50.007 Machine Learning

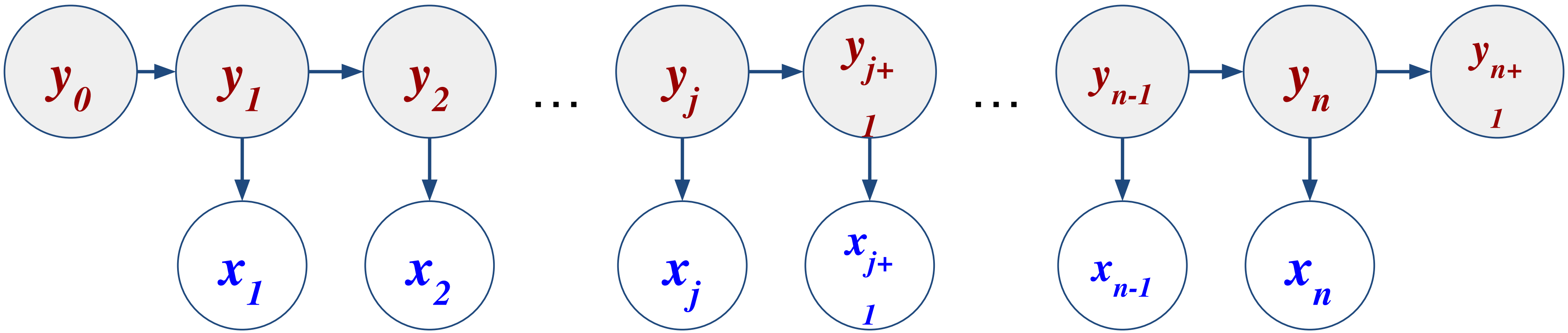
Hidden Markov Model

(Continue…)

Roy Ka-Wei Lee
Assistant Professor, DAI/ISTD, SUTD

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

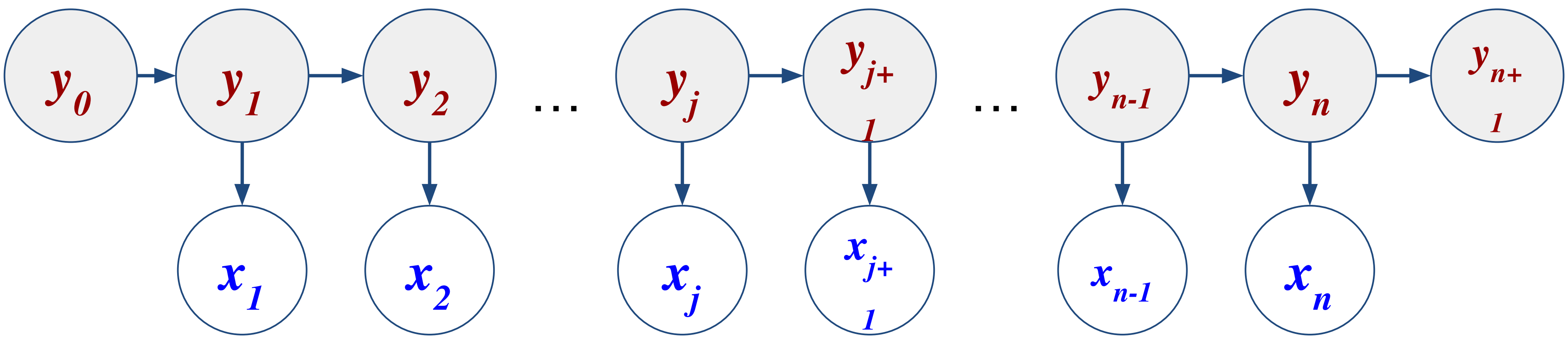
HMM Parameterization



$$p(x_1, x_2, \ldots, x_{n-1}, x_n, y_0, y_1, y_2, \ldots, y_{n-1}, y_n, y_{n+1})$$

$$\underbrace{\prod_{j=0}^{n} a_{y_j, y_{j+1}}}_{\textbf{Transition Probabilities}} \times \underbrace{\prod_{j=1}^{n} b_{y_j}(x_j)}_{\textbf{Emission Probabilities}}$$

HMM Supervised Learning



$$p(x_1, x_2, \ldots, x_{n-1}, x_n, y_0, y_1, y_2, \ldots, y_{n-1}, y_n, y_{n+1})$$

$$a_{u,v} = \frac{\text{count}(u,v)}{\text{count}(u)} \qquad b_u(o) = \frac{\text{count}(u \rightarrow o)}{\text{count}(u)}$$

Hidden Markov Model (Example)

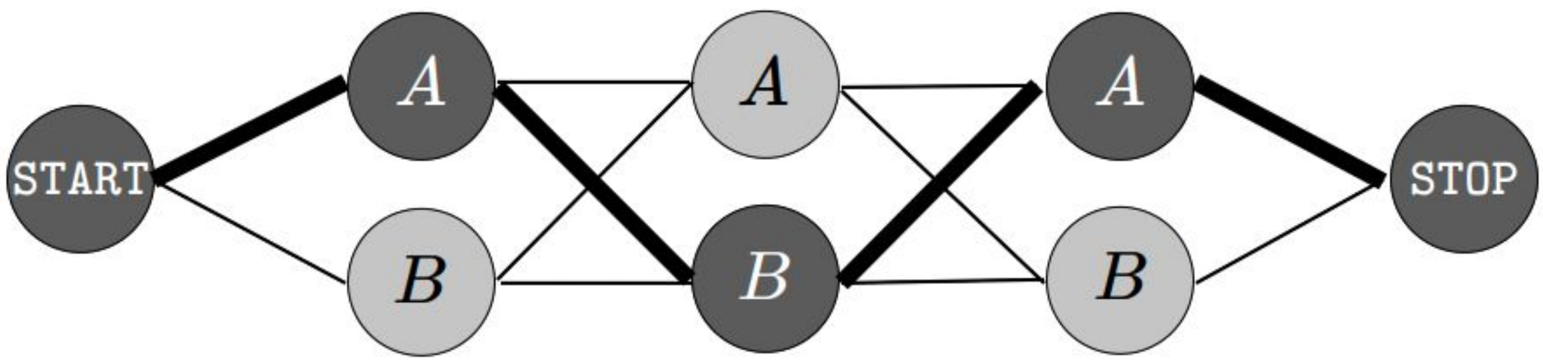
$$a_{u,v}$$

$u \backslash v$	A	B	STOP
START	1.0	0.0	0.0
A	0.5	0.5	0.0
B	0.0	0.8	0.2

$$b_u(o)$$

$u \backslash o$	"the"	"dog"
A	0.9	0.1
B	0.1	0.9

$$(\mathbf{x}, \mathbf{y}) = \text{the}/A, \text{dog}/B, \text{the}/A$$



$$a_{\texttt{START},A} \times b_A(\text{``the''}) \times a_{A,B} \times b_B(\text{``dog''}) \times a_{B,A} \times b_A(\text{``the''}) \times a_{A,\texttt{STOP}}$$

Hidden Markov Model (Example)

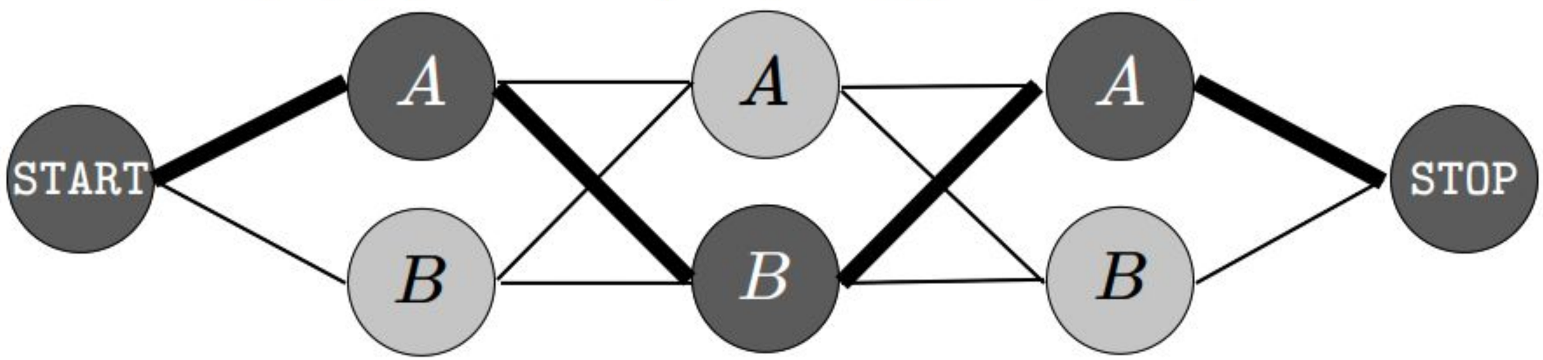
$a_{u,v}$

$u \backslash v$	A	B	STOP
START	1.0	0.0	0.0
A	0.5	0.5	0.0
B	0.0	0.8	0.2

$b_u(o)$

$u \backslash o$	"the"	"dog"
A	0.9	0.1
B	0.1	0.9

$$(\mathbf{x}, \mathbf{y}) = \text{the}/B, \text{dog}/B, \text{the}/B$$



What about this new y label sequence?

Hidden Markov Model (Example)

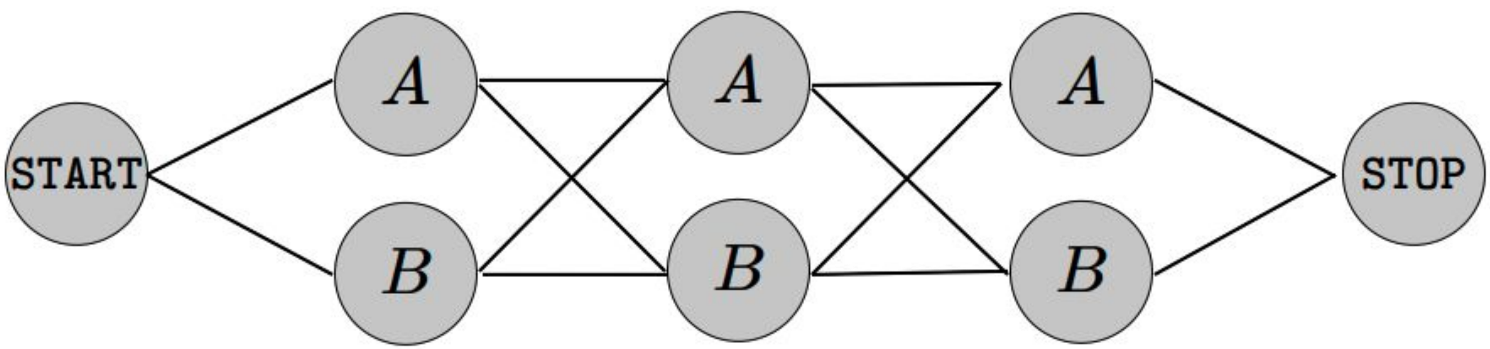
$a_{u,v}$

$u \backslash v$	A	B	STOP
START	1.0	0.0	0.0
A	0.5	0.5	0.0
B	0.0	0.8	0.2

$b_u(o)$

$u \backslash o$	"the"	"dog"
A	0.9	0.1
B	0.1	0.9

$$\mathbf{x} = \text{the dog the}$$



Which label sequence y is the most probable given the word sequence x?

Hidden Markov Model (Example)

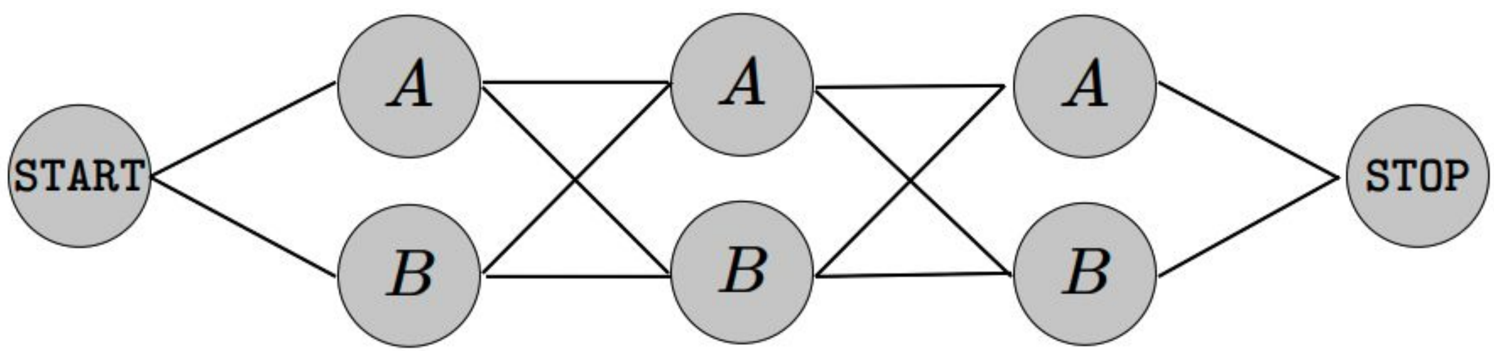
$$a_{u,v}$$

$u \backslash v$	A	B	STOP
START	1.0	0.0	0.0
A	0.5	0.5	0.0
B	0.0	0.8	0.2

$$b_u(o)$$

$u \backslash o$	"the"	"dog"
A	0.9	0.1
B	0.1	0.9

$$\mathbf{x} = \text{the dog the}$$

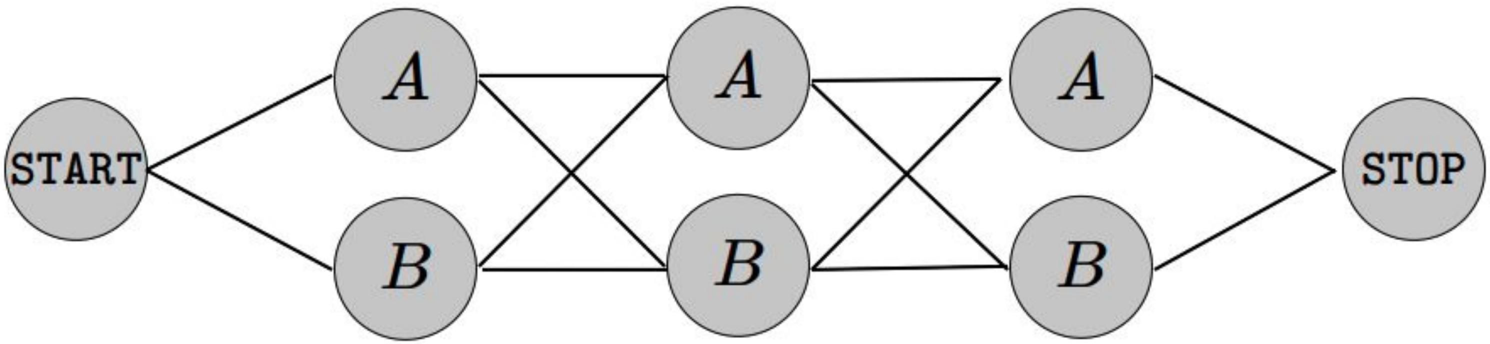


$$\mathbf{y}^* = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$$

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Hidden Markov Model (Example)

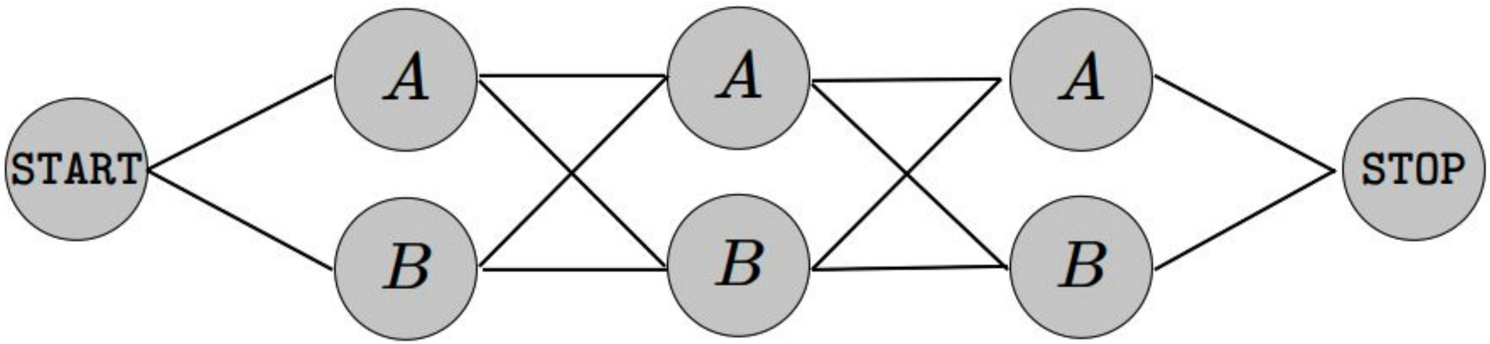
$$\mathbf{x} = \text{the dog the}$$



$$\mathbf{y}^* = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$$

Hidden Markov Model (Example)

$$\mathbf{x} = \text{the dog the}$$

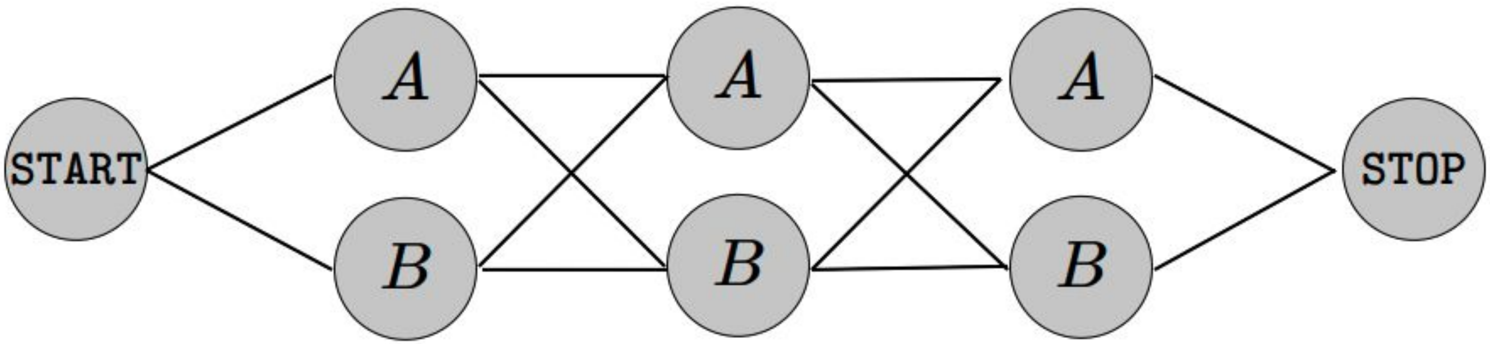


$$\mathbf{y}^* = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$$
$$= \arg\max_{\mathbf{y}} p(\mathbf{x}, \mathbf{y})/p(\mathbf{x})$$

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Hidden Markov Model (Example)

$$\mathbf{x} = \text{the dog the}$$

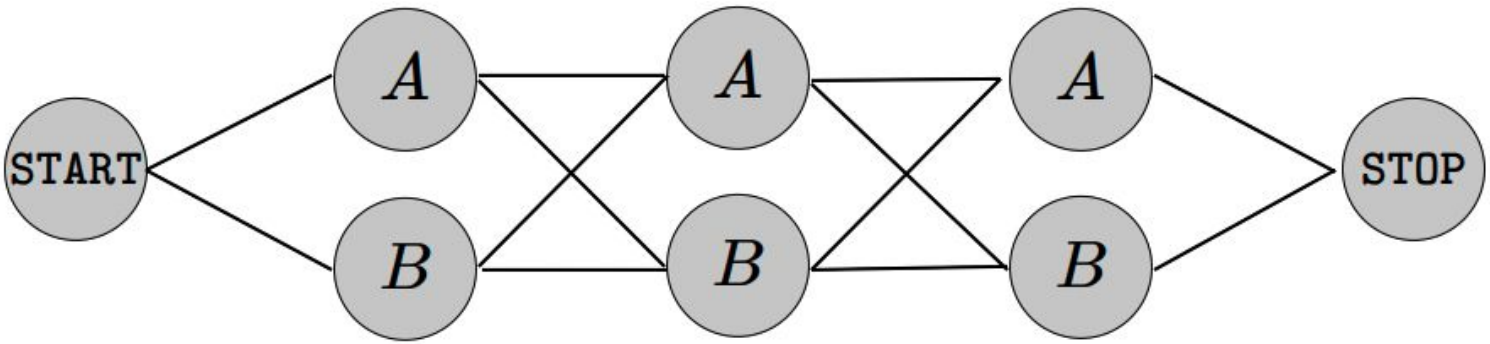


$$\begin{aligned} \mathbf{y}^* &= \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x}) \\ &= \arg\max_{\mathbf{y}} p(\mathbf{x}, \mathbf{y})/p(\mathbf{x}) \\ &= \arg\max_{\mathbf{y}} p(\mathbf{x}, \mathbf{y}) \end{aligned}$$

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Hidden Markov Model (Example)

$$\mathbf{x} = \text{the dog the}$$

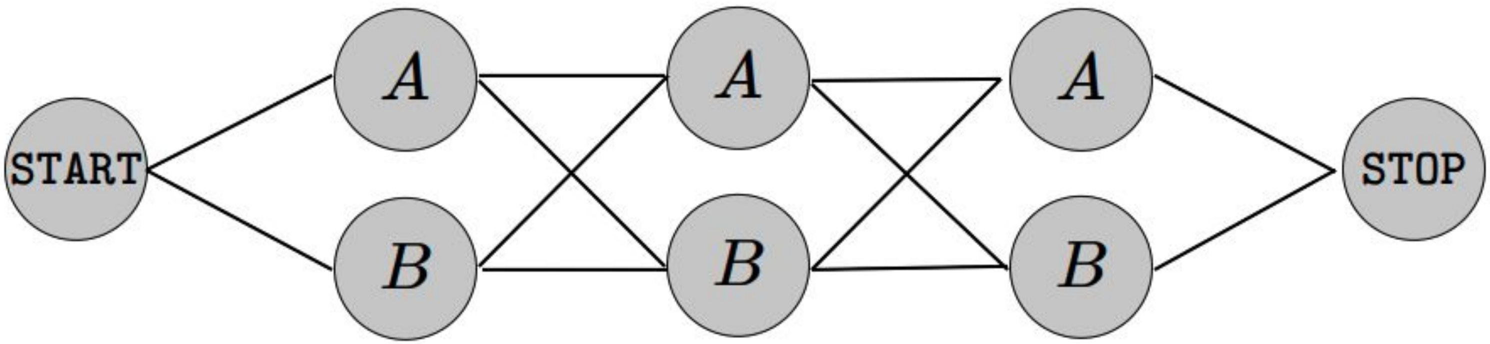


$$\begin{aligned} \mathbf{y}^* &= \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x}) \\ &= \arg\max_{\mathbf{y}} p(\mathbf{x}, \mathbf{y})/p(\mathbf{x}) \\ &= \arg\max_{\mathbf{y}} p(\mathbf{x}, \mathbf{y}) \end{aligned}$$

We can try one y at a time, and see which gives the highest score!

Hidden Markov Model (Example)

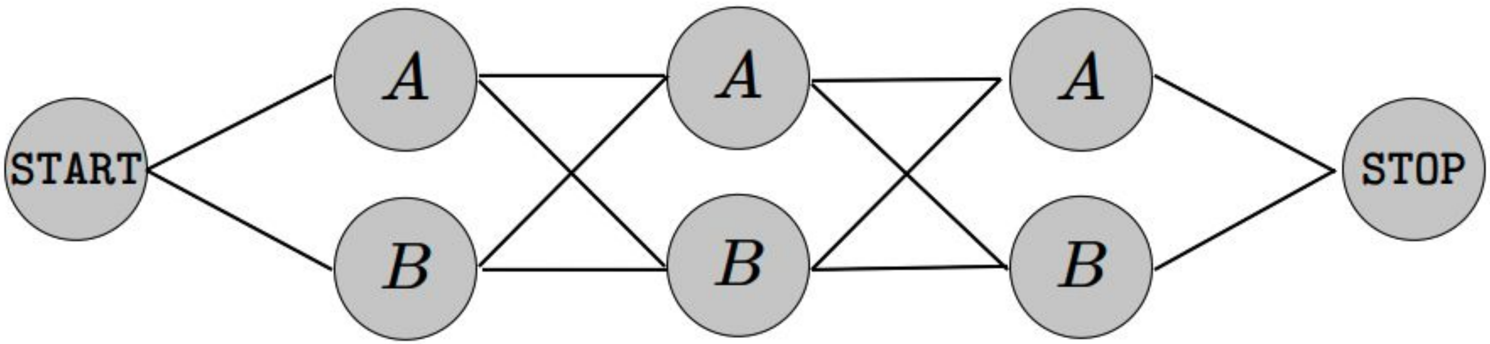
$$\mathbf{x} = \text{the dog the}$$



$$\begin{aligned}\mathbf{y}^* &= \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x}) \\ &= \arg\max_{\mathbf{y}} p(\mathbf{x},\mathbf{y})/p(\mathbf{x}) \\ &= \arg\max_{\mathbf{y}} p(\mathbf{x},\mathbf{y})\end{aligned}$$

Hidden Markov Model (Example)

$$\mathbf{x} = \text{the dog the}$$



$$\mathbf{y}^* = \arg\max_{\mathbf{y}} p(\mathbf{x}, \mathbf{y})$$

Number of words in the sentence

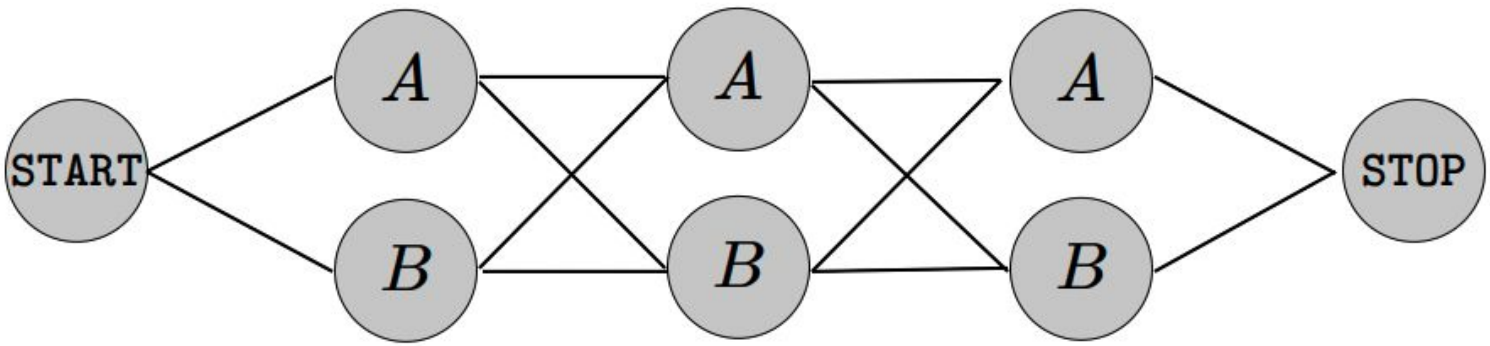
There are $O(|\mathcal{T}|^n)$ possible $\mathbf{y}$'s!

Number of possible tags at each word/position

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

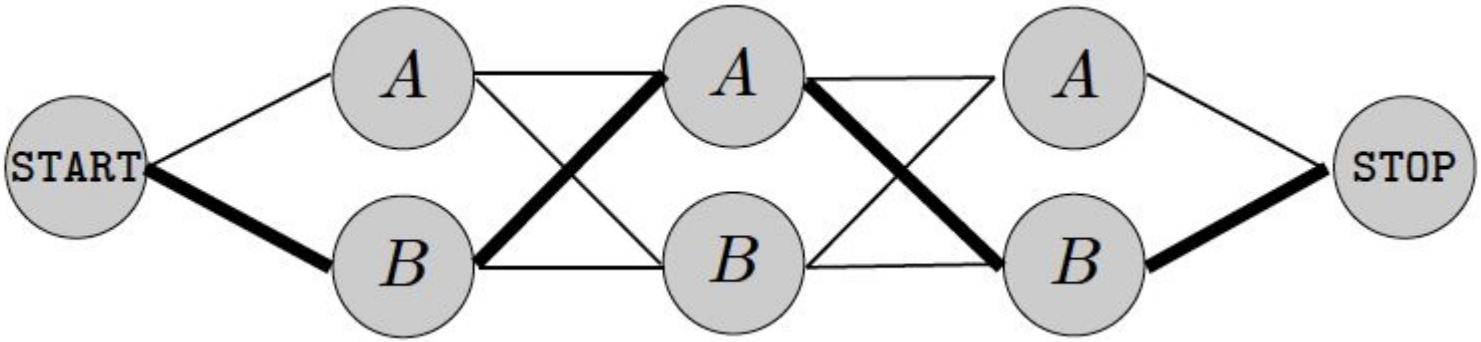
Finding Highest Scoring Path

We are facing a problem of finding the highest scoring path connecting START and STOP



Finding Highest Scoring Path

We are facing a problem of finding the highest scoring path connecting START and STOP

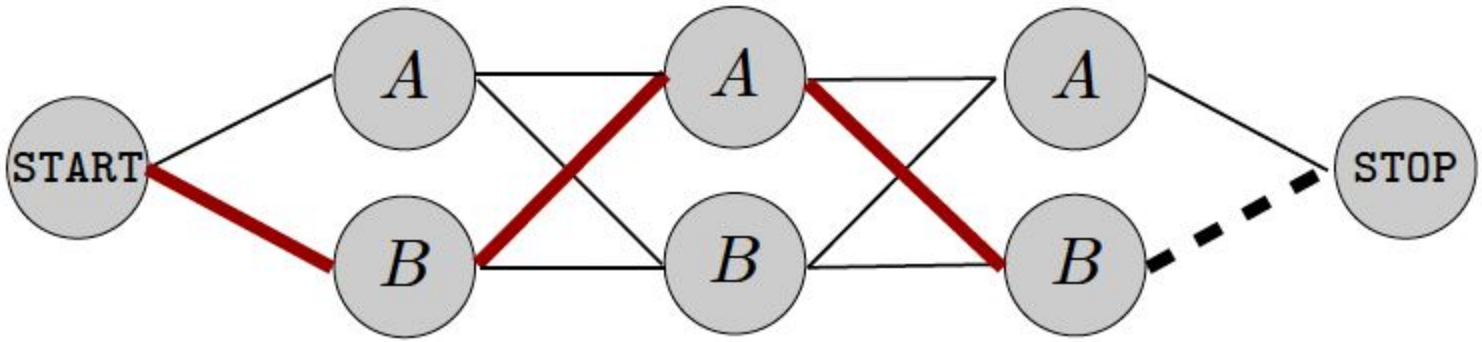


Let's assume this is the highest scoring path

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Finding Highest Scoring Path

We are facing a problem of finding the highest scoring path connecting START and STOP

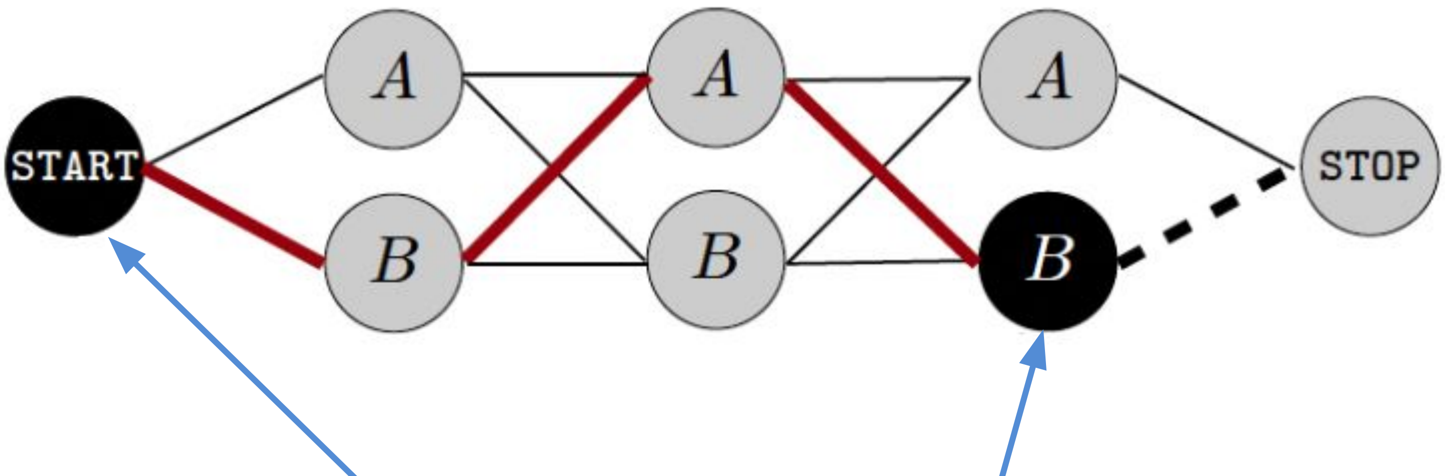


Let's assume this is the highest scoring path

Then, what can we say about this partial path?
(What types of properties do we know for this path?)

Finding Highest Scoring Path

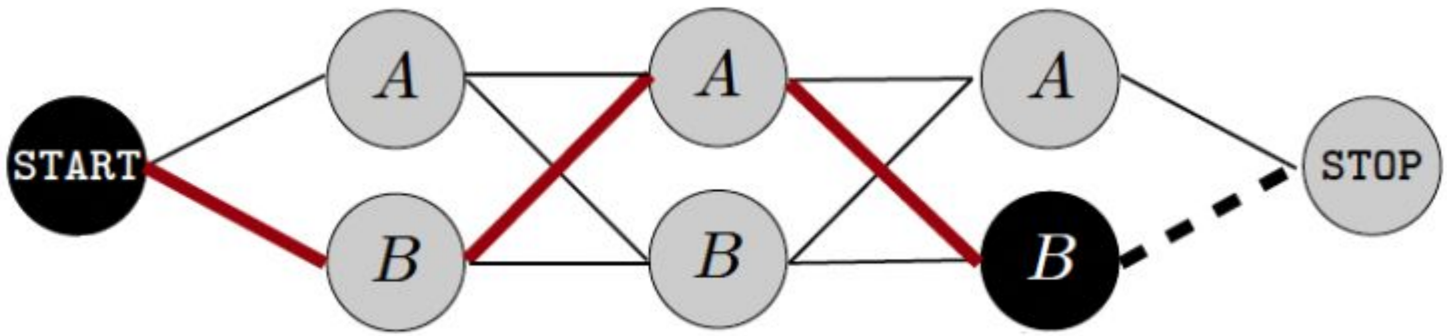
We are facing a problem of finding the highest scoring path connecting START and STOP



The highest scoring path among all paths connecting these two nodes!

Finding Highest Scoring Path

We are facing a problem of finding the highest scoring path connecting START and STOP

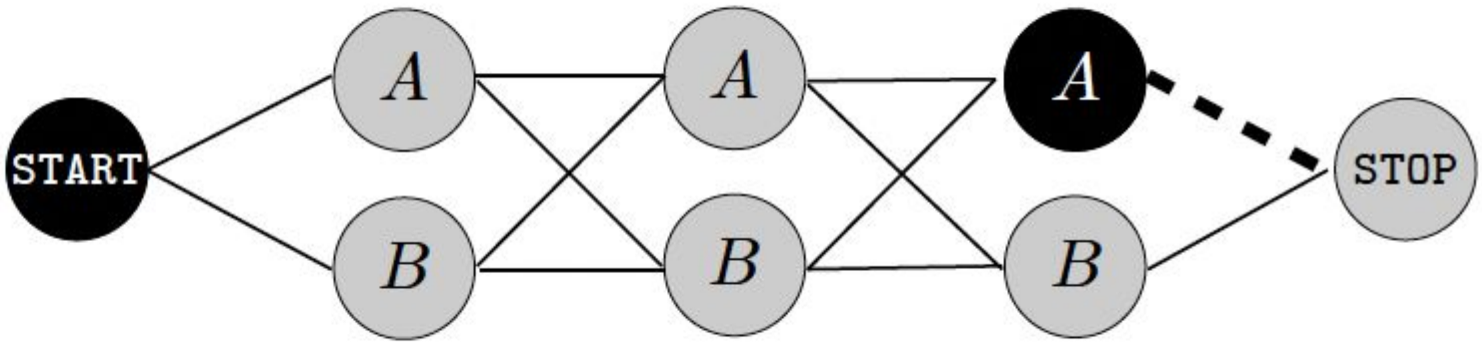


Is it possible to solve the original problem, if we already know the solutions to such sub-problems?

Finding Highest Scoring Path

Case A

The second last node in the highest scoring path is A.



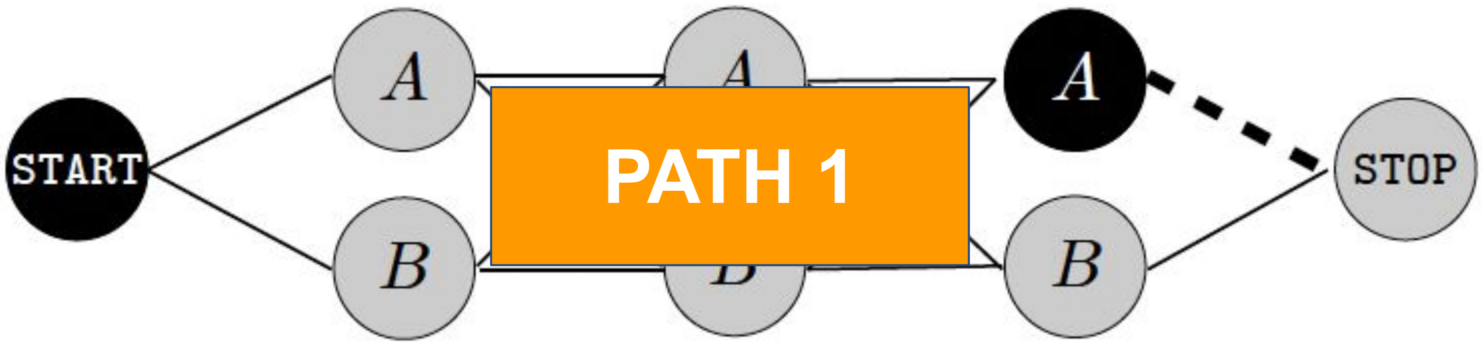
Find the highest scoring path from START to A at position n

A single edge from node A to STOP

Finding Highest Scoring Path

Case A

The second last node in the highest scoring path is A.



Find the highest scoring path from START to A at position n

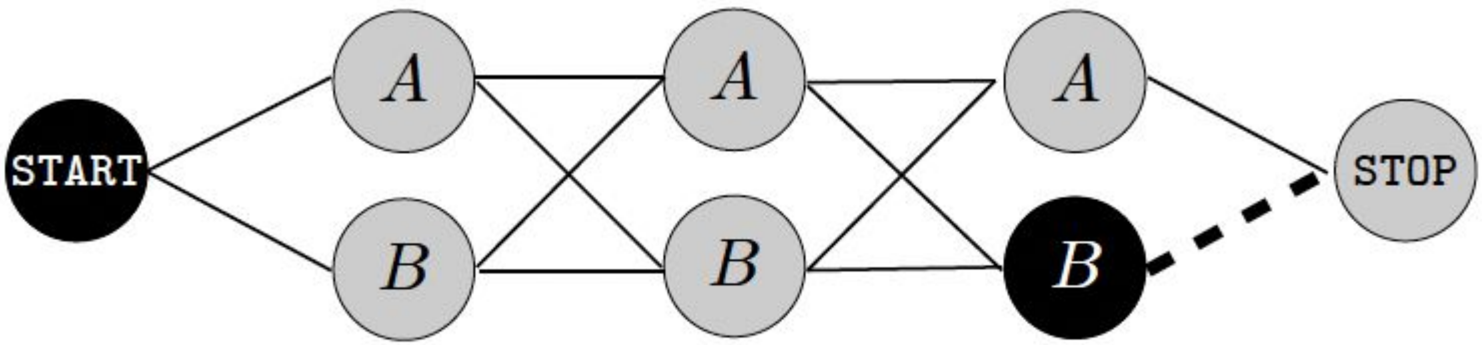
+

A single edge from node A to STOP

Finding Highest Scoring Path

Case B

The second last node in the highest scoring path is B.



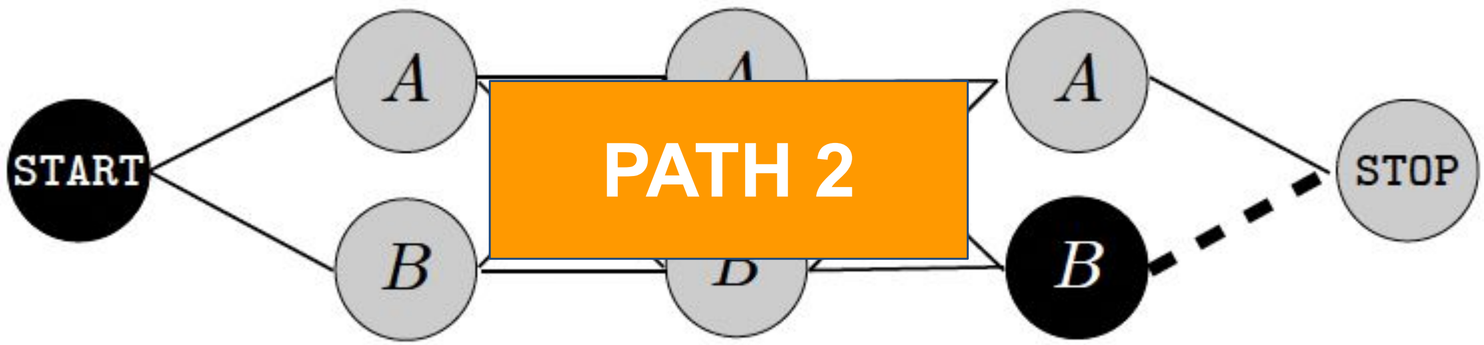
Find the highest scoring path from START to B at position n

A single edge from node B to STOP

Finding Highest Scoring Path

Case B

The second last node in the highest scoring path is B.

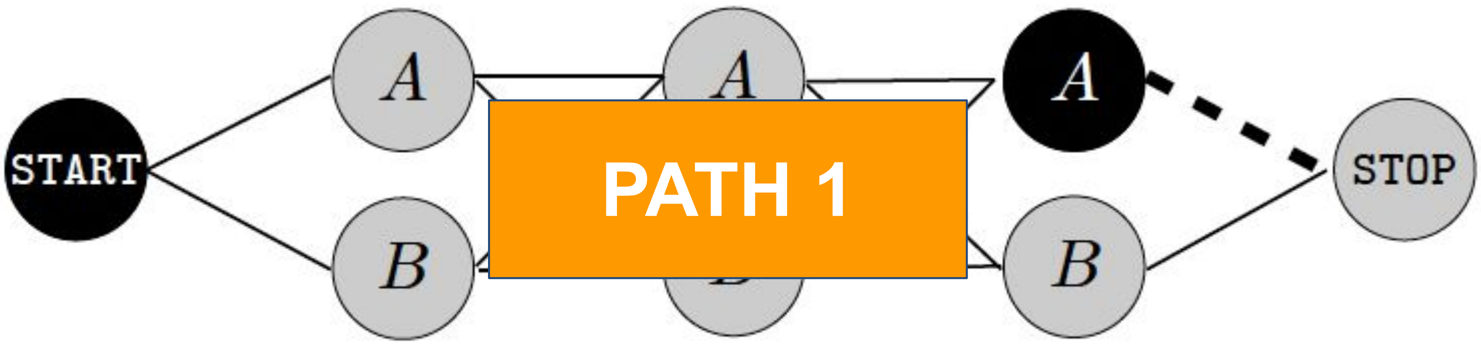


Find the highest scoring path from START to B at position n

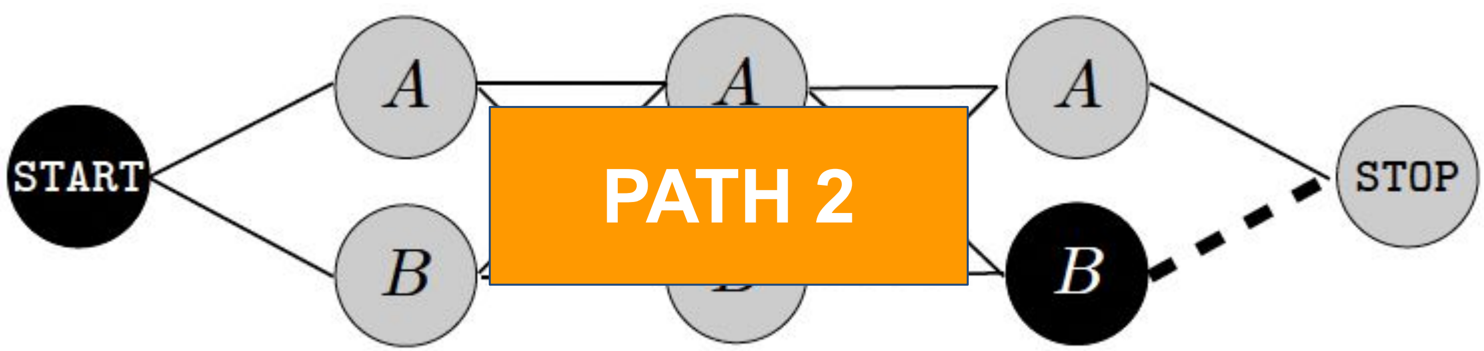
A single edge from node B to STOP

Finding Highest Scoring Path

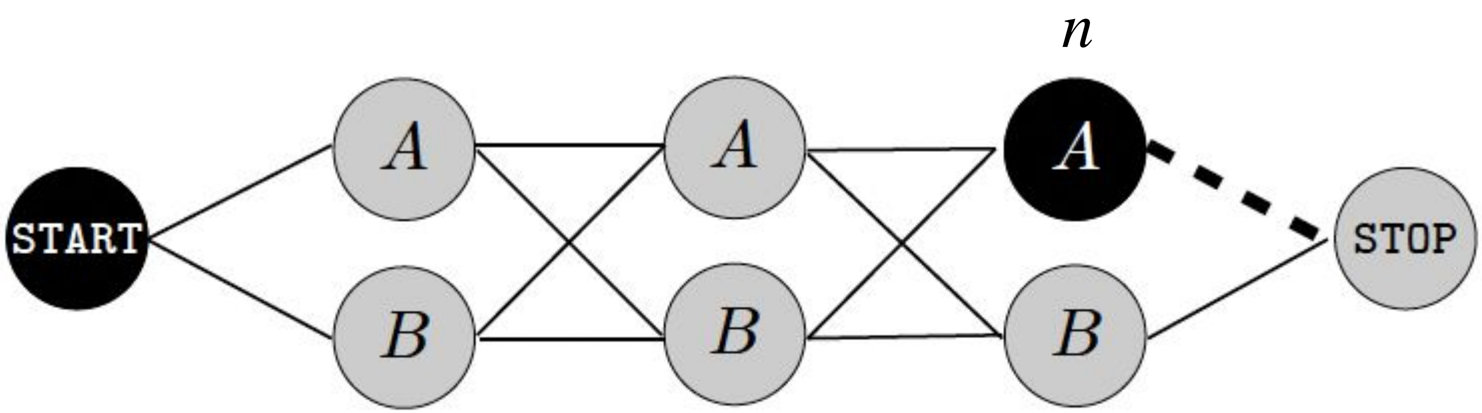
The highest scoring path from START to STOP



Or

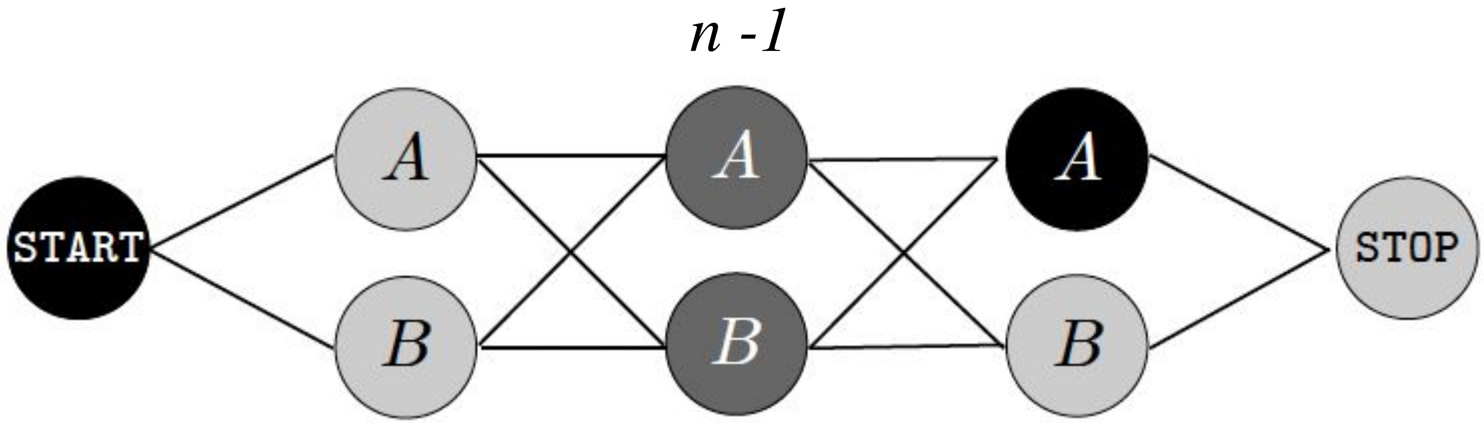


Finding Highest Scoring Path



How do you find the highest scoring path from START to node A at position n?

Finding Highest Scoring Path



How do you find the highest scoring path from START to node A at position n?

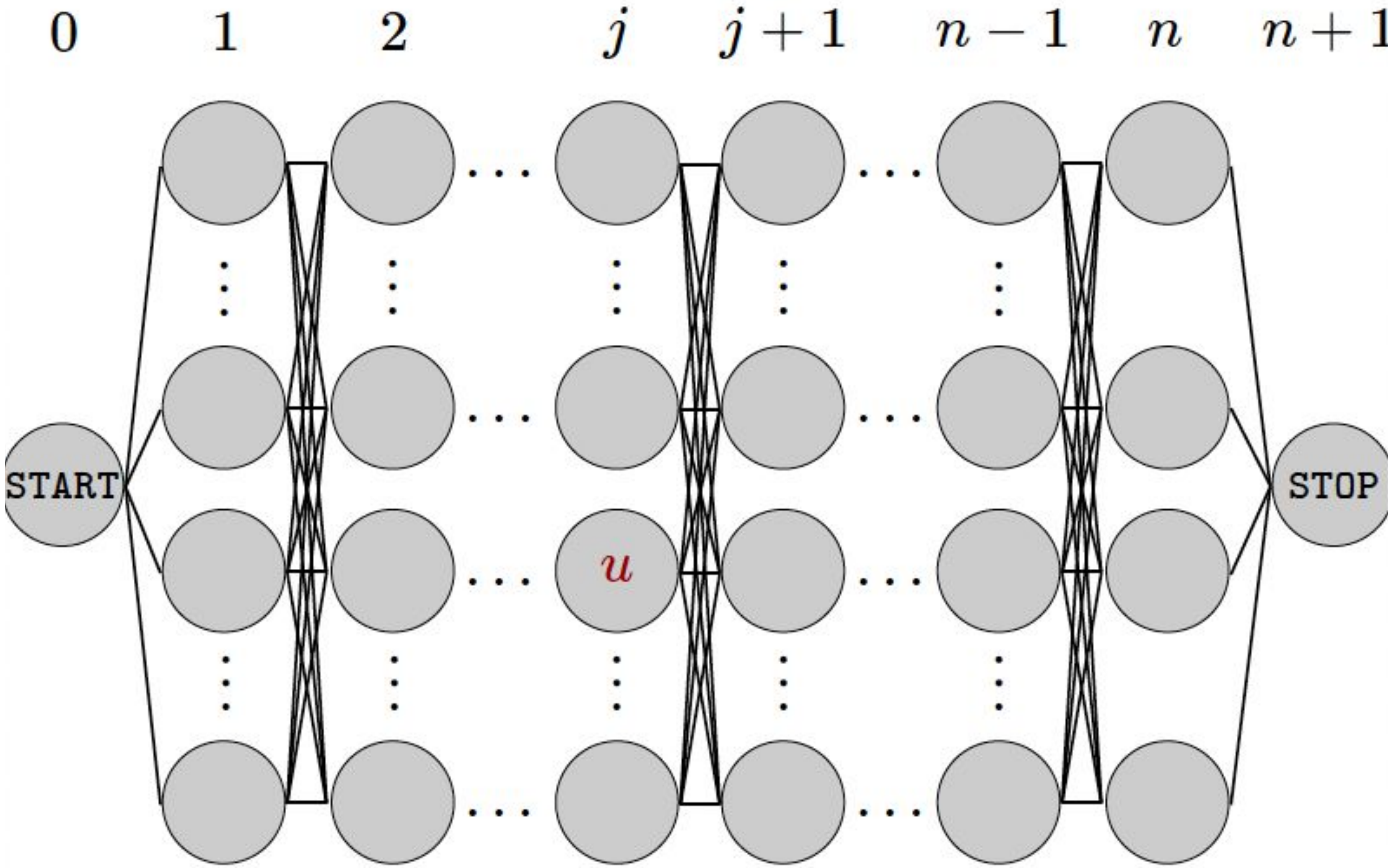
We shall again rely on the partial paths from START to the two nodes at position (n – 1)

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

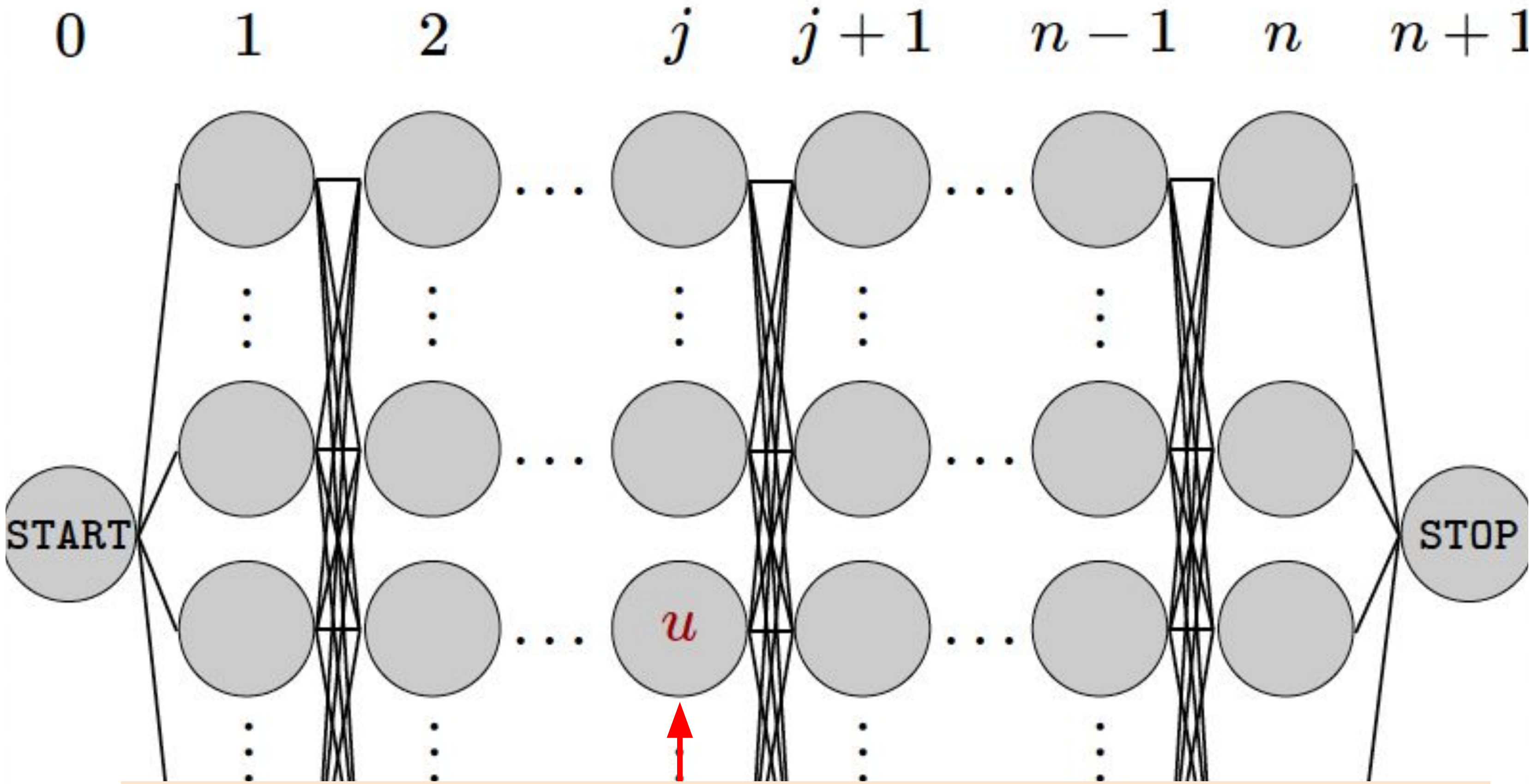
Question

Now, can we develop an efficient algorithm based on such observations?

Viterbi Algorithm

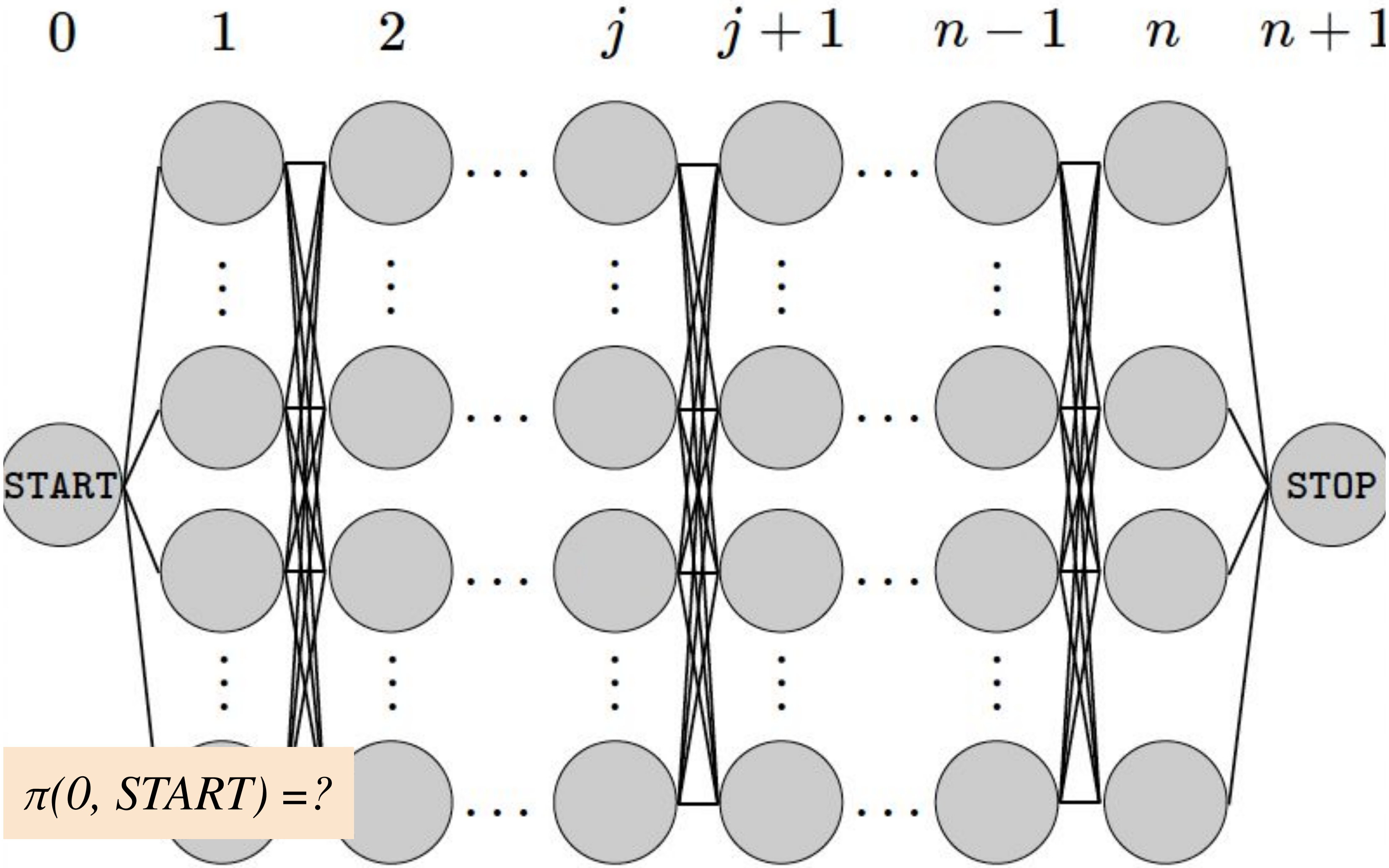


Viterbi Algorithm



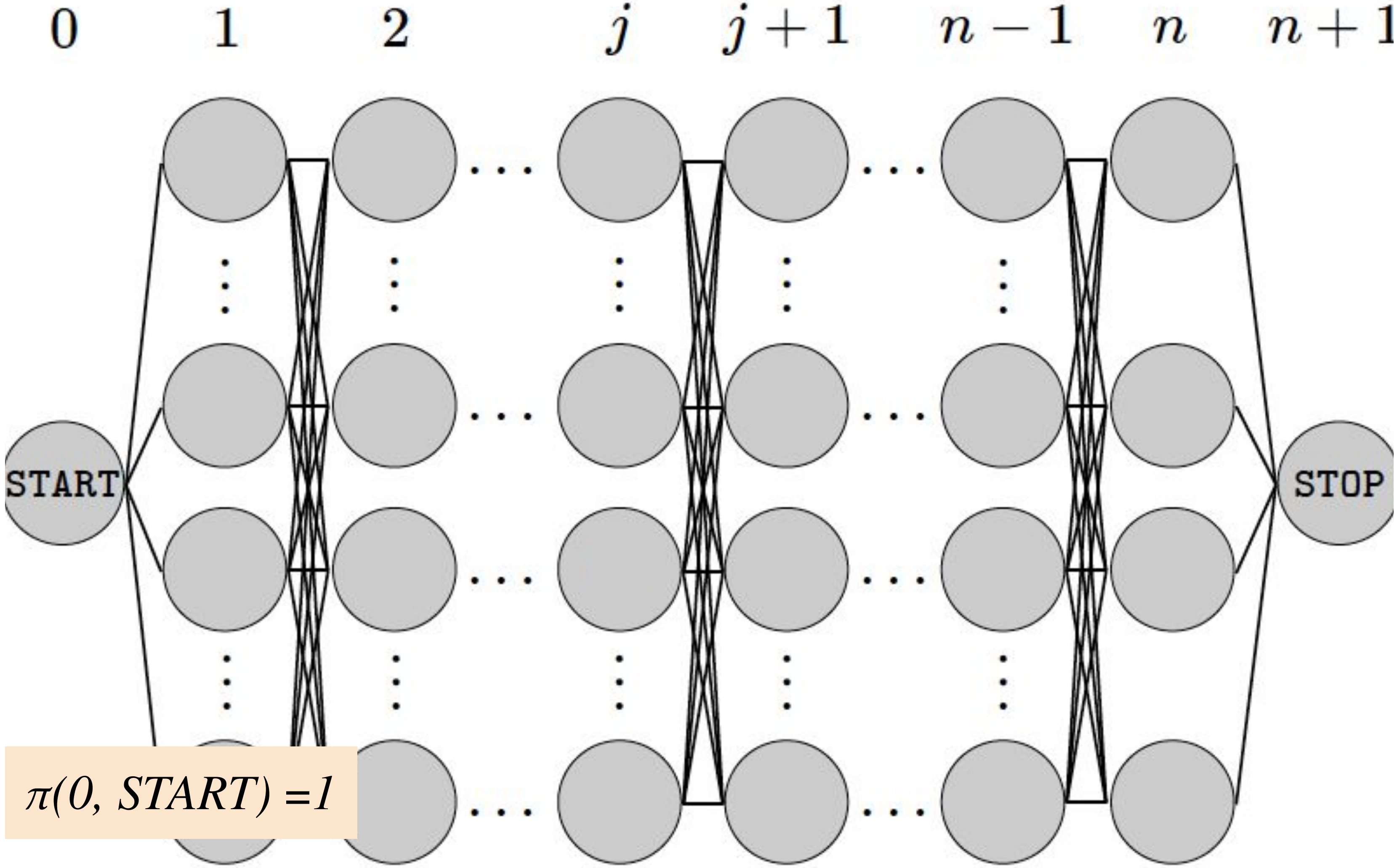
Store inside it $\pi(j, u)$ – the score of the highest scoring path from START to this node (j, u)

Viterbi Algorithm



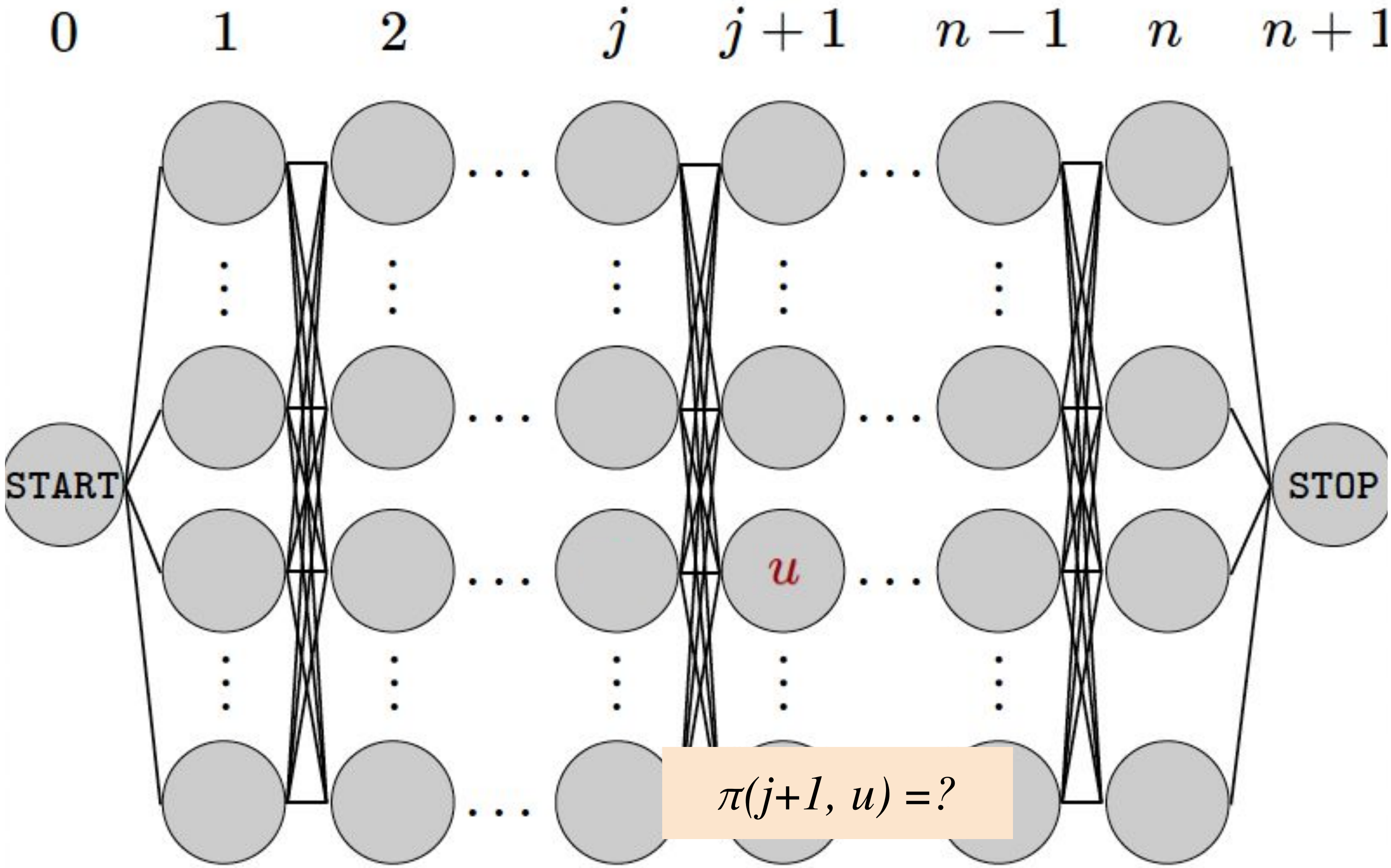
$\pi(0, START) = ?$

Viterbi Algorithm



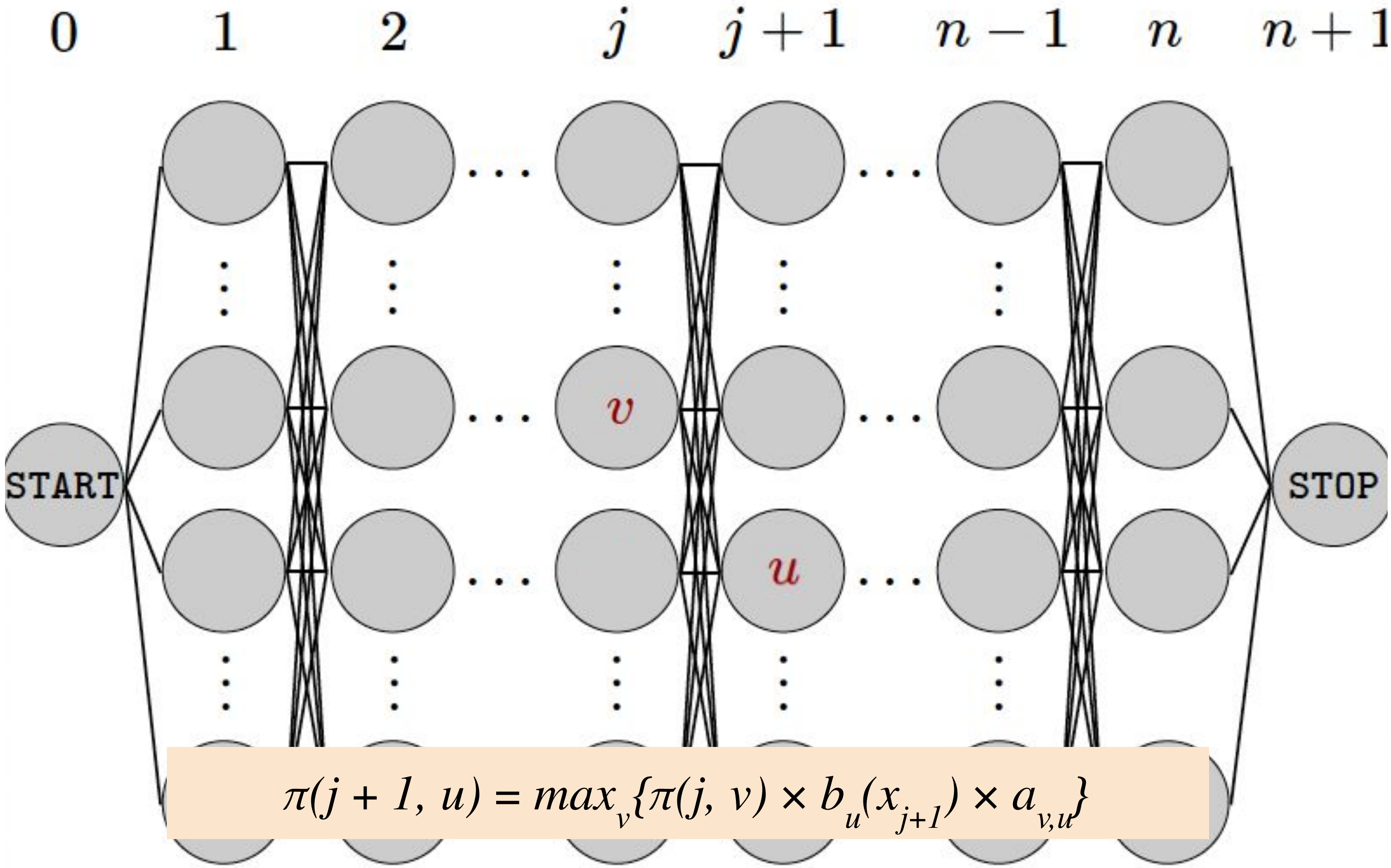
$\pi(0, START) = 1$

Viterbi Algorithm



$\pi(j+1, u) = ?$

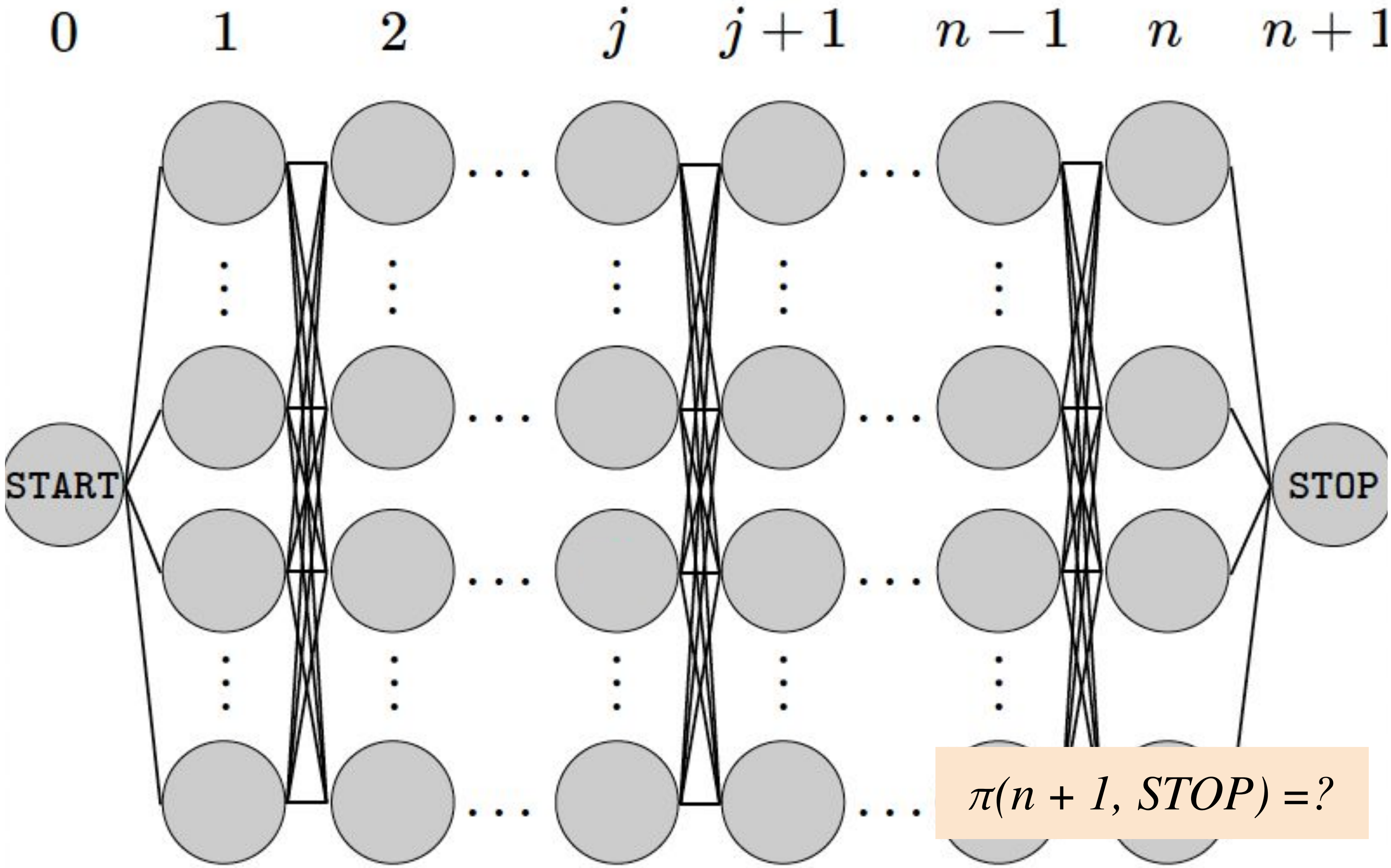
Viterbi Algorithm



$$\pi(j+1, u) = \max_v \{\pi(j, v) \times b_u(x_{j+1}) \times a_{v,u}\}$$

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Viterbi Algorithm



$\pi(n + 1, STOP) = ?$

Viterbi Algorithm

$0 \quad 1 \quad 2 \quad j \quad j+1 \quad n-1 \quad n \quad n+1$

STAR

STOP

1. Initialization

$$\pi(0, u) = \begin{cases} 1 & \text{if } u = \texttt{START} \\ 0 & \text{otherwise} \end{cases}$$

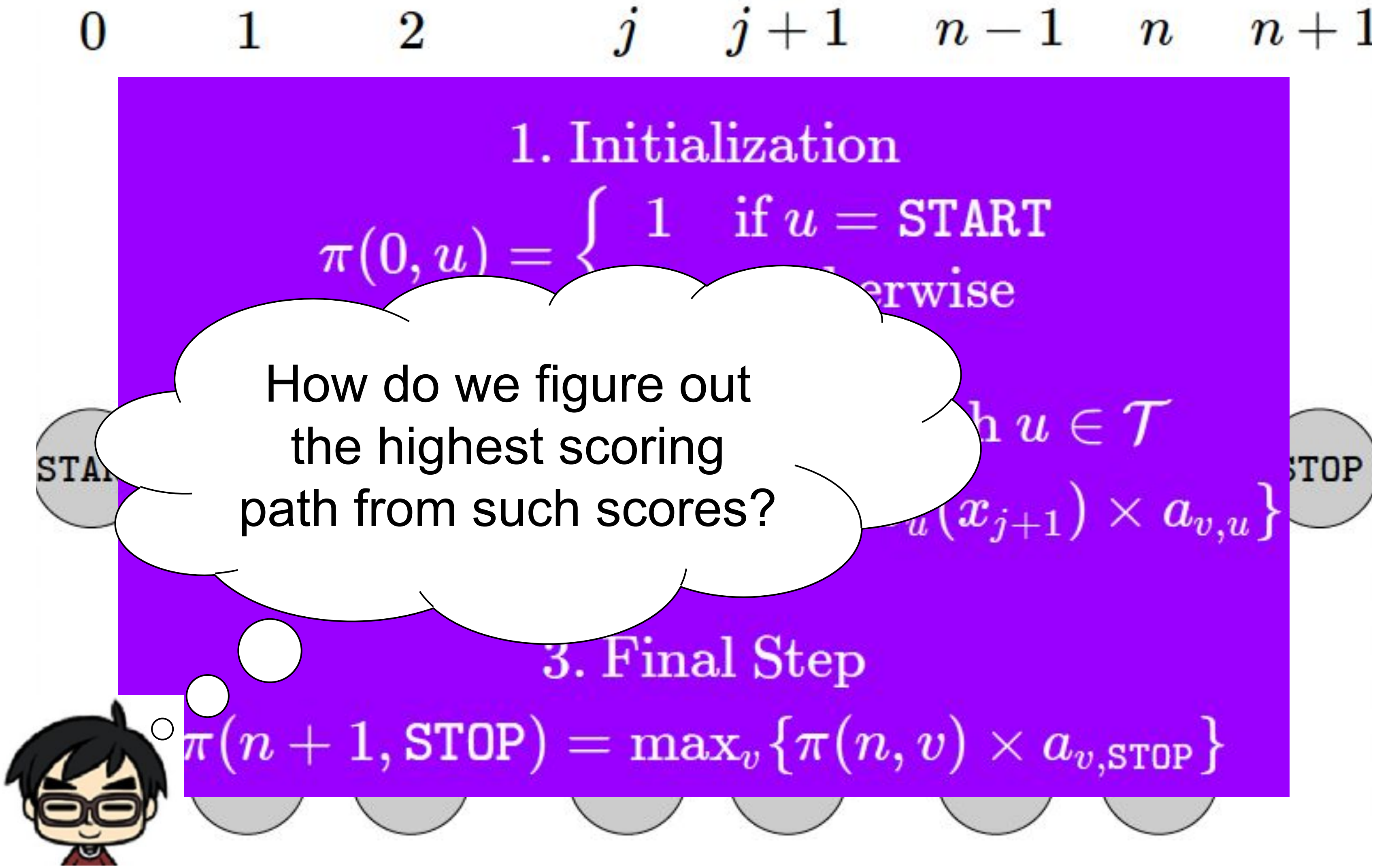
2. For $j = 0 \ldots n-1$, for each $u \in \mathcal{T}$

$$\pi(j+1, u) = \max_v \{\pi(j, v) \times b_u(x_{j+1}) \times a_{v,u}\}$$

3. Final Step

$$\pi(n+1, \texttt{STOP}) = \max_v \{\pi(n, v) \times a_{v,\texttt{STOP}}\}$$

Viterbi Algorithm



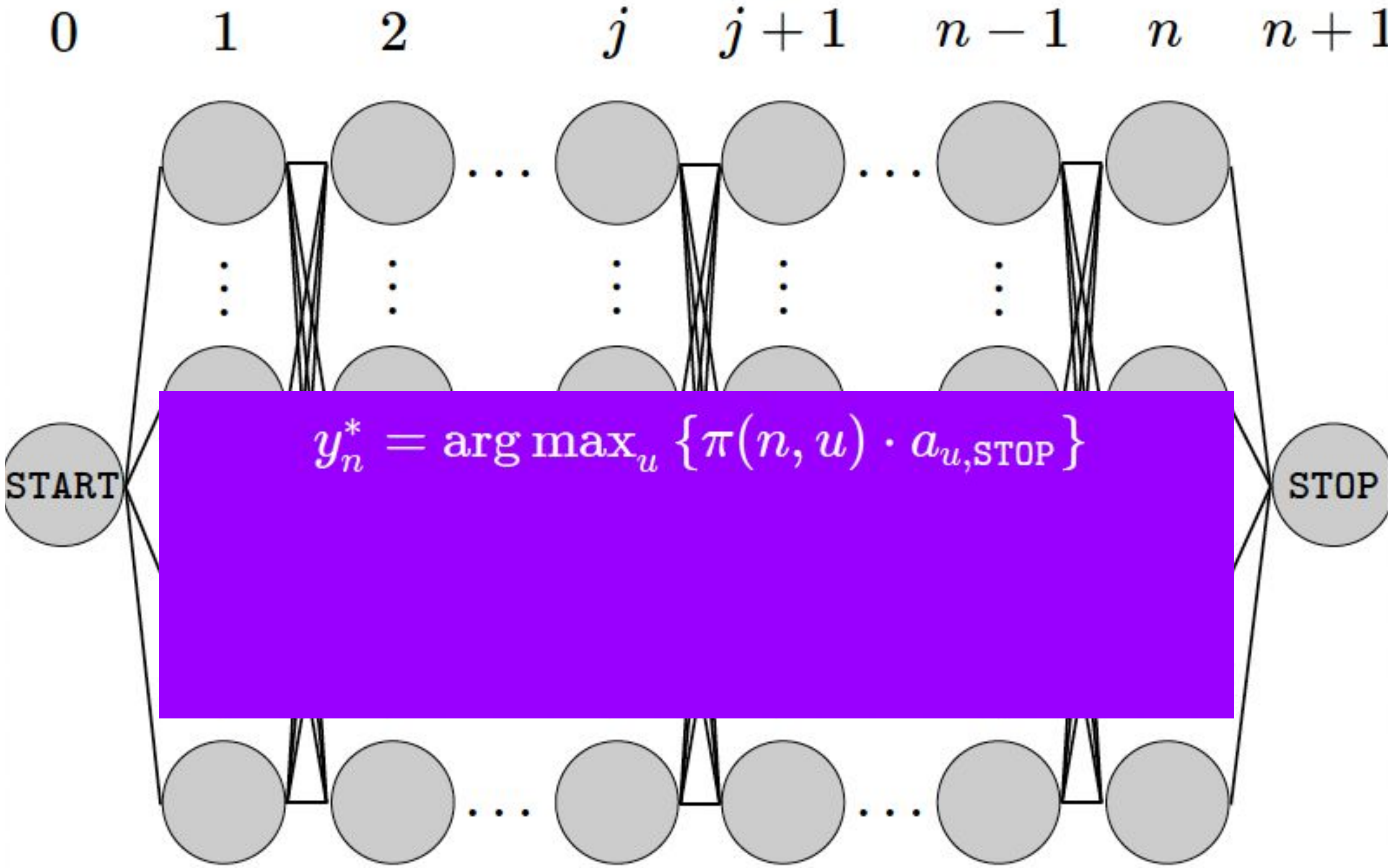
1. Initialization

$$\pi(0,u) = \begin{cases} 1 & \text{if } u = \texttt{START} \end{cases}$$

3. Final Step

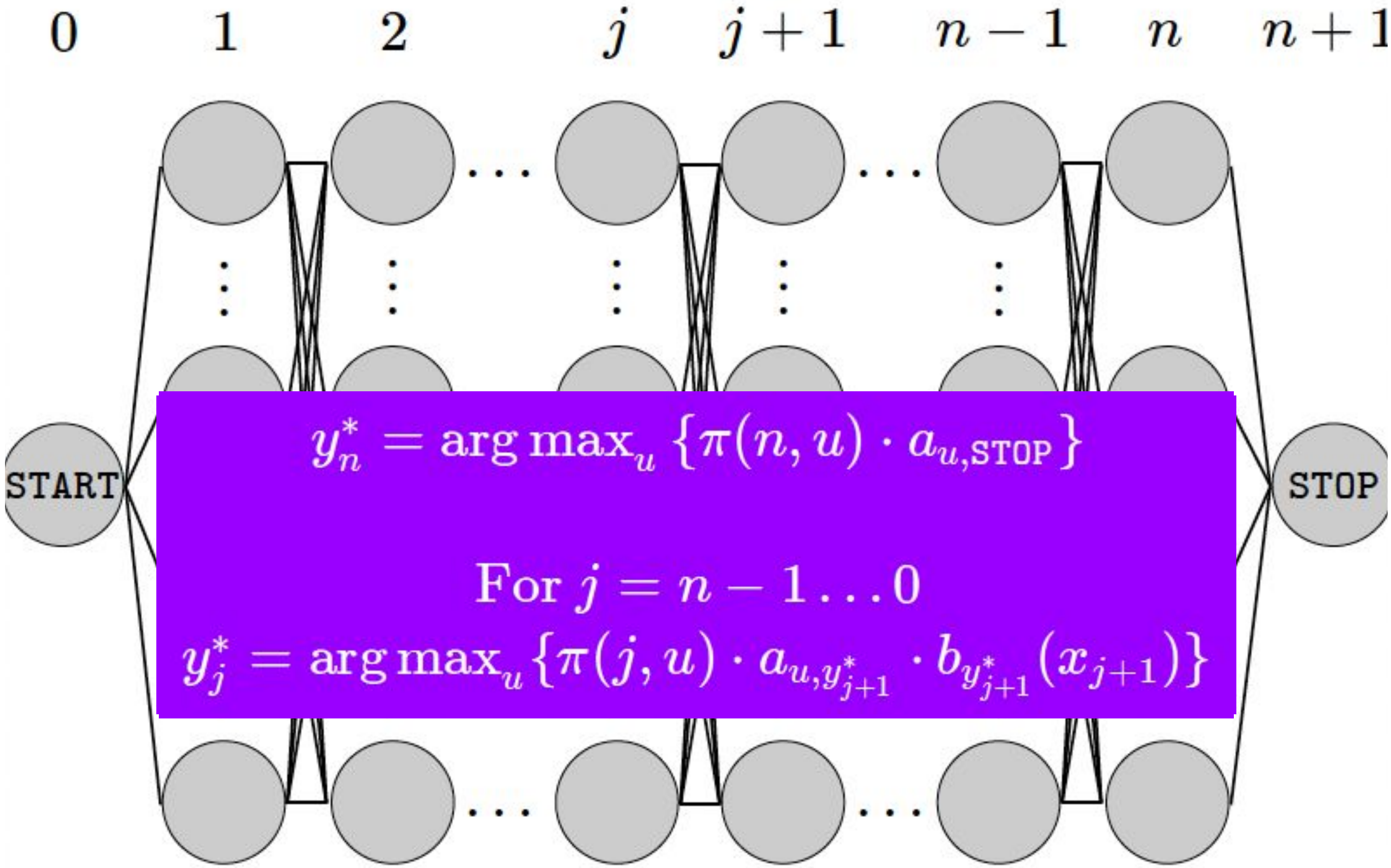
$$\pi(n+1, \texttt{STOP}) = \max_v \{\pi(n,v) \times a_{v,\texttt{STOP}}\}$$

Viterbi Algorithm



$$y_n^* = \arg\max_u \left\{ \pi(n, u) \cdot a_{u,\mathrm{STOP}} \right\}$$

Viterbi Algorithm

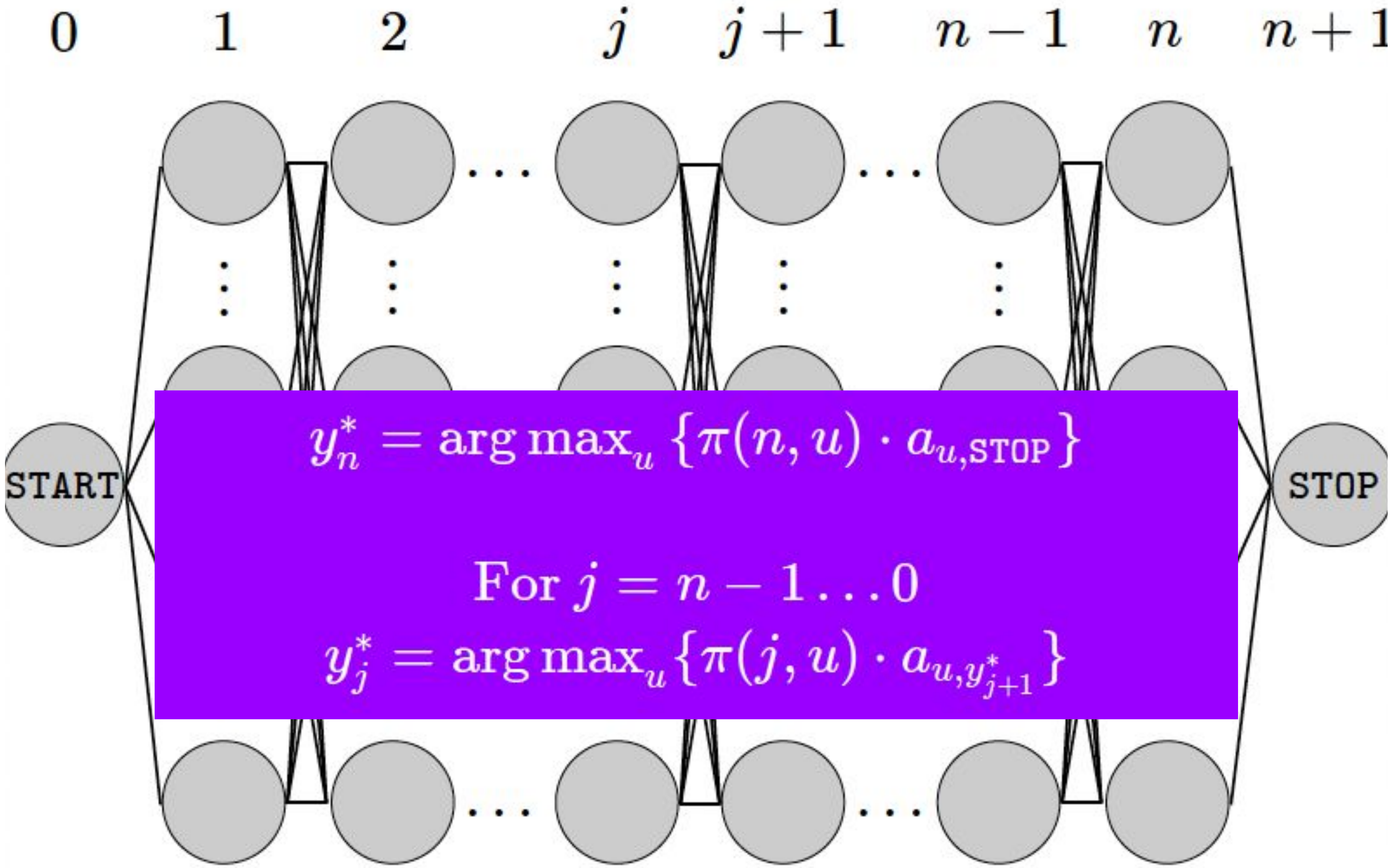


$$y_n^* = \arg\max_u \{\pi(n, u) \cdot a_{u,\text{STOP}}\}$$

For $j = n - 1 \ldots 0$

$$y_j^* = \arg\max_u \{\pi(j, u) \cdot a_{u, y_{j+1}^*} \cdot b_{y_{j+1}^*}(x_{j+1})\}$$

Viterbi Algorithm



$$y_n^* = \arg\max_u \{\pi(n,u) \cdot a_{u,\text{STOP}}\}$$

For $j = n-1 \ldots 0$

$$y_j^* = \arg\max_u \{\pi(j,u) \cdot a_{u,y_{j+1}^*}\}$$

Question

What is the time complexity of the Viterbi algorithm?