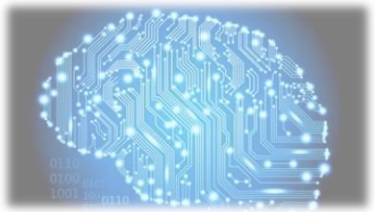
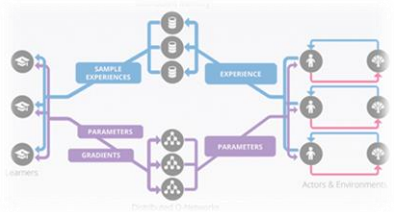
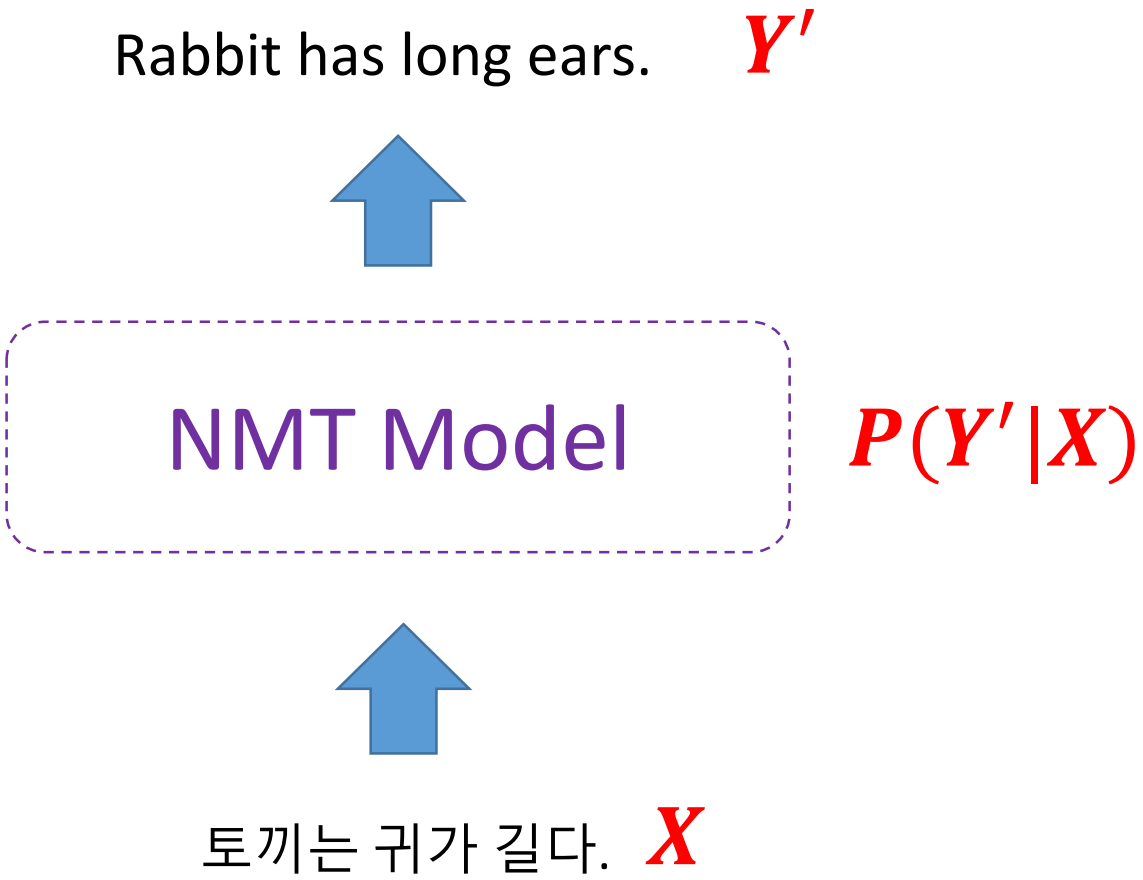



Neural Machine Translation using GANs

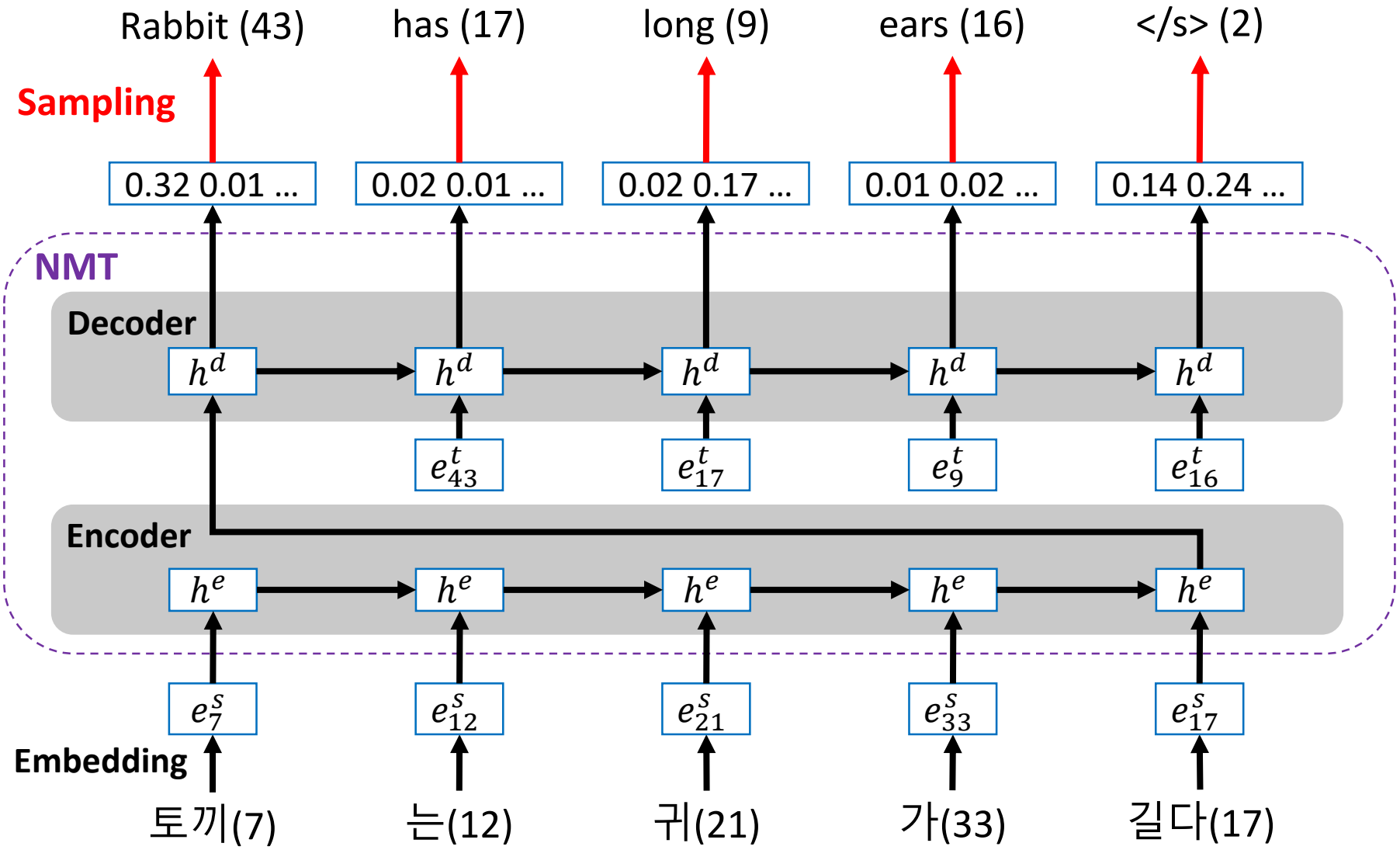
박재균

2019. 07. 02.

1.1 Neural Machine Translation (NMT)

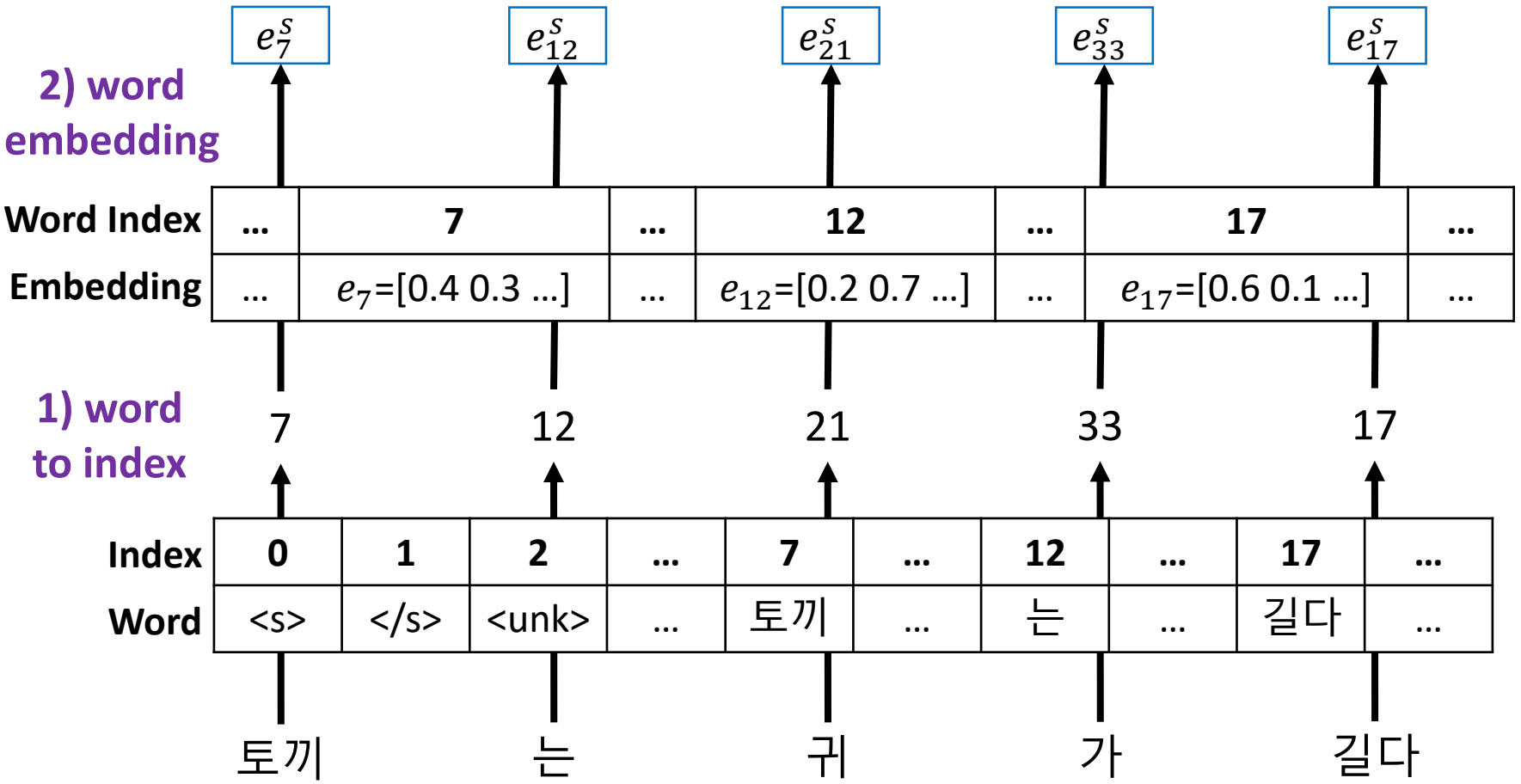


1.1. Neural Machine Translation (NMT)



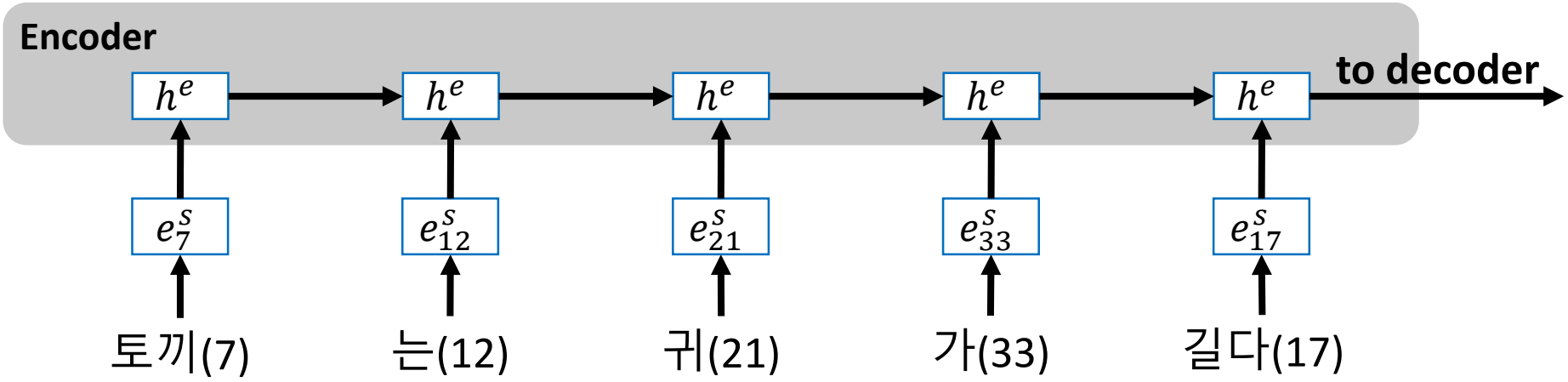
1.1. Neural Machine Translation (NMT)

- **Word Embedding**



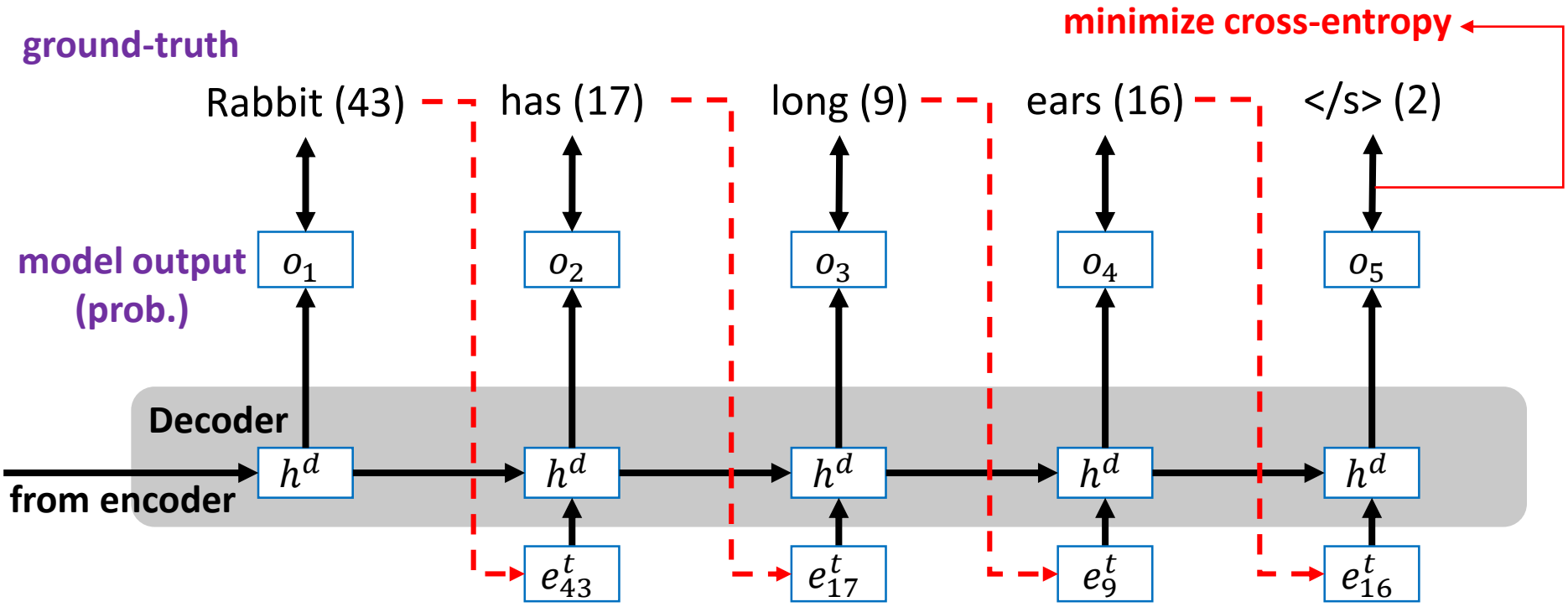
1.1. Neural Machine Translation (NMT)

- **Encoding**



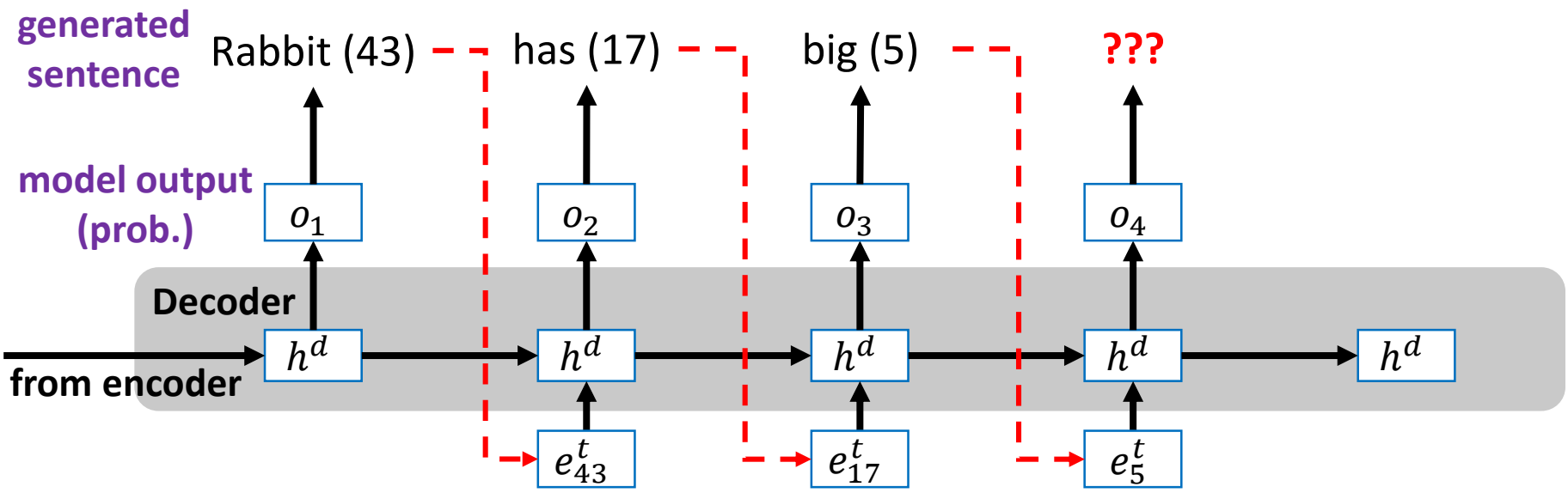
1.1. Neural Machine Translation (NMT)

- **Decoder : Training step**



1.1. Neural Machine Translation (NMT)

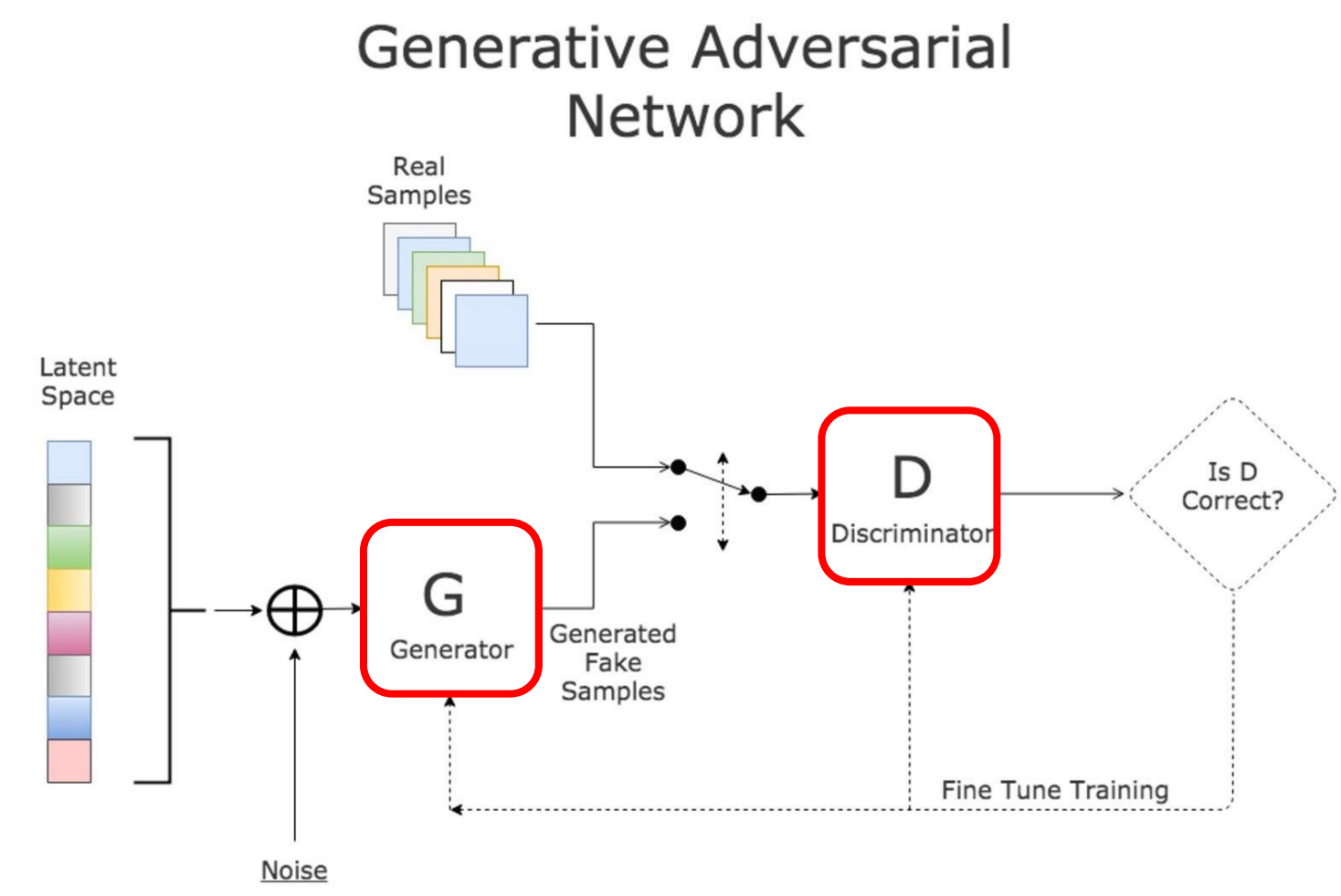
- **Decoder : Inference (test) step**
  - if “Rabbit has big” has never been the words in training data, the following could happen:



- **Exposure bias** : Because the number of training data is limited, it happens when the model generates words that have never been exposed during the training step.

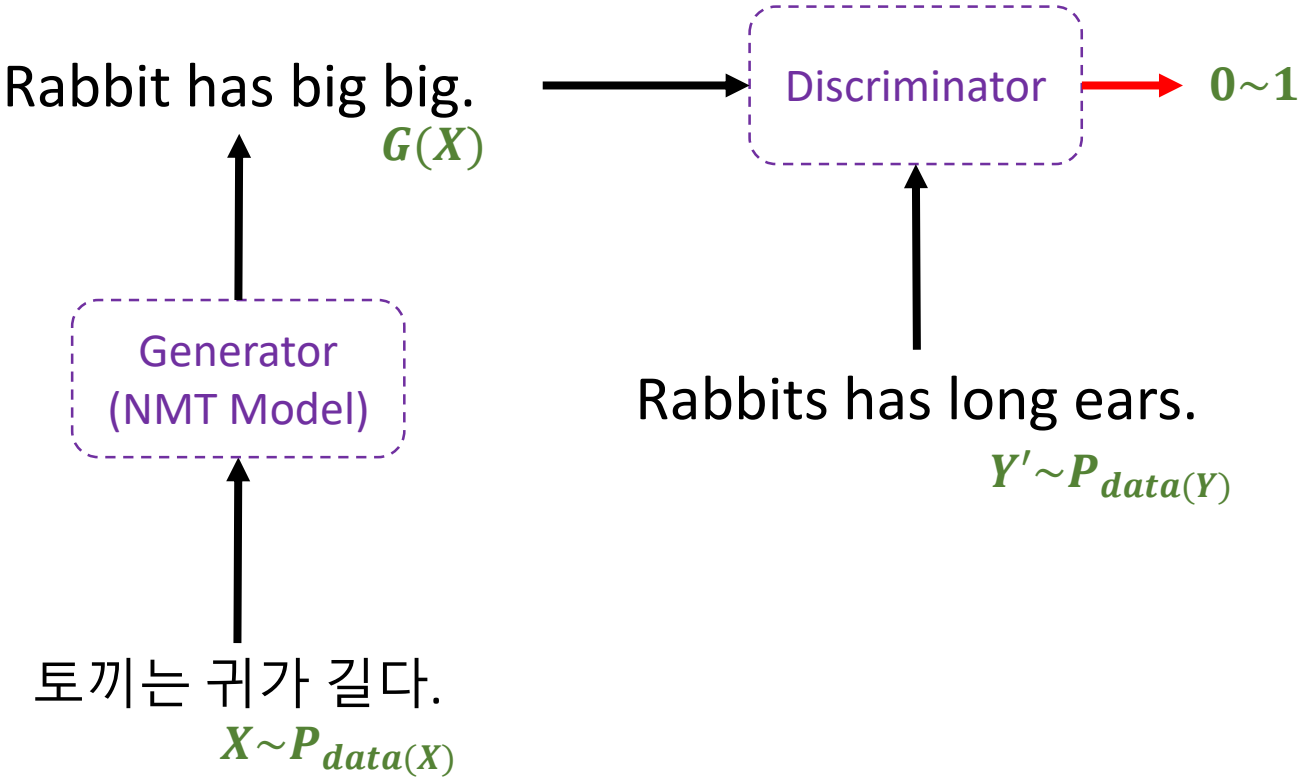
1.2. Generative Adversarial Networks (GANs)

- It is a generative model that generates data that is similar to the existing one by learning the data distribution.



2.1. Training NMT model by using GANs

$$\min_G \max_D V(D,G) = \mathbb{E}_{X,Y' \sim P_{data(X,Y')}}[\log D(X,Y')] + \mathbb{E}_{X \sim P_{data(X)}}[\log(1 - D(X, G(X)))]$$



2.1. Training NMT model by using GANs

$$\min_G \max_D V(D,G) = \mathbb{E}_{X,Y' \sim P_{data(X,Y')}}[\log D(X,Y')] + \mathbb{E}_{X \sim P_{data(X)}}[\log(1 - D(X, G(X)))]$$

$$G(X) = [a^{(1)}, \ldots, a^N]$$

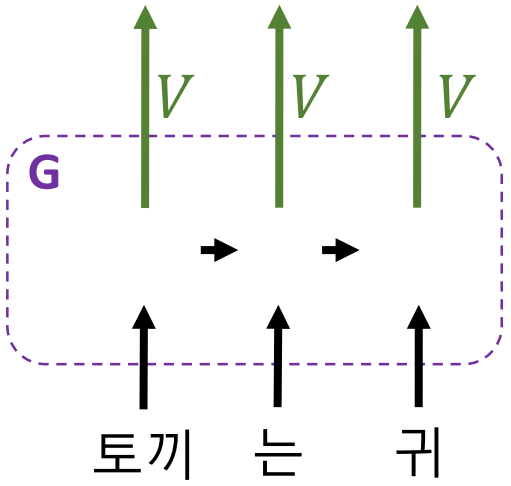
$a^{(t)}$ | 1 | 0 | 0 | ...

Sampling by stochastic selection

$o^{(t)}$ | 0.21 | 0.01 | 0.13 | ...

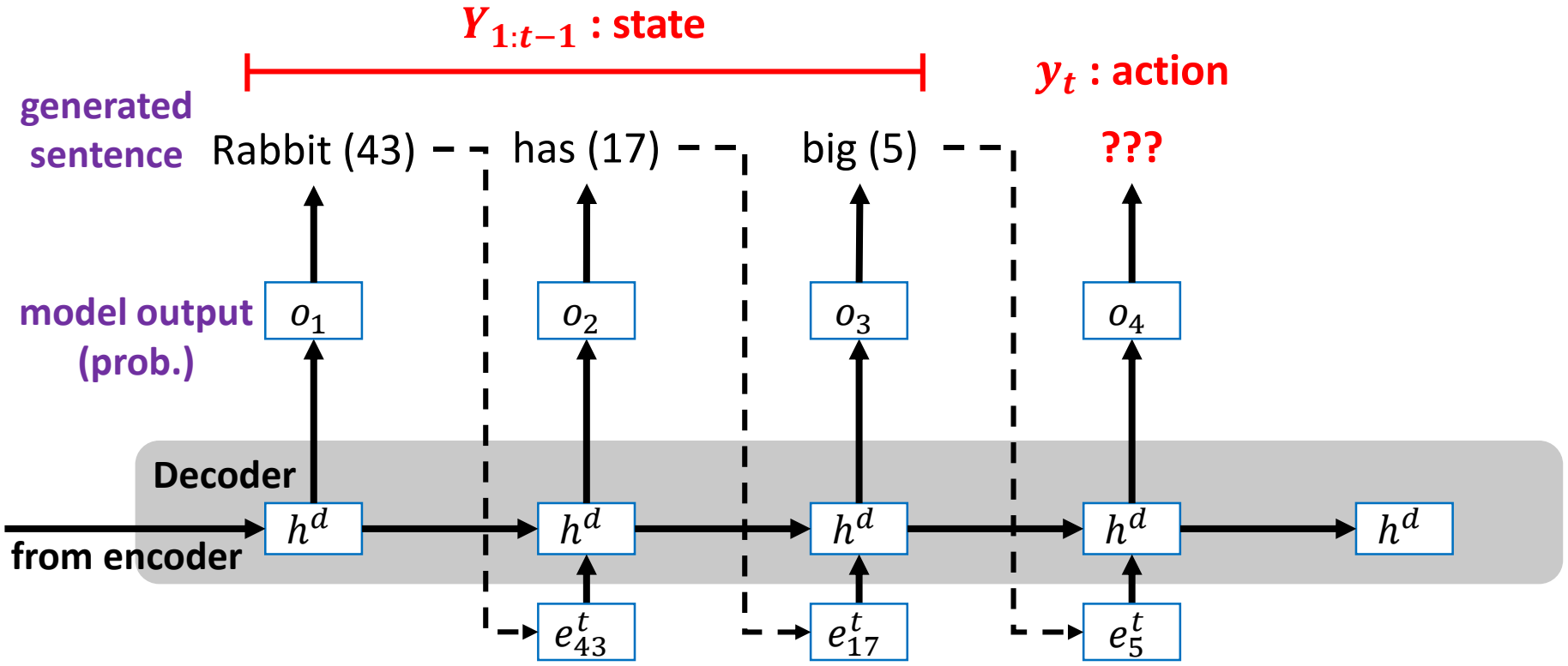
$$\frac{\partial D(G(X))}{\partial V} = \frac{\partial D(G(X))}{\partial G(X)} \sum_t \frac{\partial G(x)}{\partial a^{(t)}} \cdot \frac{\partial a^{(t)}}{\partial o^{(t)}} \cdot \frac{\partial o^{(t)}}{\partial V}$$

Non-differentiable



2.1. Training NMT model by using GANs

- **Generator Training by Reinforcement learning** (Yu et al., 2017)

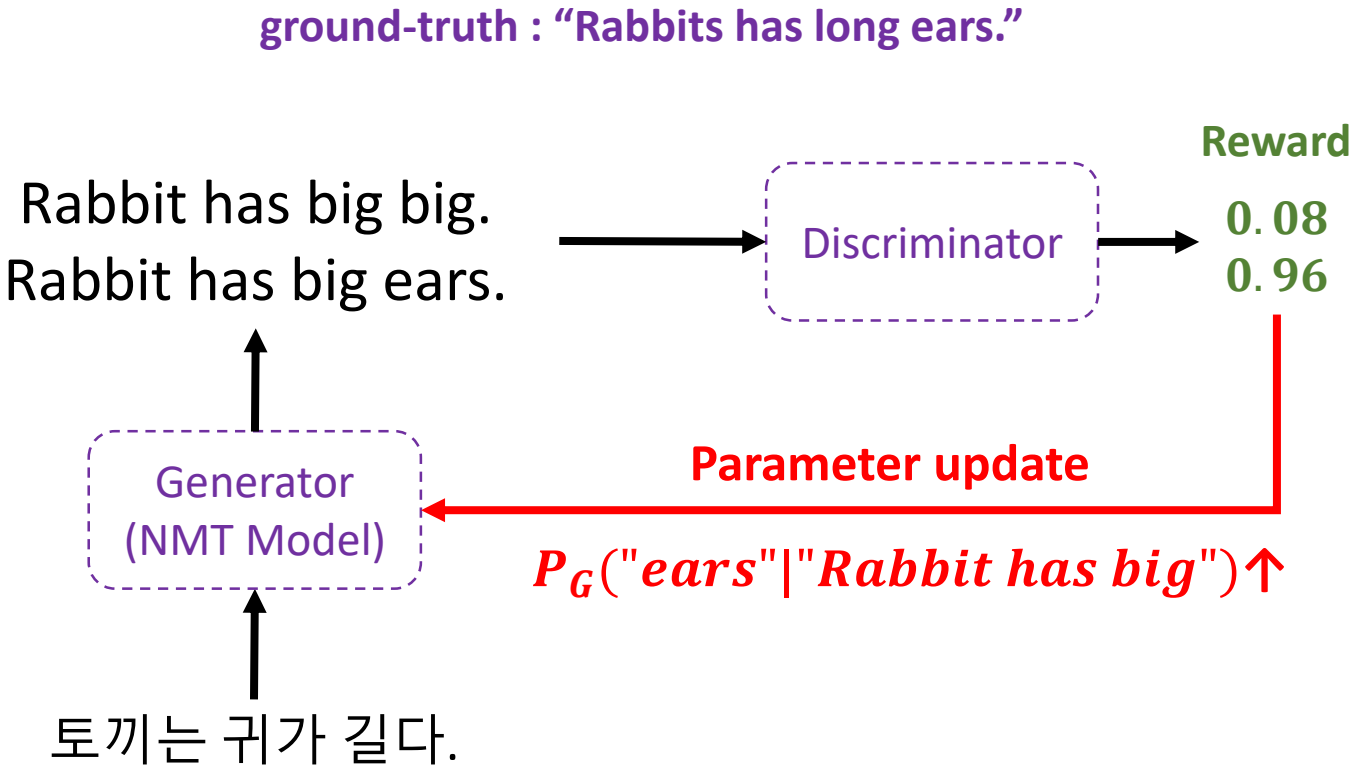


- y_t is generated by NMT model (policy)

2.1. Training NMT model by using GANs

- **Generator Training by Reinforcement learning**

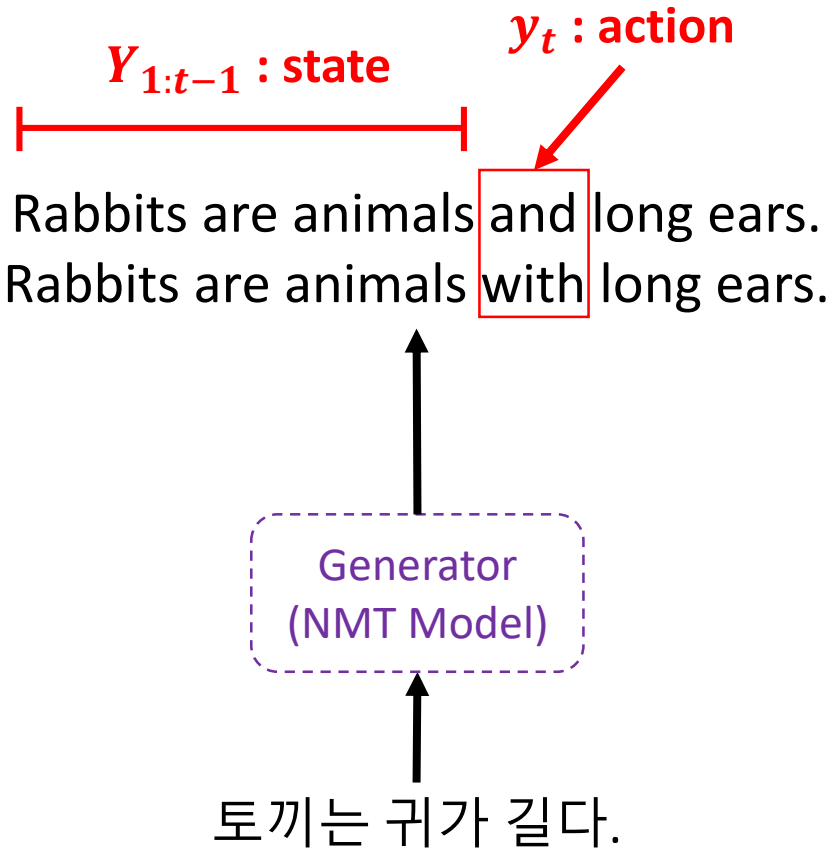
  When $t = T$



2.1. Training NMT model by using GANs

- **Generator Training by Reinforcement learning**

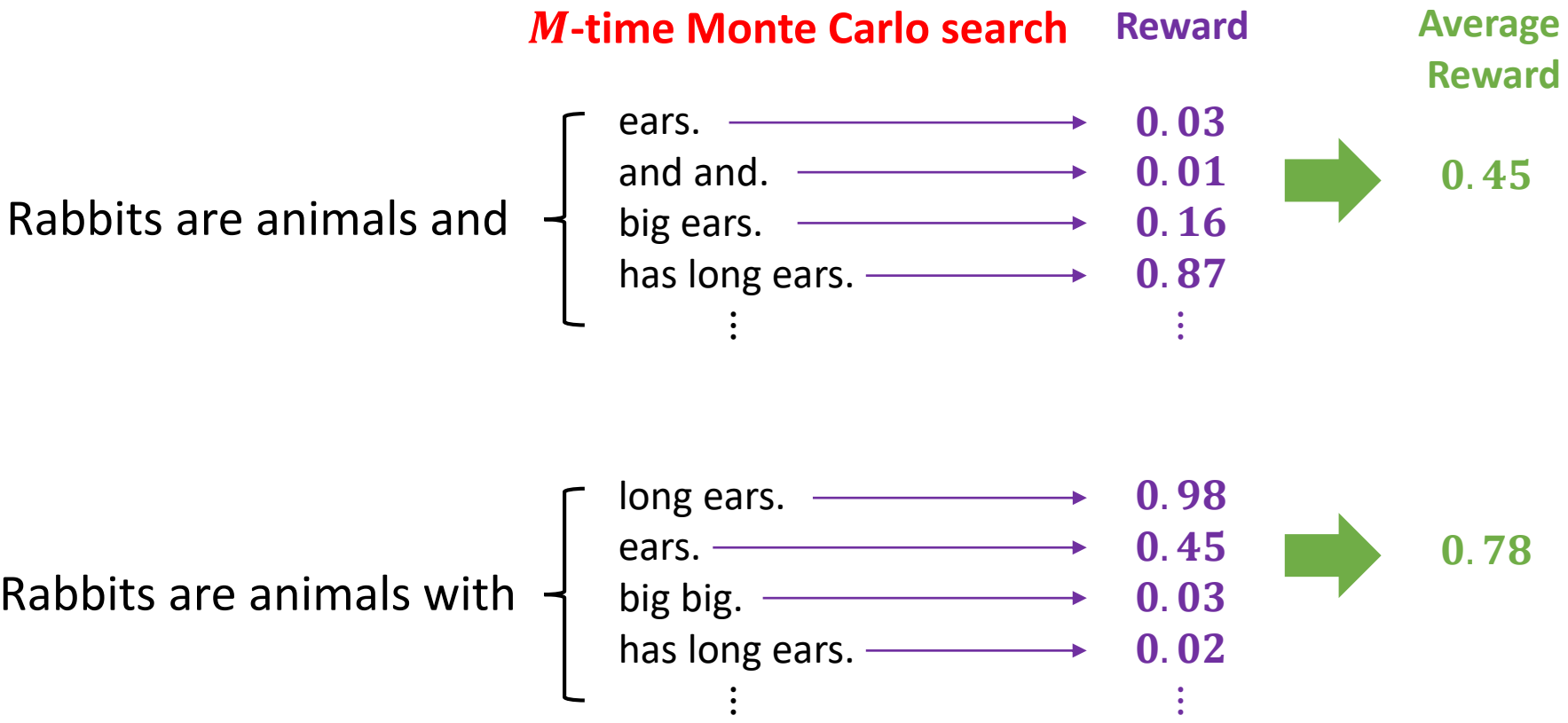
  When $t < T$



2.1. Training NMT model by using GANs

- **Generator Training by Reinforcement learning**

  When $t < T$



$P_G("with"|"Rabbit\ are\ animals")\uparrow$

2.2. The issue of M-time Monte Carlo search

- **It takes too much time to compute rewards for generated sentence.**

When "Rabbits are animal with long ears." is generated by NMT model.

Rabbits are animal with long ears.
MC search

Rabbits are animal with long ears.
MC search

$$O(T * M)$$

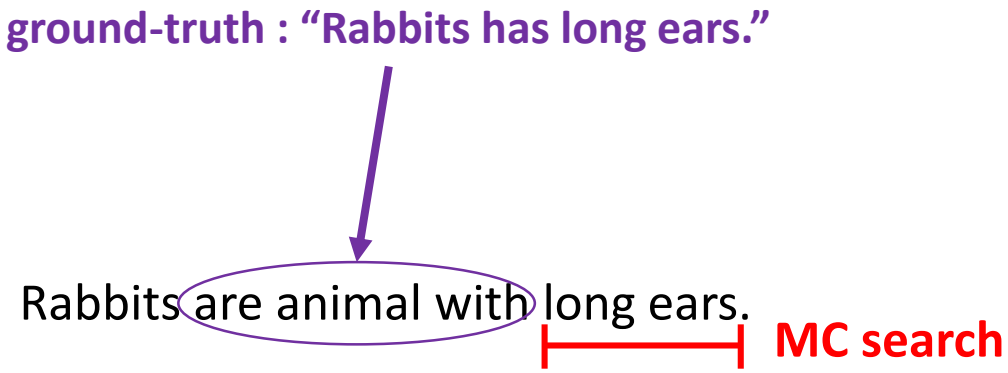
Rabbits are animal with long ears.
MC search

Rabbits are animal with long ears.
MC search

Rabbits are animal with long ears.
MC search

3. Proposed method

- **It computes the reward only for the words that correspond to the exposure bias.**



4. Experiment

- **Data**
  - IWSLT'15 English-Vietnamese data
  - Train (133K sentence pairs)
  - Vocab. (Eng. 17K and Viet. 7.7K). The words whose frequencies are less than 5 by <unk>

- **Generator**
  - Embedding dim. : 512
  - Max length : 50
  - Recurrent unit : gated recurrent unit (GRU)
  - The number of units : 512
  - Attention mechanism : (Luong et al., 2015)
  - Gradient clipping : -5, 5
  - The number of layers : 1

4. Experiment

- **Discriminator**
  - Embedding dim. : 512
  - Same as (Yu et al., 2017)

- **GANs Training**
  - Mini-batch size : 128
  - MC search : 10

4. Experiment

1. **Pre-training NMT model (Complete)**

   - epoch 30 (8 hours)

2. **Pre-training Discriminator (Complete)**

   - Termination condition : Accuracy > 0.82

3. **Training GAN (Ongoing)**

   1) Training NMT model by MC search : 1 epoch

   2) Training discriminator until Accuracy > 0.82

   1), 2) repetition