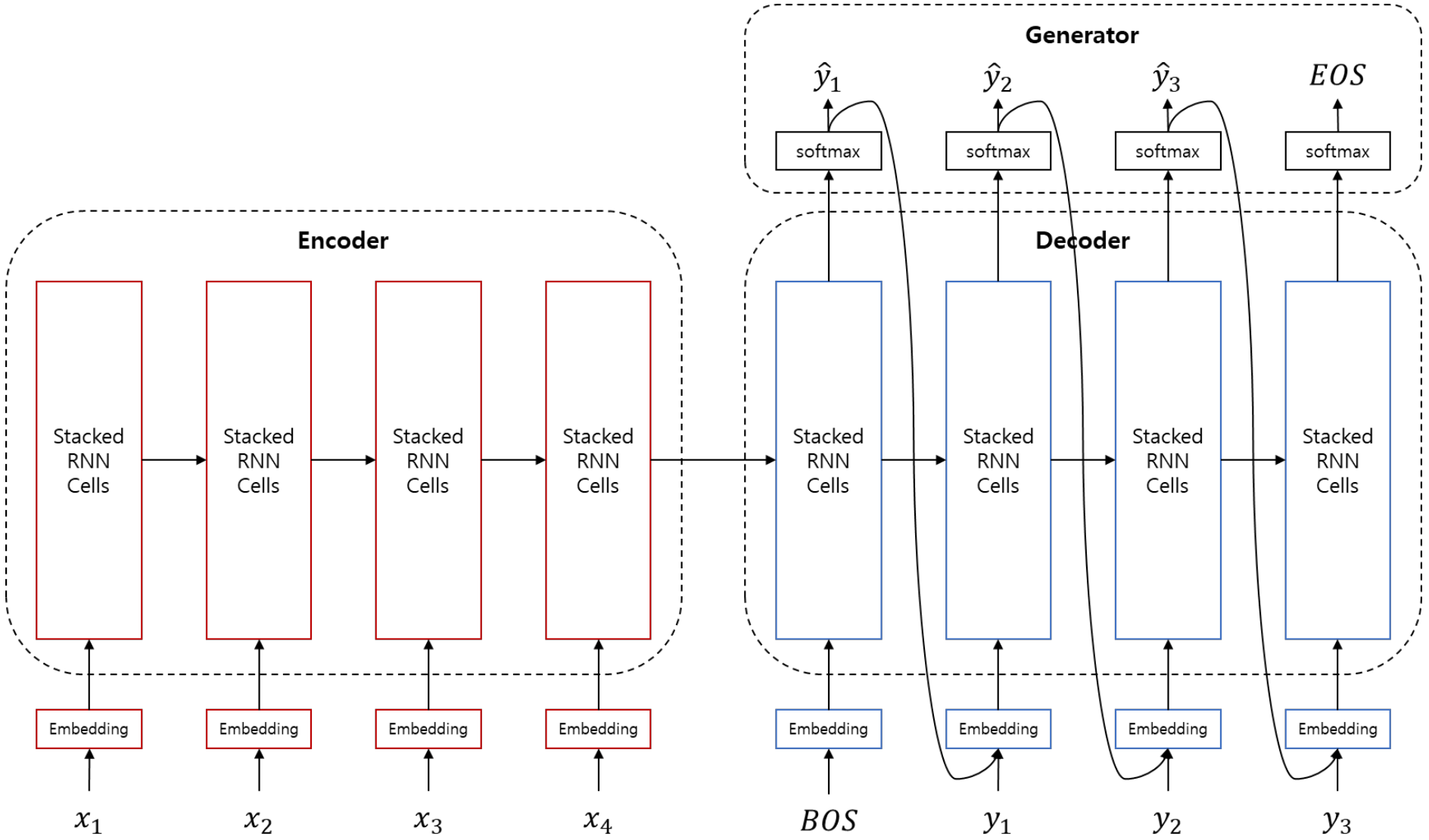
Sequence to Sequence: Generator

Ki Hyun Kim

nlp.with.deep.learning@gmail.com

Sequence to Sequence



Equations

- Given dataset,

$$\mathcal{D} = \{x^i, y^i\}_{i=1}^N$$
$$x^i = \{x_1^i, \cdots, x_m^i\} \text{ and } y^i = \{y_0^i, y_1^i, \cdots, y_n^i\},$$
$$\text{where } y_0 = <\text{BOS}> \text{ and } y_n = <\text{EOS}>.$$

- Hidden states from decoder can be calculated like as below:

$$h_t^{\text{dec}} = \text{RNN}_{\text{dec}}(\text{emb}_{\text{dec}}(\hat{y}_{t-1}), h_{t-1}^{\text{dec}}),$$
$$\text{where } h_0^{\text{dec}} = h_m^{\text{enc}}.$$

- Generator returns a probability distribution of current output token.

$$\hat{y}_t = \text{softmax}(h_t^{\text{dec}} \cdot W_{\text{gen}}),$$
$$\text{where } h_t^{\text{dec}} \in \mathbb{R}^{\text{batch\_size} \times 1 \times \text{hidden\_size}} \text{ and } W_{\text{gen}} \in \mathbb{R}^{\text{hidden\_size} \times |V|}.$$

Loss Function

- We need to minimize negative log-likelihood,

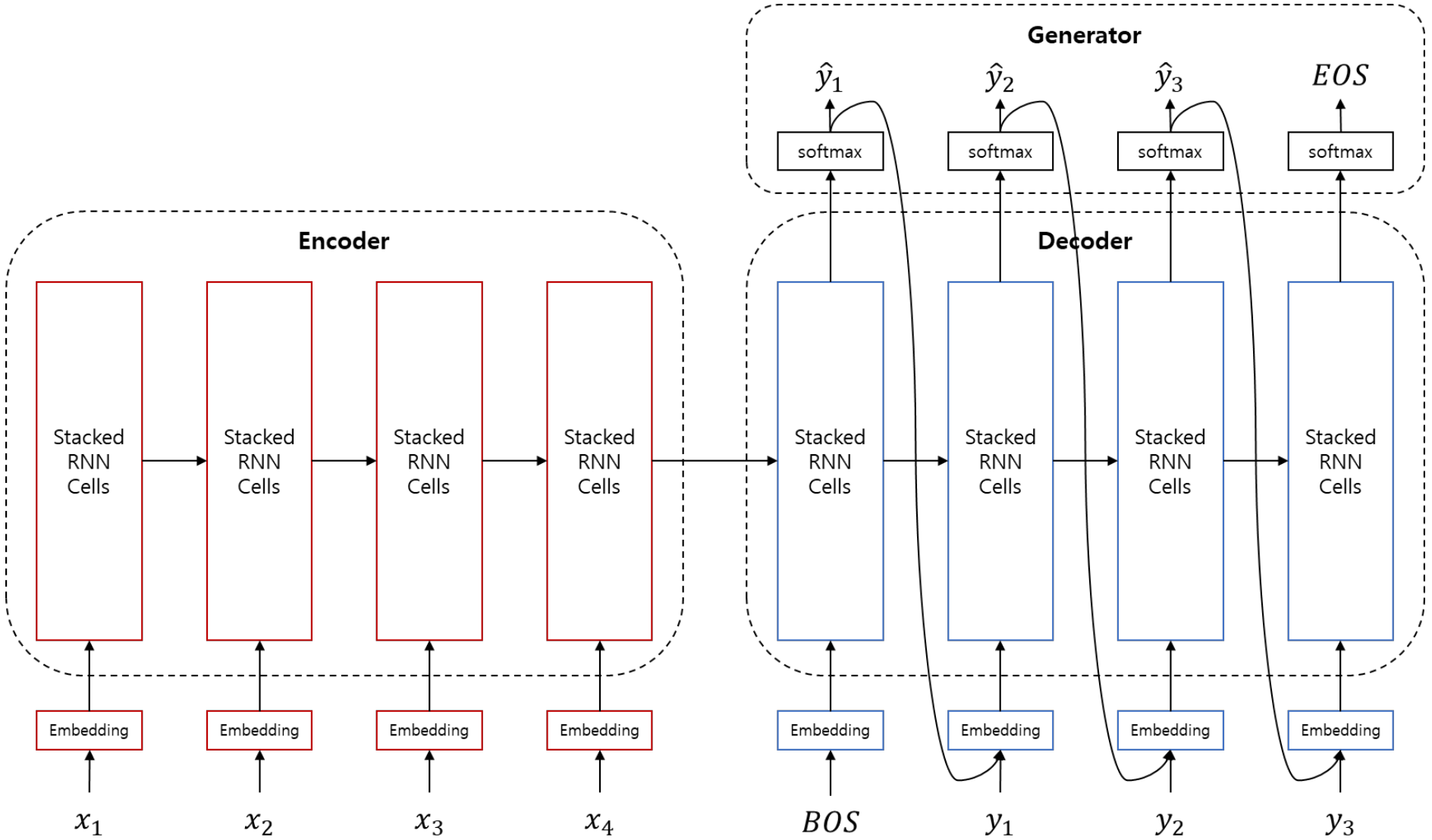
$$\mathcal{L}(\theta) = -\sum_{i=1}^{N} \log P(y^i|x^i;\theta)$$
$$= -\sum_{i=1}^{N}\sum_{j=1}^{n} \log P(y^i_j|x^i, y^i_{<j};\theta)$$

- Log-likelihood can be calculated like as below:

$$\log P(y_t|x, y_{<t}) = y_t^T \cdot \log \hat{y}_t,$$

where y_t is one-hot vector, and $\hat{y}_t$ is a probability distribution from softmax.

<BOS> and <EOS>



Summary

- Generator는 디코더의 hidden state를 받아
  현재 time-step의 출력 token에 대한 확률 분포(multinoulli distribution) 반환

- 단어를 선택하는 문제이므로 cross entropy loss를 통해 최적화 가능
  - GT 분포와 모델 분포 사이의 차이를 최소화 하기 위함
  - 조건부 언어모델로 볼 수 있으므로, PPL로 치환 가능