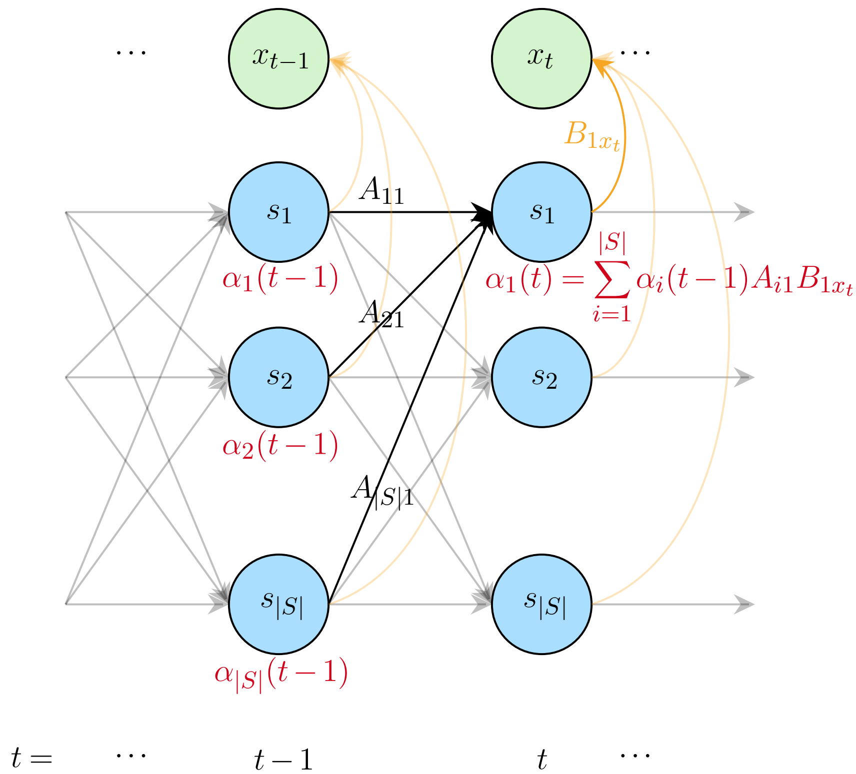
1 Visualize The Forward Procedure

The following is the visualized process for the forward procedure described in `http://cs229.stanford.edu/section/cs229-hmm.pdf`.



2 Derivation of The Viterbi Algorithm Procedure

The following shows how we uncovered the hidden state from an observed output using the famous dynamic programming algorithm, the Viterbi algorithm. First we need to define the notations used in this material.

- V: the output alphabet $V = \left\{v_1, v_2, \cdots, v_{|V|}\right\}$;
- S: the state alphabet $S = \left\{s_1, s_2, \cdots, s_{|S|}\right\}$;
- x: a series of observed outputs $x = (x_1, x_2, \cdots, x_T)$ such that $x_i \in V,\ i = 1, 2, \cdots, T$;
- z: a series of hidden states $z = (z_1, z_2, \cdots, z_T)$ such that $z_i \in S,\ i = 1, 2, \cdots, T$;
- π: an array of initial distribution $\pi = \left(\pi_1, \pi_2, \cdots, \pi_{|S|}\right)$ such that π_i stores the probability that $z_1 = s_i$;
- A: the transition matrix A of size $|S| \times |S|$ such that A_{ij} stores the probability of transiting from state s_i to state s_j;
- B: the emission matrix B of size $|S| \times |V|$ such that B_{ij} stores the probability of observing v_j from state s_i.

2.1 Define Subproblems to Be Solved

Define:

- $\text{OPT\_VALUE}\,(j,t)$: stores the maximum probability that can be achieved for the fact that the hidden state at time t is s_j, which means $z_t = s_j$, with the corresponding observed output is x_t;
- $\text{OPT}(j,t)$: stores the hidden state at time $t-1$ which later produces $\text{OPT\_VALUE}\,(j,t)$ at time t.

2.2 The Recurrence Relationship for Subproblems

Based on the defined subproblems in section 2.1, one can write down the recurrence equations below.

$$\begin{aligned} \text{OPT\_VALUE}(j,t) &= \max_i \Big(\{\text{OPT\_VALUE}\,(i,t-1) \cdot A_{ij} \cdot B_{jx_t},\ i = 1,2,\cdots,|S|\} \Big) \\ \text{OPT}(j,t) &= \arg\max_i \Big(\{\text{OPT\_VALUE}\,(i,t-1) \cdot A_{ij} \cdot B_{jx_t},\ i = 1,2,\cdots,|S|\} \Big) \end{aligned} \tag{1}$$

2.3 Pseudocode

Based on the recurrence formulas in section (1), one can write down the pesudocode below.

Algorithm 1 Viterbi Algorithm

function $\textsc{Viterbi}(V, S, \pi, x, A, B)$: z
    # Initialize base case
    for each state $i = 1, 2, \cdots, |S|$ **do**
        $\text{OPT\_VALUE}(i, 1) = \pi_i \cdot B_{ix_1}$
        $\text{OPT}(i, 1) = 0$ # Here, 0 means that we use s_0, which is not in the state alphabet $|S|$, to represent the initial hidden state before the first output is observed at time 1.
    end for

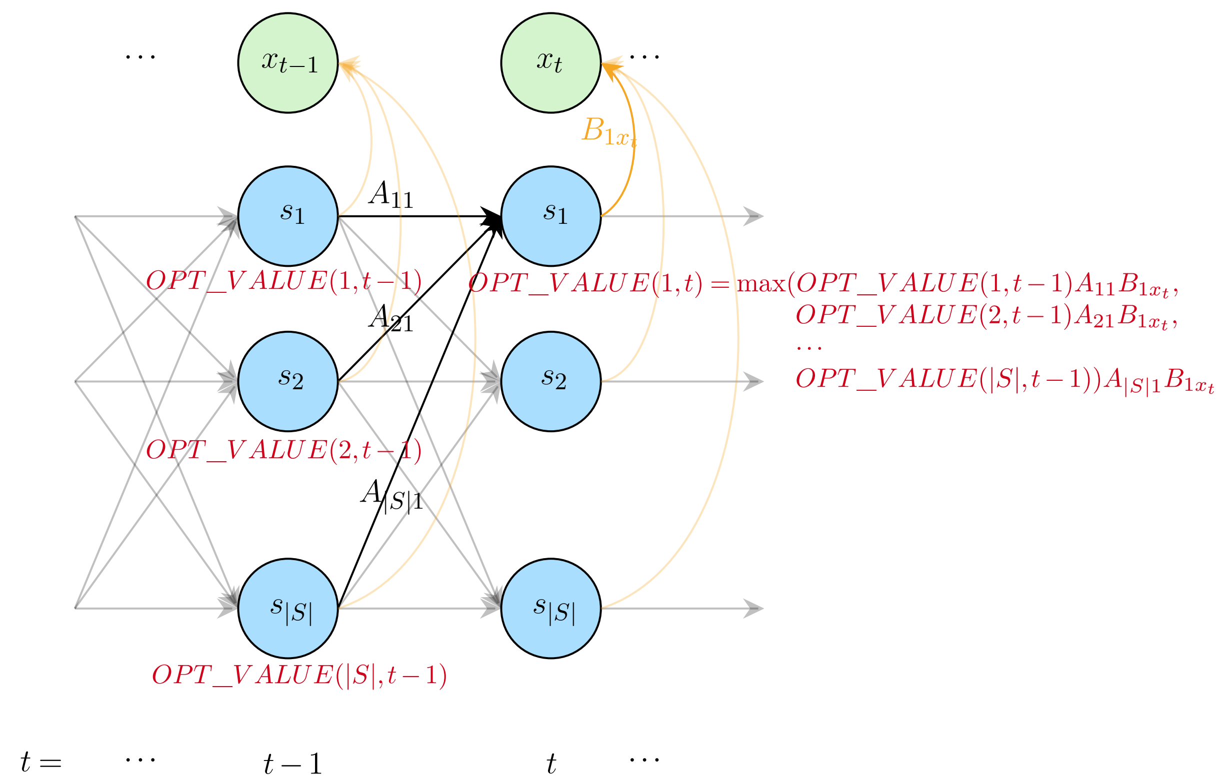
    # Recurrence process
    for time $t = 2, 3, \cdots, T$ **do**
        for each state $j = 1, 2, \cdots, |S|$ **do**
            $\text{OPT\_VALUE}(j, t) = \max\limits_i \Big(\{\text{OPT\_VALUE}(i, t-1) \cdot A_{ij} \cdot B_{jx_t},\ i = 1, 2, \cdots, |S|\} \Big)$
            $\text{OPT}(j, t) = \arg\max\limits_i \Big(\{\text{OPT\_VALUE}(i, t-1) \cdot A_{ij} \cdot B_{jx_t},\ i = 1, 2, \cdots, |S|\} \Big)$
        end for
    end for

    # Uncover the hidden states
    init an empty array z of size T
    $k \leftarrow \arg\max\limits_m \Big(\{\text{OPT\_VALUE}(i, T),\ i = 1, 2, \cdots, |S|\} \Big)$
    $z_T \leftarrow s_k$
    for time $t = T, T-1, \cdots, 2$ **do**
        $k \leftarrow \text{OPT}(k, t)$
        $z_{t-1} = s_k$
    end for

    return z
end function

3 Visualize The Viterbi Algorithm Procedure

The following is the visualized process for the forward procedure described in section 2.3.



$\cdots$
x_{t-1}
x_t
$\cdots$
B_{1x_t}
s_1
A_{11}
s_1
$OPT\_VALUE(1,t-1)$
$OPT\_VALUE(1,t) = \max(OPT\_VALUE(1,t-1)A_{11}B_{1x_t},$
$OPT\_VALUE(2,t-1)A_{21}B_{1x_t},$
$\ldots$
$OPT\_VALUE(|S|,t-1))A_{|S|1}B_{1x_t}$
A_{21}
s_2
s_2
$OPT\_VALUE(2,t-1)$
$A_{|S|1}$
$s_{|S|}$
$s_{|S|}$
$OPT\_VALUE(|S|,t-1)$
$t =$
$\cdots$
$t-1$
t
$\cdots$