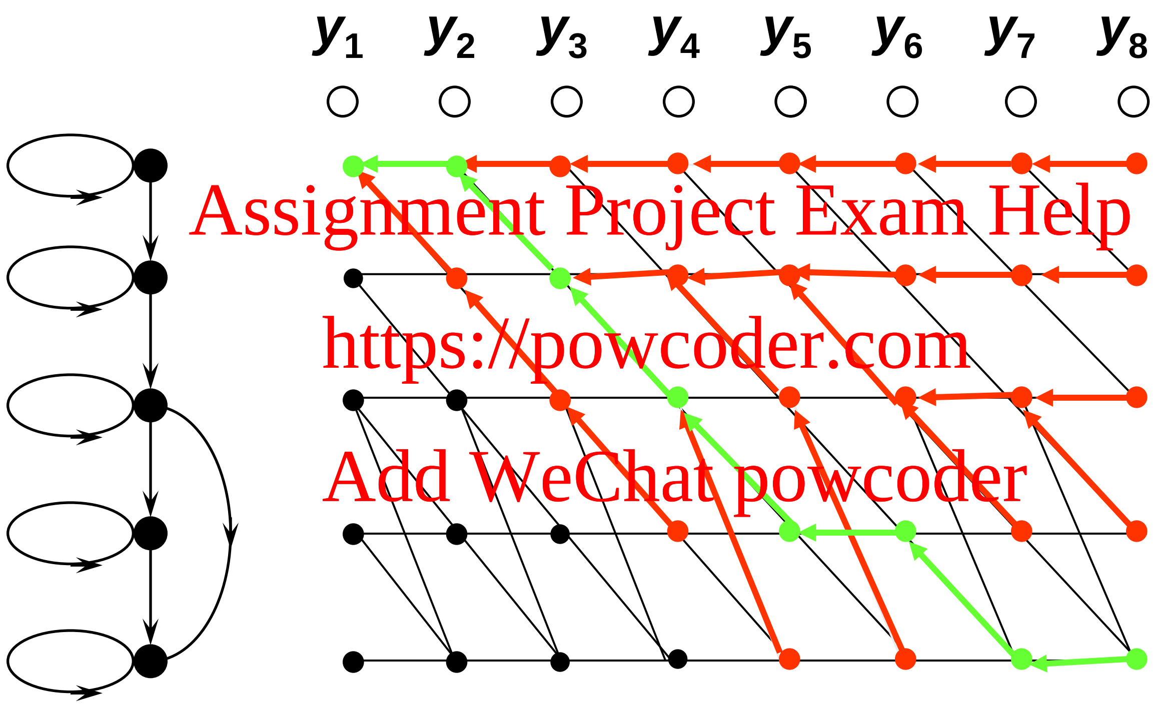
Viterbi Decoding – Trace-back



Viterbi Decoding – Trace-back

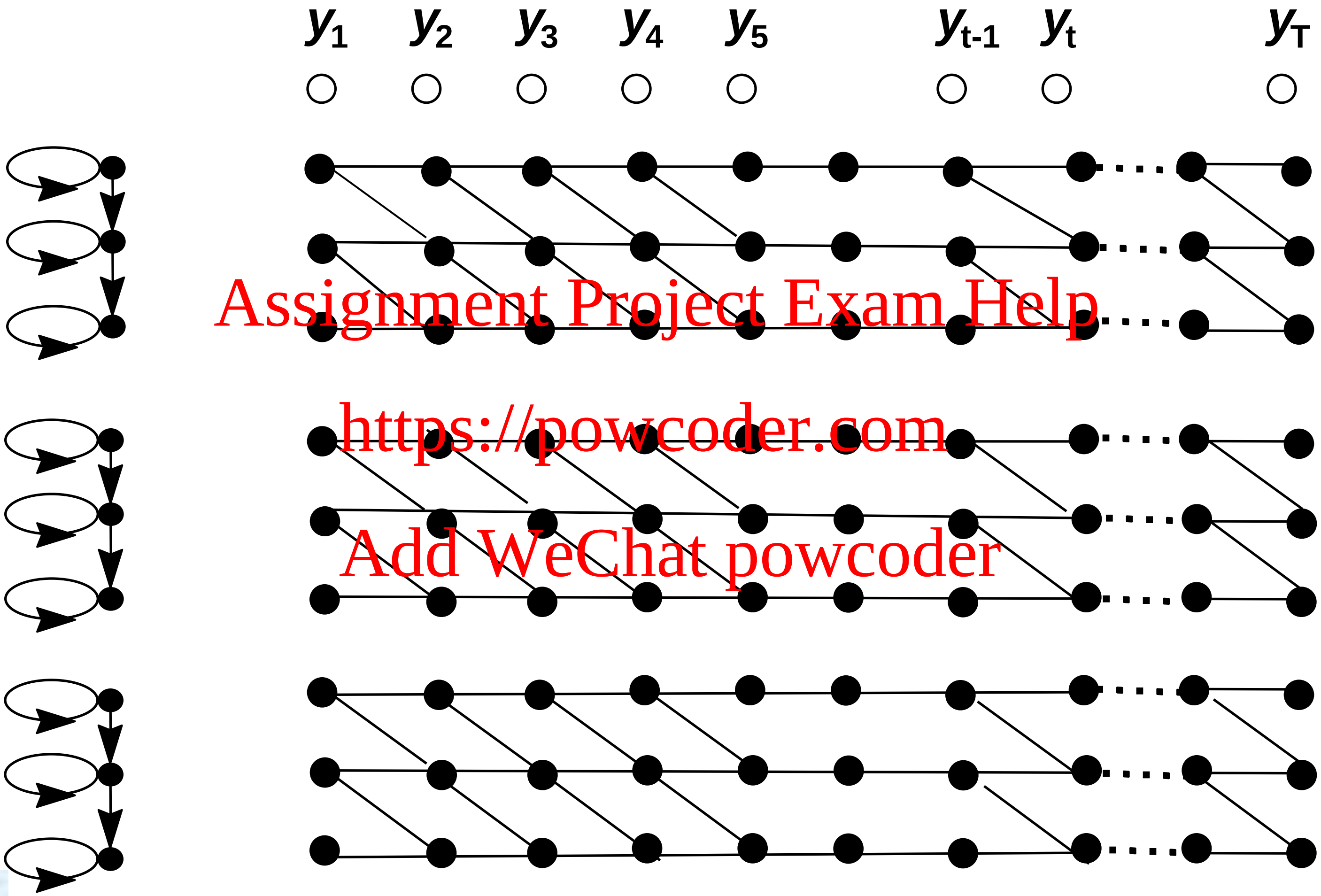


Viterbi Decoding – Trace-back



UNIVERSITY OF BIRMINGHAM

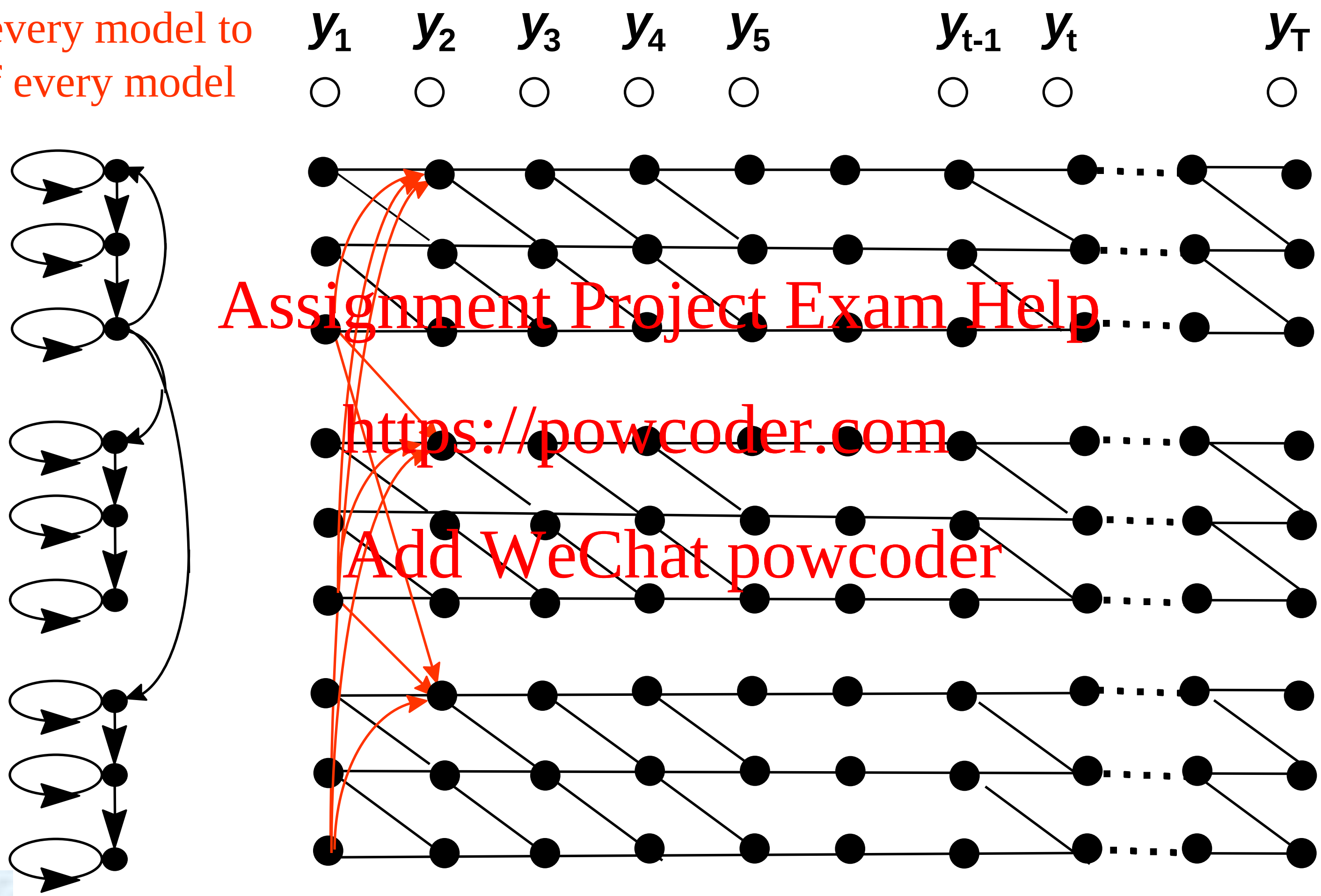
Isolated Speech Recognition



UNIVERSITY OF BIRMINGHAM

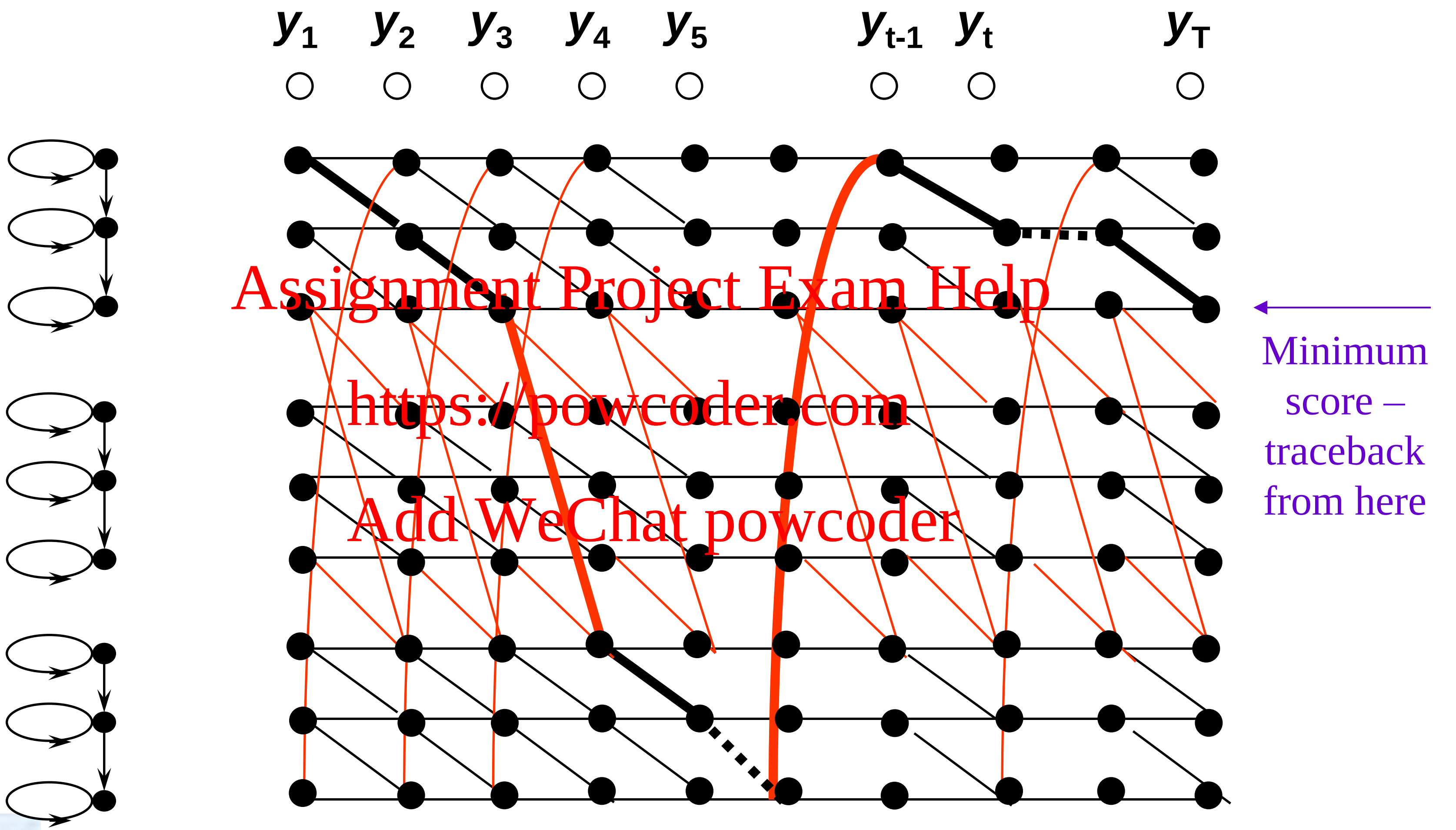
Connected Speech Recognition

New transitions connect end of every model to start of every model



UNIVERSITY OF BIRMINGHAM

Connected Speech Recognition



UNIVERSITY OF BIRMINGHAM

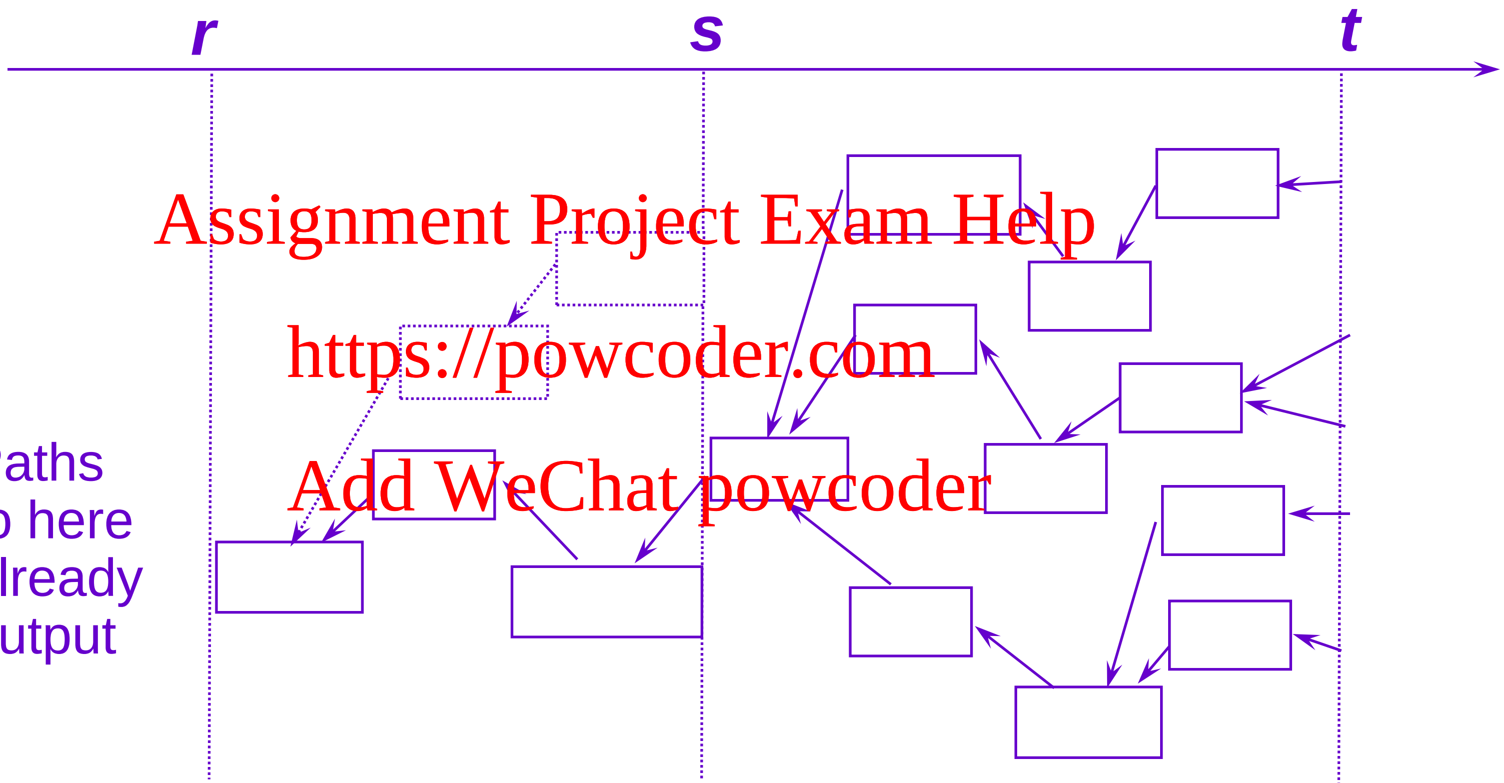
Partial Traceback (1)

- In continuous speech recognition, cannot trace-back from the end of the utterance (there is no end!)
- Instead **partial traceback** operates as follows:

  - For each time *t* and state *i* a *word link record* describes the sequence of words on the best path to *(t,i)*.

  - At regular intervals all active paths are traced back to see if they converge at some time *s* in the past
  - If so, the best path up to time *s* cannot change, and the sequence of words up to *s* can be output

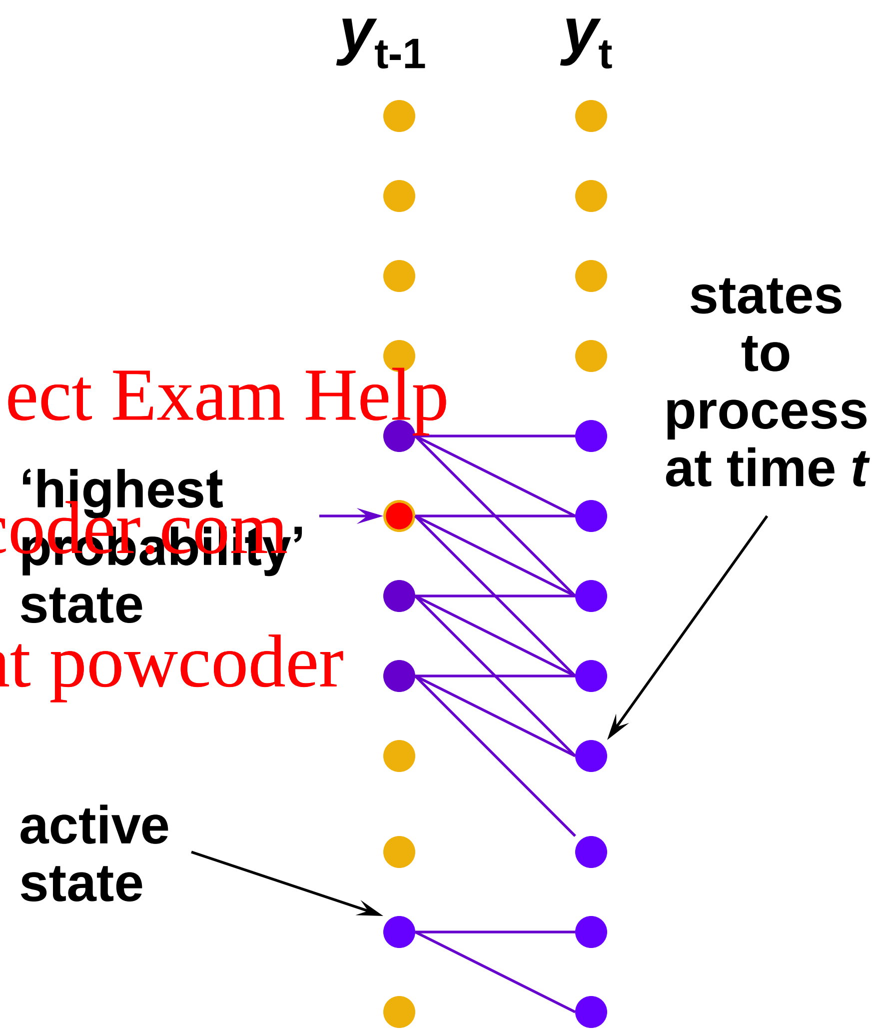
Partial Traceback (2)



Paths
to here
already
output

Beam Pruning

- Choose threshold T
- The active states at time $t\text{-}1$ are those whose partial path scores are within T of the best path score at time $t\text{-}1$
- At time t, only process those states which link back to an active state



Partial Traceback, Beam Pruning & Recognition ‘Speed’

- Partial traceback introduces a ‘lag’ into recognition process - **not** due to inadequate processor speed
- Lag worse when models are poor
- Beam Pruning less effective for ambiguous input
- Severe Beam Pruning will degrade performance
- Proper management of Partial Traceback and Beam Pruning is essential for optimal performance

Summary

- Speech recognition using HMMs
  - Viterbi decoding
  - Isolated & Connected/Continuous speech recognition
  - Beam pruning
  - Partial traceback