Language Model & Fusioning

Winter Vacation Capstone Study

TEAM Kai.Lib

발표자 : 배세영

2020.02.17 (MON)

언어 모델(Language Model)이란

- 단어 시퀀스에 확률을 할당하는 일을 하는 모델 (가장 자연스러운 단어 시퀀스를 찾아낸다)
- 가장 보편적으로 사용되는 방법은 이전 단어들이 주어졌을 때 다음 단어를 예측하도록 하는 것
- 기계 번역(Machine Translation)
  - P('나는 버스를 탔다') > P('나는 버스를 태운다')
- 오타 교정(Spell Correction)
  - 선행 문장 : "선생님이 교실로 부리나케"
  - P('달려갔다') > P('잘려갔다')
- 음성 인식(Speech Recognition)
  - P('나는 메롱을 먹었다') < P('나는 메론을 먹었다')
- 이처럼 확률 값을 기반으로 보다 적절한 문장을 판단하는 역할을 한다

언어 모델(Language Model)이란

- 기본적으로는 조건부 확률을 사용

$$P(w_n|w_1,...,w_{n-1})$$

- 전체 단어 시퀀스 W의 확률은 모든 단어가 예측되고 나서야 알 수 있으므로

$$P(W) = P(w_1, w_2, w_3, w_4, w_5, \ldots w_n) = \prod_{i=1}^{n} P(w_n|w_1,...,w_{n-1})$$

ex)

$$P(\text{An adorable little boy is spreading smiles}) =$$
$$P(\text{An}) \times P(\text{adorable}|\text{An}) \times P(\text{little}|\text{An adorable}) \times P(\text{boy}|\text{An adorable little}) \times P(\text{is}|\text{An adorable little boy})$$
$$\times P(\text{spreading}|\text{An adorable little boy is}) \times P(\text{smiles}|\text{An adorable little boy is spreading})$$

통계적 언어 모델(Statistical Language Model, SLM)

- 출현 빈도(Count) 기반

$$P(\text{is}|\text{An adorable little boy}) = \frac{\text{count}(\text{An adorable little boy is})}{\text{count}(\text{An adorable little boy })}$$

- 모델이 학습한 데이터에서 'An adorable little boy'라는 문장과
  'An adorable little boy is'라는 문장이 등장한 횟수를 비교하여 확률 값을 결정

ex)
count(An adorable little boy) = 100, count(An adorable little boy is) = 300이라고 가정하면

P(is I An adorable little boy) = 30/100 = 30%

통계적 언어 모델(Statistical Language Model, SLM)

- 희소 문제(Sparsity Problem)
  - 훈련 데이터에 없는 단어 시퀀스에 대한 확률 계산값에 오류가 발생하는 문제

$$P(\text{is}|\text{An adorable little boy}) = \frac{\text{count}(\text{An adorable little boy is})}{\text{count}(\text{An adorable little boy })}$$

  - 분자, 혹은 분모가 0이 되어 전체 확률값이 0이 되거나 정의되지 않는 문제가 생김

- 희소 문제는 단어 시퀀스가 길어질수록 더 심해짐

- 이를 해결하기 위하여 도입되는 개념이 n-gram 언어 모델

n-gram 언어 모델(n-gram Language Model)

- SLM에서 다음으로 올 단어 예측에 필요한 단어의 수를 조정하여 희소 문제가 발생할 가능성을 줄이는 기법

$$P(\text{is}|\text{An adorable little boy}) \approx P(\text{is}|\text{boy})$$

$$P(\text{is}|\text{An adorable little boy}) \approx P(\text{is}|\text{little boy})$$

- 몇 개의 단어까지 참고하여 결정할 것인지 (window size)에 따라
  - uni-gram
  - bi-gram
  - tri-gram
  - n-gram
  - .....

n-gram 언어 모델(n-gram Language Model)

- n-gram 언어 모델의 한계
  - 희소 문제가 줄어들 뿐 여전히 존재함
  - n을 선택하는 것의 trade-off (5 이하로 잡아야 한다고 권장)
    - n을 높게 설정하면 :
      - 보다 넓은 window를 통해 다음 단어를 보다 높은 정확도로 예측 가능
      - 고려해야 하는 단어 시퀀스의 길이가 길어져 해당 시퀀스가 데이터 상에 없을 가능성이 높아짐
    - n을 낮게 설정하면 :
      - window size가 작아지므로 단어 예측의 정확도가 떨어짐
      - 고려해야 하는 단어 시퀀스의 길이가 짧아지므로 희소 문제가 완화됨

-	Unigram	Bigram	Trigram
Perplexity	962	170	109

피드포워드 신경망 언어 모델(Neural Network Language Model)

- 희소 문제(Sparsity Problem)은 단어 간 유사도를 파악할 수 있다면 해결할 수 있음
  - 학습 데이터 : "보도 자료를 살펴보다" / "마라탕을 냠냠하다"
    - P(톺아보다 | 보도 자료를) = 0
    - P(냠냠하다 | 보도 자료를) = 0.000000001

- 단어 간 유사도를 반영하는 이 개념은 워드 임베딩(word embedding)의 기반이 됨

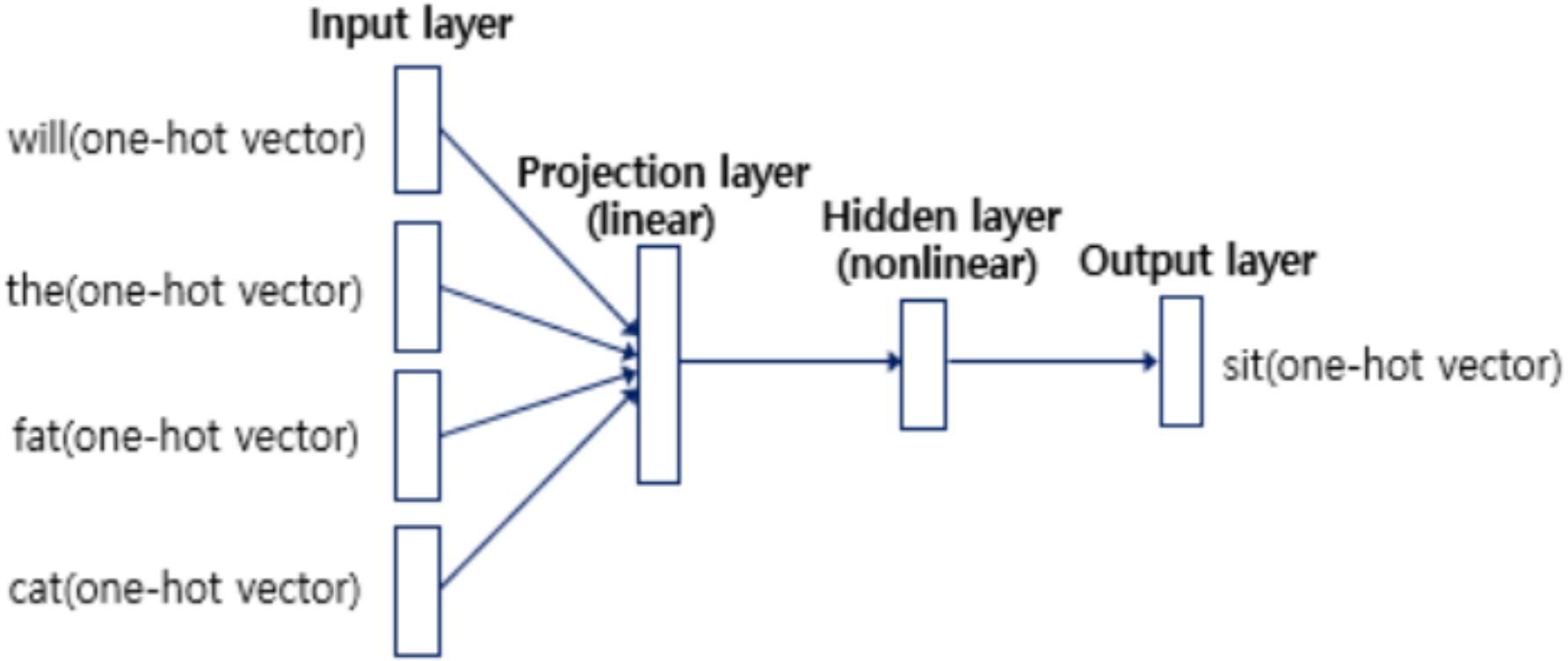
피드포워드 신경망 언어 모델(Neural Network Language Model)

- NNLM의 학습 과정
  - 예문 : "what will the fat cat sit on"
  - 1. 모든 단어를 인코딩 (one-hot vector)

```
what = [1, 0, 0, 0, 0, 0, 0]
will = [0, 1, 0, 0, 0, 0, 0]
the = [0, 0, 1, 0, 0, 0, 0]
fat = [0, 0, 0, 1, 0, 0, 0]
cat = [0, 0, 0, 0, 1, 0, 0]
sit = [0, 0, 0, 0, 0, 1, 0]
on = [0, 0, 0, 0, 0, 0, 1]
```

피드포워드 신경망 언어 모델(Neural Network Language Model)

- NNLM의 학습 과정
  - 예문 : "what will the fat cat sit on"
  - 4-gram



피드포워드 신경망 언어 모델(Neural Network Language Model)

- NNLM의 학습 과정
  - 예문 : "what will the fat cat sit on"
  - 4-gram

  - 1. 단어 인코딩 (one-hot-vector)

```
what = [1, 0, 0, 0, 0, 0, 0]
will = [0, 1, 0, 0, 0, 0, 0]
the = [0, 0, 1, 0, 0, 0, 0]
fat = [0, 0, 0, 1, 0, 0, 0]
cat = [0, 0, 0, 0, 1, 0, 0]
sit = [0, 0, 0, 0, 0, 1, 0]
on = [0, 0, 0, 0, 0, 0, 1]
```

피드포워드 신경망 언어 모델(Neural Network Language Model)

- NNLM의 학습 과정
  - 예문 : "what will the fat cat sit on"
  - 4-gram

  - 2. window size만큼의 단어를 투사층(projection layer, size = N)에 통과

x_{fat} × $W_{V\times M}$ = e_{fat}

0	0	0	1	0	0	0

×

0.5	2.1	1.9	1.5	0.8
0.8	1.2	2.8	1.8	2.1
0.1	0.8	1.2	0.9	0.7
2.1	1.8	1.5	1.7	2.7

=

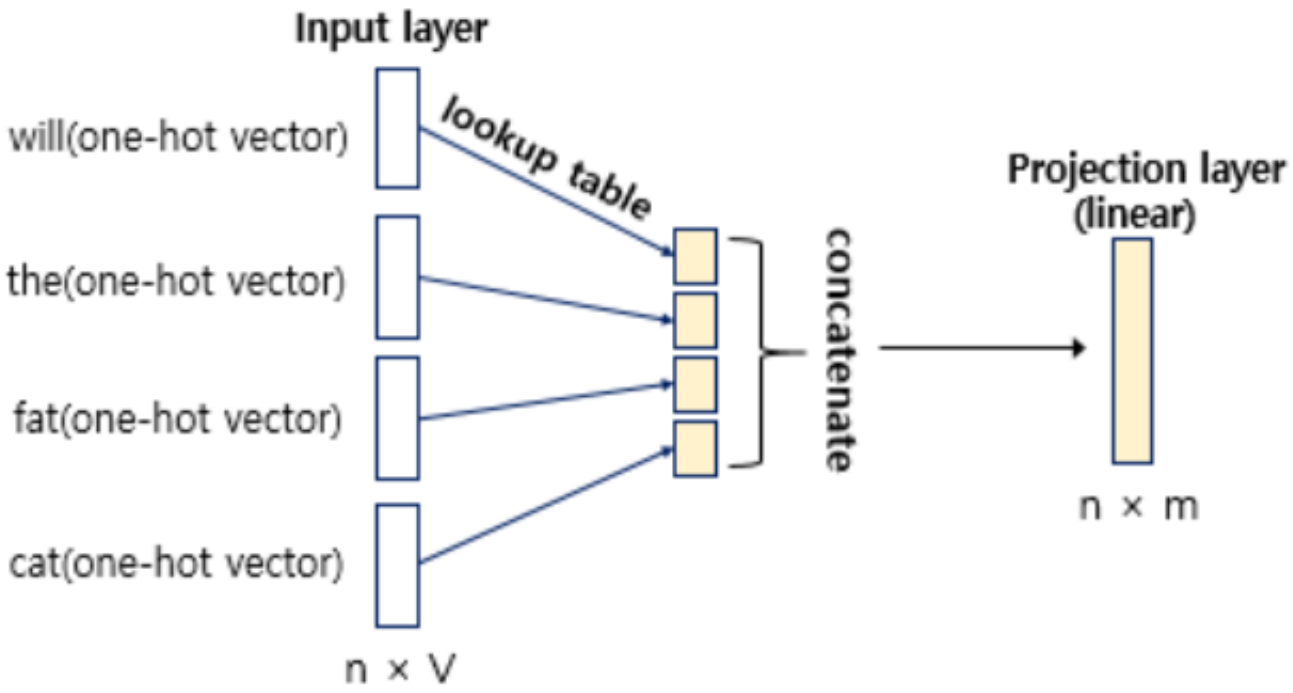
2.1	1.8	1.5	1.7	2.7

lookup table
(Embedding Vector)

피드포워드 신경망 언어 모델(Neural Network Language Model)

- NNLM의 학습 과정
  - 예문 : "what will the fat cat sit on"
  - 4-gram

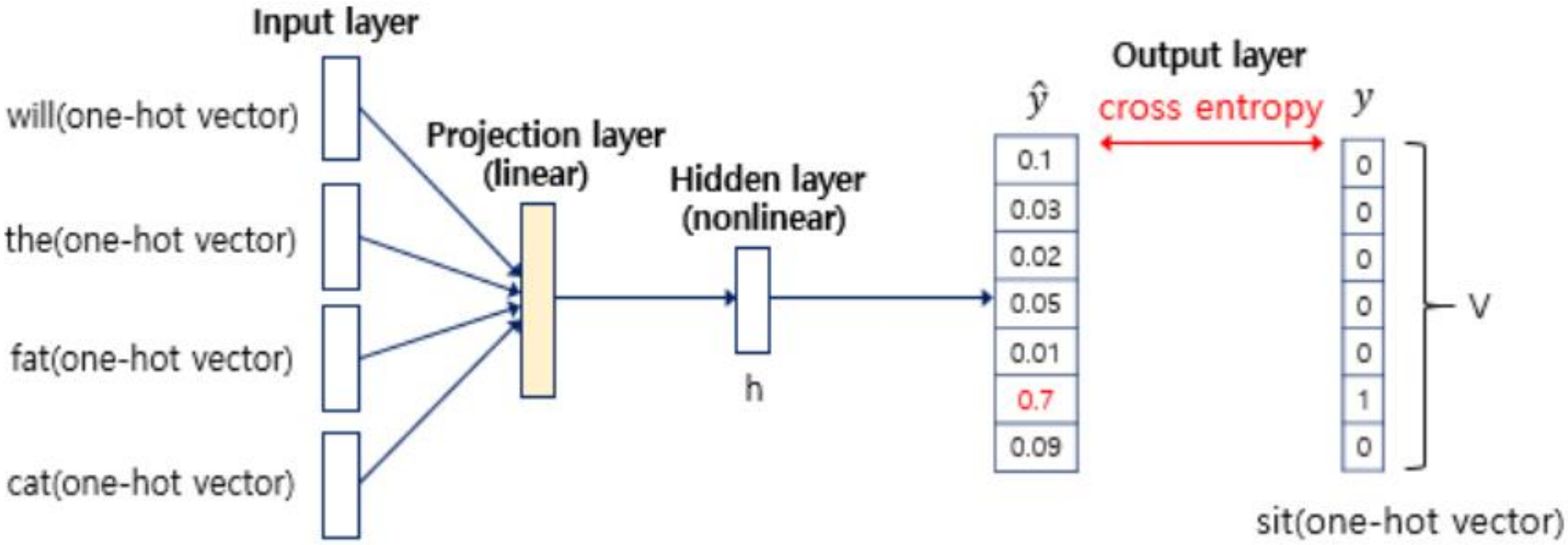
  - 3. n개의 embedding vector를 concatenate



피드포워드 신경망 언어 모델(Neural Network Language Model)

- NNLM의 학습 과정
  - 예문 : "what will the fat cat sit on"
  - 4-gram

  - 4. concatenated vector에 대하여 hidden layer 통과, softmax 수행

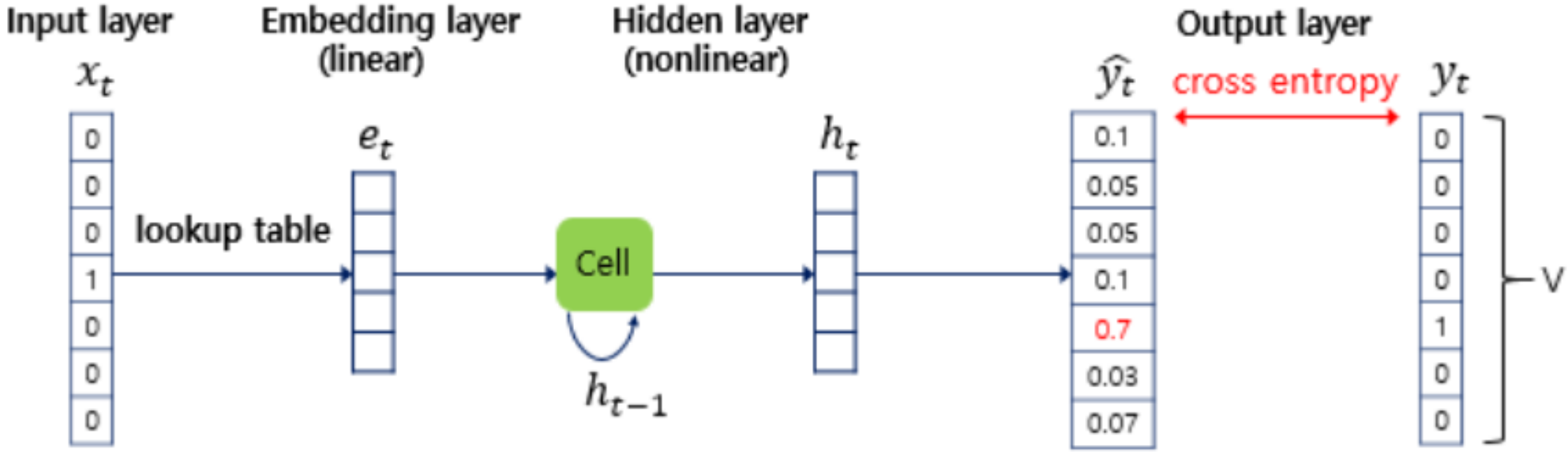


피드포워드 신경망 언어 모델(Neural Network Language Model)

- NNLM의 개선점과 한계
  - 개선점
    - Lookup table을 통한 단어 간의 유사성 학습 가능
    - 희소 문제 (sparsity problem) 해소
    - 일반 n-gram model보다 작은 크기의 저장 공간 필요
  - 한계
    - n을 결정하며 정해지는 window size에 따라 버려지는 단어들의 문맥 정보는 고려 불가
    - 고정 길이의 입력만 처리 가능
      -> RNN Language Model

순환 신경망 언어 모델(Recurrent Neural Network Language Model)

- NNLM의 고정 입력 한계를 탈피하고자 제안
- 기본적인 과정은 NNLM과 동일. RNN을 사용하여 가변 길이 입력을 처리할 수 있게 됨



Fusioning

- **Shallow/Deep/Cold Fusion 관련 논문 [Tencent AI Lab. 2019.May]**

COMPONENT FUSION: LEARNING REPLACEABLE LANGUAGE MODEL COMPONENT FOR END-TO-END SPEECH RECOGNITION SYSTEM

Changhao Shan[1,2*], Chao Weng[4], Guangsen Wang[3], Dan Su[3], Min Luo[2], Dong Yu[4], Lei Xie[1†]

[1]School of Computer Science and Engineering, Northwestern Polytechnical University, Xian, China
[2]Tencent AI Platform Department, Shenzhen, China
[3]Tencent AI Lab, Shenzhen, China
[4]Tencent AI Lab, Bellevue, USA

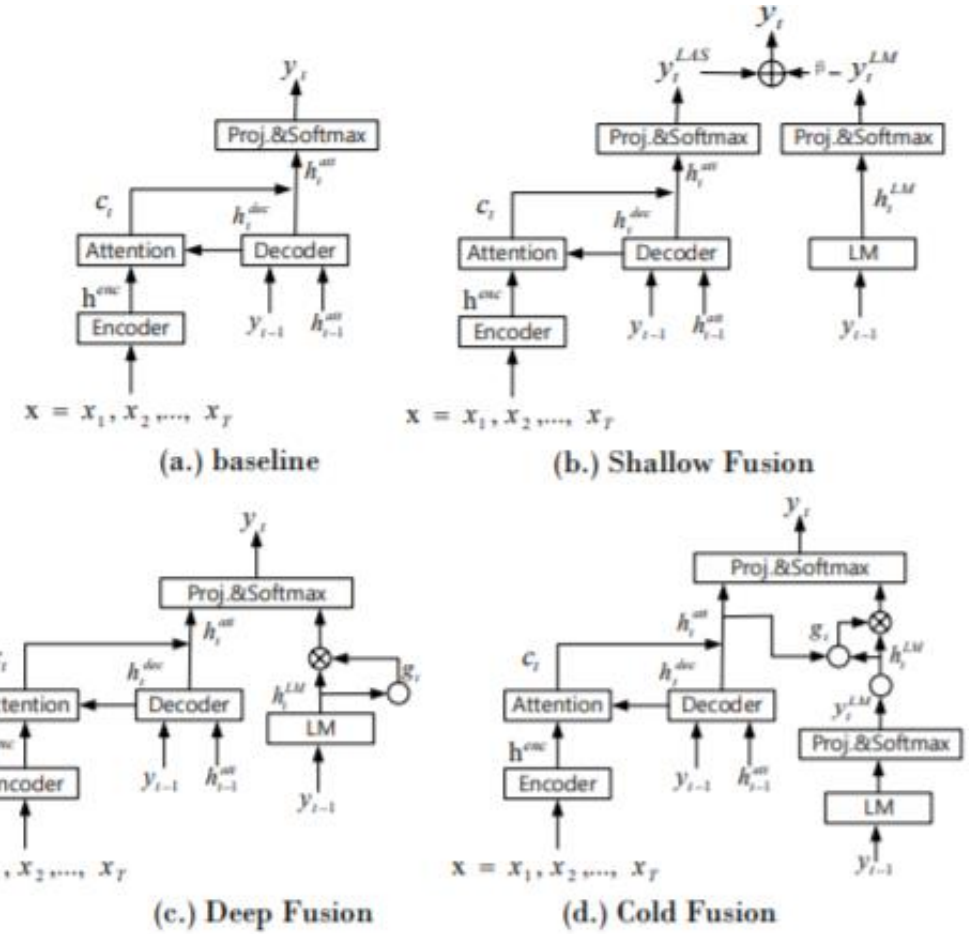
{chshan, lxie}@nwpu-aslp.org, {cweng, vincegswang, dansu, selwynluo, dyu}@tencent.com

Fusioning

- Shallow/Deep/Cold Fusion 관련 논문 [Tencent AI Lab. 2019.May]

- Attention 기반 seq2seq model (LAS)에서 LM을 활용하는 세 가지 방법 소개

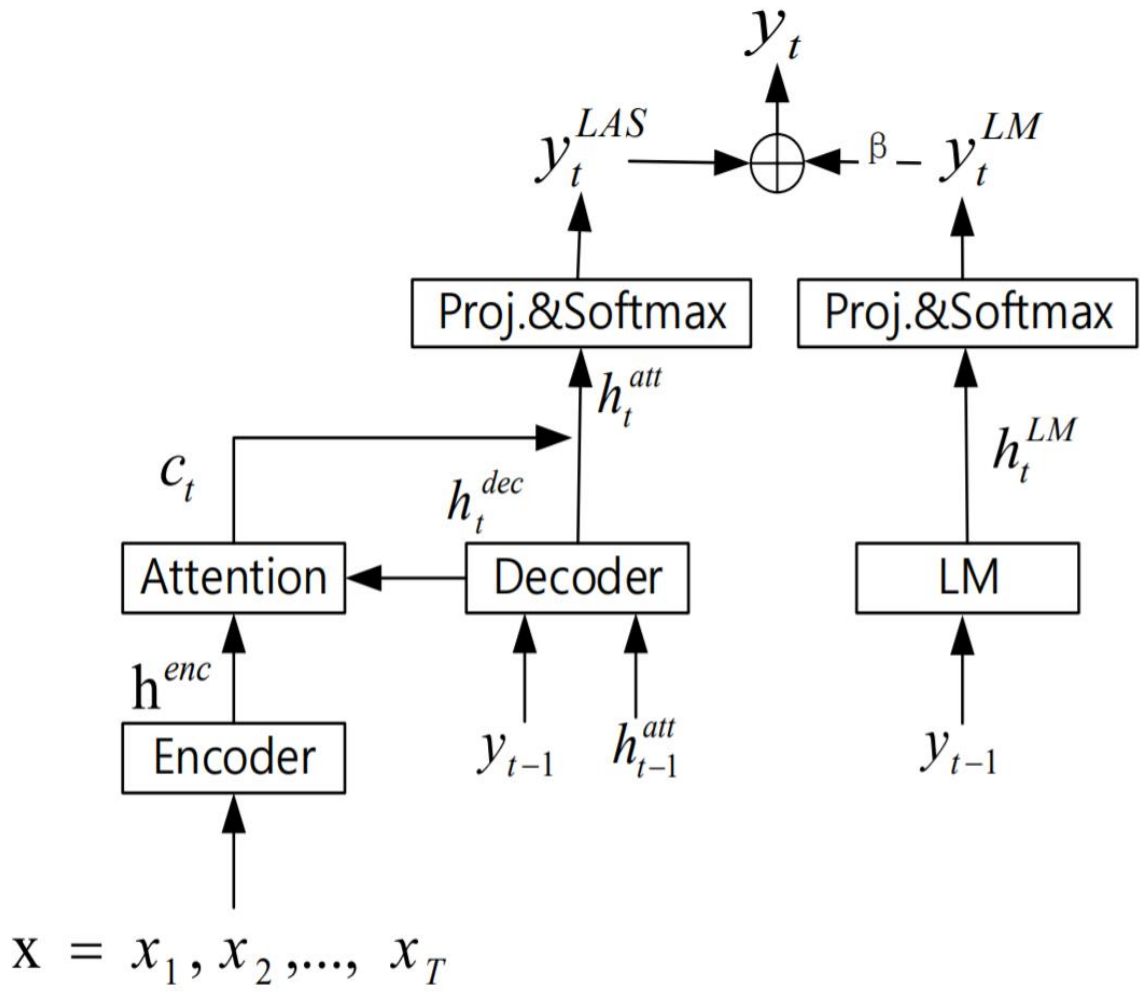
- 전통적인 Shallow/Deep Fusion에 이어 Cold Fusion 방식 소개



Shallow Fusion

- 가장 단순한 방식의 LM 사용법
- LM과 Seq2seq model은 별개로 학습됨
- Decoding 과정에서 LM의 probs와 Seq2seq model의 probs를 단순 선형 결합하여 최종 선택

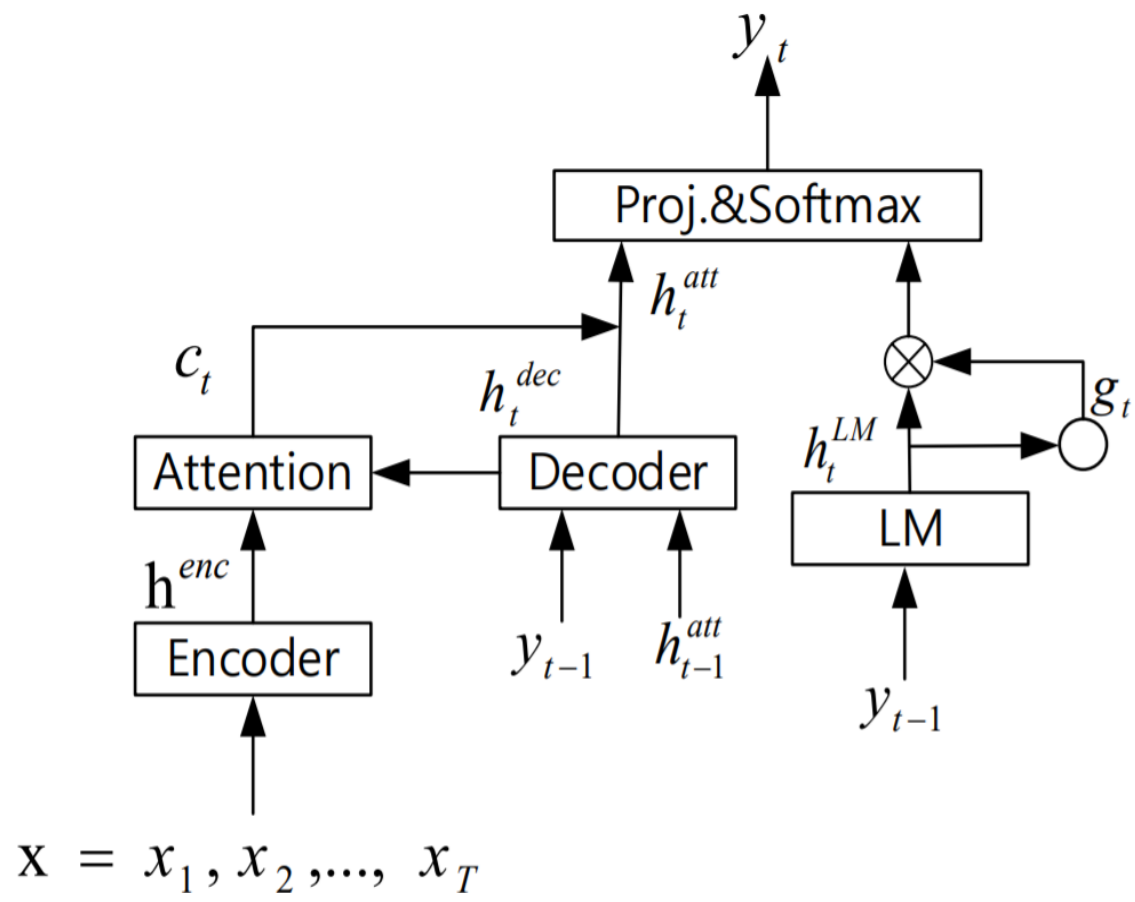
$$logP(y_t) = logP_{Att}(y_t) + \beta logP_{LM}(y_t),$$



Deep Fusion

- Shallow Fusion의 성능 향상 도모
- LM과 Seq2seq model은 별개로 학습됨
- 단순 선형 결합이 아닌, Gate를 사용하는 보다 복잡한 연산이 필요

$$
\begin{aligned}
g_t &= sigmoid(\boldsymbol{U_g} s_t^{LM} + b), \\
\widehat{h}_t^{att} &= [h_t^{att}; g_t s_t^{LM}], \\
y_t &= softmax(\boldsymbol{W_o'} \widehat{h}_t^{att}),
\end{aligned}
$$



Shallow/Deep Fusion 방식의 문제점

- 언어 모델을 사용하지만, Seq2seq model을 학습시킬 때는 LM 없이 학습시키므로
  Seq2seq model의 학습 과정에서 "내부적인" LM의 학습이 필요함

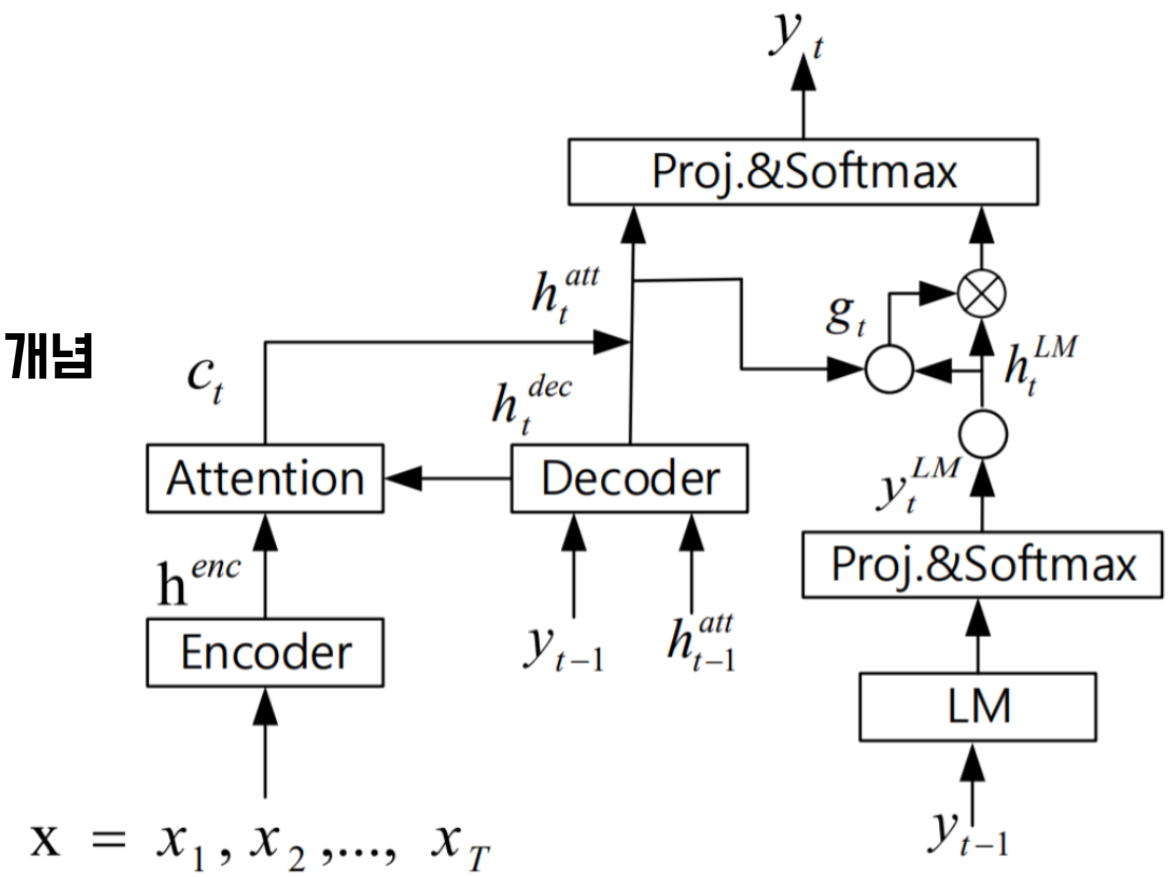
- 이 "내부적인" LM으로 인해 디코더의 capacity 일부가 사용되므로 task 자체를 학습하는 능력이 저하됨

- 이 "내부적인" LM은 학습 데이터의 도메인에 따라 편향되므로 다른 도메인의 데이터를 사용해 예측한다면
  overfitting으로 인한 성능 저하가 우려됨

Cold Fusion

- Shallow/Deep Fusion에서의 문제점을 개선
- Seq2seq model의 학습 과정에 LM을 함께 사용
- 입력이 specific하거나 noisy한 경우 LM을 참조
- 즉, LM의 사용법을 Seq2seq model이 학습한다는 개념

$$
\begin{aligned}
h_t^{LM} &= DNN(l_t^{LM}), \\
g_t &= sigmoid(\boldsymbol{U_g}[h_t^{LM}; h_t^{att}] + b), \\
\widehat{h}_t^{att} &= [h_t^{att}; g_t h_t^{LM}], \\
y_t &= softmax(\boldsymbol{W_o'}\widehat{h}_t^{att}).
\end{aligned}
$$



Cold Fusion

- **성능 향상**
  ("Cold Fusion: Training Seq2Seq Models Together with Language Models" [Anuroop Sriram, 2017])

Model	Prediction
Ground Truth	where's the sport in that greer snorts and leaps greer hits the dirt hard and rolls
Plain Seq2Seq	where is the sport **and** that **through snorks** and leaps **clear its** the dirt **card** and **rules**
Deep Fusion	where is the sport **and** that **there is north some beliefs through its** the dirt **card** and **rules**
Cold Fusion	where's the sport in that greer snorts and leaps greer hits the dirt hard and rolls
Cold Fusion (Fine-tuned)	where's the sport in that greer snorts and leaps greer hits the dirt hard and rolls
Ground Truth	jack sniffs the air and speaks in a low voice
Plain Seq2Seq	**jacksonice** the air and **speech** in a **logos**
Deep Fusion	**jacksonice** the air and **speech** in a **logos**
Cold Fusion	jack sniffs the air and speaks in a low voice
Cold Fusion (Fine-tuned)	jack sniffs the air and speaks in a low voice
Ground Truth	skipper leads her to the dance floor he hesitates looking deeply into her eyes
Plain Seq2Seq	**skip er leadure** to the dance floor he **is it takes** looking deeply into her eyes
Deep Fusion	**skip er leadure** to the dance floor he **has it takes** looking deeply into her eyes
Cold Fusion	skipper leads **you** to the dance floor he **has a tates** looking deeply into her eyes
Cold Fusion (Fine-tuned)	skipper leads her to the dance floor he hesitates looking deeply into her eyes