

---

# 담화 분석 (Discourse Analysis)

---

건국대학교 컴퓨터공학부 /  
KAIST 전산학부 (겸직)

김학수

# Core Layers of NLU

단계	설명	예제: 나는 그 과자를 먹었다.
형태소 분석	문장을 형태소별로 분리하고 품사를 부착하는 단계	나/대명사+는/조사 그/대명사 과자/명사+를/조사 먹/동사+었/선어말어미+다/어미+./.기호
구문 분석	문장의 문법적 적합성과 어절의 구문적 역할(주어, 목적어 등)을 찾는 단계	[SUBJ: 나는 [[MOD: 그 [OBJ: 과자를]] 먹었다]]
의미 분석	문장을 구성하는 술어와 논항들 사이의 의미적 적합성을 분석하는 단계	PREDICATE: 먹다 AGENT: 나/ANIMATE OBJECT: 그 과자/EATABLE
담화 분석	대화 문맥을 파악하여 상호참조를 해결하고 의도를 파악하는 단계	SPEECH ACT: STATEMENT PREDICATE: 먹다 AGENT: 홍길동/ANIMATE OBJECT: 꼬깔콘/EATABLE



# 담화 분석

- 담화 분석이란?
  - 문맥(대화 히스토리)를 고려하여 화자의 의도를 파악하는 과정

Are there any action movie  
to see this weekend?

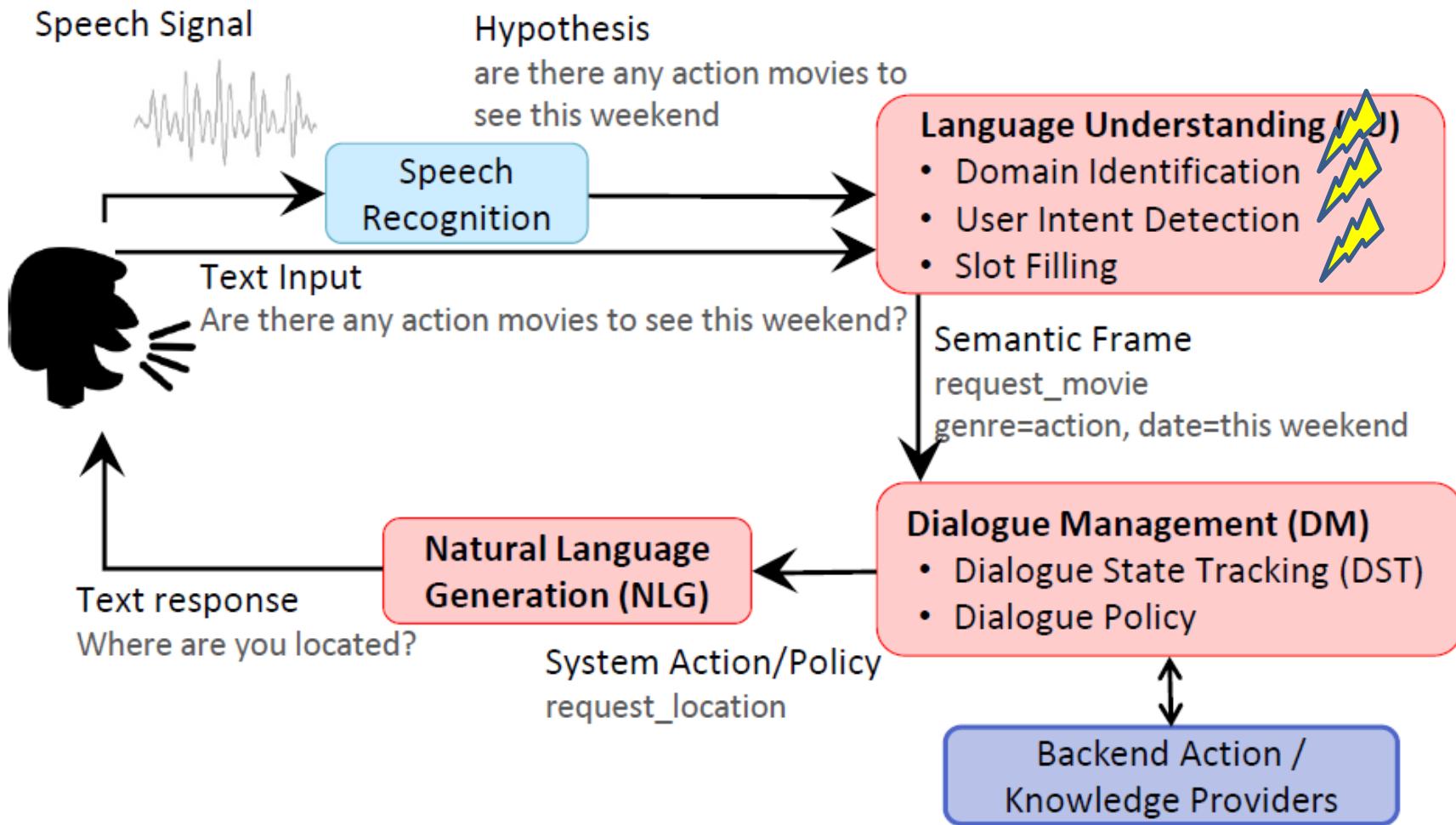


request\_movie  
genre=action, date=this weekend

- 담화 분석의 구성
  - 화자 의도 분석
    - Intent = Dialogue Act
    - Dialogue Act = Speech Act + Predicator(Main Action)
  - 슬롯 태깅(Slot Tagging)
    - 태스크(task)에 따라 정의된 영역 프레임(domain frame)을 구성하는 슬롯 정보를 추출하는 과정



# Task-Oriented Dialogue System



# Intent Detection

User



find a good eating place for taiwanese food



Intelligent  
Agent



FIND\_RESTAURANT

FIND\_PRICE

FIND\_TYPE

:

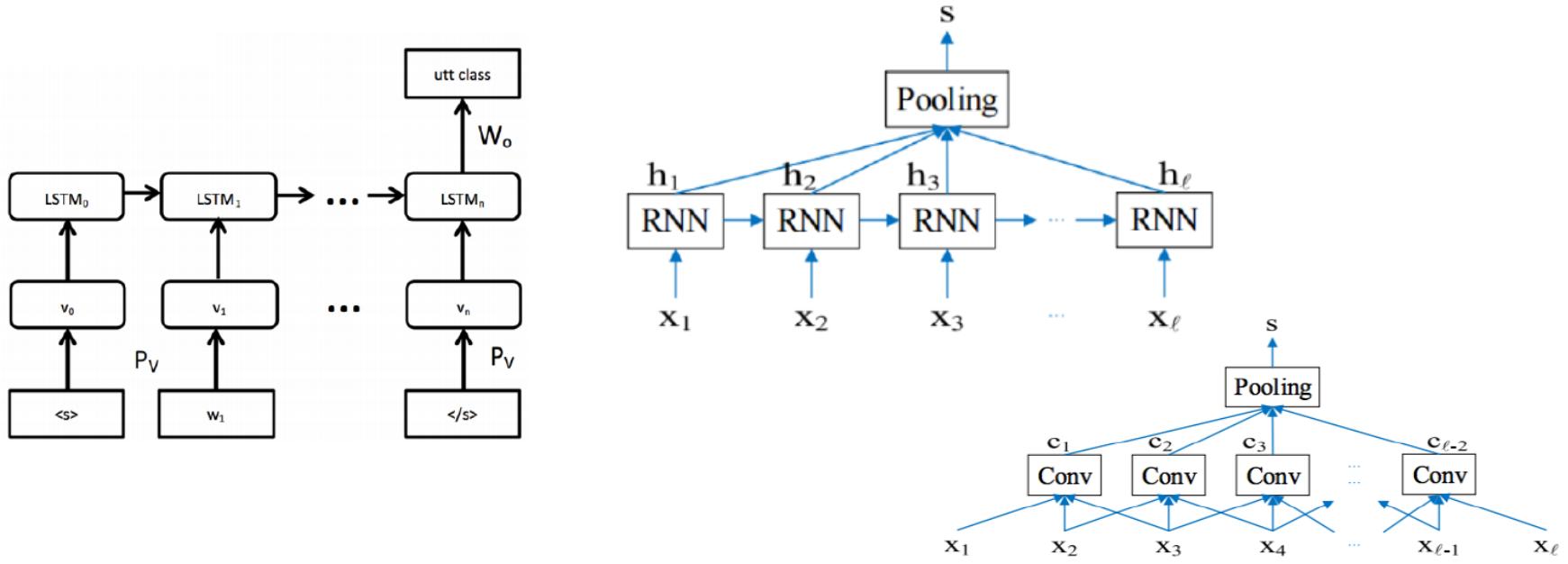
**Classification!**

Requires Predefined Schema



# DNN for Domain/Intent Classification

- Deep belief nets (Sarikaya et al., 2011)
- Deep convex networks (Tur et al., 2012, Deng et al., 2012)
- RNN and LSTMs for utterance classification (Ravuri & Stolcke, 2015; Lee & Dernoncourt, 2016)



# 화행, 서술자, 감정 통합 분석 모델

- **화행(speech act)**

- 화자의 목적이 무엇인가?
  - 발화(utterance)를 요청, 제안, 질문, 응답 등으로 분류

“내일 날씨 말해줘”  
“내일 날씨 말해줄래?”  
“내일 날씨 좀 알 수 있을까?”

Request(요청)



# 화행, 서술자, 감정 통합 분석 모델

- 발화는 화자의 감정(emotion), 서술자(predicator), 화행을 내포
- 감정, 서술자, 화행은 발화 이해를 위한 중요한 정보
- 발화 이해를 통해 더 적절한 응답 가능

“좀 떨리더라”

화행 ----- Statement

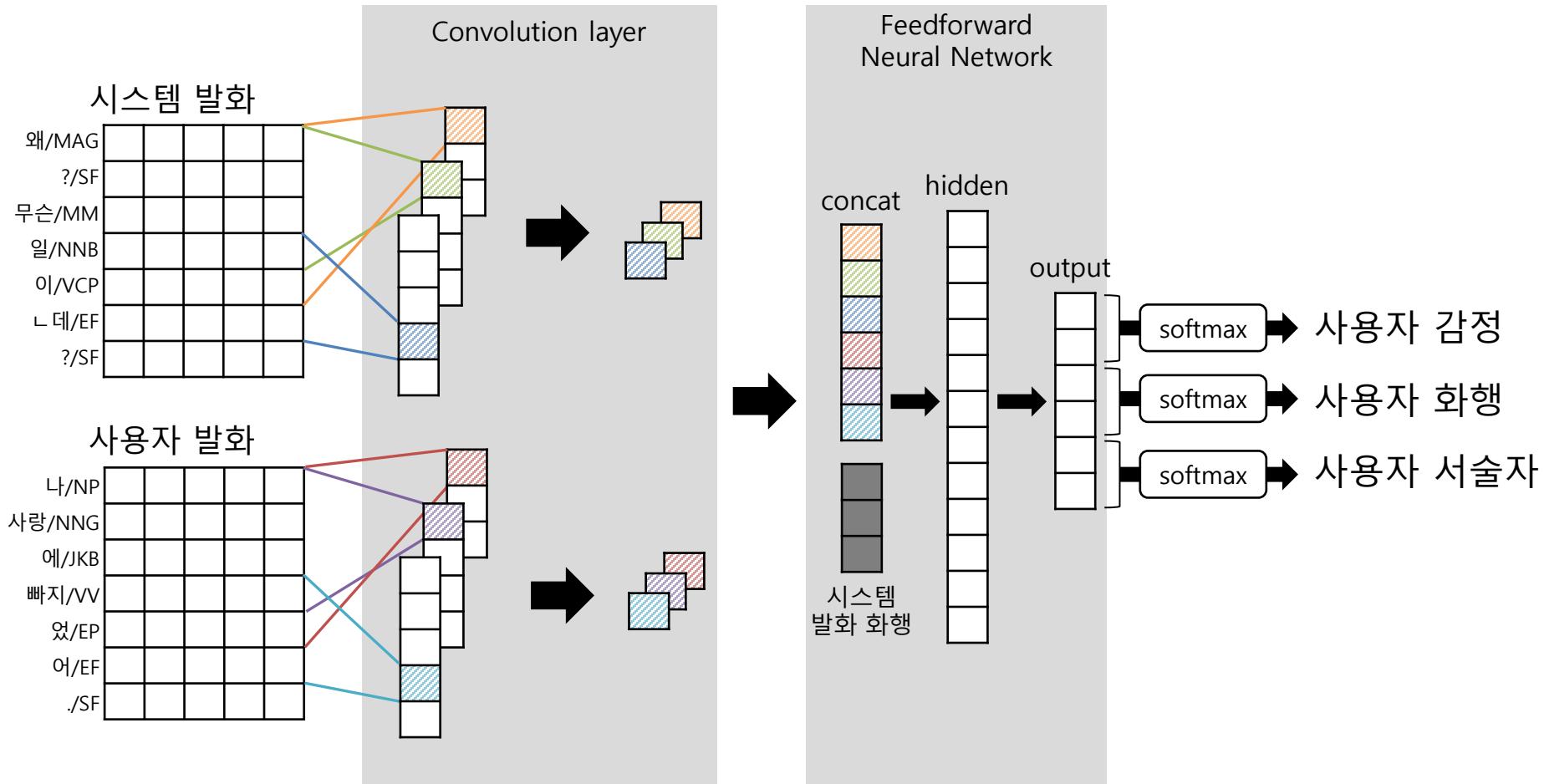
서술자 ----- Expression

감정 ----- Love      Fear

(데이트)  
“우와 부럽다”      (회사 면접)  
“괜찮아 잘했어”



# 화행-서술자-감정 통합 분석 모델 1



# 실험 및 평가

---

- 실험 데이터
  - 대화 영역: 사랑
  - 총 대화 수: 441개
  - 총 발화 수 7,490개
  - 대화 당 평균 발화 수: 17개
- 범주 구성
  - 화행 범주: 13개
  - 서술자 범주: 68개
  - 감정 범주: 10개

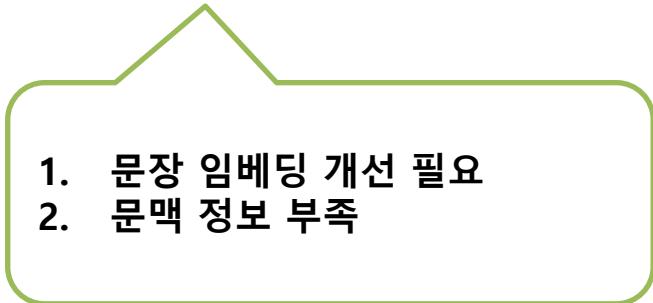


# 실험 및 평가

- 성능 평가

- $$- \text{정밀도} = \frac{\text{시스템이 맞게 예측한 수}}{\text{시스템 결과}}$$

모델	감정	화행	서술자
개별 모델	0.490	0.836	0.544
통합 모델	0.567	0.838	0.335

- 
1. 문장 임베딩 개선 필요
  2. 문맥 정보 부족



# 문장 임베딩 개선 방안

$$IM(D) \stackrel{\text{def}}{=} \arg \max_{S_{1,n}, P_{1,n}, E_{1,n}} P(S_{1,n}, P_{1,n}, E_{1,n} | U_{1,n})$$

Chain rule

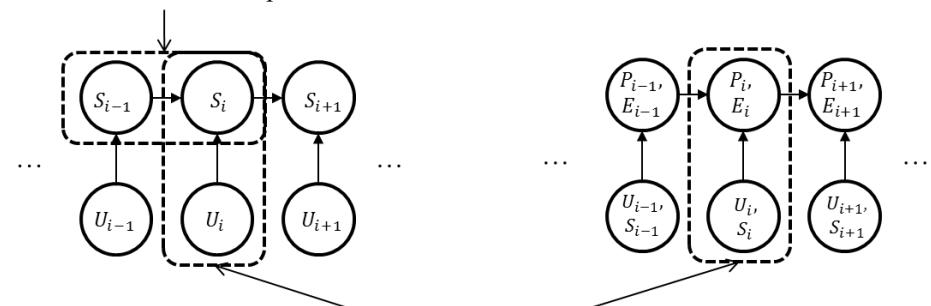
$$IM(D) \stackrel{\text{def}}{=} \arg \max_{S_{1,n}, P_{1,n}, E_{1,n}} P(S_{1,n} | U_{1,n}) P(P_{1,n}, E_{1,n} | U_{1,n}, S_{1,n})$$



Markov & Independent

$$IM(D) \stackrel{\text{def}}{=} \arg \max_{S_{1,n}, P_{1,n}, E_{1,n}} \max \prod_{i=1}^n \left\{ \frac{P(S_i | U_i) P(S_i | S_{i-1})}{P(P_i, E_i | U_i, S_i)} \right\}$$

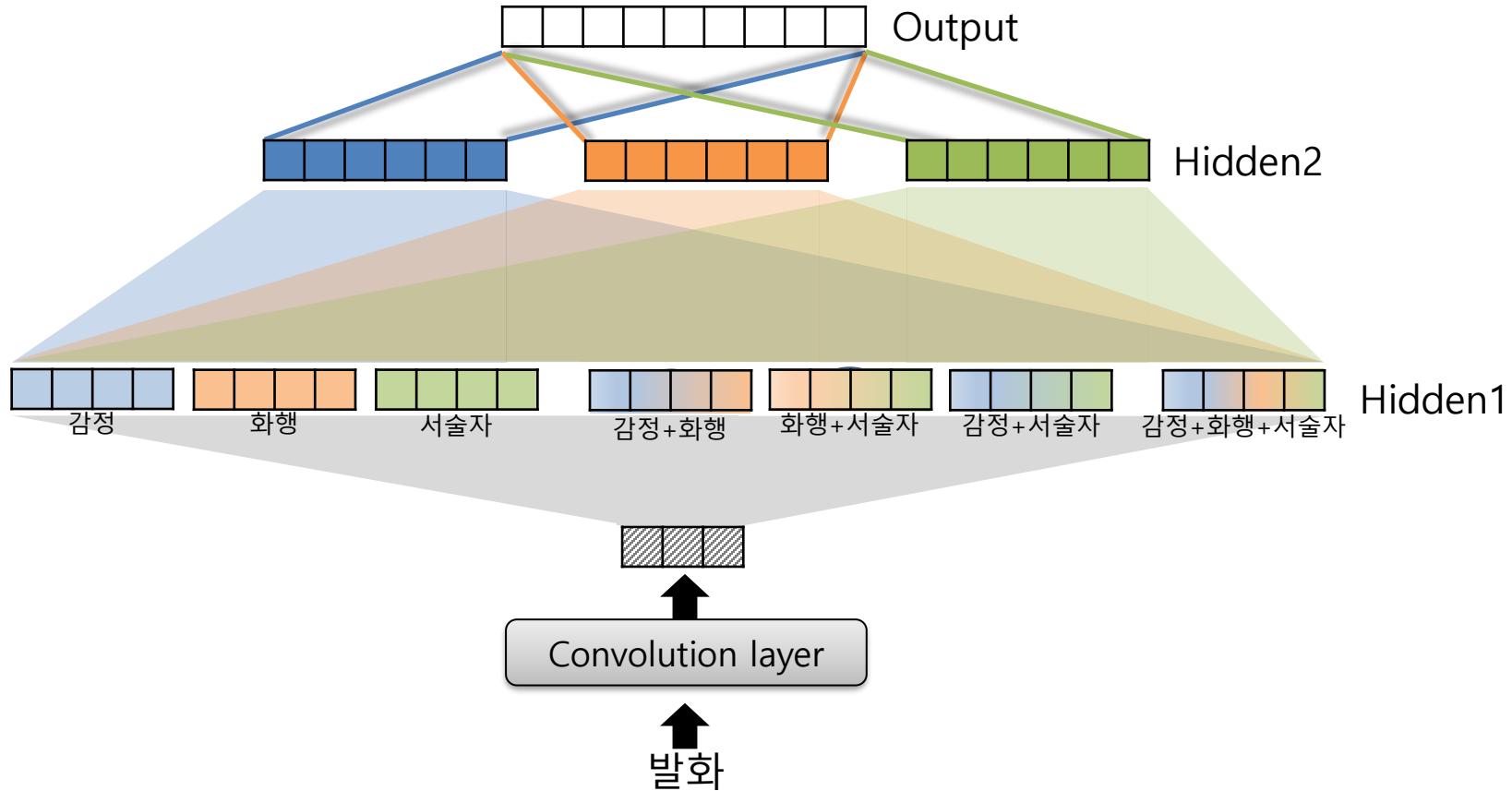
1<sup>st</sup> order Markov assumption



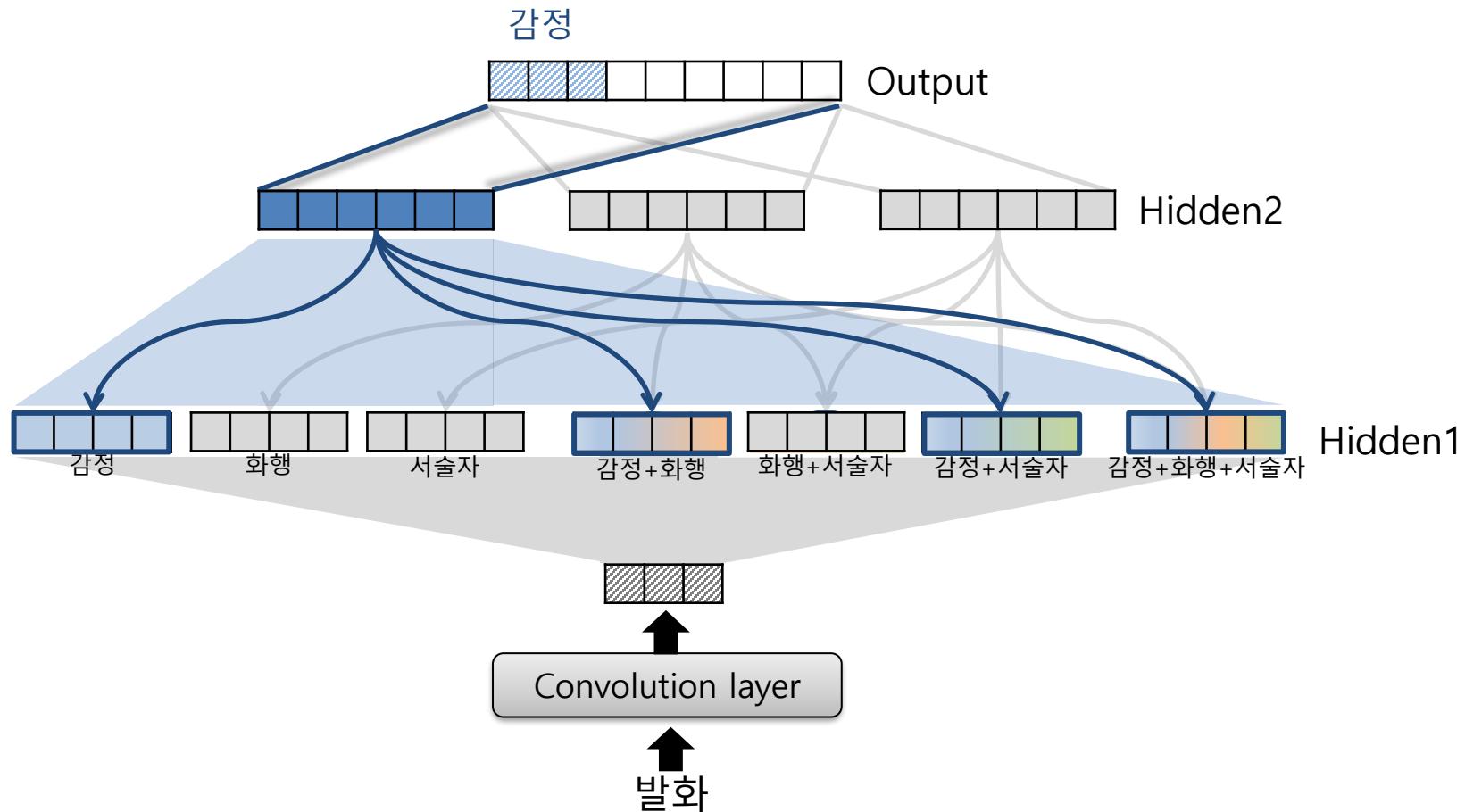
Conditional independent assumption



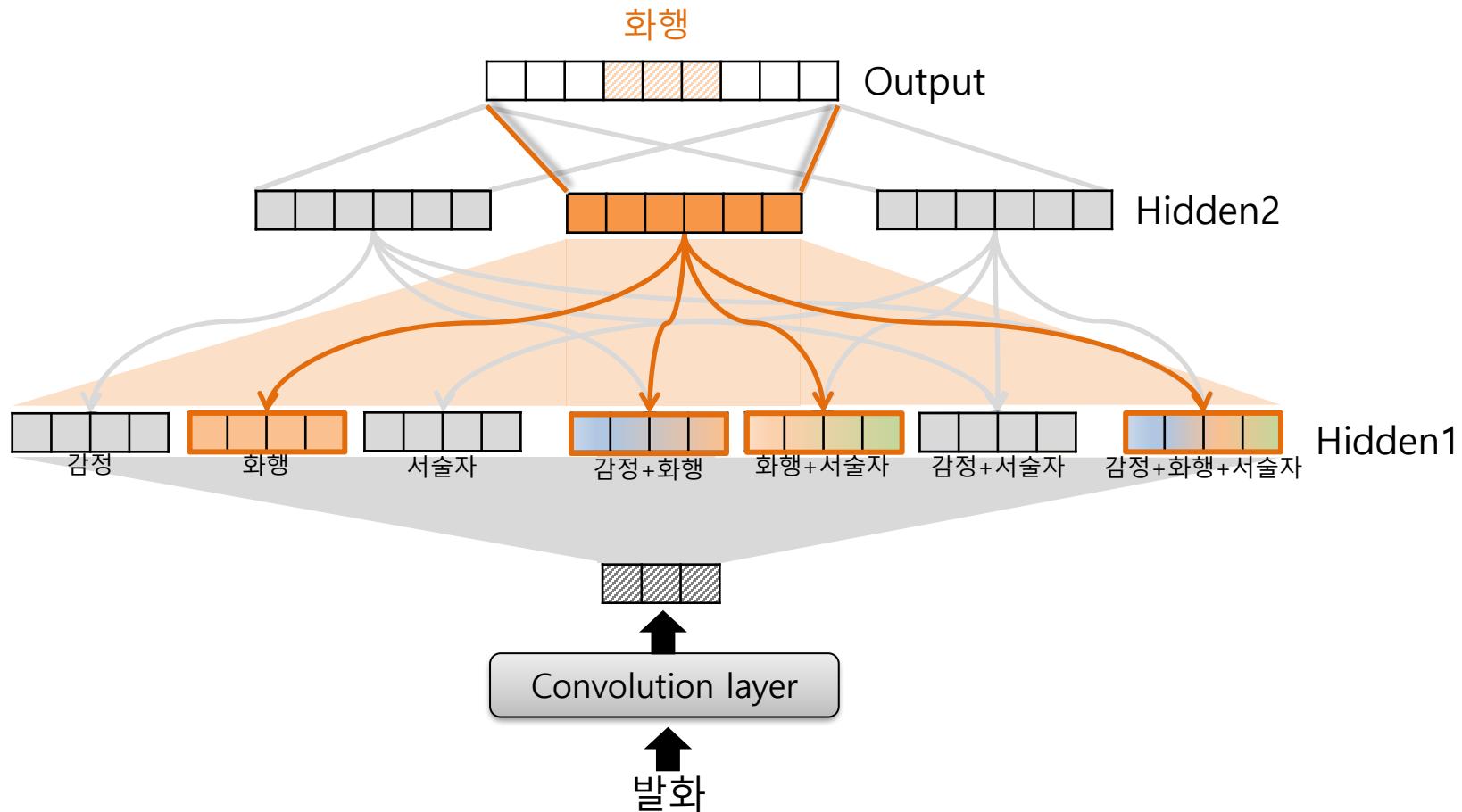
# 문장 임베딩 개선 방안



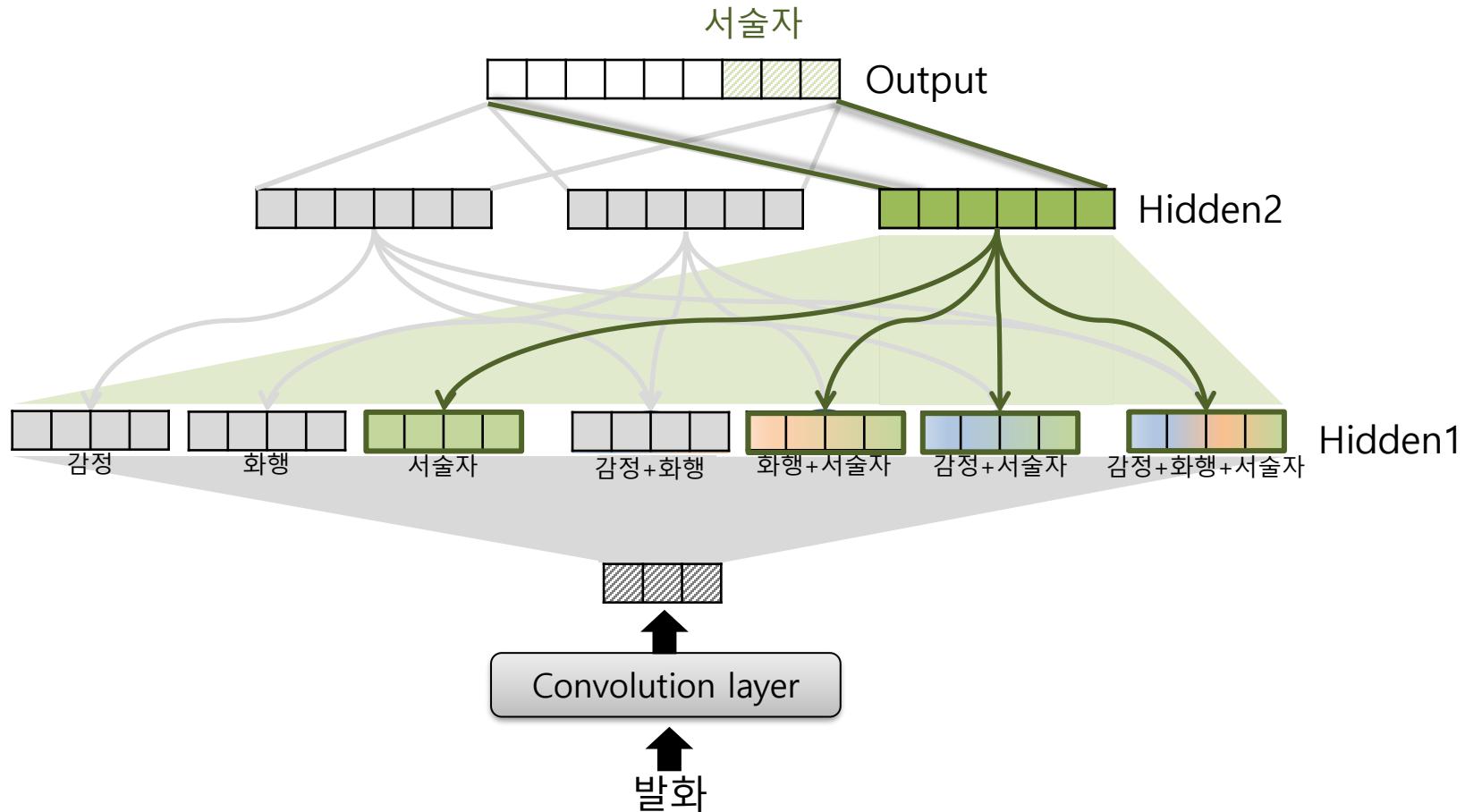
# 문장 임베딩 개선 방안



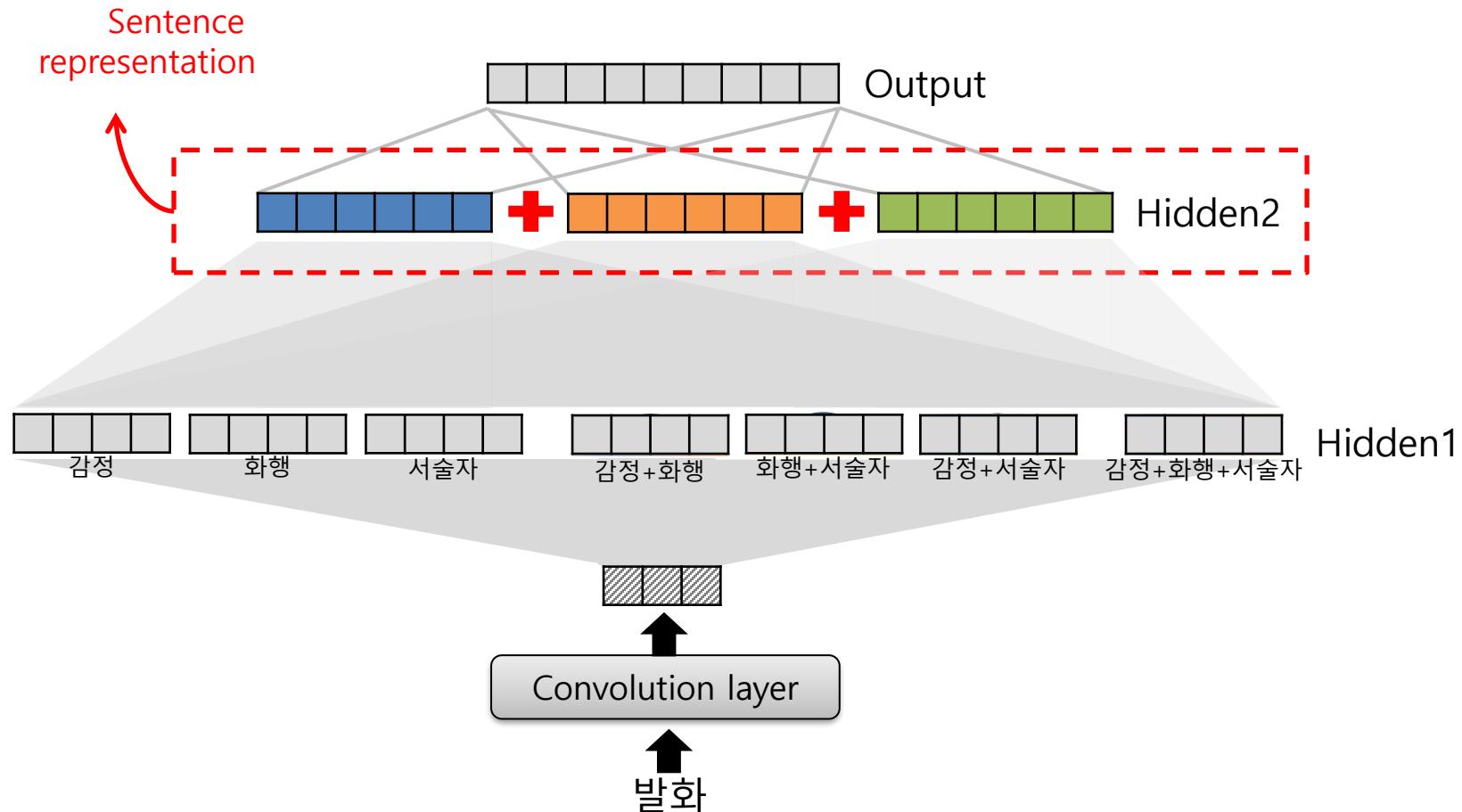
# 문장 임베딩 개선 방안



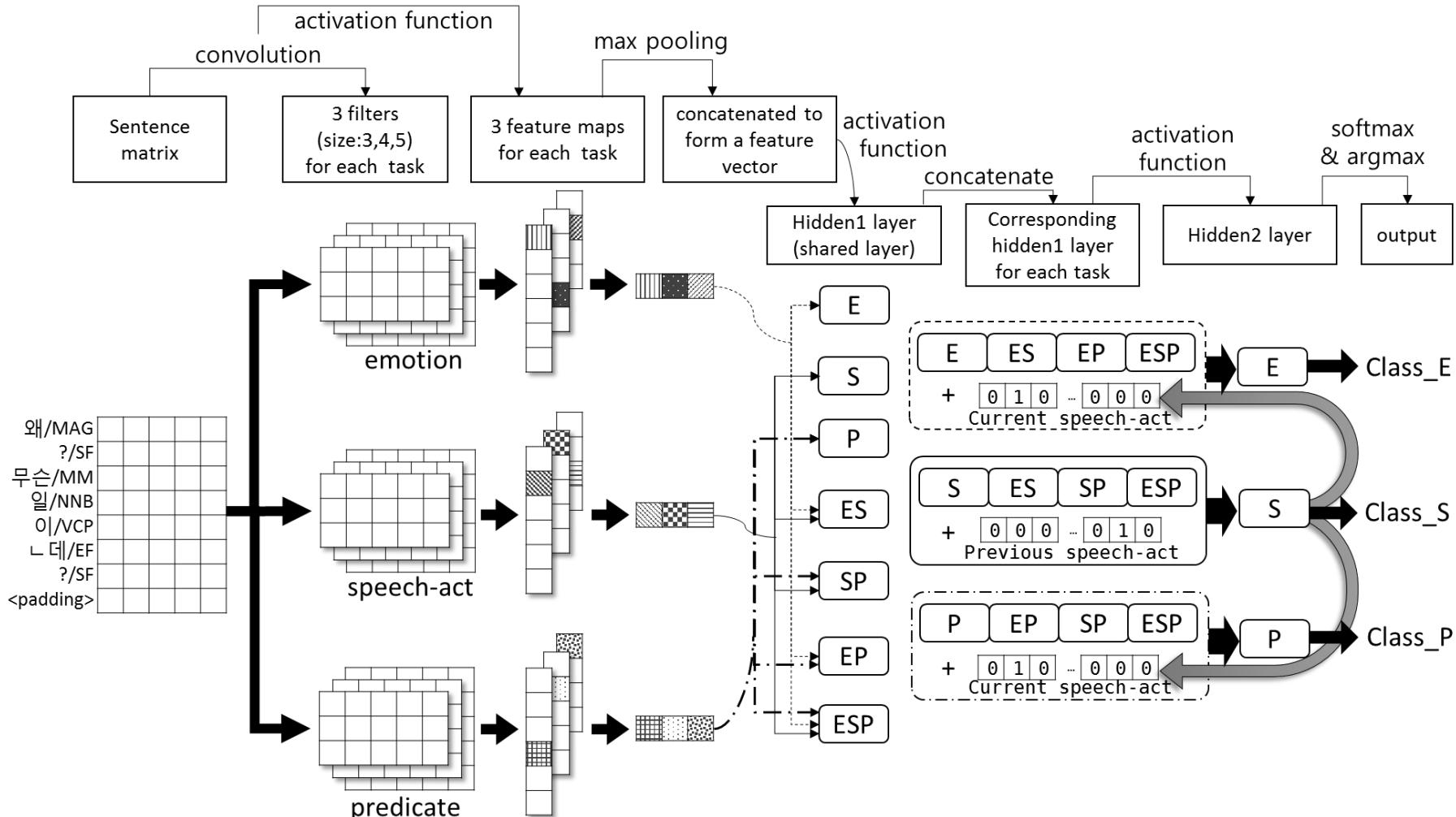
# 문장 임베딩 개선 방안



# 문장 임베딩 개선 방안

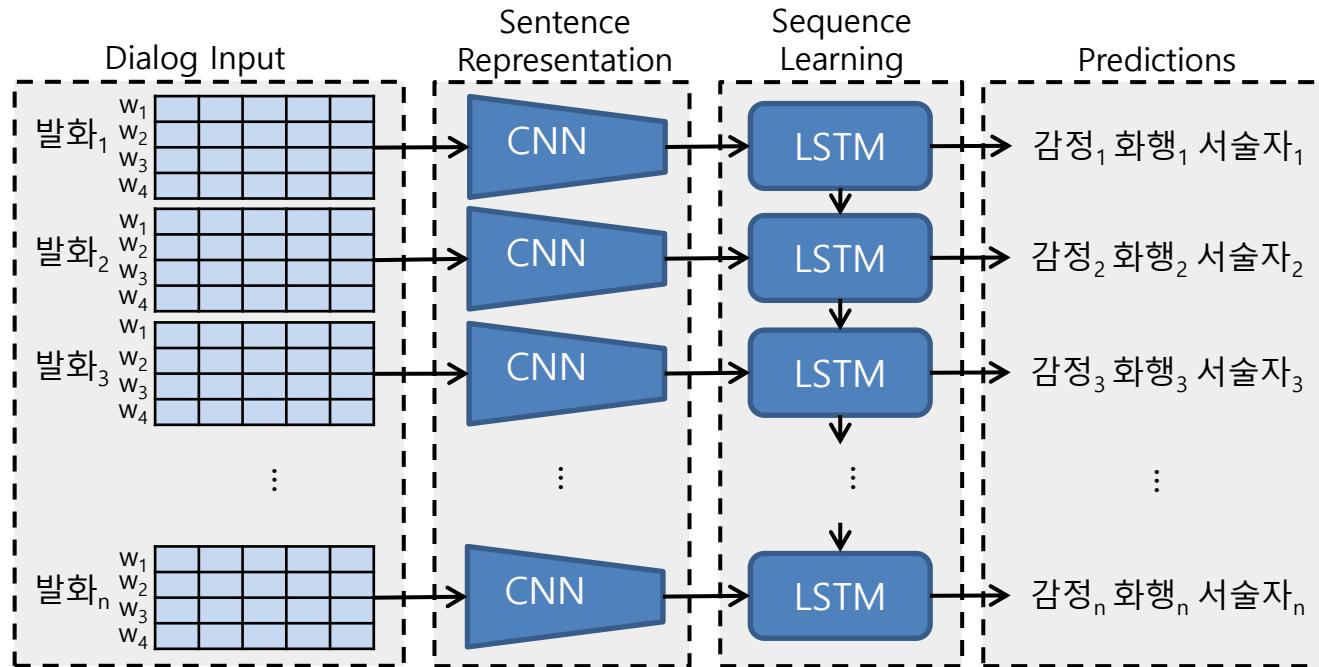


# 화행-서술자-감정 통합 분석 모델 2



# 문맥정보 개선 방안

- CNN을 이용하여 문장 표현을 학습
- LSTM의 각 time-step의 입력 → 대화의 순서 관계 학습



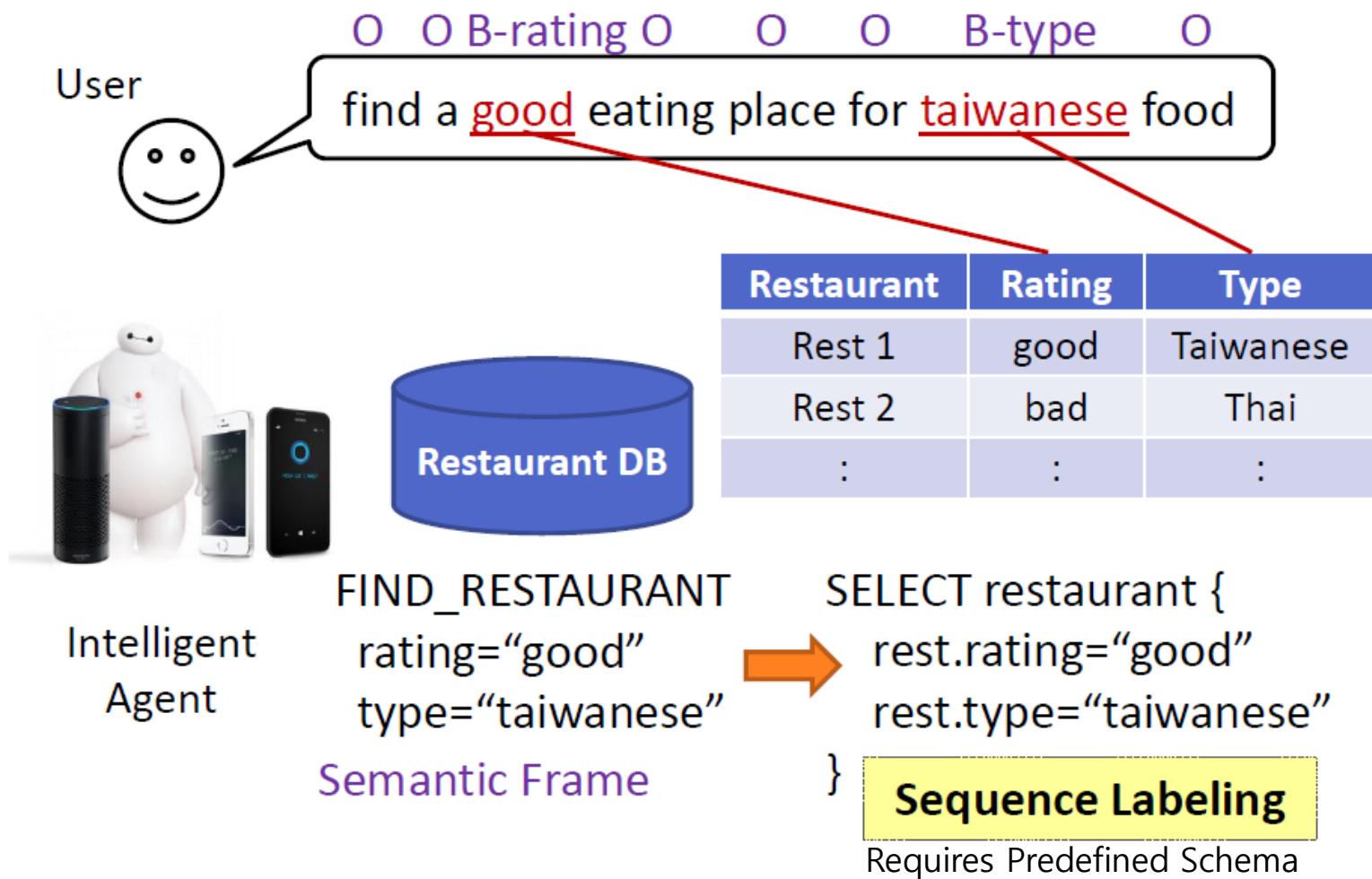
# 실험 및 평가

- 성능 평가 척도
  - 10배 교차검증을 이용한 성능평가

	Accuracy	Macro precision	Macro recall	F1-score
독립CNN-E	0.638	0.607	0.527	0.516
공유CNN-E	0.661	0.594	0.559	0.565
공유CNN-LSTM-E	<b>0.952</b>	<b>0.969</b>	<b>0.952</b>	<b>0.959</b>
독립CNN-S	0.744	0.799	0.657	0.646
공유CNN-S	0.855	0.862	0.720	0.714
공유CNN-LSTM-S	<b>0.974</b>	<b>0.964</b>	<b>0.919</b>	<b>0.904</b>
독립CNN-P	0.698	0.837	0.618	0.579
공유CNN-P	0.731	0.835	0.691	0.641
공유CNN-LSTM-P	<b>0.924</b>	<b>0.916</b>	<b>0.832</b>	<b>0.796</b>



# Slot Tagging



# Slot Tagging

As a sequence  
tagging task

- Given a collection tagged word sequences,  $S = \{((w_{1,1}, w_{1,2}, \dots, w_{1,n_1}), (t_{1,1}, t_{1,2}, \dots, t_{1,n_1})), ((w_{2,1}, w_{2,2}, \dots, w_{2,n_2}), (t_{2,1}, t_{2,2}, \dots, t_{2,n_2})) \dots\}$  where  $t_i \in M$ , the goal is to estimate tags for a new word sequence.

flights from Boston to New York today

	flights	from	Boston	to	New	York	today
Entity Tag	O	O	B-city	O	B-city	I-city	O
Slot Tag	O	O	B-dept	O	B-arrival	I-arrival	B-date

Domain-specific NER

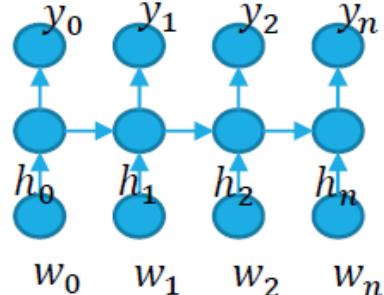


# RNN for Slot Tagging

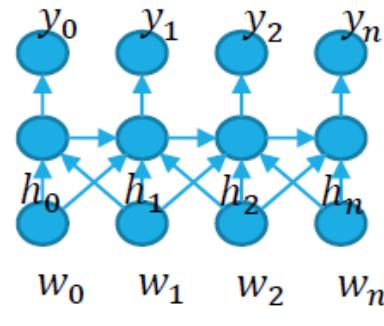
(Yao et al., 2013; Mesnil et al., 2015)

## □ Variations:

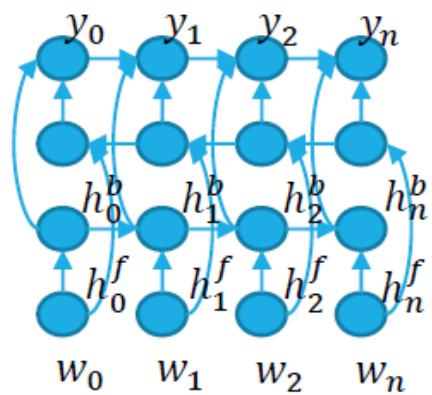
- a. RNNs with LSTM cells
- b. Input, sliding window of n-grams
- c. Bi-directional LSTMs



(a) LSTM



(b) LSTM-LA



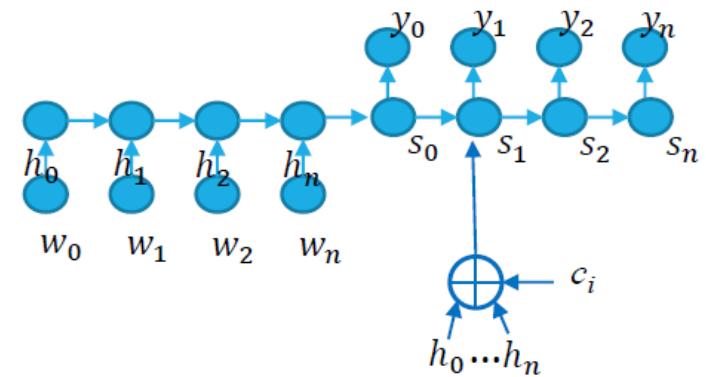
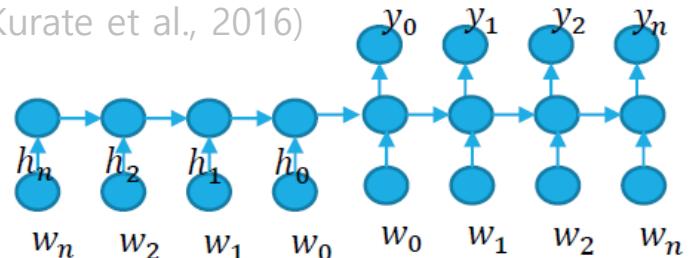
(c) bLSTM



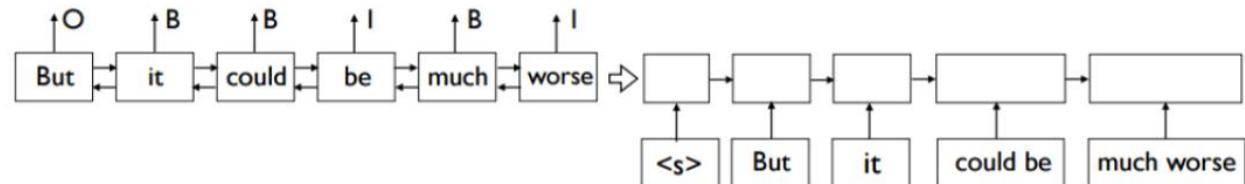
# RNN for Slot Tagging

- Encoder-decoder networks
  - Leverages sentence level information
- Attention-based encoder-decoder
  - Use of attention (as in MT) in the encoder-decoder network
  - Attention is estimated using a feed-forward network with input:  $h_t$  and  $s_t$  at time  $t$
- Encoder that segments
- Decoder that tags the segments

(Kurate et al., 2016)



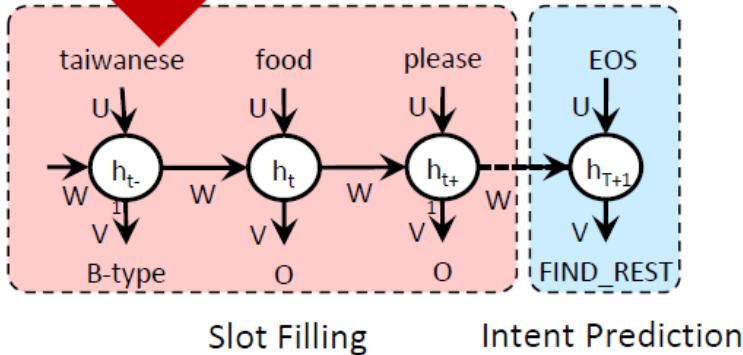
(Zhai et al., 2016)



# Joint Semantic Frame Parsing

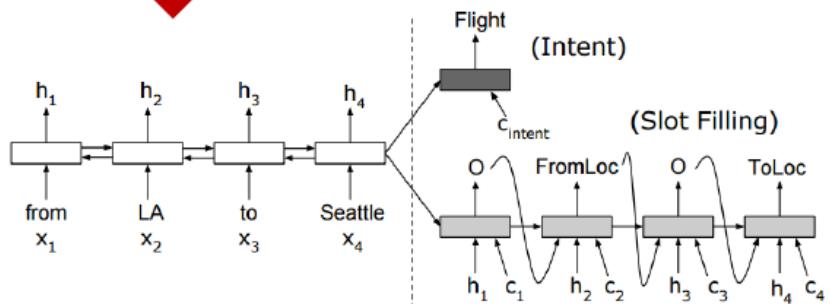
Sequence-based  
(Hakkani-Tur et al., 2016)

- Slot filling and intent prediction in the same output sequence



Parallel  
(Liu and Lane, 2016)

- Intent prediction and slot filling are performed in two branches



---

# Co-reference Resolution

---

건국대학교 컴퓨터공학부 /  
KAIST 전산학부 (겸직)

김학수

# 상호참조 해결

| 상호참조해결: 주어진 문서에서 같은 개체를 가리키는 단어(멘션)들의 관계 정의

- 멘션(mention): 상호참조해결의 후보, 고유한 개체를 표현하는 구(중심어와 수식어로 구성)
- 엔티티(entity): 개체를 나타내는 멘션들의 클러스터(cluster)
- 상호참조해결 과정: 멘션 탐지 → 상호참조해결

| Step 1: 멘션 탐지 (의존구문분석 결과로 쉽게 찾을 수 있음)

[세종]은 [[조선의 4대 군주]였으며 언어학자]였다. [[[그]의 업적에 대한 존경의 의미를 담은 명칭]인 세종대왕]으로 자주 일컬어진다. [세종]은 [1443년 누구나 쉽게 배울 수 있는 훈민정음]을 창제하였다. [훈민정음]은 [한국어의 표기 문자 체계]를 말하며 [쓰기 편한 28자의 소리글자]이다.

| Step 2: 상호참조해결 (문제 해결 어려움)

[세종]은 [[조선의 4대 군주]였으며 언어학자]였다. [[[그]의 업적에 대한 존경의 의미를 담은 명칭]인 세종대왕]으로 자주 일컬어진다. [세종]은 [1443년 누구나 쉽게 배울 수 있는 훈민정음]을 창제하였다. [훈민정음]은 [한국어의 표기 문자 체계]를 말하며 [쓰기 편한 28자의 소리글자]이다.



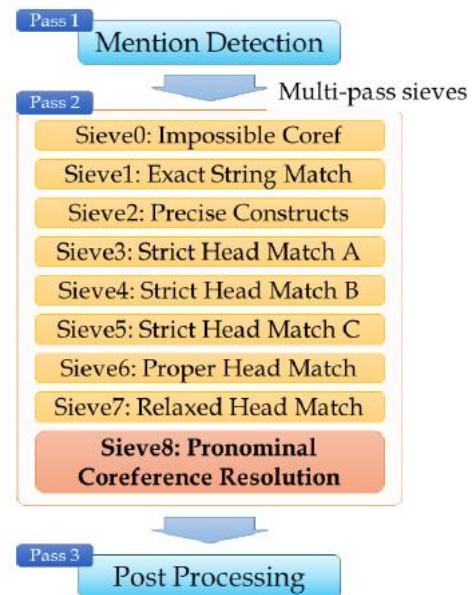
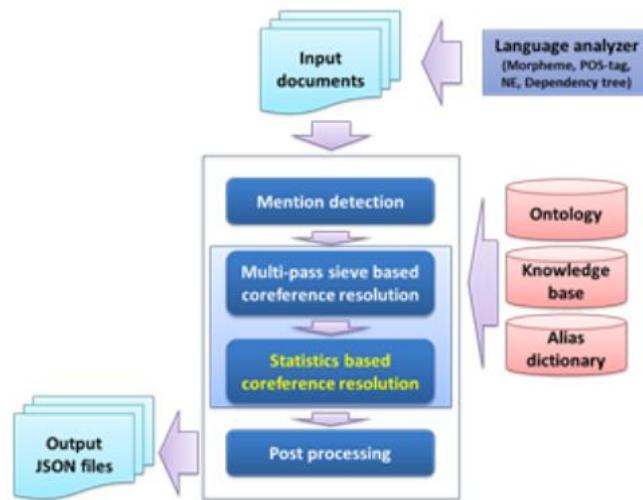
# 규칙 기반 상호참조 해결

## Rule-based Coref (multi-pass sieve)

- Extracting Mention
- Processing 9 conditions (9 Sieves)

## 단점

- 멘션 탐지를 수행해야 함
- 사람이 직접 규칙을 정의  
→ 모든 자연어에 대한 규칙 정의 어려움



# Sieve Rule (1/7)

---

- 전체 문자열 비교
  - 멘션이 고유명사를 포함할 경우에 적용
  - 전체 문자열이 일치하거나 동의어인 경우 상호참조

류현진

류현진의 커브

박용근의 플레이

대전구장 좌측

류현진

류현진의 커브

박용근의 플레이

대전구장 좌측



# Sieve Rule (2/7)

---

- 약어 비교
  - 한 멘션의 약어와 다른 멘션의 전체 문자열이 일치하는지 비교

김응룡 감독

- 감독 김응룡
- 감독님
- **김 감독**
- 김 감독님
- 응룡
- 응룡이

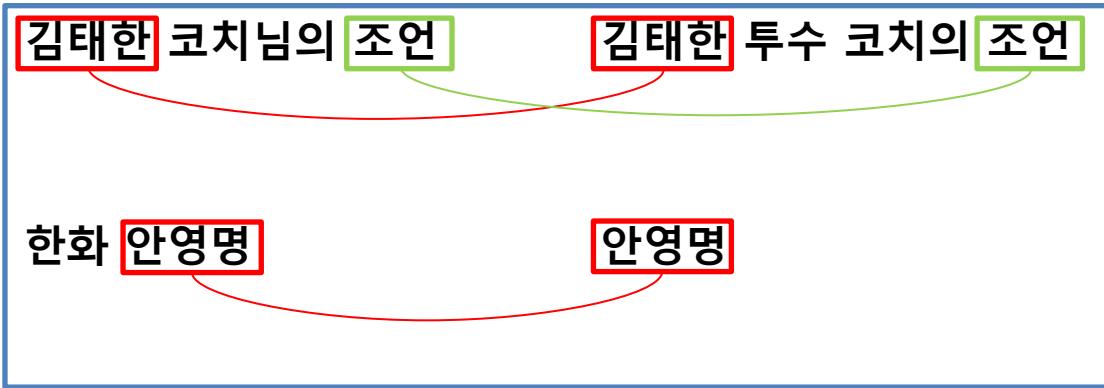
김 감독

...



# Sieve Rule (3/7)

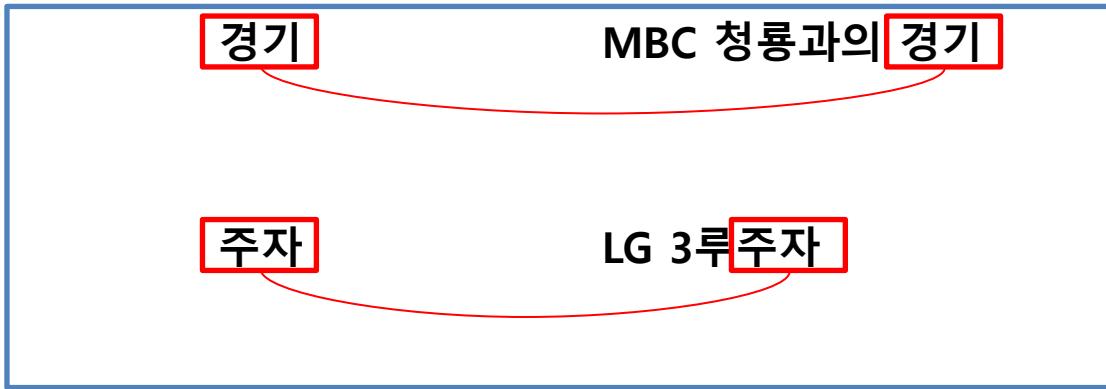
- 중심어 일치
  - 두 멘션 모두 고유명사를 포함할 경우에 적용
  - 고유명사는 고유명사끼리 일반명사는 일반명사끼리 비교



# Sieve Rule (4/7)

---

- 마지막 어절 일치
  - 한쪽이라도 고유명사를 포함할 경우에 적용
  - 마지막 어절 어휘가 일치하는지 검사



# Sieve Rule (5/7)

- 대명사 처리
  - 대명사의 타입과 다른 멘션의 개체명 타입을 비교

장민석(PER)

다저스(ORG)

고척구장(LOC)

그(PER)

우리(ORG)

여기(LOC)

장민석은 넥센 시절이던 2008년 투수에서 타자로 전향했다. 뒤늦게 포지션을 바꾼 만큼 그는...

엘리스는 “류현진은 이제 클럽하우스 안에서 사랑받는 존재다. 이제 우리 모두 류현진이 선발 등판하는 날은 이길 수 있는 좋은 기회의 날이라는 것을 잘 알고 있다”고 말했다.



# Sieve Rule (6/7)

---

- 한정 명사구 어휘 비교
  - 멘션이 관형사를 포함한 경우 다른 멘션과 중심어 비교

이 사건으로 인해 한동안 경기가 펼쳐지지 못했다.

광주 경기는 3연전의 첫머리이던 전날부터 심판의 오심 사건이 터지며 뒤숭숭한 분위기다.



# Sieve Rule (7/7)

---

- 타입 비교
  - 한 멘션의 개체명 타입과 다른 멘션의 중심어 타입을 비교

승리 투수(PIT)  
심판(UMP)

류현진(PIT)  
박근영(UMP)



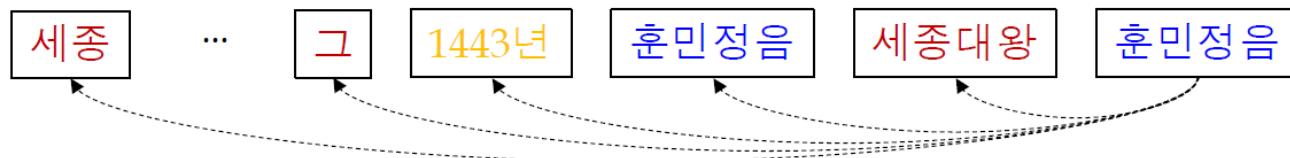
# 기계학습 기반 상호참조 해결: 멘션 페어

## | 멘션 페어(mention pair)

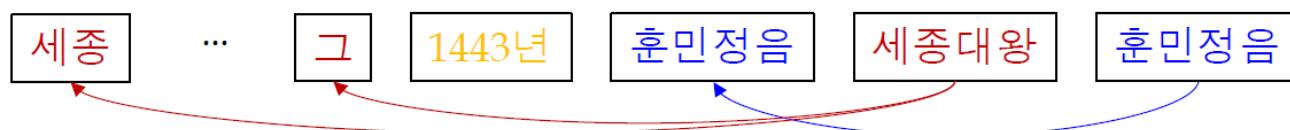
- Coref or not: 두 멘션을 비교하여 상호참조인지 아닌지 판별 (이진 분류)
- Coref score  $\rightarrow s = p(m_i, m_j)$

세종은 조선의 4대 군주였으며 언어학자였다. 그는 1443년 누구나 쉽게 배울 수 있는 훈민정음을 창제하였다. 세종대왕이 만든 훈민정음은 한국어의 표기 문자 체계를 말하며 쓰기 편한 28자의 소리글자이다.

Train time



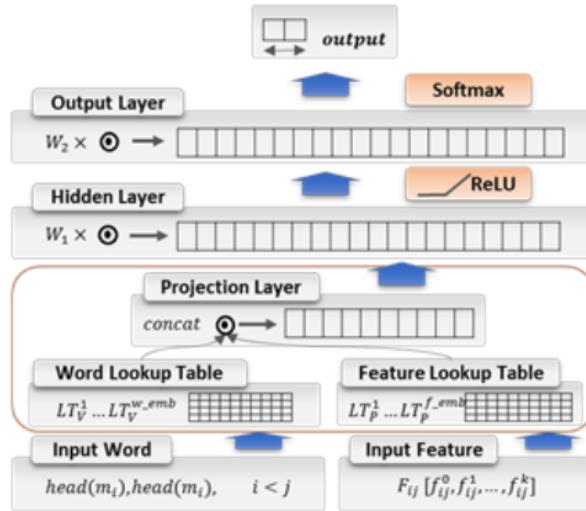
Test time



# 기계학습 기반 상호참조 해결: 멘션 페어

## | Binary Classification → SVM or FFNN

- Mention pair: Coref or not (binary classification)



Feature name	Explanation	Feature name	Explanation
pronoun_1	Y if antecedent is a pronoun; else N	number	C if the mentions agree in number; I if they disagree
subject_1	Y if antecedent is a subject; else N	animacy	C if the mentions match in animacy; else I
left_pos_1	The POS tag of just before antecedent mention.	both_pronouns	C if both mentions are pronouns; I if neither are pronouns; else NA
right_pos_1	The POS tag of just after antecedent mention.	both_proper_nouns	C if both mentions are proper nouns; I if neither are proper nouns; else NA
nested_1	Y if antecedent is a nested NP; else N	span	Word distance between the mentions
number_2	SINGULAR or PLURAL, determined using a lexicon	sem_class	If the mentions have the same semantic class; I if they don't
pronoun_2	Y if anaphoric is a pronoun; else N	distance	Sentence distance between the mentions
nested_2	Y if anaphoric is a nested NP; else N	same_lemma	C if the mentions' dependency heads have same lemma; else I
sem_class_2	The semantic class of anaphoric; can be one of NA, human, thing, time, place by pronoun_dic and an NT recognizer	number_12	SINGULAR or PLURAL of two mentions ( e.g., number_12=singular/plural )
animacy_2	Y if anaphoric is determined as HUMAN by pronoun_dic and an NE recognizer	pronoun_12	The concatenation of the PRONOUN 2 feature values of two mentions
left_pos_2	The POS tag of just before anaphoric mention.	nested_12	The concatenation of the NESTED 1 feature values of two mentions
right_pos_2	The POS tag of just after anaphoric mention.	sem_class_12	The concatenation of the SEMCLASS 2 feature values of two mentions
head_match	C if the mentions have the same head noun; else I	animacy_12	The concatenation of the ANIMACY 2 feature values of two mentions
str_match	C if the mentions are the same string; else I	determiner_1	It distinguishes a determiner phrase from an antecedent
subter_match	C if one mention is a substring of the other; else I	determiner_2	It distinguishes a determiner phrase from a current mention
pro_str_match	C if both mentions are pronominal and are the same string; else I	deter_str_match	It distinguishes the same string of a determiner phrase from the both antecedents and a current mention
nonpro_str_match	C if both mentions are proper names and are the same string; else I	both_determiners	It distinguishes the determiner phrase from the both antecedents and a current mention
pn_str_match	C if the two mentions are both non-pronominal and are the same string; else I	i-within-i	It distinguishes a possessive relationship from the both antecedents and a current mention

Guided Coreference Resolution (ETRIJ 16): 38 features

## | 멘션 페어의 단점

- 멘션 페어는 한번 예측할 때 하나의 쌍에 대해서만 확인 가능

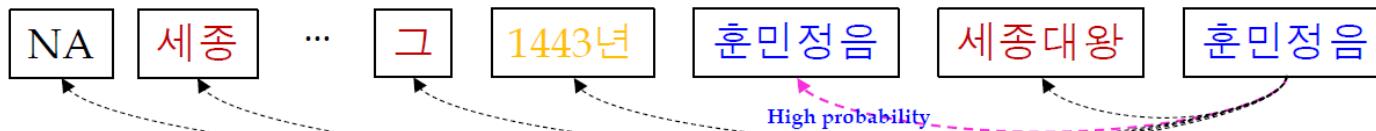


# 기계학습 기반 상호참조 해결: 멘션 랭킹

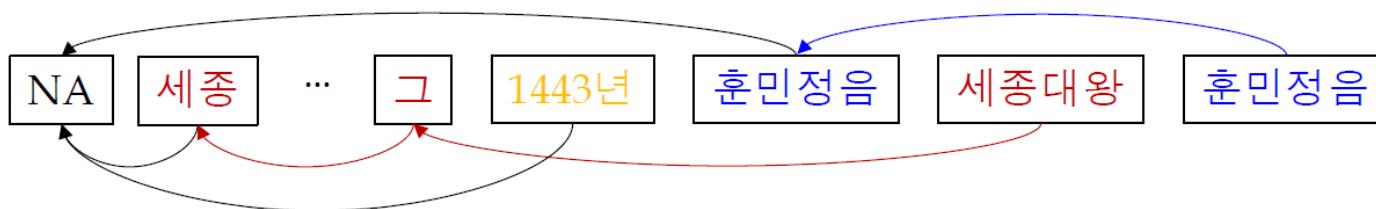
## | 멘션 랭킹 (mention ranking)

- Highest score: 각 멘션은 가장 높은 스코어를 가지는 후보 멘션과 연결 됨
- Coref:  $\operatorname{argmax} (p(m_i, m_j))$

Train time



Test time



## | 기계 학습 모델의 단점

- 모든 문맥 정보를 고려하는데 한계
- 사람이 직접 자질 추출 및 조합을 수행해야 함



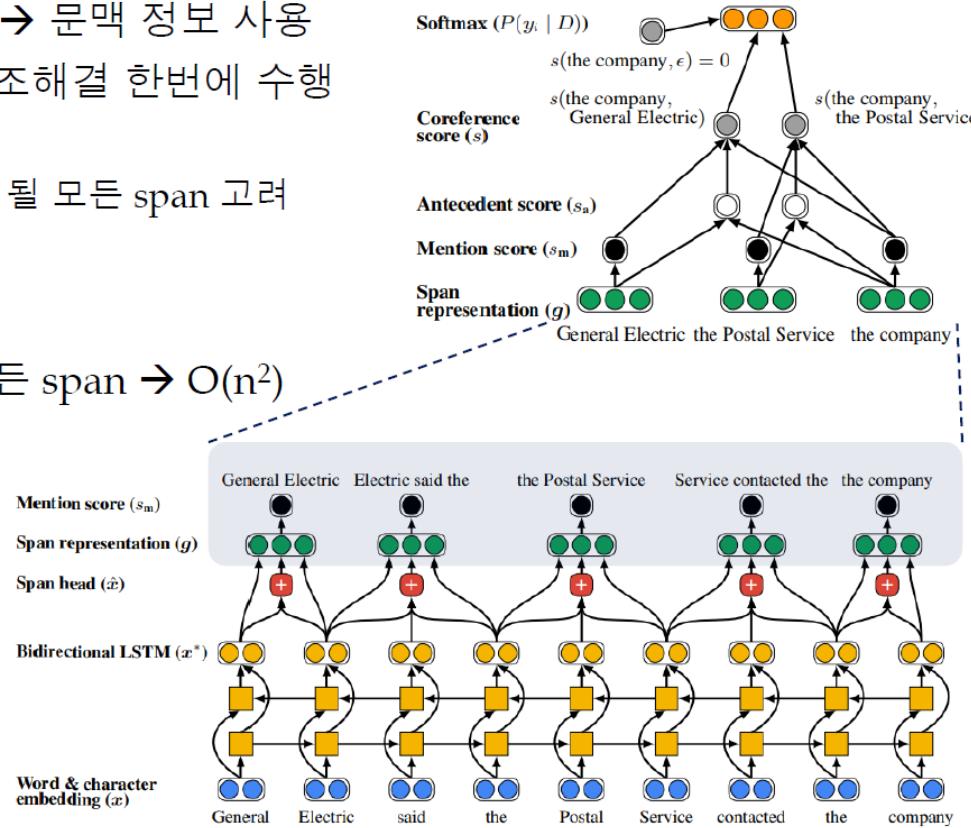
# DNN 기반 상호참조 해결 모델

## End-to-end neural coref. (E2E-coref, EMNLP 17)

- 멘션 랭킹 모델
- LSTM, 어텐션 사용 → 문맥 정보 사용
- 멘션 탐지와 상호참조해결 한번에 수행
  - 멘션 탐지 단계 X
  - 멘션 후보로 사용 될 모든 span 고려

## E2E-coref 단점

- 문서에 등장하는 모든 span  $\rightarrow O(n^2)$
- 시간 복잡도:  $O(n^4)$
- Pruning 적용  
(coarse-to-fine)  
 $\rightarrow O(n^3)$

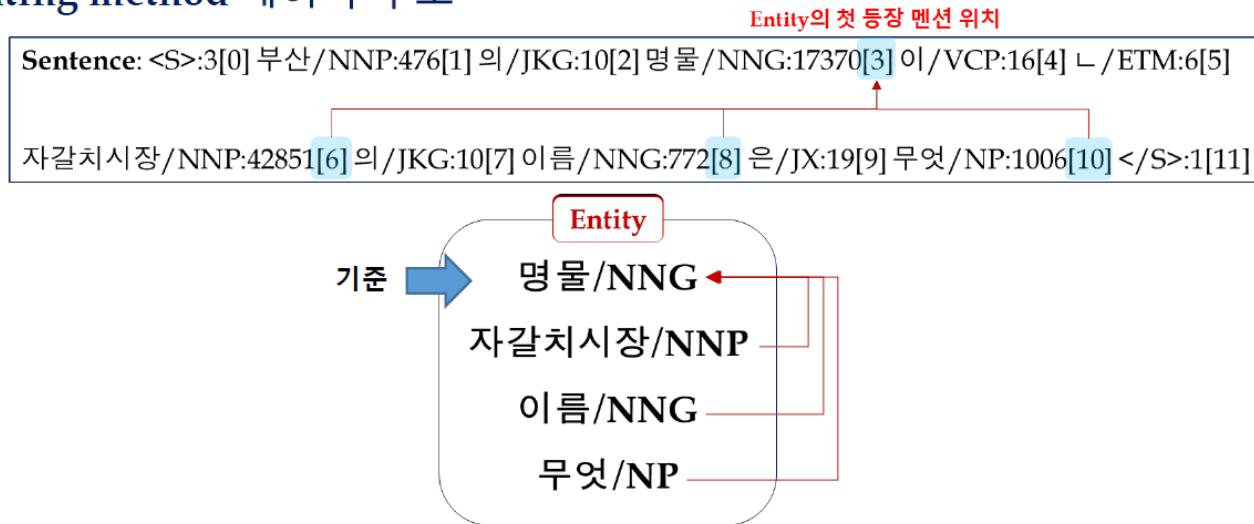


# DNN 기반 상호참조 해결 모델

## | 포인터 네트워크를 이용한 상호참조해결 데이터 구조

- 포인터 네트워크: RNN Encoder-Decoder 기반
- 인코더 입력: 입력 문서
- 디코더 입력: 입력 문서에 포함된 중심어
- 디코더 출력: 입력 문서의 위치
  - Pointing method: 모든 중심어는 엔티티의 첫 번째 위치한 중심어를 가리킴

## Pointing method 데이터 구조



# DNN 기반 상호참조 해결 모델

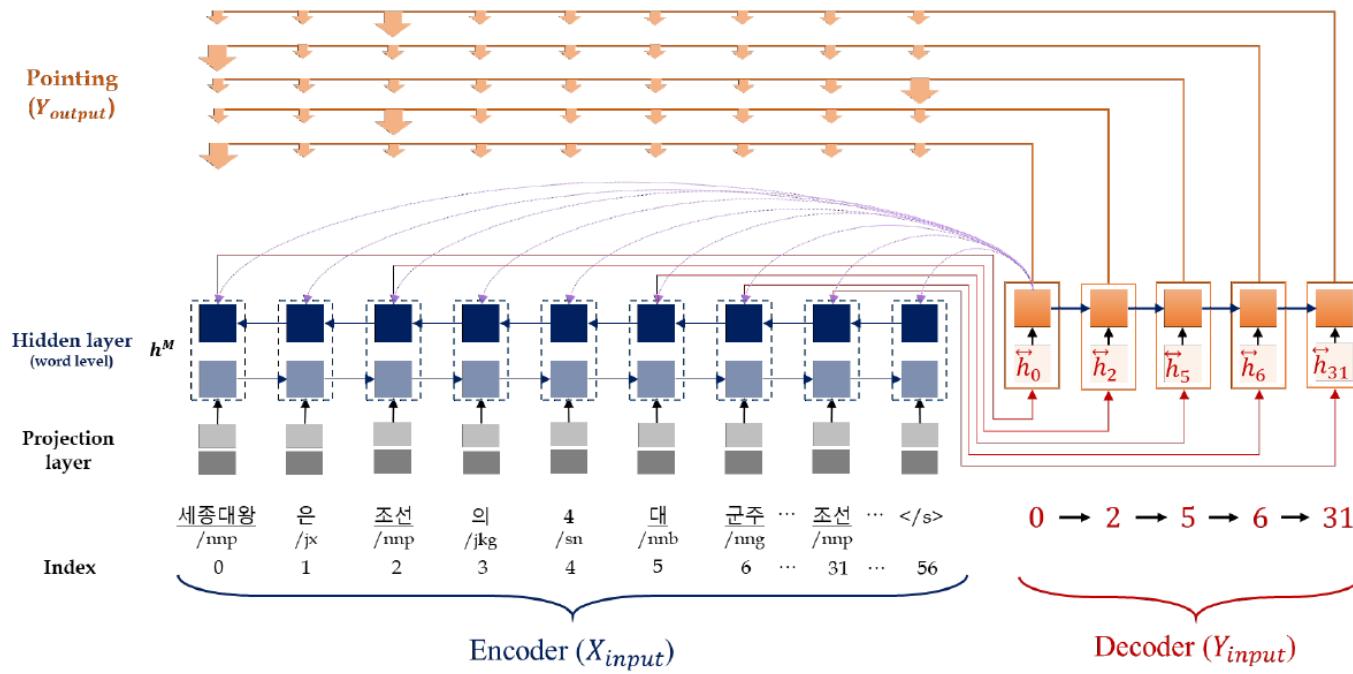
## | 포인터 네트워크를 이용한 상호참조해결의 데이터 구조

Encoder input sentence	Decoder				
	index	input sentence	sequence	output sentence	sequence
<S>[0] 이것/NP[1] 은/JX[2] 세종대왕/NNP[3] 이/JKS[4] 훈민정음/NNP[5] 창제/NNG[6] 이/ 후/NNG[7] ,/SP[8] 훈란/NNG[9] 상태/NNG[10] 에/JKB[11] 있/VA[12] 던/ETM[13] 우리/NNP[14] 나라/NNG[15] 의/JKG[16] 한자/NNG[17] 음/NNG[18] 을/JKO[19] 바로잡/VV[20] 아/EC[21] 통일/NNG[22] 되/XSV[23] ←/ETM[24] 표 준/NNG[25] 음/NNG[26] 을/JKO[27] 정하/VV[28] 려는/ETM[29] 목적/NNG[30] 에서/JKB[31] 신속 주/NNP[32] 등/NNP[33]에게/JKB[34] 명하/VV[35] 여/EC[36] 편찬/NNG[37] 하/XSV[38] 게/EC[39] 하/VV[40] ←/ETM[41] 책/NNG[42] 이/VCP[43] 다/EF[44] ./SF[45] </S>[46] <S>[47] '/SS[48] 우 리/NNP[49] 나라/NNG[50] 의/JKG[51] 바 르/VA[52] ←/ETM[53] 음/NNG[54] '/SS[55] 이/VCP[56] 라는/ETM[57] 뜻/NNG[58] 의/JKG[59] 이것/NP[60] 은/JX[61] 무엇/NP[62] 이/VCP[63] ≡ 까/EF[64] ?/SF[65] </S>[66]	0	이것/NP	1	이것/NP	1
	1	세종대왕/NNP	3	</S>	66
	2	훈민정음/NNG	5	</S>	66
	3	창제/NNG	6	</S>	66
	4	이후/NNG	7	</S>	66
	5	훈란/NNG	9	</S>	66
	6	상태/NNG	10	</S>	66
	7	우리/NNP	14	</S>	66
	8	나라/NNG	15	</S>	66
	9	한자/NNG	17	</S>	66
	10	음/NNG	18	</S>	66
	11	통일/NNG	22	</S>	66
	12	표준/NNG	25	</S>	66
	13	음/NNG	26	</S>	66
	14	목적/NNG	30	</S>	66
	15	신속주/NNP	32	</S>	66
	16	등/NNB	33	</S>	66
	17	편찬/NNG	37	</S>	66
	18	책/NNG	42	이것/NP	1
	19	우리/NNP	49	</S>	66
	20	나라/NNG	50	</S>	66
	21	음/NNG	54	</S>	66
	22	뜻/NNG	58	이것/NP	1
	23	이것/NP	60	이것/NP	1
	24	무엇/NP	62	이것/NP	1



# 포인터 네트워크 모델

| 질문 문서 데이터셋 사용 (Train: 1,819, Test set: 221)



Model (TEST)	Pre	Rec	F1
규칙기반	60.52	56.77	58.57
규칙기반 + 포인터 네트워크 (대명사)	70.45	61.76	65.77
포인터 네트워크	85.62	58.58	69.53

+10.96



# BERT 기반 상호참조 해결 모델

멘션 탐지와 상호참조해결을 함께 수행:  
multi-task learning

Deep biaffine score 적용

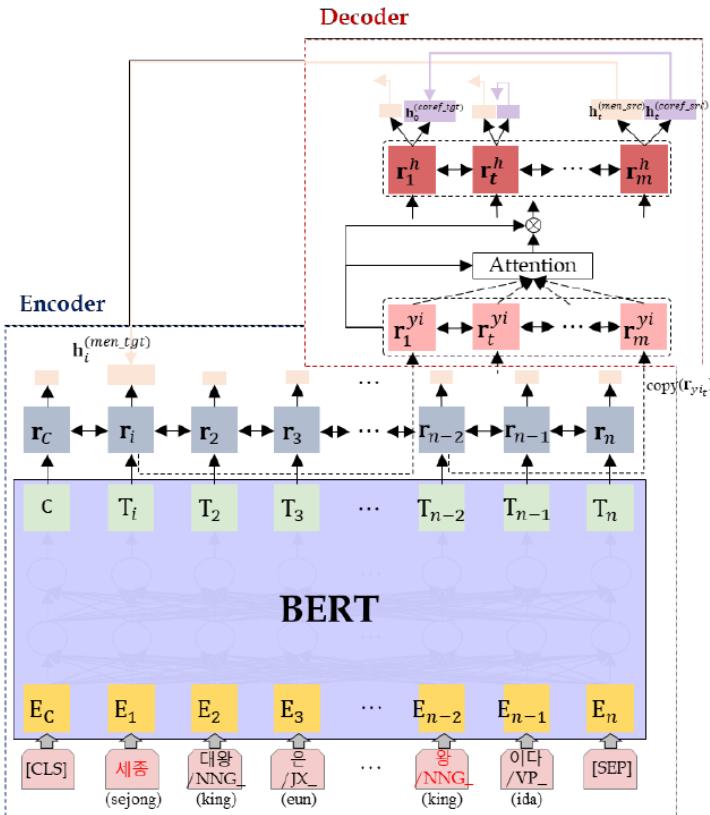
Attention mask 적용

- 상호참조해결의 포인팅 대상은 앞쪽에 등장
- 현재 타임 중심어의 뒤에 등장하는 히든 상태에 mask 적용 → 앞쪽으로 집중

중심어 리스트 타겟 클래스 사용

- 기존 방법: 입력 열의 위치를 타겟 클래스로 사용  
→ 타겟 클래스 크기: 입력 문서 전체
- 제안 방법: 입력 열에서 중심어(명사)를 추려낸 리스트를 타겟 클래스로 사용  
→ 타겟 클래스 크기: 중심어 리스트
- 효과: 타겟 클래스의 크기가 줄어듦

질문-정답 쌍 데이터: ETRI 상호참조해결 (Train: 2,819, Test set: 571)



Model (DEV)	MUC	B <sup>3</sup>	CEAFe	F1
규칙기반 (baseline)	44.25	45.21	50.91	46.79
포인터 네트워크	59.07	55.16	53.58	55.94
<b>BERT-coref</b>	<b>71.00</b>	<b>69.09</b>	<b>70.43</b>	<b>70.17</b>
Model (TEST)	MUC	B <sup>3</sup>	CEAFe	F1
포인터 네트워크	51.42	47.34	43.21	47.32
<b>BERT-coref</b>	<b>65.76</b>	<b>63.85</b>	<b>65.23</b>	<b>64.95</b>

+14.23

+17.63



---

# Dialogue Management

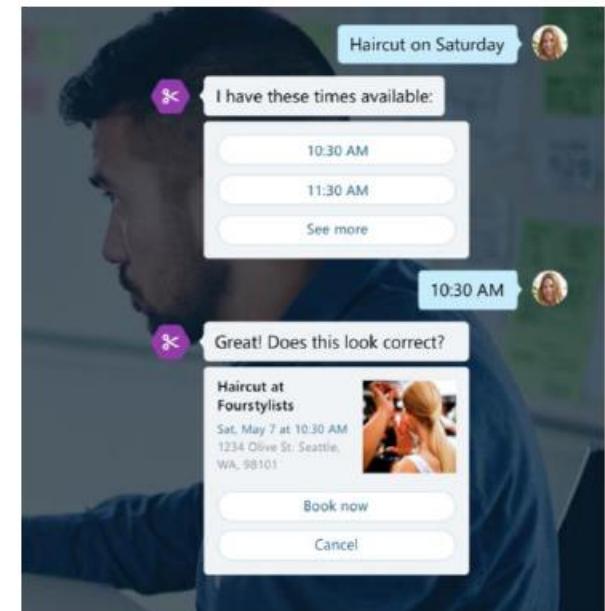
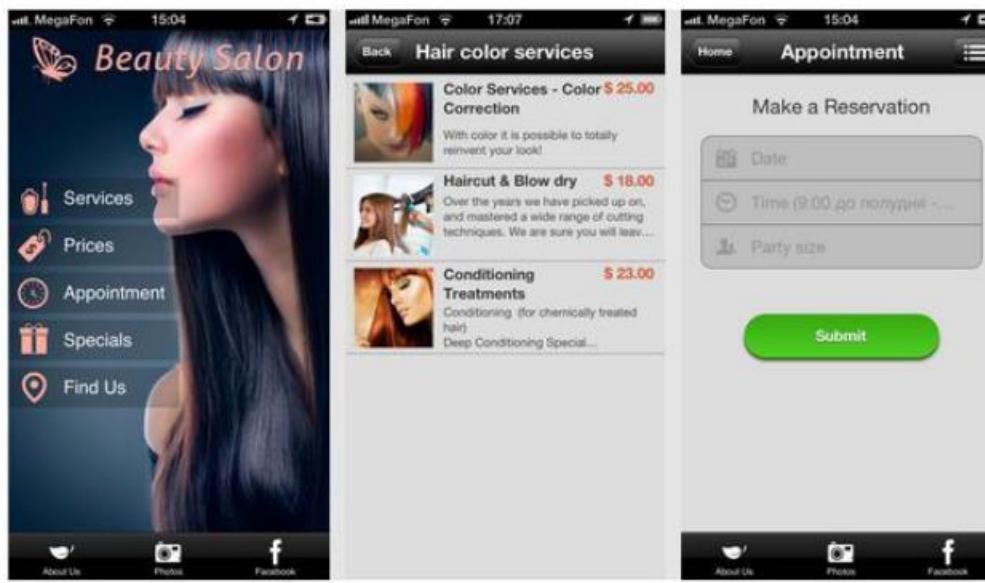
---

건국대학교 컴퓨터공학부 /  
KAIST 전산학부 (겸직)

김학수

# App → Bot

- A **bot** is responsible for a “single” domain, similar to an app



Users can initiate dialogues instead of following the GUI design



# GUI vs. CUI (Conversational UI)

---

	Website/APP's GUI	Msg's CUI
<b>Situation</b>	Navigation, no specific goal	Searching, with specific goal
<b>Information Quantity</b>	More	Less
<b>Information Precision</b>	Low	High
<b>Display</b>	Structured	Non-structured
<b>Interface</b>	Graphics	Language
<b>Manipulation</b>	Click	mainly use <b>texts or speech</b> as input
<b>Learning</b>	Need time to learn and adapt	No need to learn
<b>Entrance</b>	App download	Incorporated in any msg-based interface
<b>Flexibility</b>	Low, like machine manipulation	High, like converse with a human



# Two Branches of Bots

## Task-Oriented Bot

- Personal assistant, helps users achieve a certain task
- Combination of rules and statistical components
  - ▣ POMDP for spoken dialog systems (Williams and Young, 2007)
  - ▣ End-to-end trainable task-oriented dialogue system (Wen et al., 2016)
  - ▣ End-to-end reinforcement learning dialogue system (Li et al., 2017; Zhao and Eskenazi, 2016)



식당 예약, 기차표 예약 등

## Chit-Chat Bot

- No specific goal, focus on natural responses
- Using variants of seq2seq model
  - ▣ A neural conversation model (Vinyals and Le, 2015)
  - ▣ Reinforcement learning for dialogue generation (Li et al., 2016)
  - ▣ Conversational contextual cues for response ranking (Al-Rfou et al., 2016)



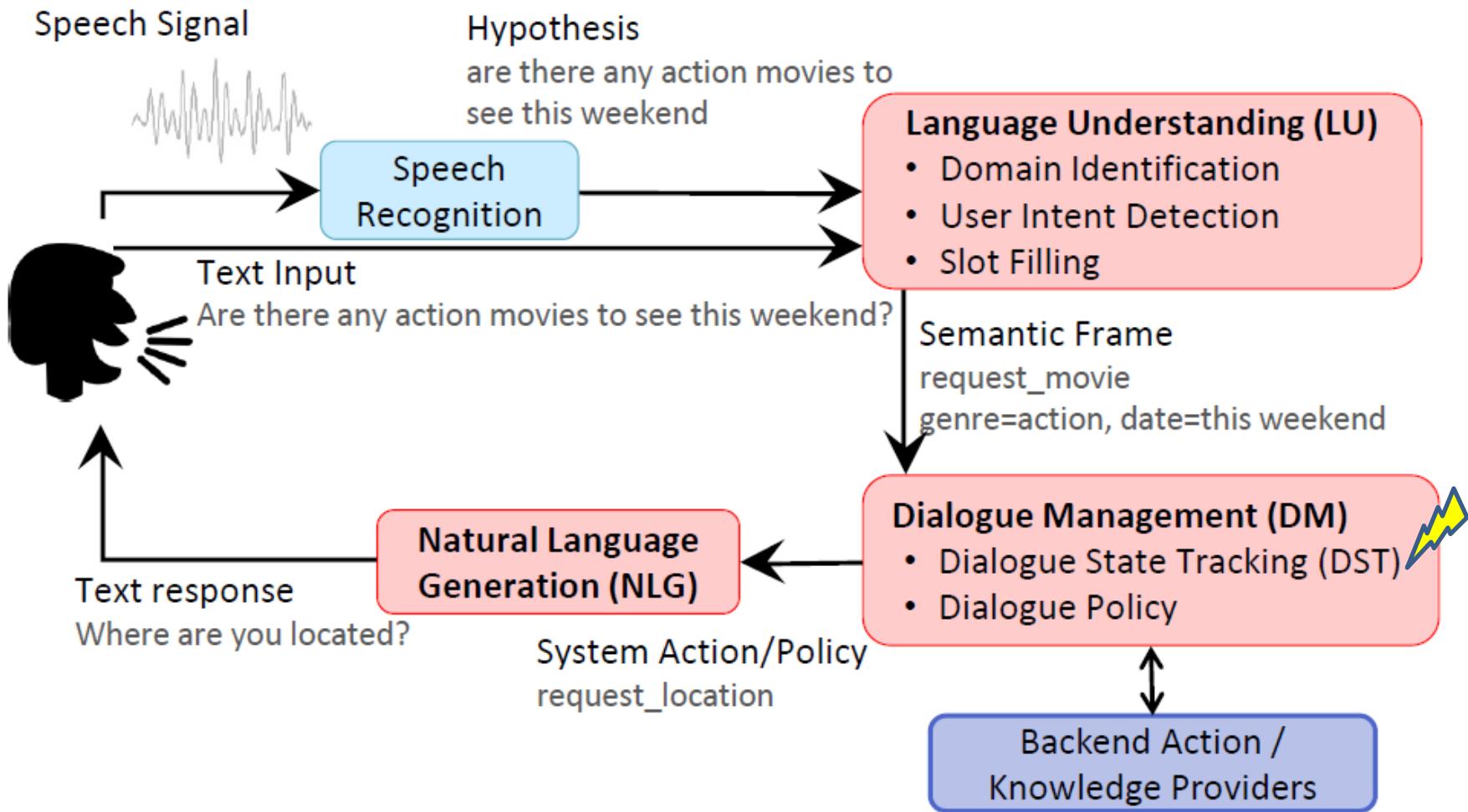
신변잡기 대화, 튜링 테스트 등



# Task-Oriented Bot

---

# Task-Oriented Dialogue System



# State Tracking

User

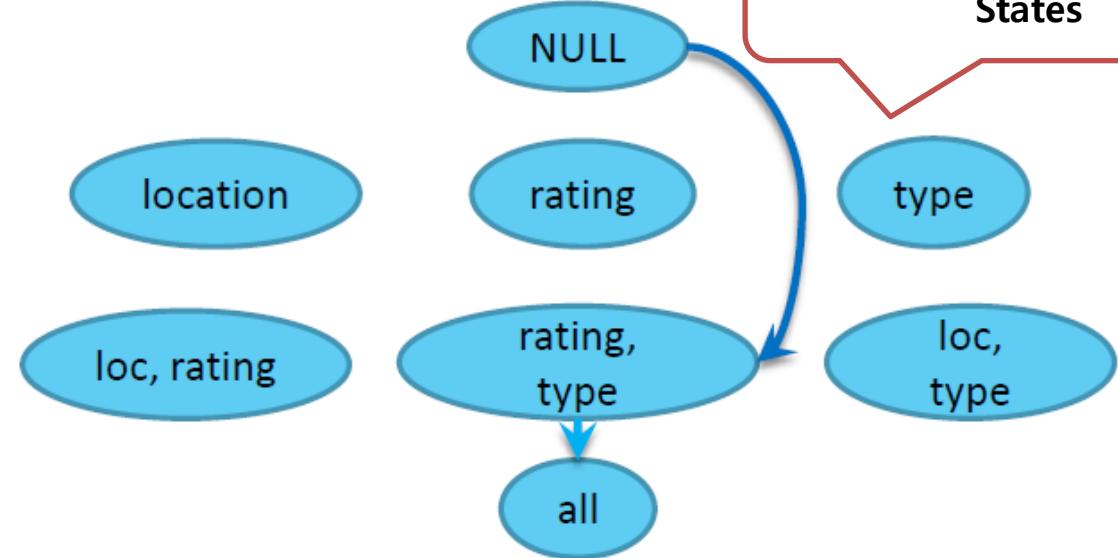


find a good eating place for taiwanese food

i want it near to my office



Intelligent  
Agent



# State Tracking

User



find a good eating place for taixxxx food

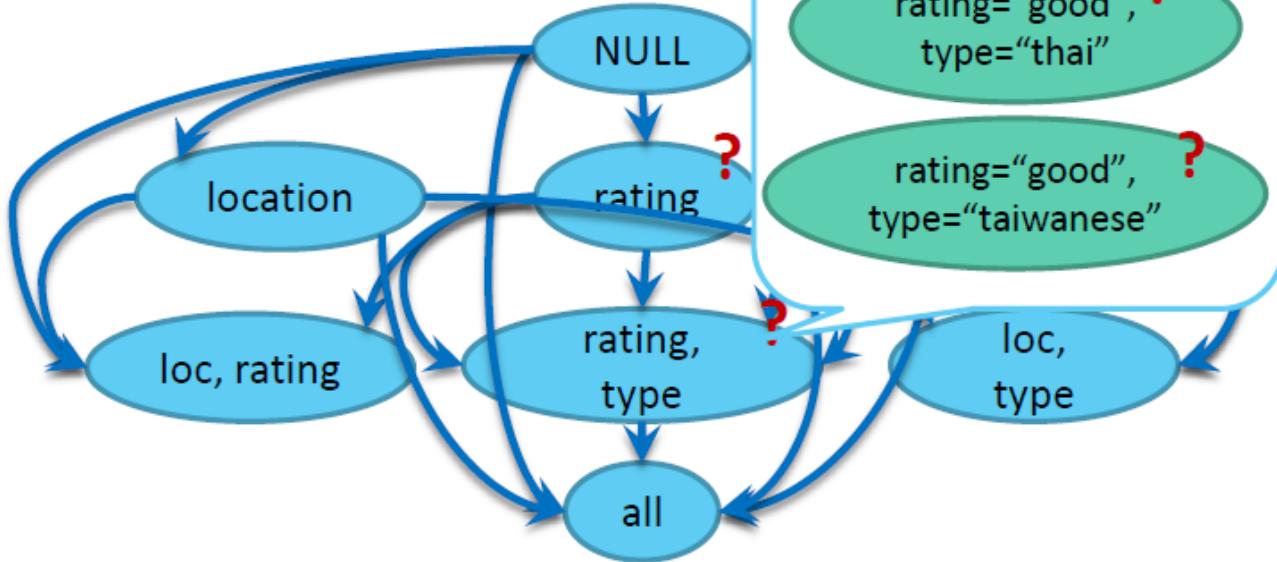
FIND\_RESTAURANT  
rating="good"  
type="taiwanese"

FIND\_RESTAURANT  
rating="good"  
type="thai"

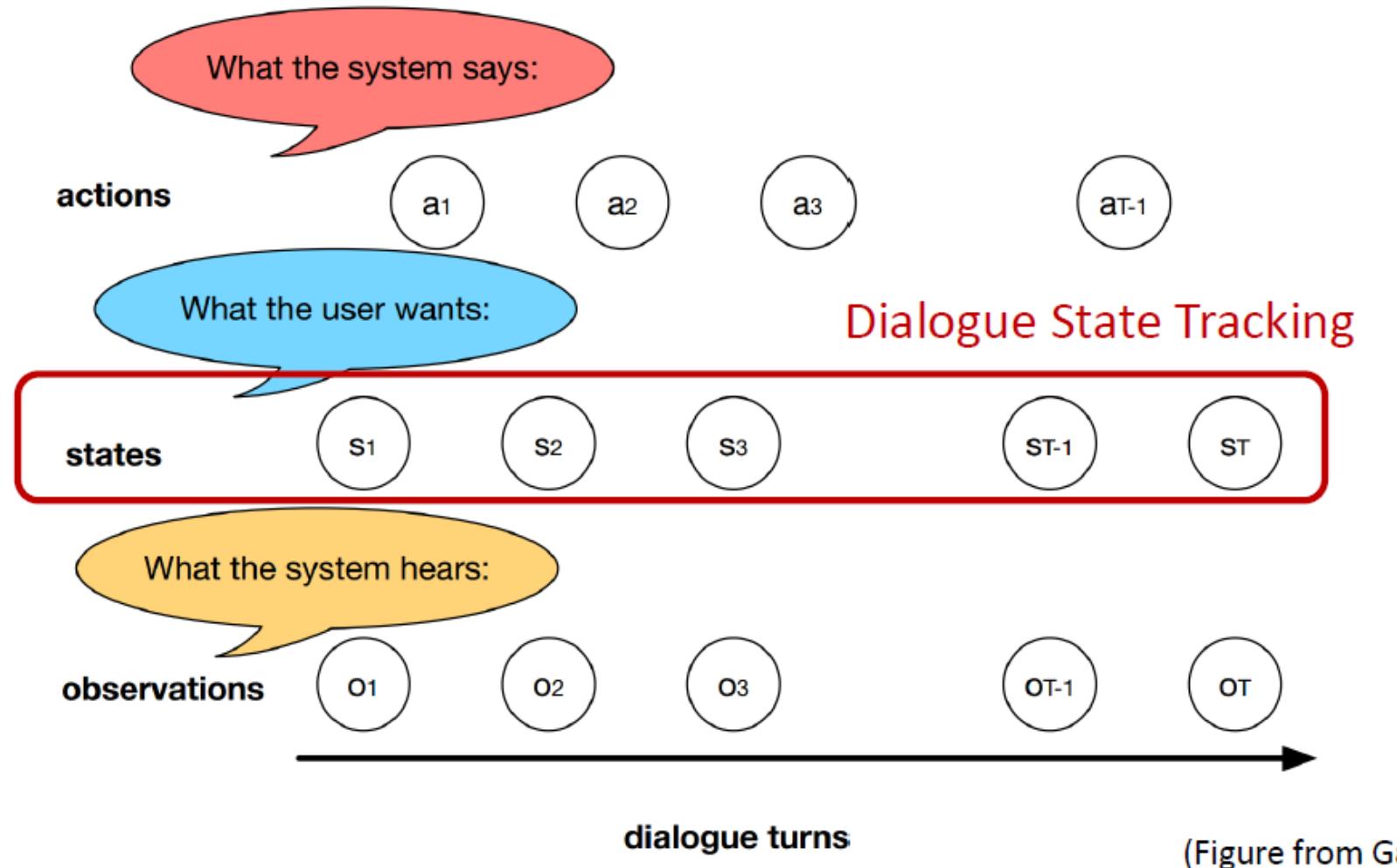
FIND\_RESTAURANT  
rating="good"



Intelligent  
Agent



# Elements of Dialogue Management

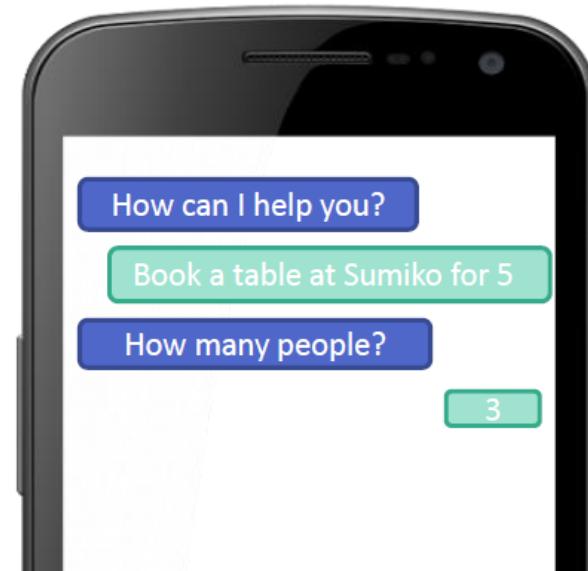


# Dialogue State Tracking

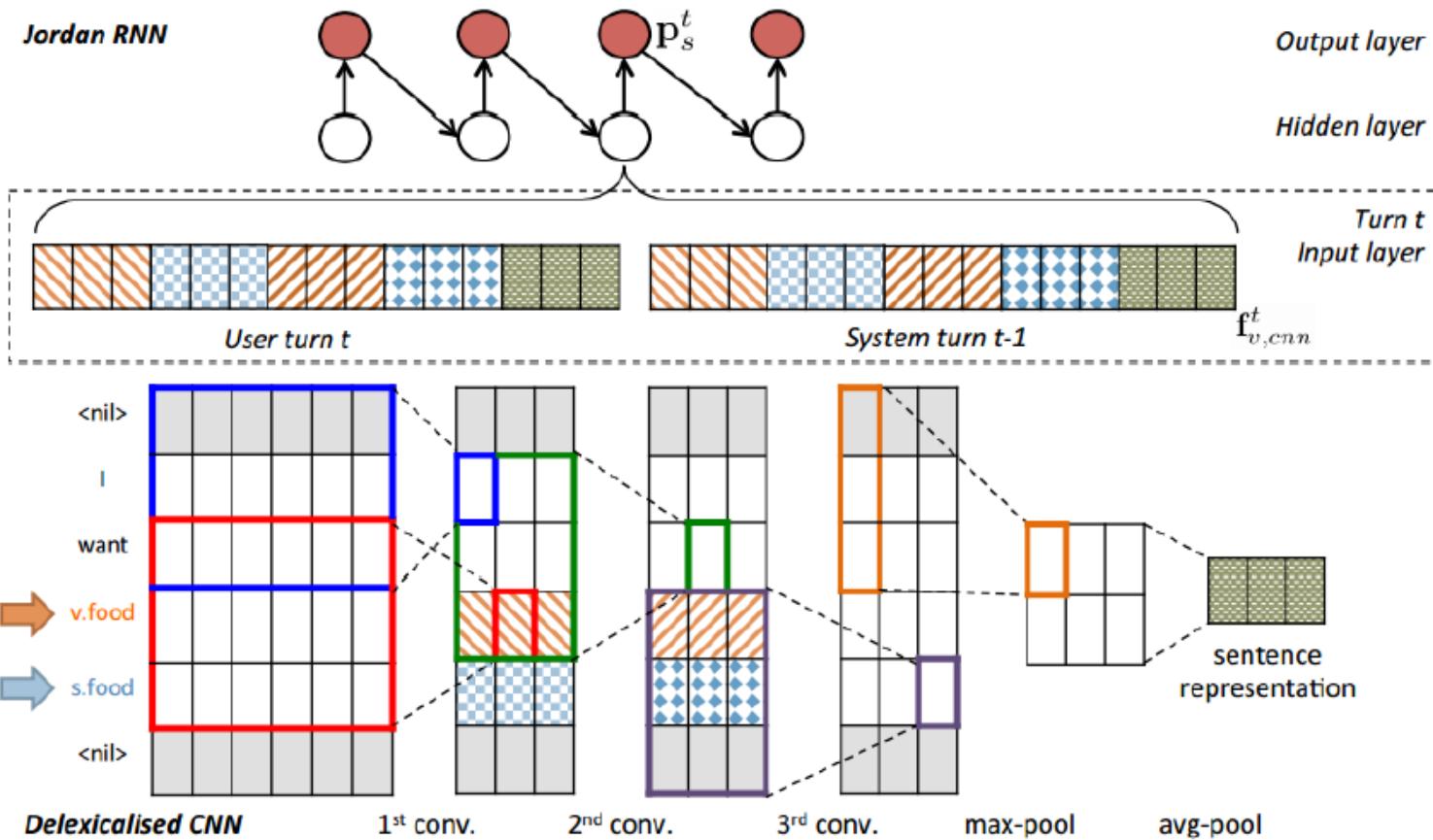
- Maintain a probabilistic distribution instead of a 1-best prediction for better robustness to SLU errors or ambiguous input

Slot	Value
# people	5 (0.5)
time	5 (0.5)

Slot	Value
# people	3 (0.8)
time	5 (0.8)



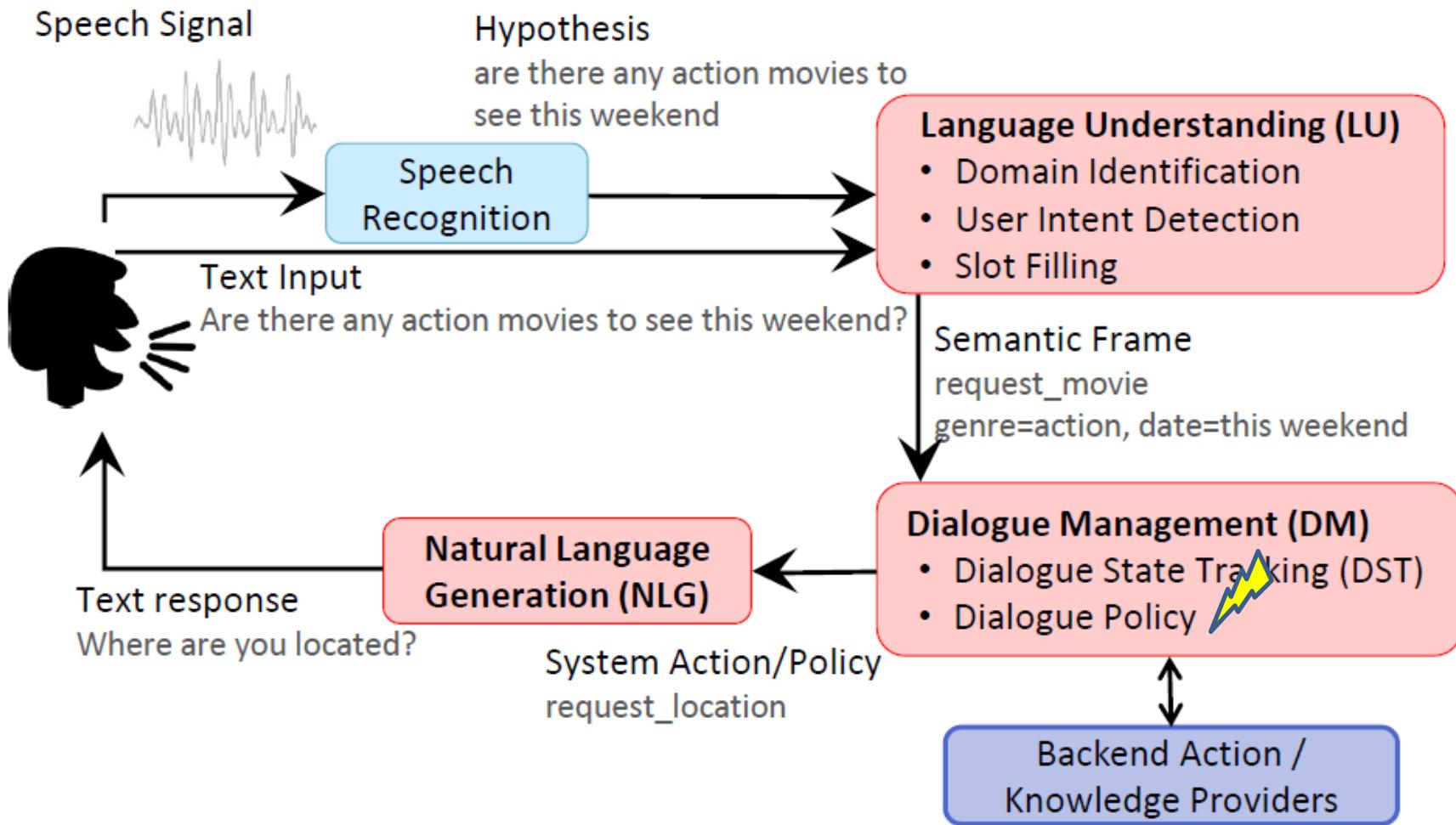
# NN-Based DST



(Figure from Wen et al, 2016)



# Task-Oriented Dialogue System



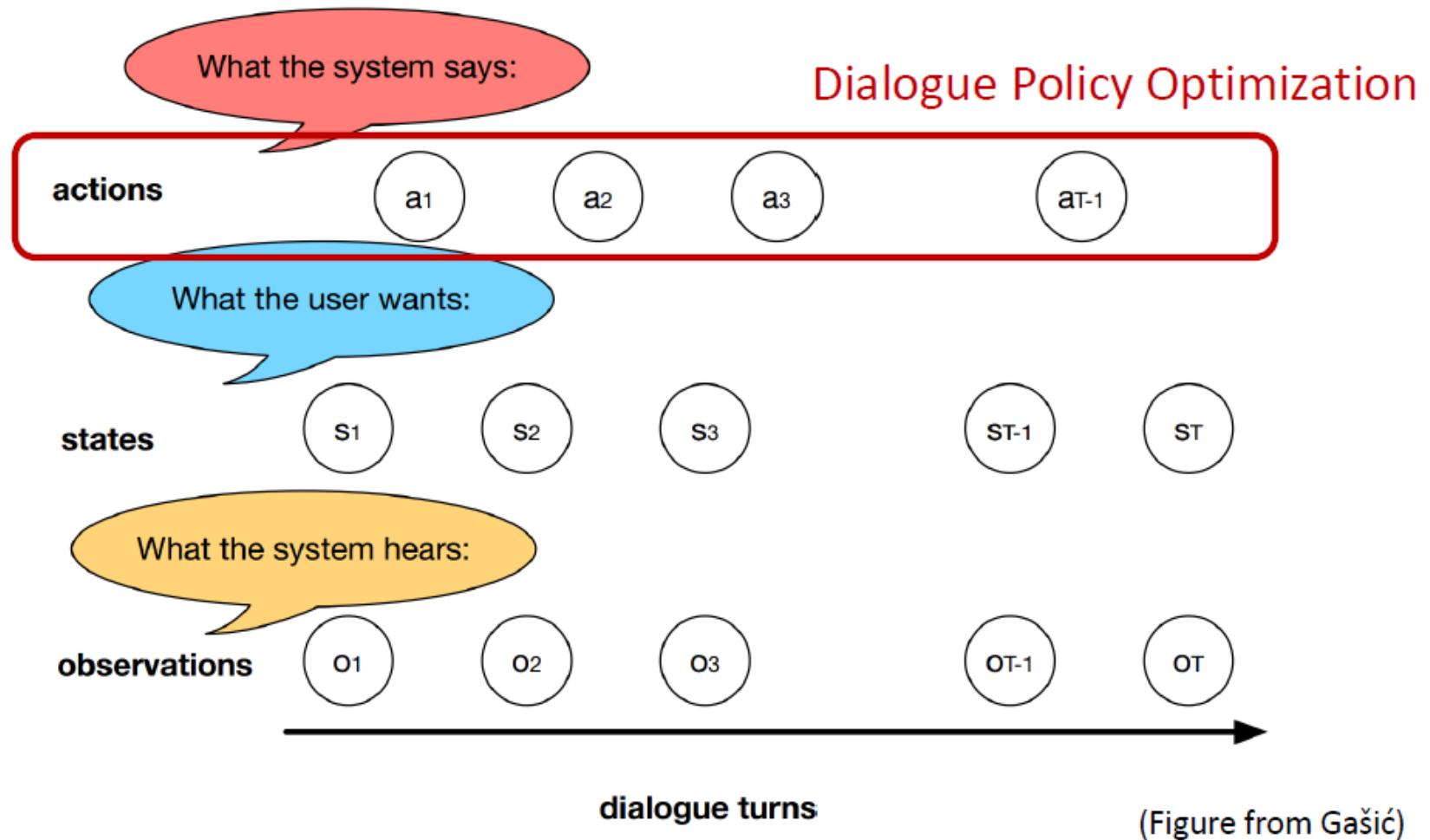
# Dialogue Policy for Agent Action

---

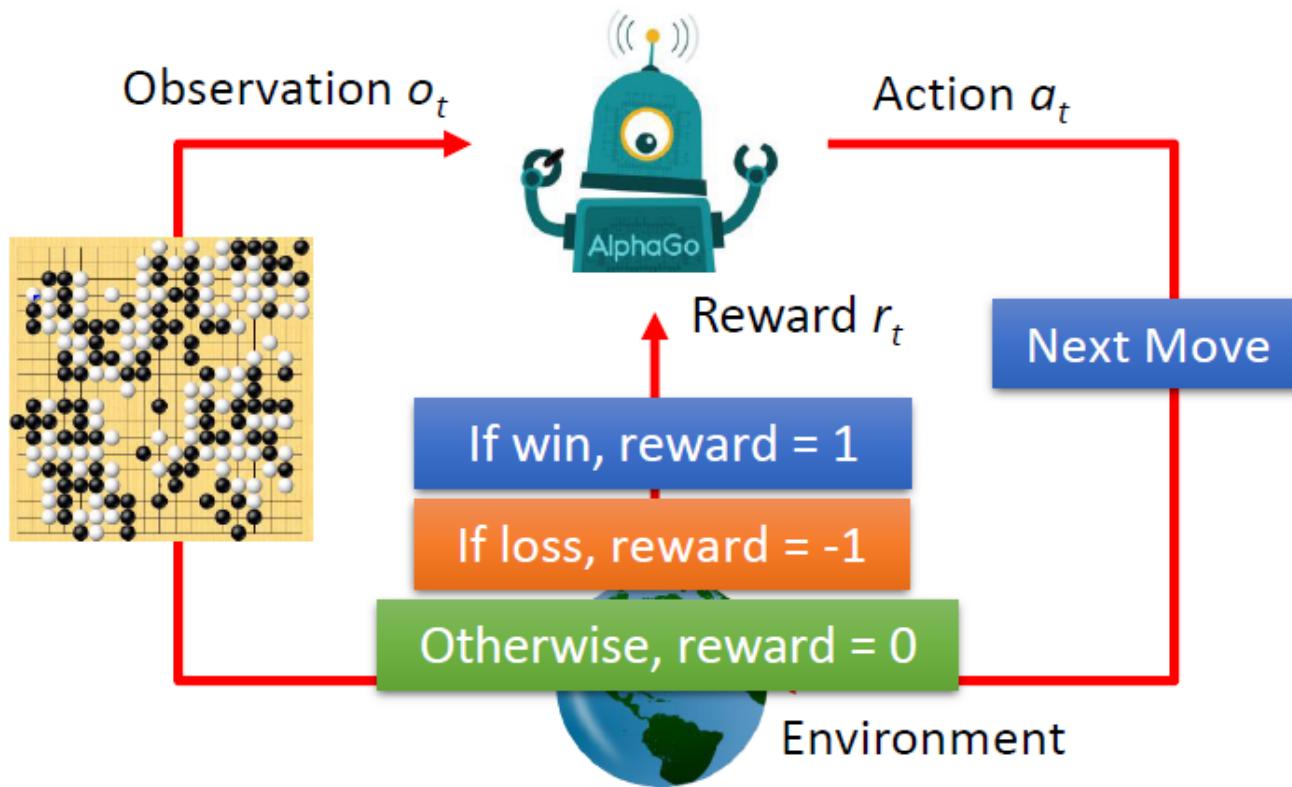
- Inform(location="Taipei 101")
  - "The nearest one is at Taipei 101"
- Request(location)
  - "Where is your home?"
- Confirm(type="taiwanese")
  - "Did you want Taiwanese food?"



# Elements of Dialogue Management



# Scenario of Reinforcement Learning



<https://www.youtube.com/watch?v=B-CZv9WD5eM/>



# Supervised vs. Reinforcement

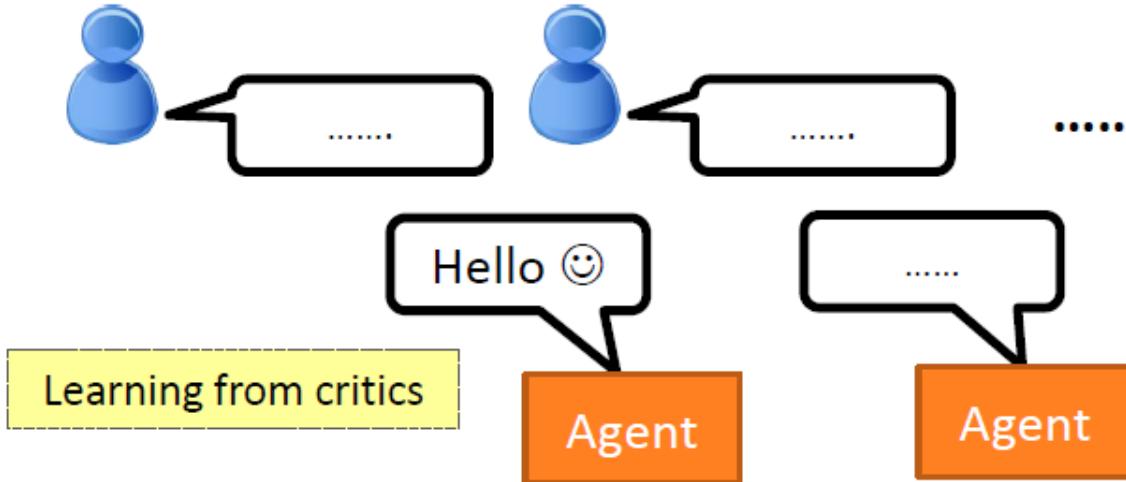
---

## □ Supervised

Learning from teacher



## □ Reinforcement



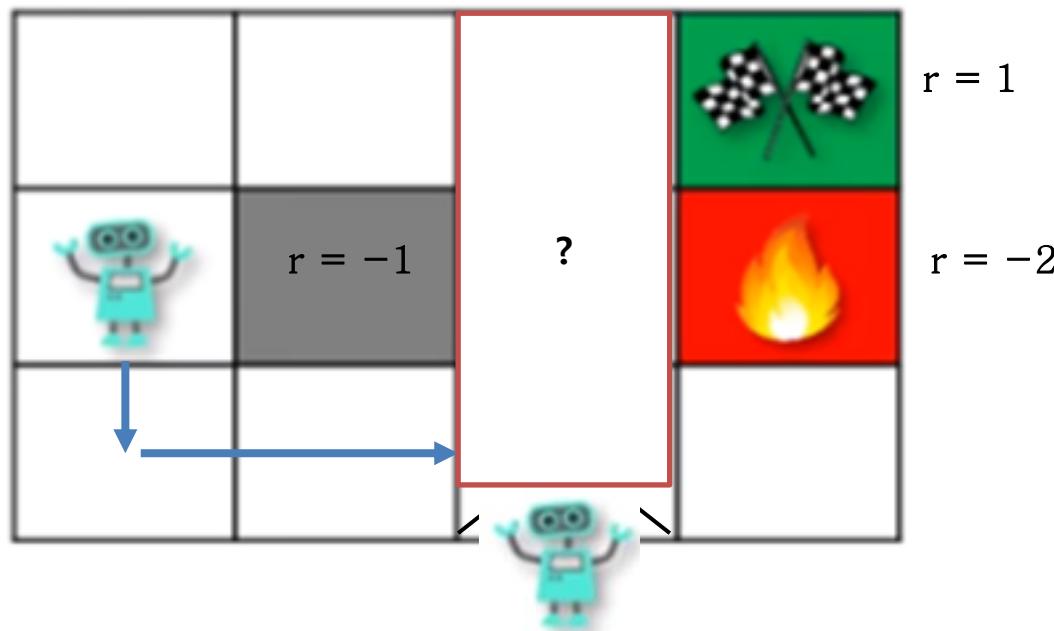
Bad



# 확인 문제: Q-Learning

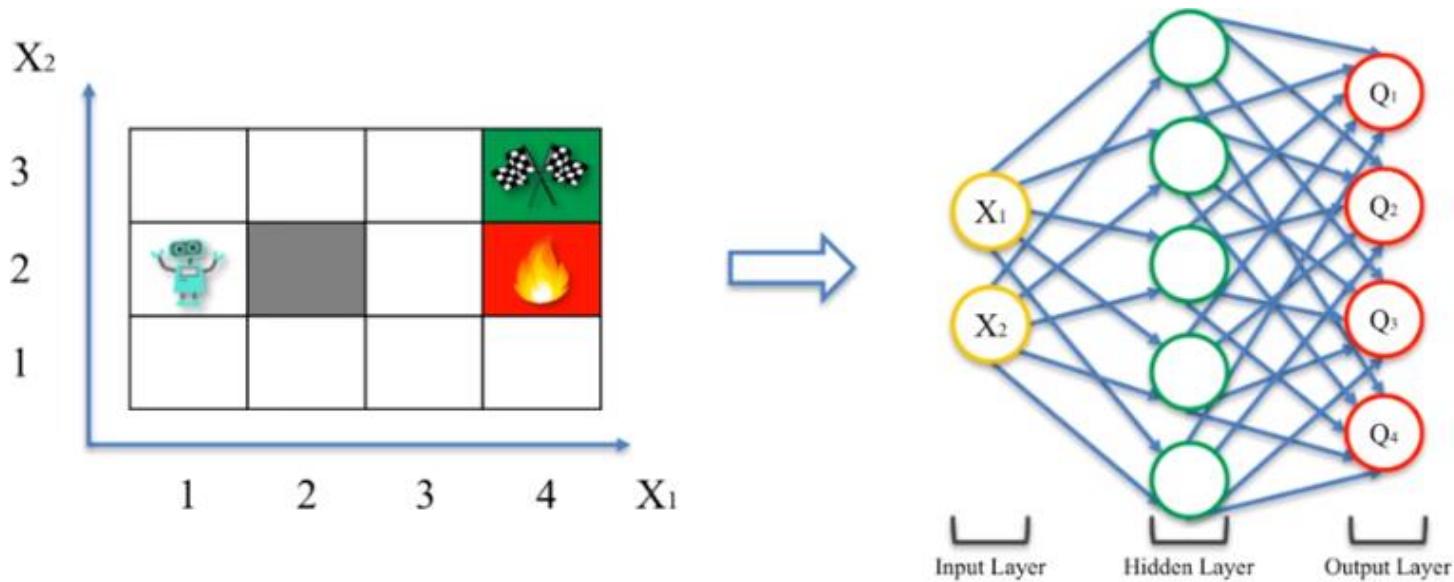
다음 Dummy Q-Learning 수식을 바탕으로 빈칸을 채우시오.

$$\hat{Q}(s, a) \leftarrow r + \gamma \max_{a'} \hat{Q}(s', a'), \text{ where } \gamma = 0.9$$

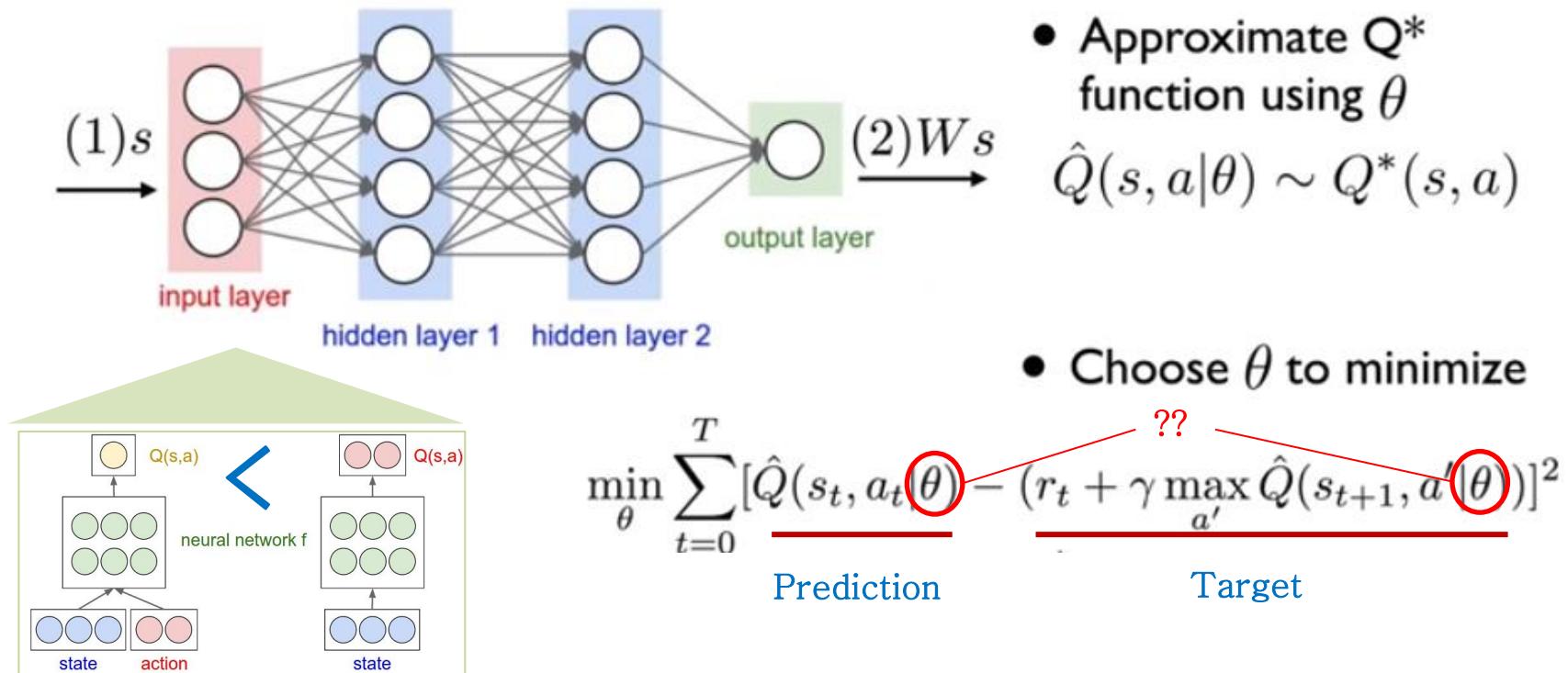


# Deep Reinforcement Learning

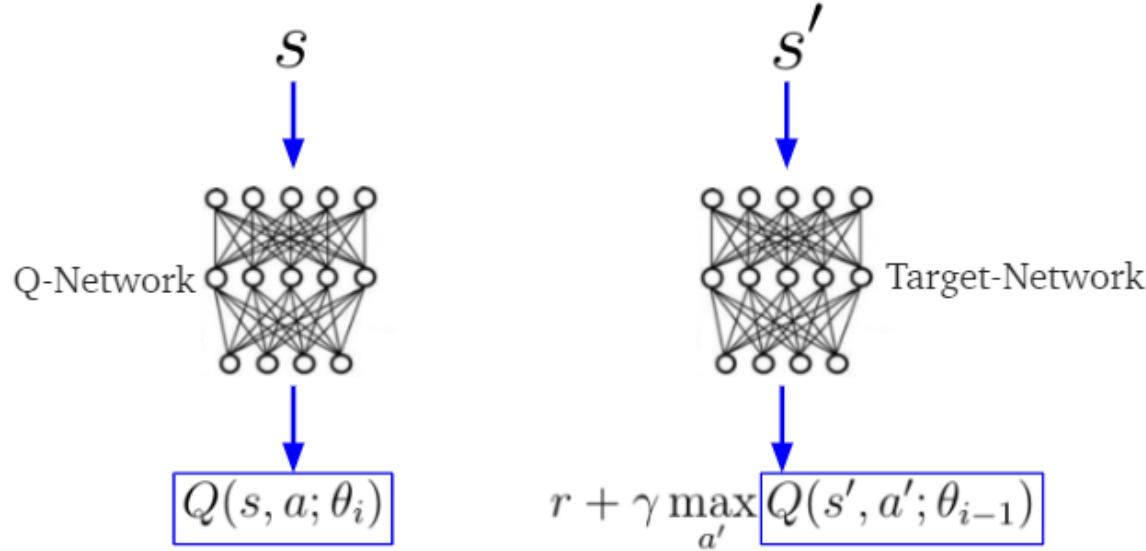
Q-Table → DQN (Deep Q-Network)



# DQN



# Nonstationary Target Problem in DQN



Separate Target



Copy

Every C steps reset  $\hat{Q} = Q$

$$\min_{\theta} \sum_{t=0}^T [\hat{Q}(s_t, a_t | \theta) - (r_t + \gamma \max_{a'} \hat{Q}(s_{t+1}, a' | \theta))]^2$$

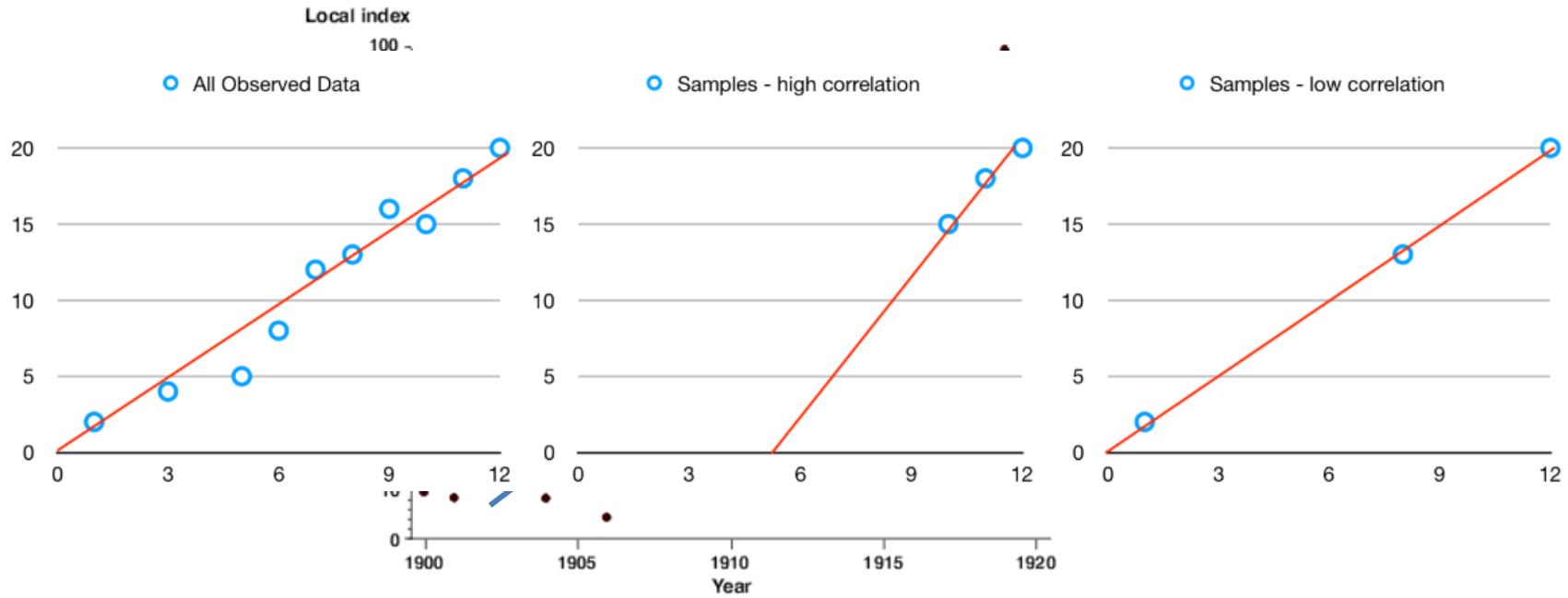


Non-stationary target problem

$$\min_{\theta} \sum_{t=0}^T [\hat{Q}(s_t, a_t | \theta) - (r_t + \gamma \max_{a'} \hat{Q}(s_{t+1}, a' | \bar{\theta}))]^2$$



# Correlation Problem in DQN

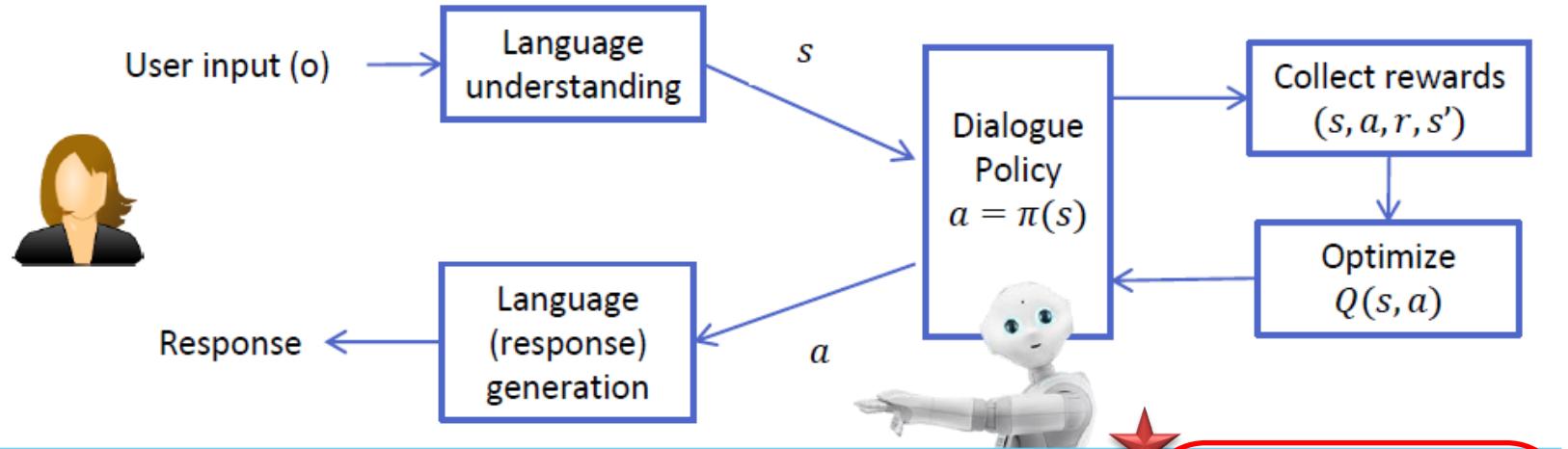


Experience replay

Store transition  $(\phi_t, a_t, r_t, \phi_{t+1})$  in  $\mathcal{D}$   
Sample random minibatch of transitions  $(\phi_j, a_j, r_j, \phi_{j+1})$  from  $\mathcal{D}$   
Set  $y_j = \begin{cases} r_j & \text{for terminal } \phi_{j+1} \\ r_j + \gamma \max_{a'} Q(\phi_{j+1}, a'; \theta) & \text{for non-terminal } \phi_{j+1} \end{cases}$   
Perform a gradient descent step on  $(y_j - Q(\phi_j, a_j; \theta))^2$  according to equation 3



# RL for Dialogue Policy Optimization



Type of Bots	State	Action	Reward
Social ChatBots	Chat history	System Response	# of turns maximized; Intrinsically motivated reward
InfoBots (interactive Q/A)	User current question + Context	Answers to current question	Relevance of answer; # of turns minimized
Task-Completion Bots	User current input + Context	System dialogue act w/ slot value (or API calls)	Task success rate; # of turns minimized

Goal: develop a generic deep RL algorithm to learn dialogue policy for all bot categories

# Dialogue Reinforcement Learning

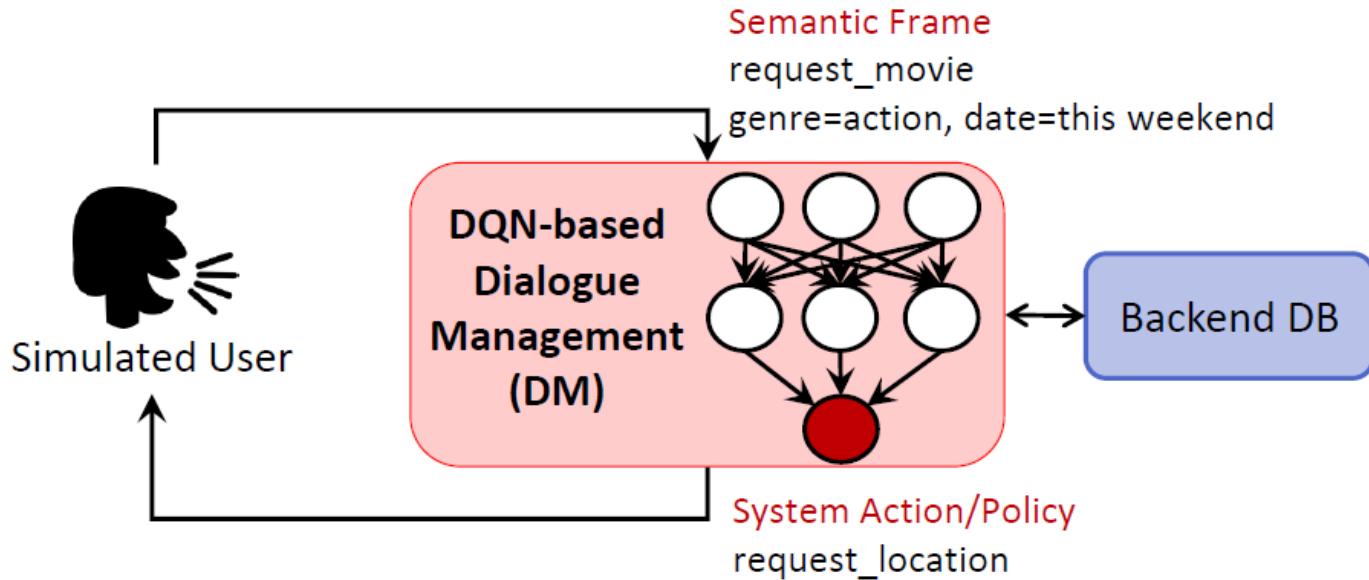
- Typical reward function
  - -1 for per turn penalty
  - Large reward at completion if **successful**
- Typically requires **domain knowledge**
  - ✓ Simulated user
  - ✗ Paid users (Amazon Mechanical Turk)
  - ✗ Real users

The user simulator is usually required for dialogue system training before deployment

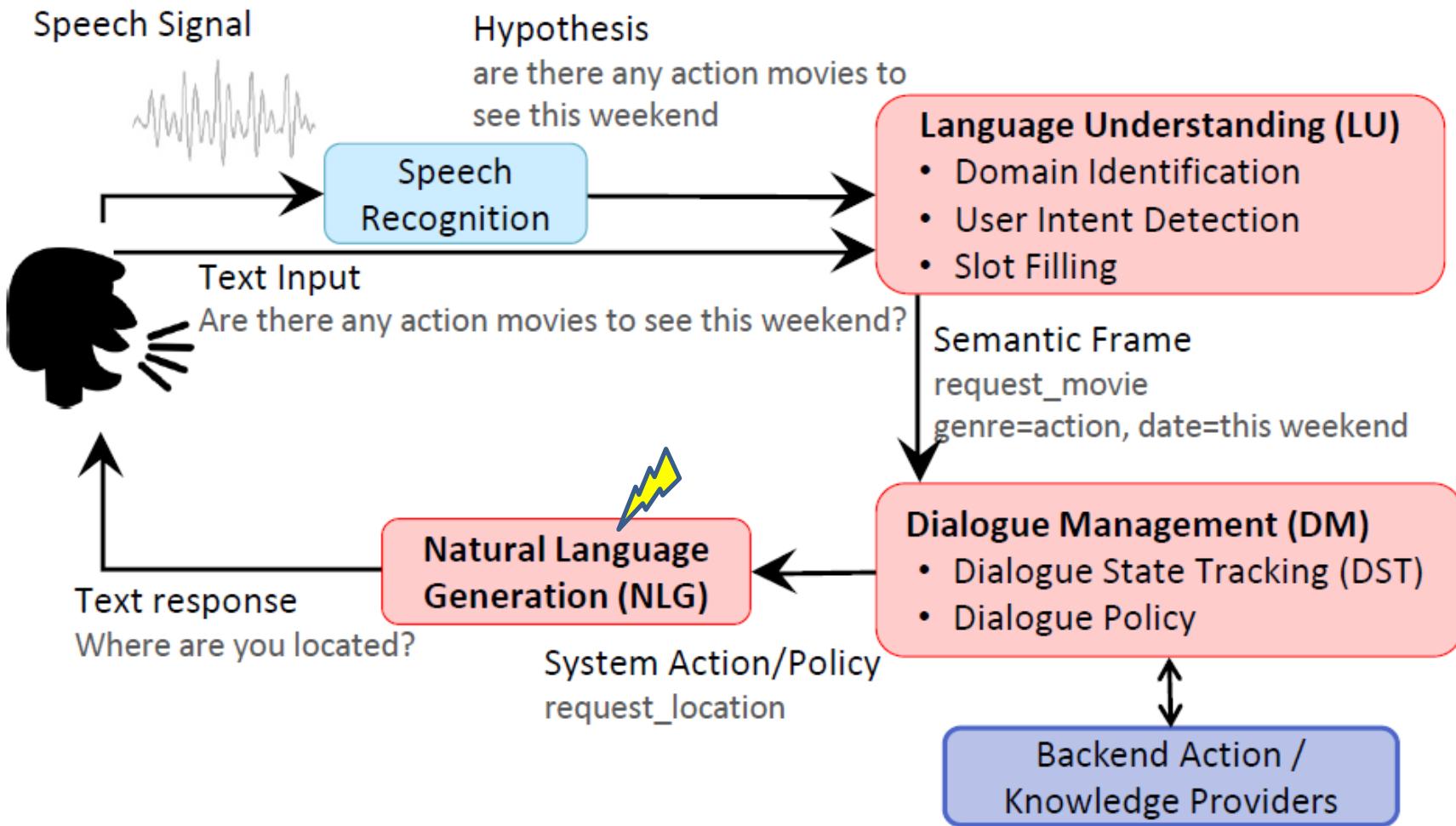


# Neural Dialogue Manager

- Deep Q-network for training DM policy (Li et al., 2017)
- Input: current semantic frame observation, database returned results
- Output: system action



# Natural Language Generation



# Natural Language Generation

---

- Mapping semantic frame into natural language
  - Inform(name=Seven-Days, food type=Chinese)  
→ Seven Days is a nice Chinese restaurant.
- Approaches
  - Template-based NLG
  - Phrase-based NLG
  - RNN-based LM NLG
  - ...



# Template-base NLG

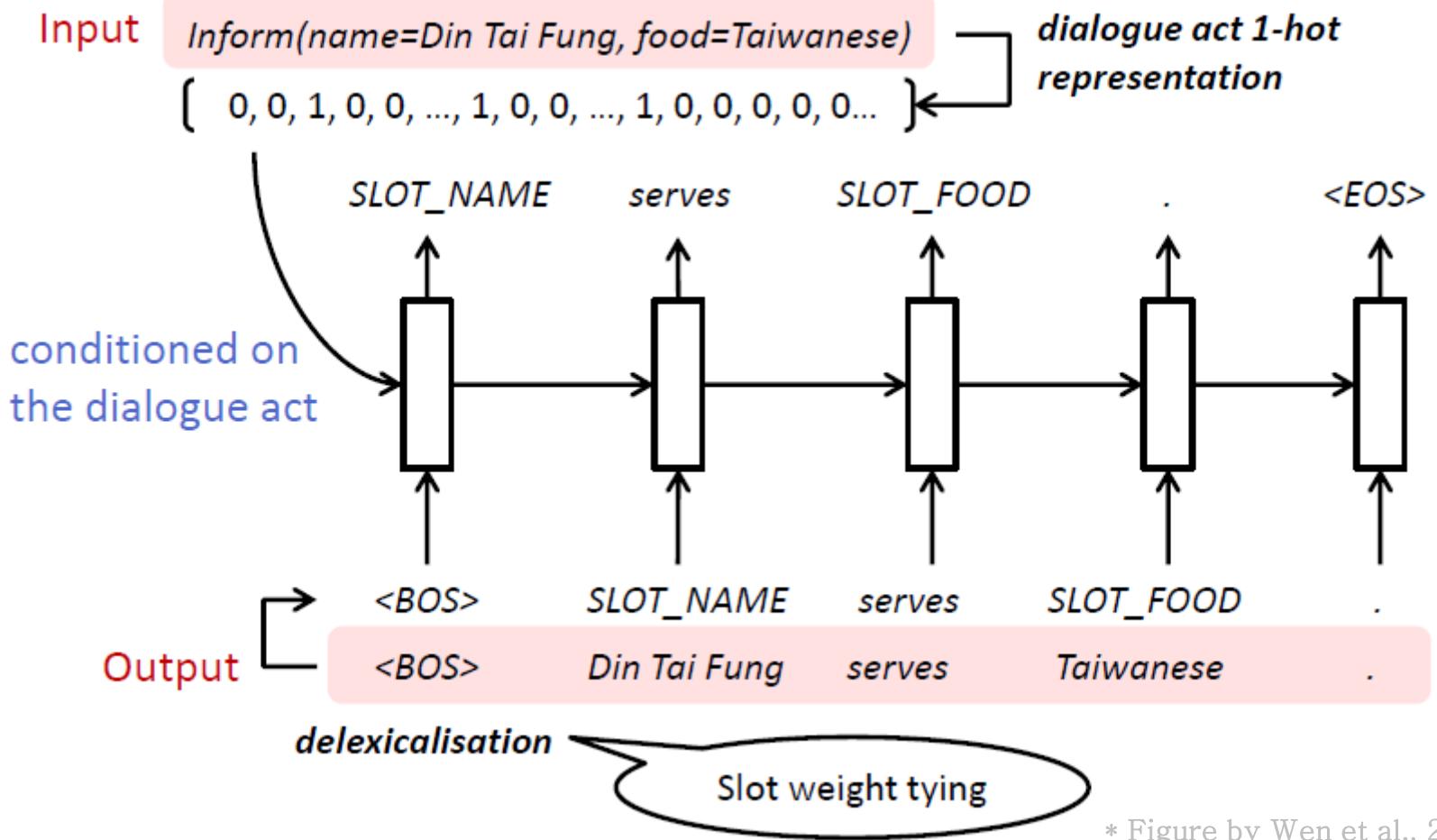
- Define a set of rules to map frames to NL

Semantic Frame	Natural Language
confirm()	"Please tell me more about the product your are looking for."
confirm(area=\$V)	"Do you want somewhere in the \$V?"
confirm(food=\$V)	"Do you want a \$V restaurant?"
confirm(food=\$V,area=\$W)	"Do you want a \$V restaurant in the \$W?"

**Pros:** simple, error-free, easy to control  
**Cons:** time-consuming, poor scalability



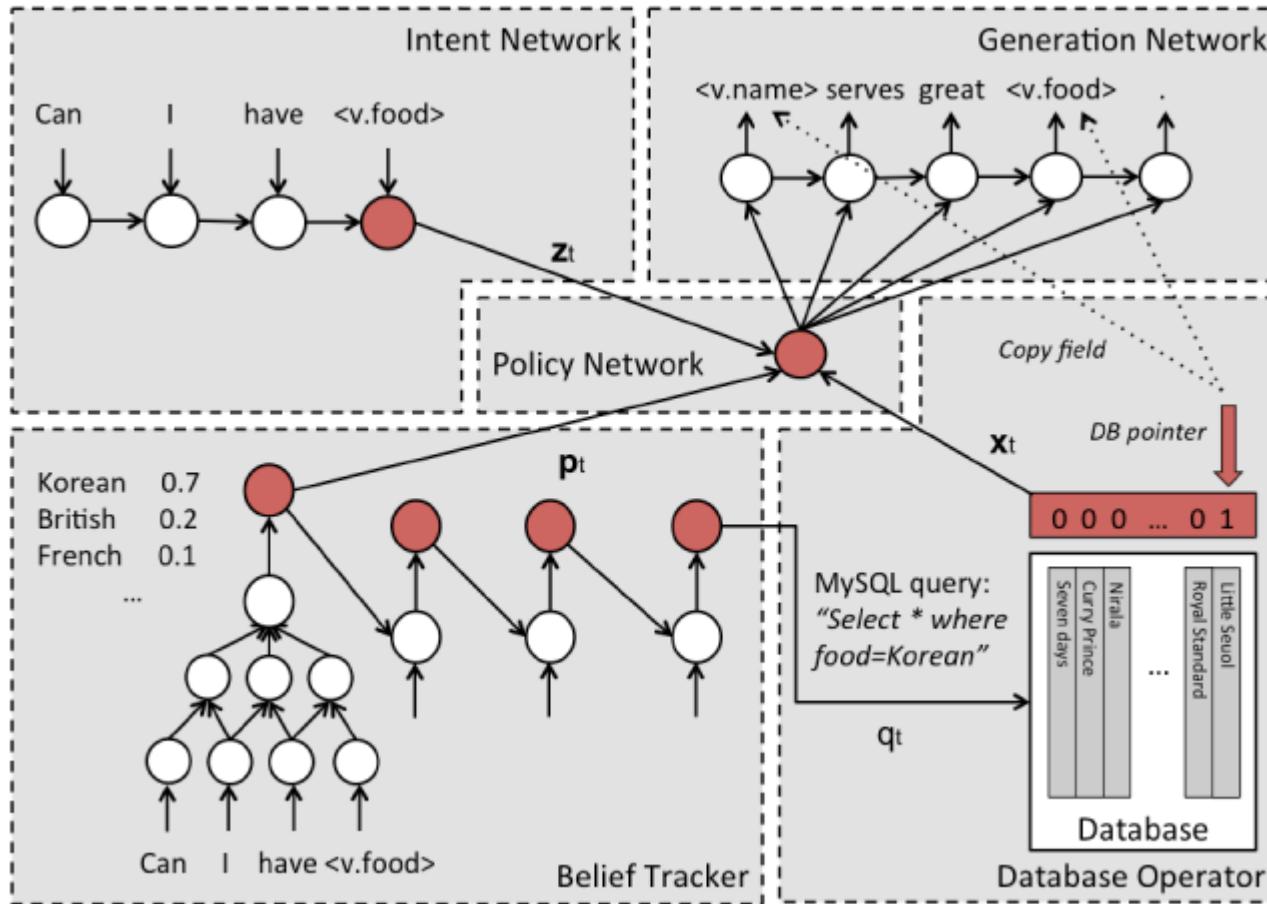
# RNN-based LM NLG



\* Figure by Wen et al., 2015

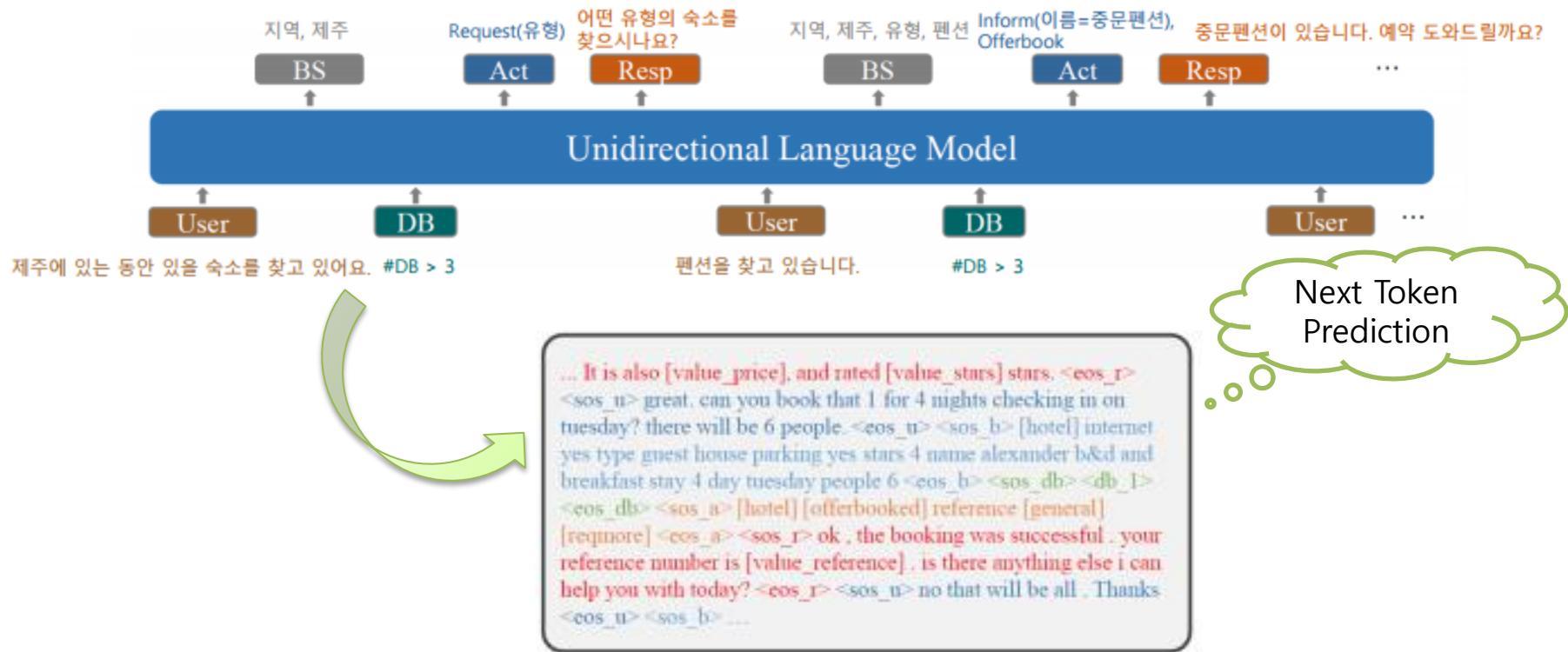


# End-to-End Dialogue System



# End-to-End Dialogue System

- GPT-2 기반 목적지향 대화 시스템 (현재 SOTA)



출처: Yang et al., UBAR: Towards Fully End-to-End Task-Oriented Dialog Systems with GPT-2, 2021  
ETRI, 자가대화형 강화학습 기반 대화처리 기술, 2021



# Quiz 1

---

- 기차표를 예약하는 대화시스템을 만들려고 할 때 Intent 와 Slot을 정의하고, 다음 대화의 각 발화에 의미표지 (Intent & Dialogue State)를 부착하시오.
  - Intent
    - Say\_Hello, Ask\_Dept, ..., Response\_Dept, ..., Confirm\_Dept, ..., Say\_Bye
  - Slot
    - Dept, Arrival, Date, Time, Num\_Pass, ...



# Quiz 1

---

## - Dialogue

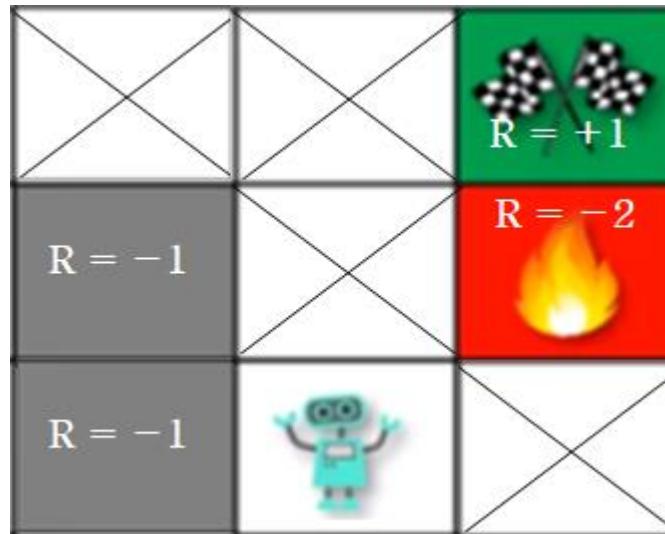
- System: 안녕하세요, 무엇을 도와드릴까요?
- User: 내일 대전가는 KTX 예약하고 싶어요.
- System: 어느 역에서 출발하실 건가요?
- User: 수서역이요.
- System: 수서역에서 출발하는 KTX는 없습니다.
- System: 수서역에서 출발하는 고속열차로는 SRT가 있습니다.
- System: 그것으로 예약해 드릴까요?
- User: 예
- System: 몇 시에 출발하실 예정인가요?
- User: 10시에 두 명이요.
- System: 두 명이요?
- User: 예
- System: 10시 20분에 대전가는 SRT로 두 명 예약해드릴까요?
- User: 조금 더 빠른 건 없나요?
- System: 9시 50분에 출발하는 것이 있습니다.
- User: 예 그걸로 예약해주세요.



# Quiz 2

- 상하좌우로 움직일 수 있는 주어진 로봇이 목적지(깃발)에 도착하고자 할 때 X 표시된 셀의 Q값을 구하시오.
  - 각 셀의 상하좌우 Q값은 0으로 가정

$$\hat{Q}(s, a) \leftarrow r + \gamma \max_{a'} \hat{Q}(s', a'), \text{ where } \gamma = 0.9$$



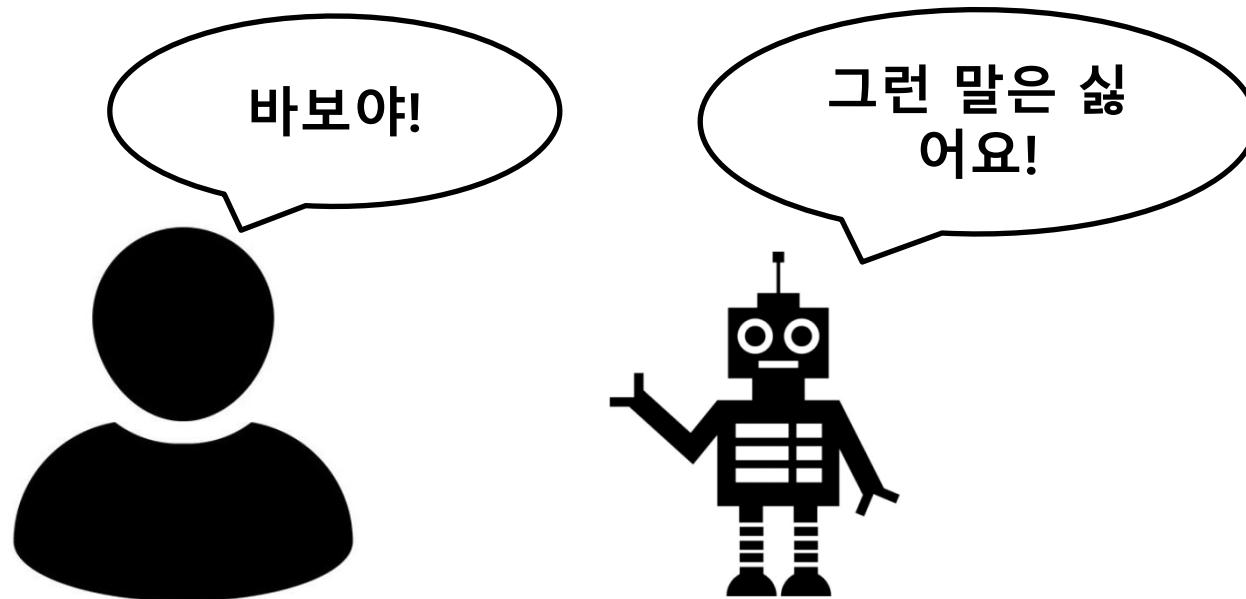
---

# Chitchat Bot

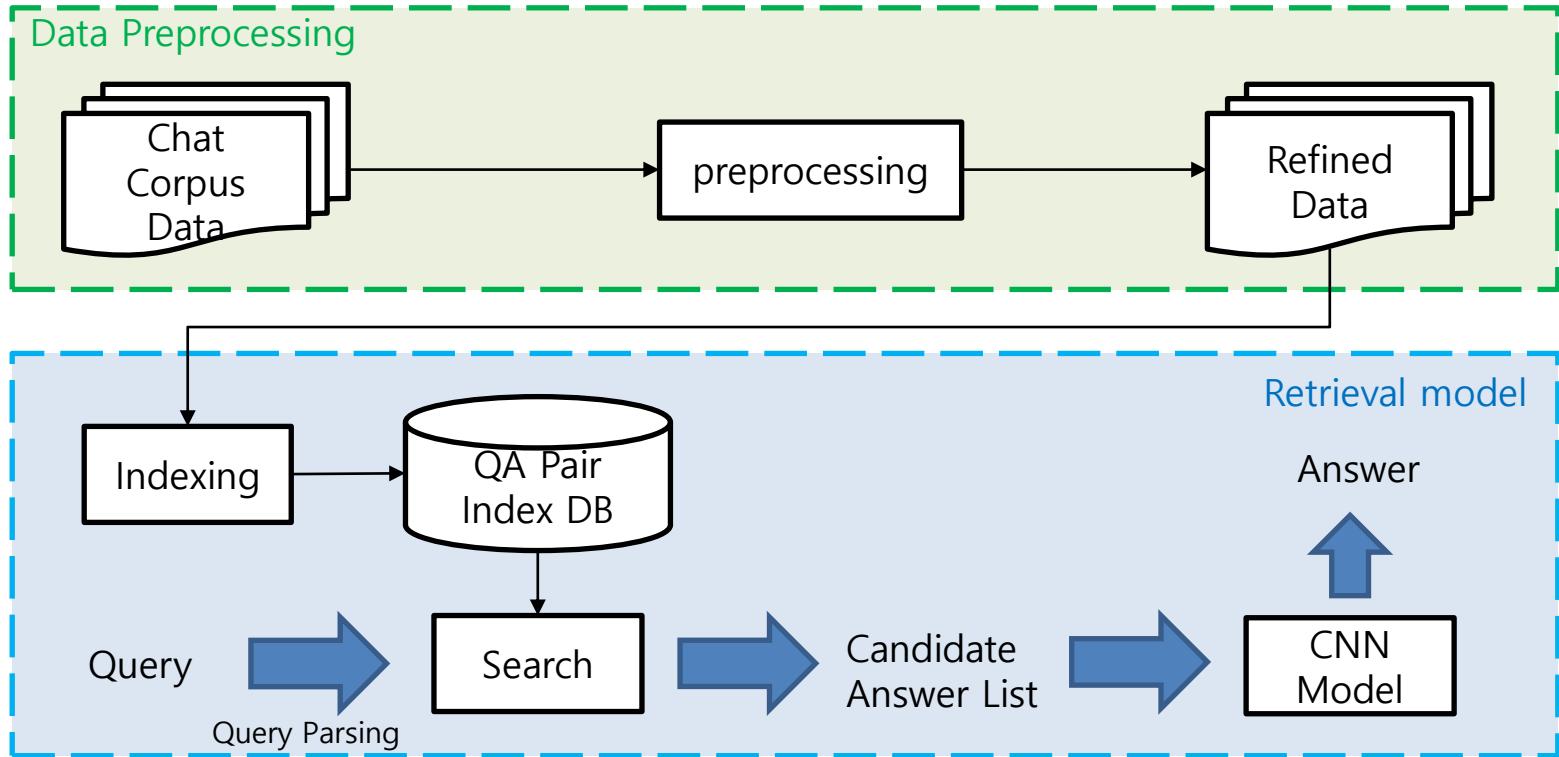
---

# 검색 기반 챗봇

- 검색 기반 챗봇이란?
  - 사용자 발화를 질의로 하여 미리 만들어둔 발화를 검색
  - 검색된 결과 중 가장 적합성이 높은 발화에 대한 기구축 응답을 제시



# 시스템 구조도



# 색인 및 검색

- 색인 템 생성
  - 문장 입력

나 어제 짜장면 먹고 싶어서 저녁에 중국집 갔었다!! \*.\*


  - 문장 정규화 과정 수행

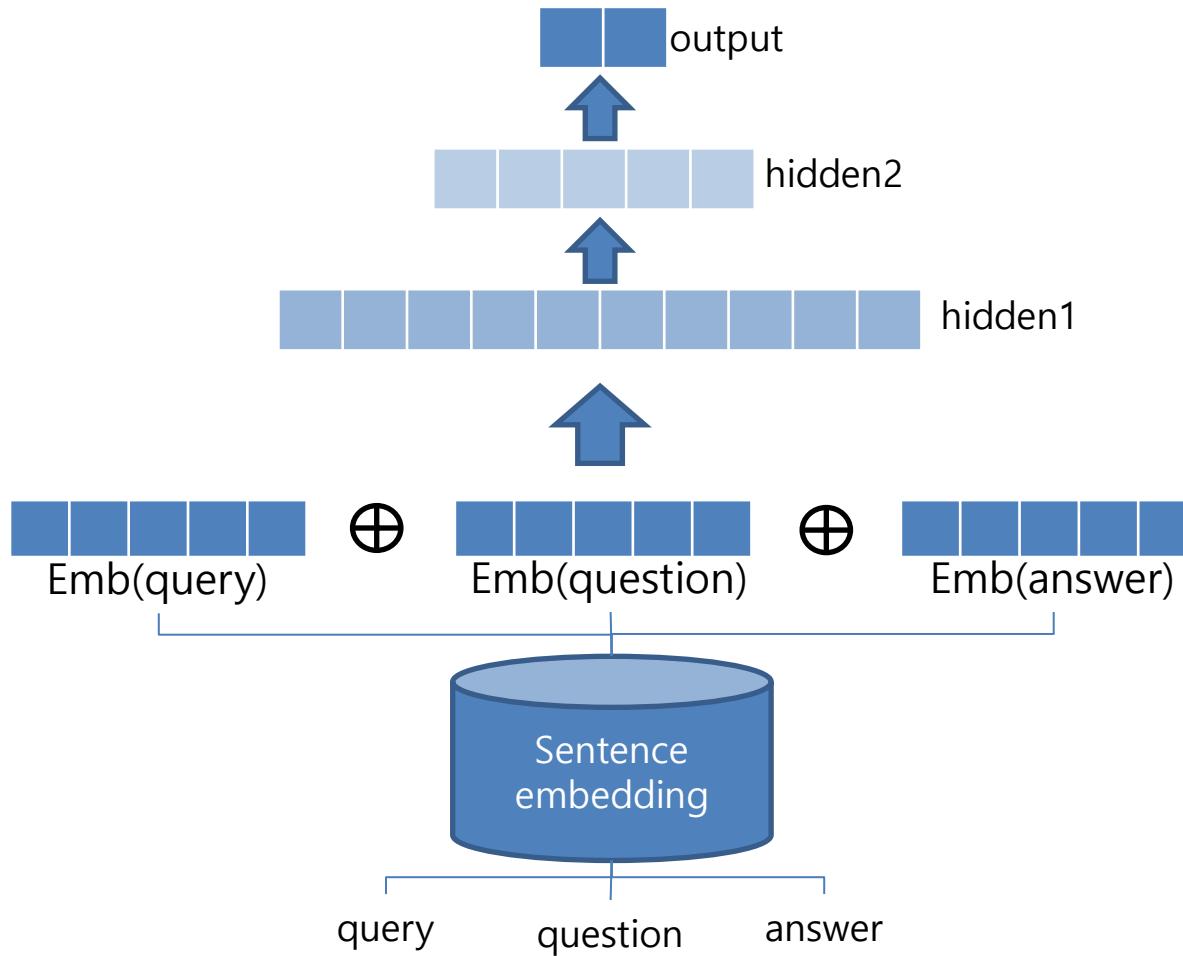
['나/NP', '어제/NNG', '짜장면/NNG', '먹/VV', '**M#34**', '저녁/NNG', '중국집/NNG', '가/VV', '**T#1**']



시제
- 생성된 색인 템을 이용해 데이터 색인
- 검색 모델을 이용한 순위화
  - BM25 모델이 가장 보편적임
  - 최근에는 문장을 그대로 활용하는 Dense Retrieval Model(예: DPR)이 적극 도입되어 활용되고 있음

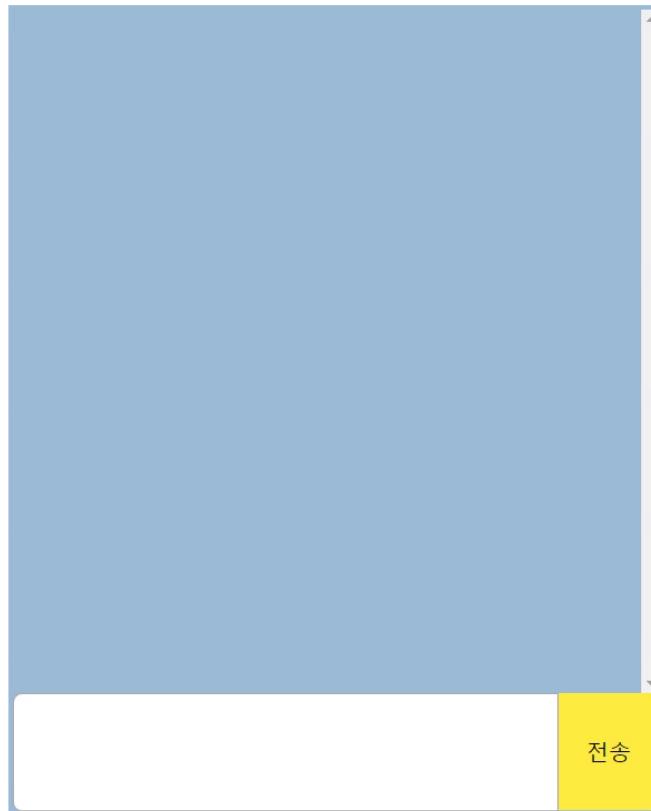


# 검색 결과 필터링



# 시연 영상

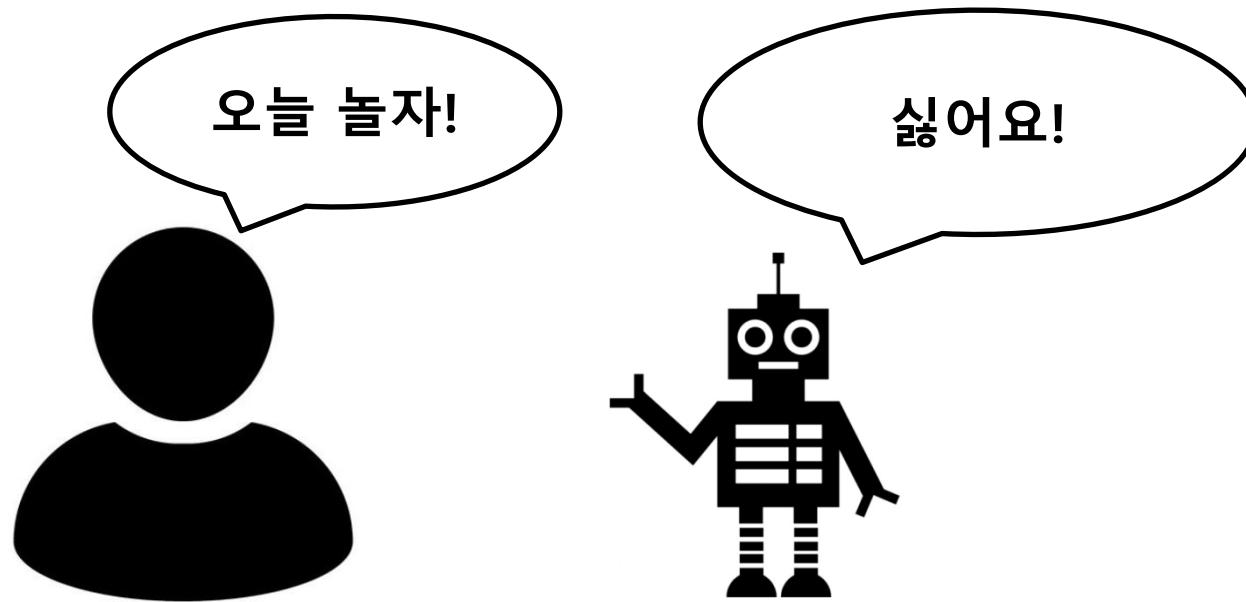
KAPS-CHATBOT



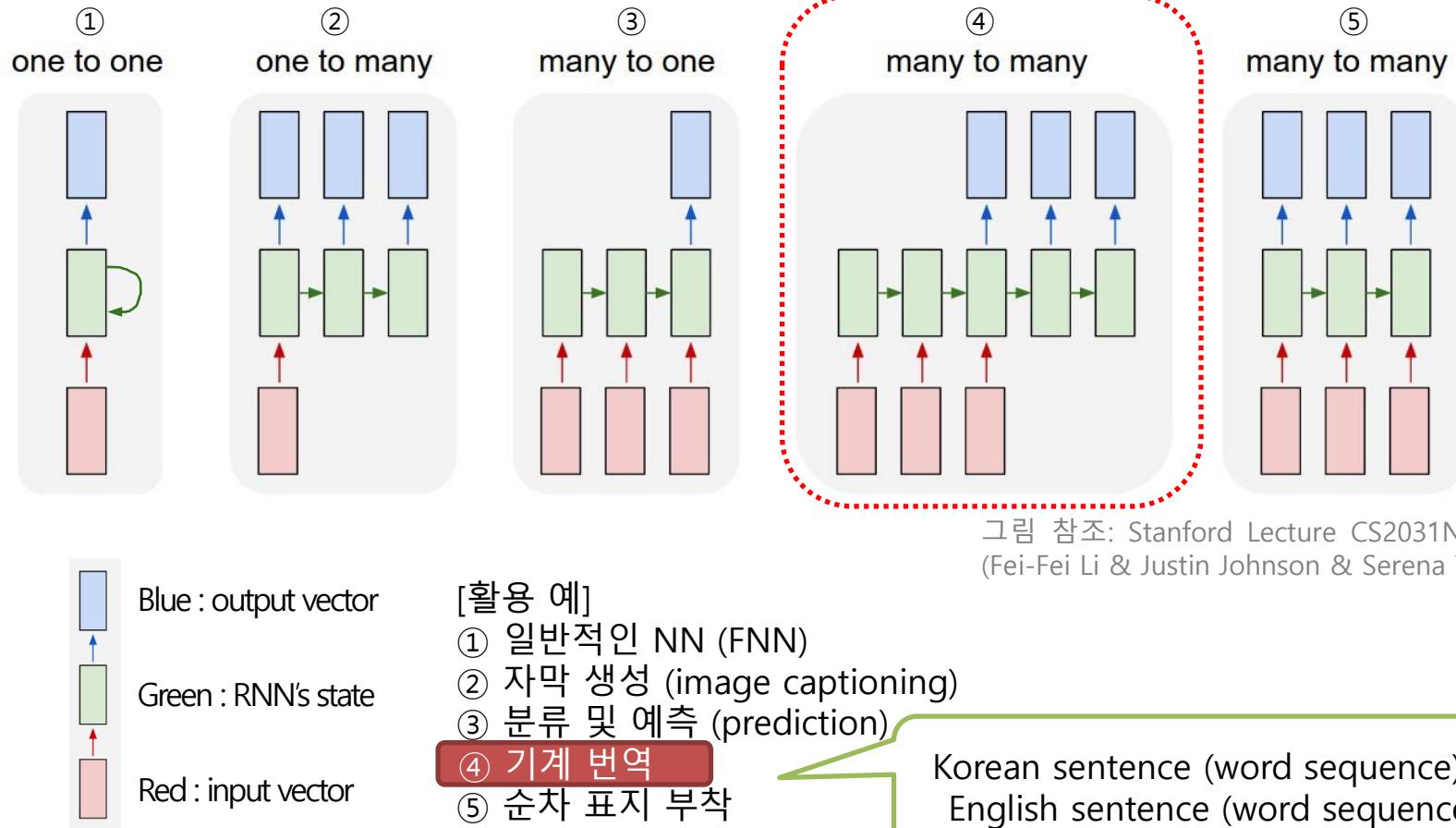
# 생성 기반 챗봇

---

- 생성 기반 챗봇이란?
  - 사용자 발화에 대한 적합한 응답을 생성해 자연스러운 대화를 하는 시스템



# Many-to-Many Model (Sequence-to-Sequence)



# Transformer

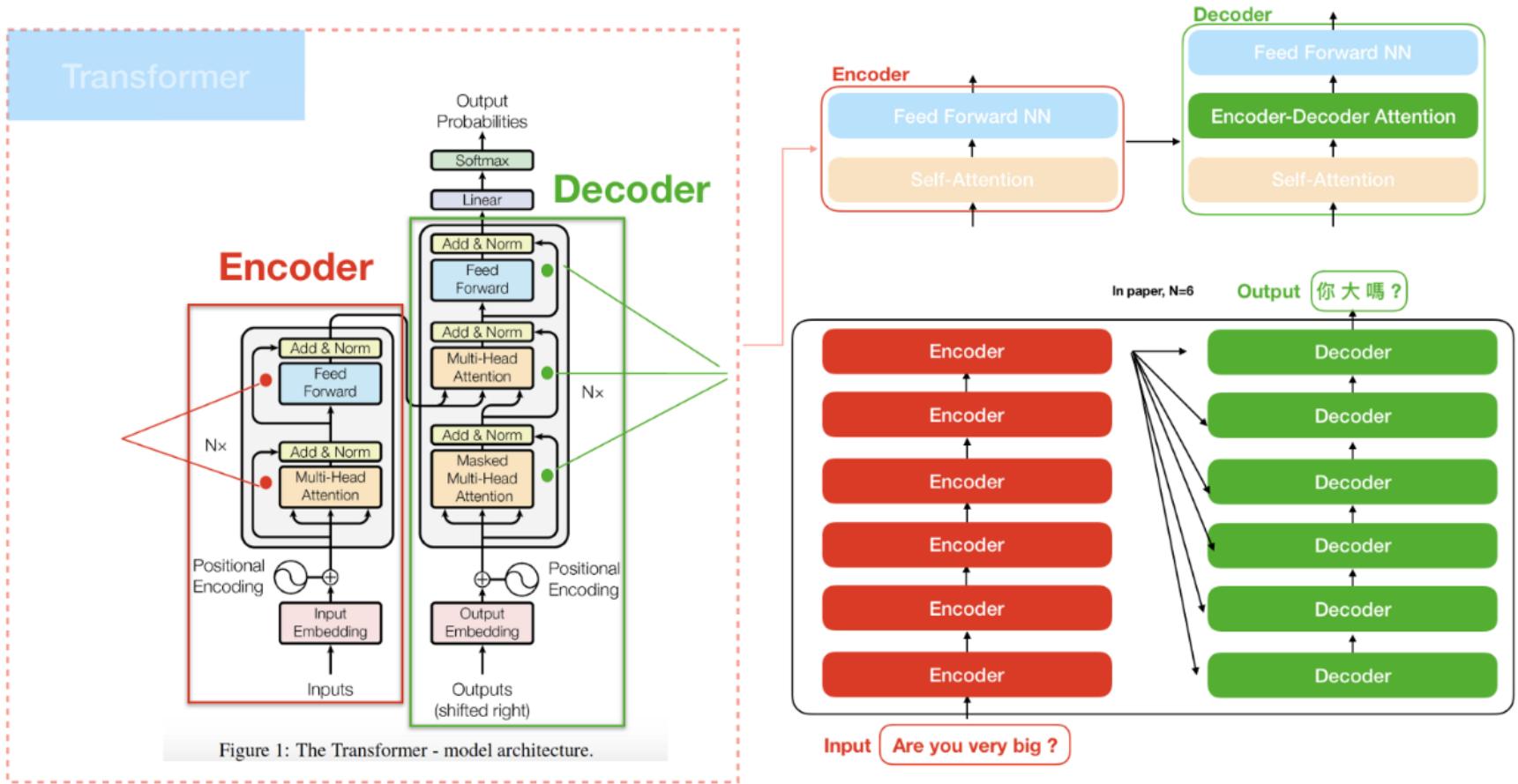
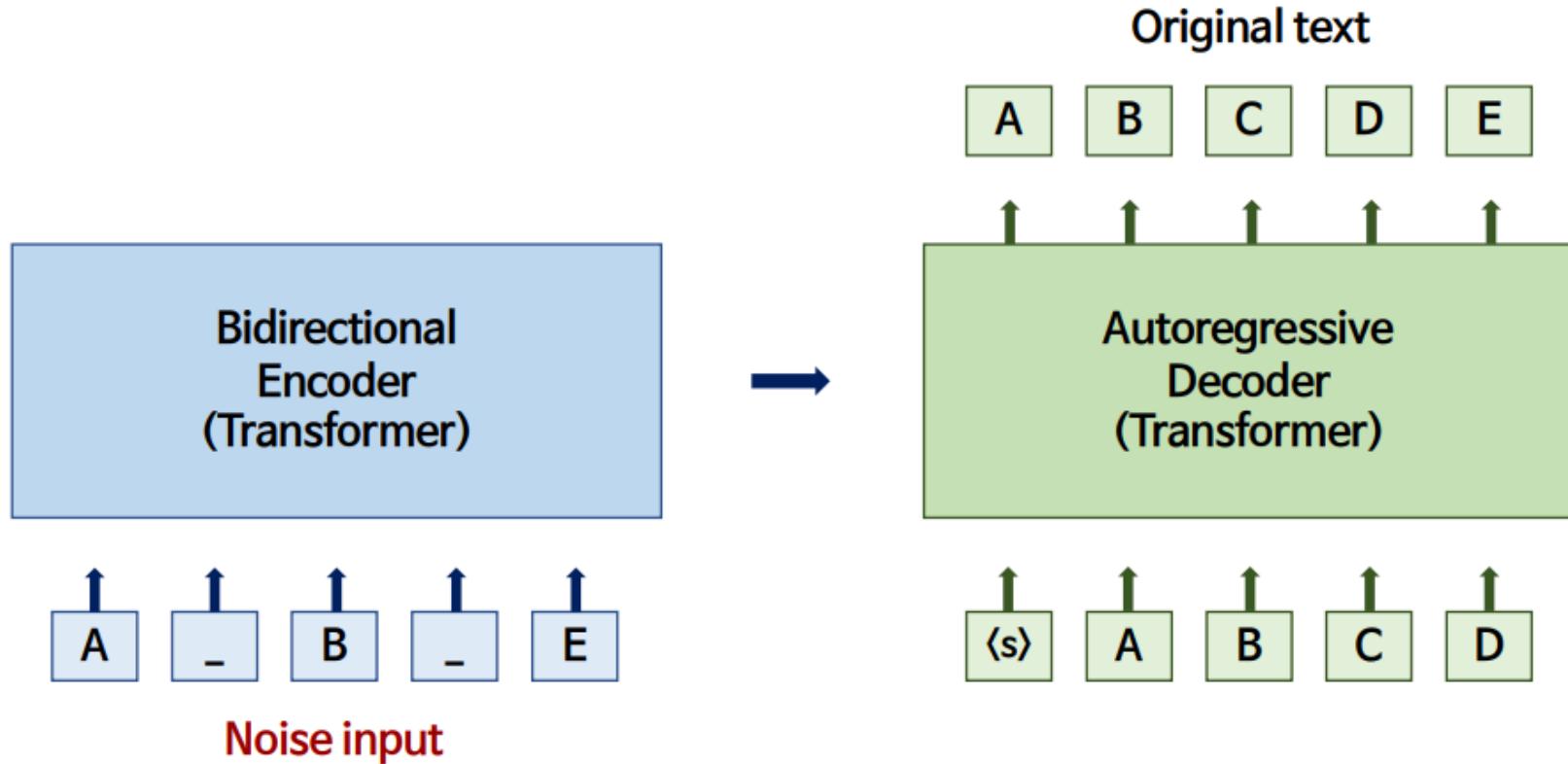


그림 출처: Ta-Chun (Bgg/Gene) Su's blog



# BART: Bidirectional and Auto-Regressive Transformer

- Denoising autoencoder for language modeling



출처: 고려대 DSBA 연구실 이유경 발표자료



# Noises for Language Modeling

Original text



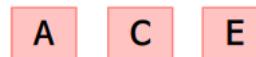
Sentence 1                                    Sentence 2

## 1. Token Masking



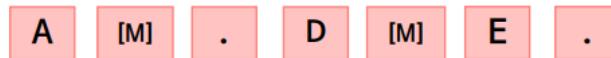
- ✓ 임의의 Token을 [MASK]로 교체함.
- ✓ BERT의 Masking을 따르기 때문에 input sequence는 유지해야함
- ✓ [MASK] token이 무엇인지 예측해야 함

## 2. Token Deletion



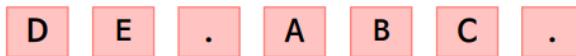
- ✓ 임의의 Token을 삭제함
- ✓ 삭제한 token의 위치를 찾아야 함

## 3. Text infilling



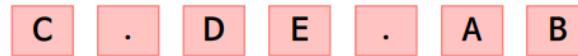
- ✓ Poisson dist. ( $\Lambda = 3$ )에서 span length를 뽑아 하나의 [MASK] token으로 대체함
- ✓ Span length가 0인 경우, 해당 위치에 [MASK] token을 추가함
- ✓ [MASK]로 대체된 token에 몇개의 token이 존재할지 예측해야 함

## 4. Sentence Permutation



- ✓ 문장의 순서를 랜덤으로 섞음

## 5. Document Rotation



- ✓ 하나의 token을 uniformly하게 뽑은 후, 그 토큰을 시작점으로 회전함
- ✓ 모델이 문서의 start point를 찾도록 학습시킴

출처: 고려대 DSBA 연구실 이유경 발표자료



# 입출력 단위 설계 (1/3)

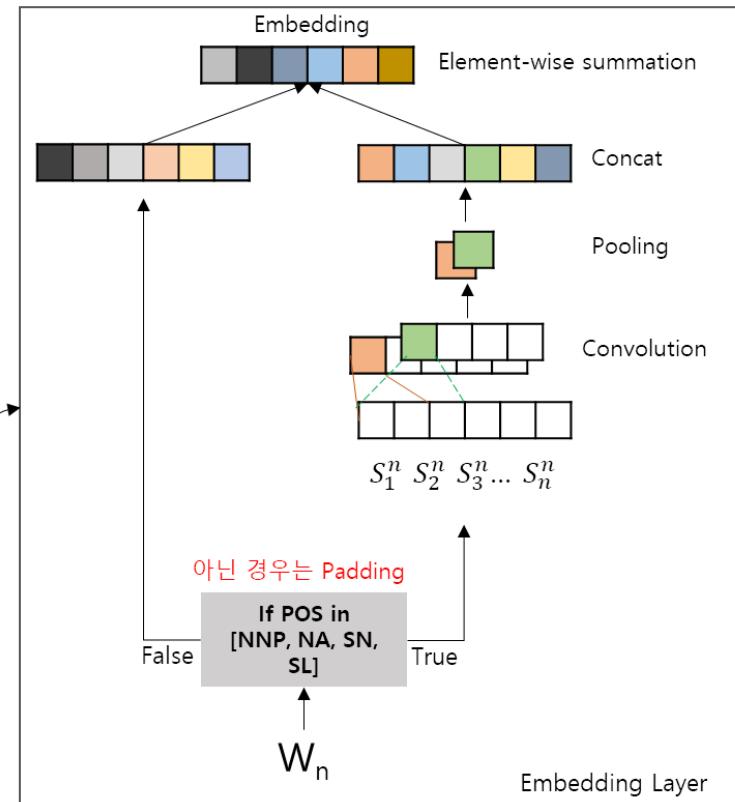
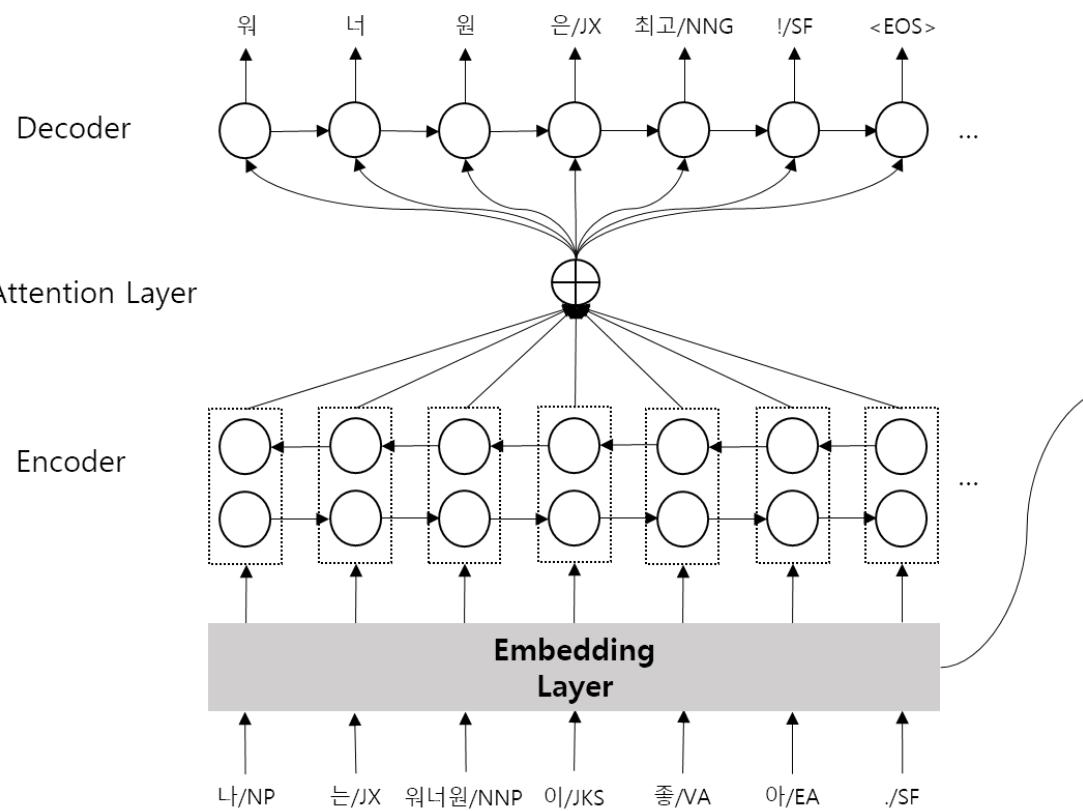
---

- 형태소 단위
  - 고유명사, 미등록어로 인한 사전 크기 증가
  - 디코딩 시에 어절 복원을 위한 후처리 필요  
예) 아름답 + 어 → 아름다워
- 음절 단위
  - 고유명사, 미등록어로 문제가 없고, 사전 크기 대폭 감소
  - 디코딩 시에 어법에 맞지 않는 형태소가 생성될 수 있음  
예) 아름덥어



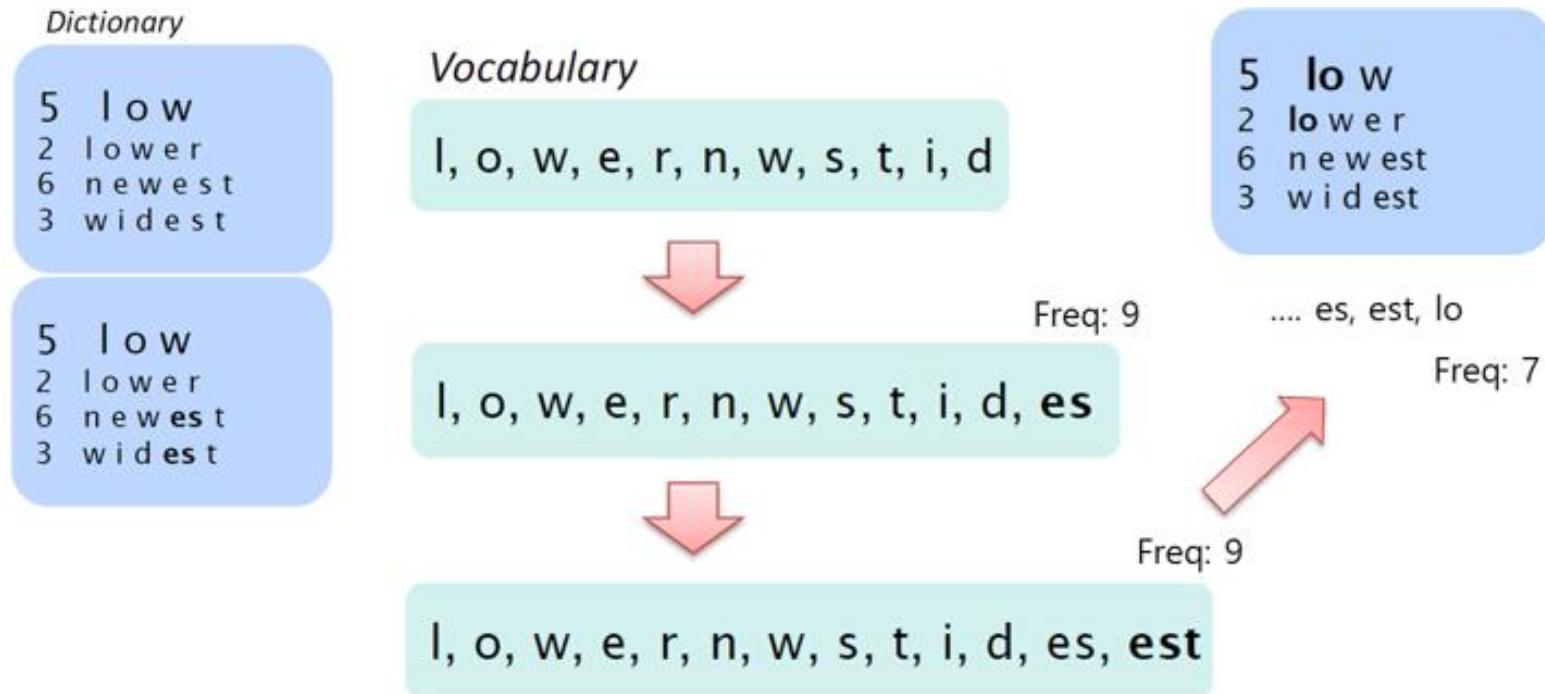
# 입출력 단위 설계 (2/3)

- 형태소+음절 단위



# 입출력 단위 설계 (3/3): Remind

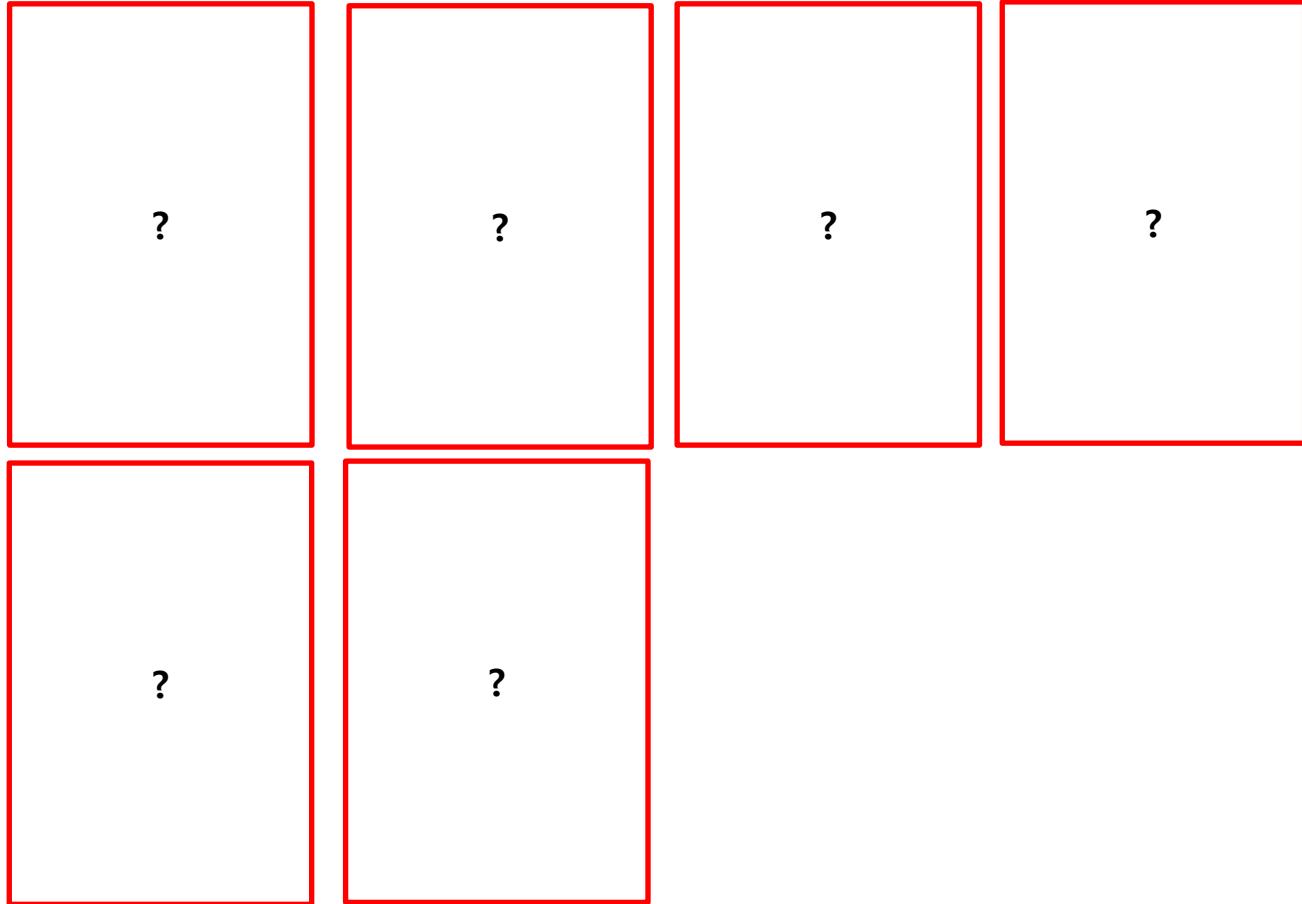
- 워드피스(WordPiece) 단위: Byte Pair Encoding



# 확인 문제: Byte Pair Encoding

다음과 같은 문자열이 주어졌을 때 7개의 subunit이 될 때까지 BPE를 수행하시오.

5 low  
2 lower  
6 newest  
3 widest

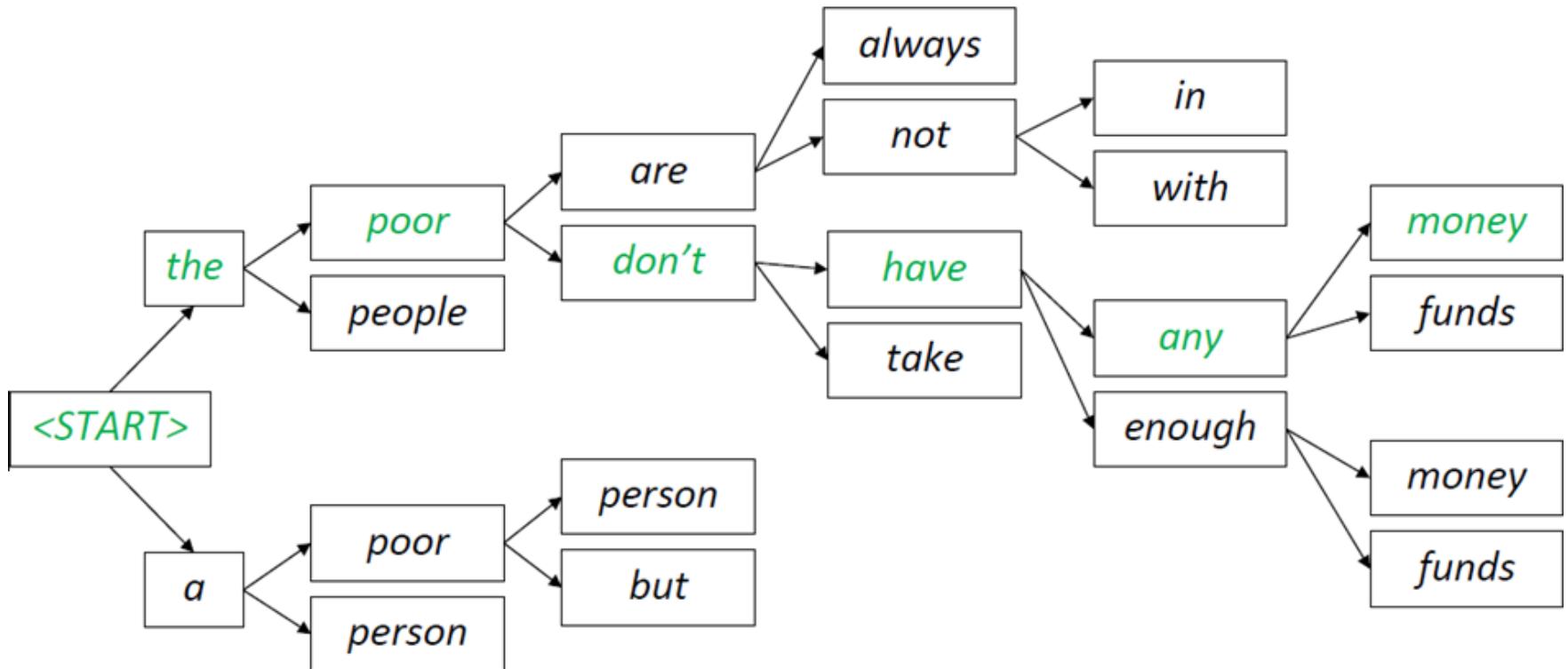


# Beam Search Decoding

$$P(y|x) = P(y_1|x) P(y_2|y_1, x) P(y_3|y_1, y_2, x) \dots, P(y_T|y_1, \dots, y_{T-1}, x)$$



Beam size = 2



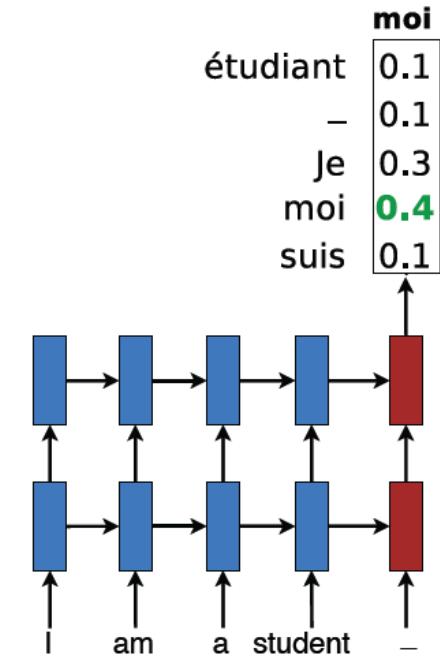
# Word Generation Problem

- Word generation problem
  - If vocabs are modest, e.g., 50K  
1단어 생성 =  $50K * 4\text{byte}(\text{float}) = 2M$

The ecotax portico in Pont-de-Buis  
Le portique écotaxe de Pont-de-Buis



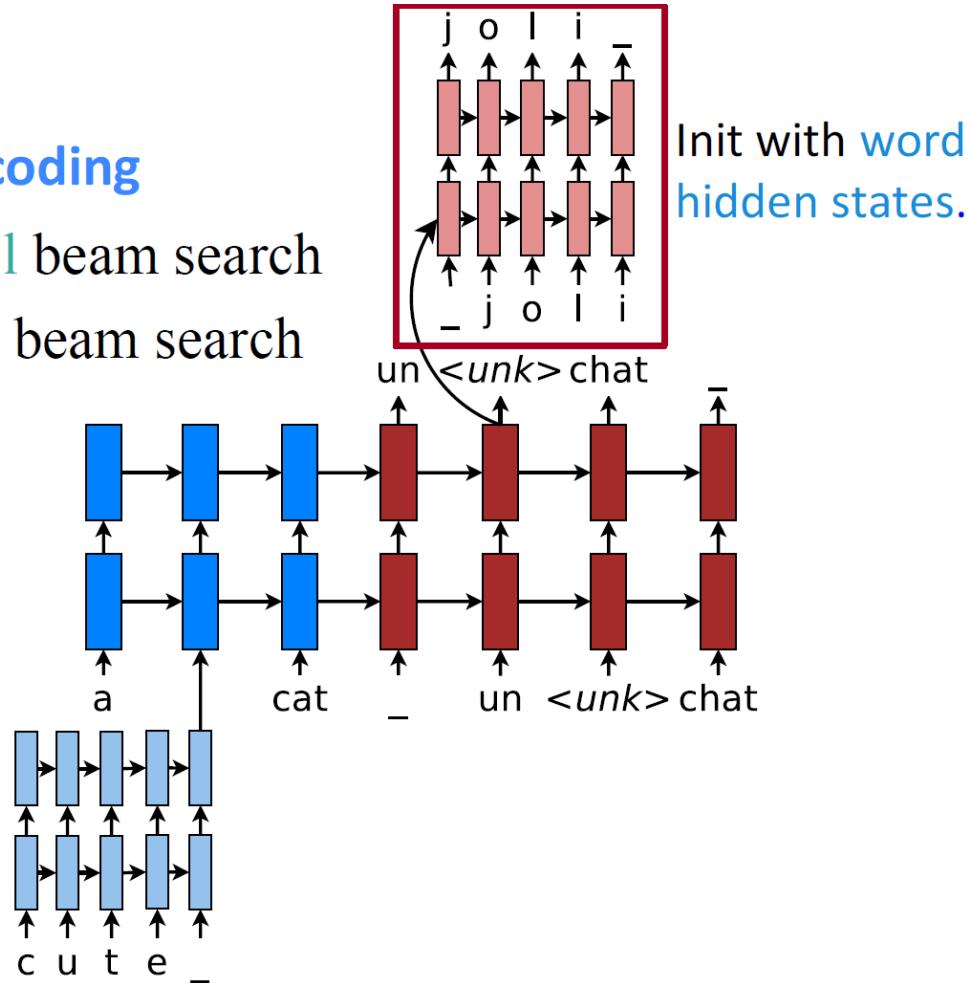
The <unk> portico in <unk>  
Le <unk> <unk> de <unk>



# 2-Stage Decoding

## 2-stage Decoding

- Word-level beam search
- Char-level beam search for  $\langle unk \rangle$ .



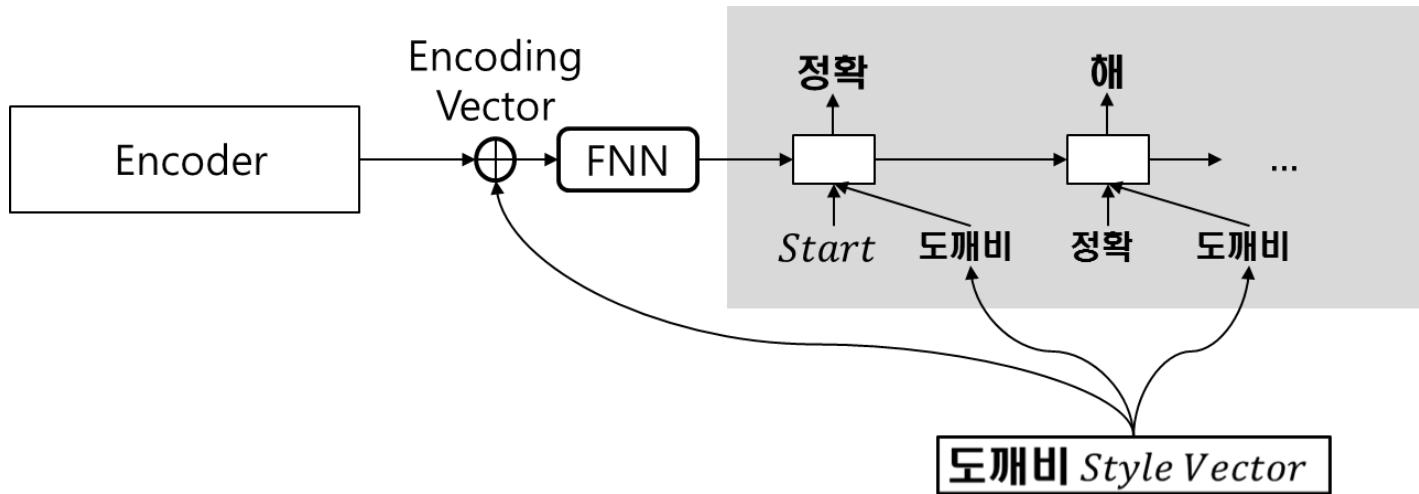
# 시연 영상

```
[I tensorflow/core/common_runtime/gpu/gpu_device.cc:972] D
I tensorflow/core/common_runtime/gpu/gpu_device.cc:982] 0
I tensorflow/core/common_runtime/gpu/gpu_device.cc:1041]
Reading model parameters from ./model/model.ckpt-1001187
Success Load!
# NLU Load Success!
I tensorflow/core/common_runtime/gpu/gpu_device.cc:975] C
Input :
```



# Chatbot with Persona

- 페르소나 반영 챗봇



- 드라마 '도깨비'의 '도깨비'와 '지은탁' 대사 600개씩 수집

발화	도깨비 어투 답변	지은탁 어투 답변
가마솥에 해먹는 밥은 맛이 일품이야.	그러게.	아니요?
1회용품 사용을 줄여야만해.	정확해.	우와.
멋있다고	하루 이틀일이 아니라.	원빈인데.



# Knowledge-Grounded Chatbot

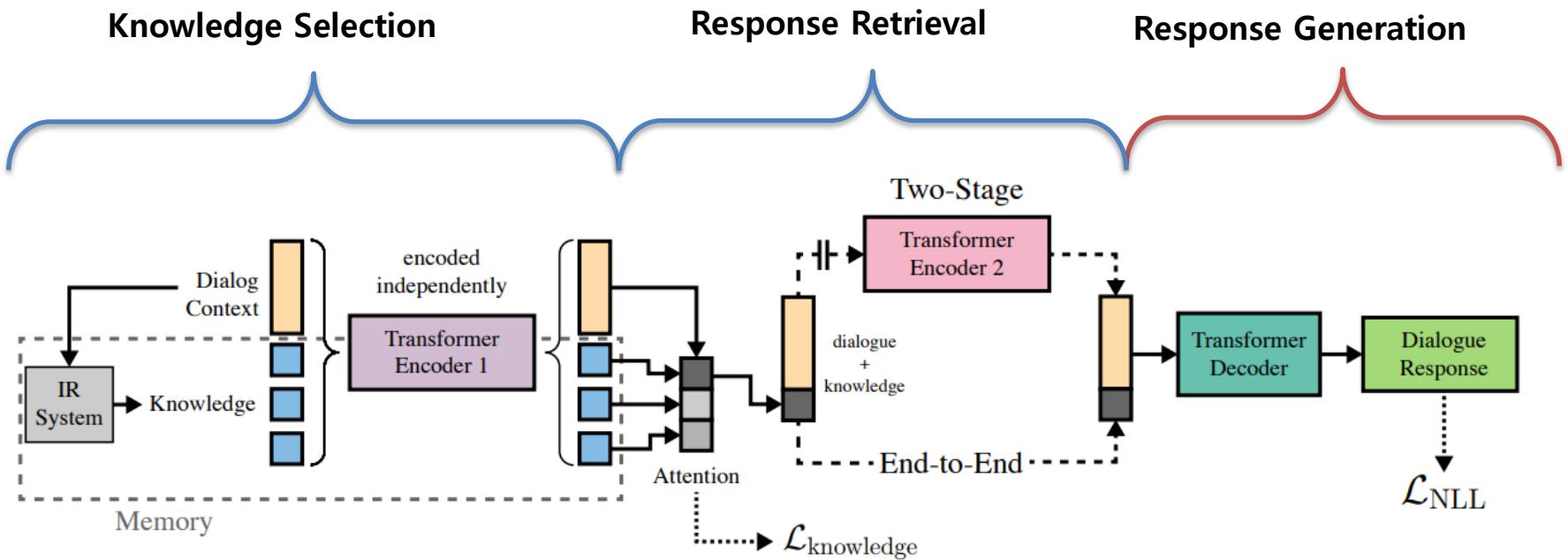
- Wizard of Wikipedia Dataset: 크라우드 소싱된 위키피디아 대상의 1,365개

<b>Topic:</b>	Lifeguard	
Apprentice:	So I am a lifeguard. Know anything about saving lives in water?	
Wizard:	I'm impressed! It's a big responsibility to supervise other people's safety in the water! Tell me more.	
Apprentice:	Well, I help make sure people do not drown or get injured while in or near the water!	
<b>Knowledge:</b>	A lifeguard is a rescuer who supervises the safety and rescue of swimmers, surfers, ... Lifeguards are strong swimmers and trained in CPR/AED first aid, certified in water ... ...	
	In some areas, the lifeguard service also carries out mountain rescues, or may function as the primary EMS provider.	
Wizard:	I've heard that in some places, lifeguards also help with other sorts of emergencies, like mountain rescues!	
	Is that part	
Apprentice:	I have! I feel	<b>Wizard of Wikipedia Task</b>
Wizard:	Oh, that's always adn	Train
		Valid
		Test Seen
		Test Unseen
	Number of Utterances	166,787
	Number of Dialogues	18,430
	Number of Topics	1,247
	Average Turns per Dialogue	9.0
	Knowledge Database	5.4M articles
		93M sentences

훈련데이터에  
속해있지 않  
은 토픽



# Wizard of Wikipedia



출처: Wizard of Wikipedia (Dinan et al., 2018)



# 실험 및 평가 (1/3)

- Knowledge Selection Task
  - The models must select the gold knowledge sentences chosen by humans given the dialogue context.

Method	Seen Test		Unseen Test	
	R@1	F1	R@1	F1
Random	2.7	13.5	2.3	13.1
IR baseline	5.8	21.8	7.6	23.5
BoW MemNet	23.0	36.3	8.9	22.9
Transformer	22.5	33.2	12.2	19.8
Transformer (+Reddit pretraining)	24.5	<b>36.4</b>	<b>23.7</b>	<b>35.8</b>
Transformer (+Reddit pretraining, +SQuAD training)	<b>25.5</b>	36.2	22.9	34.2

\* F1: 응답과 정답 사이의 유니그램 오버랩에 관한 F1-score



# 실험 및 평가 (2/3)

- Full Task: Response Retrieval
  - Models must select relevant knowledge and retrieve a response from the training set as a dialogue response.

Method	Predicted Knowledge				Gold Knowledge	
	Test Seen		Test Unseen		Seen	Unseen
	R@1	F1	R@1	F1	R@1	R@1
Random	1.0	7.4	1.0	7.3	1.0	1.0
IR baseline	17.8	12.7	14.2	11.6	73.5	67.5
BoW MemNet (no knowledge)	56.1	14.2	28.8	11.6	56.1	28.8
BoW MemNet	71.3	<b>15.6</b>	33.1	12.3	84.5	66.7
Transformer (no knowledge, w/o Reddit)	60.8	13.3	25.5	9.7	60.8	25.5
Transformer (no knowledge, w/ Reddit)	79.0	15.0	54.0	11.6	79.0	54.0
Transformer MemNet (w/ Reddit)	86.8	15.4	<b>69.8</b>	<b>12.4</b>	91.6	82.3
Transformer MemNet (w/ Reddit+SQuAD)	<b>87.4</b>	15.4	<b>69.8</b>	<b>12.4</b>	<b>92.3</b>	<b>83.1</b>

\* F1: 응답과 정답 사이의 유니그램 오버랩에 관한 F1-score



# 실험 및 평가 (3/3)

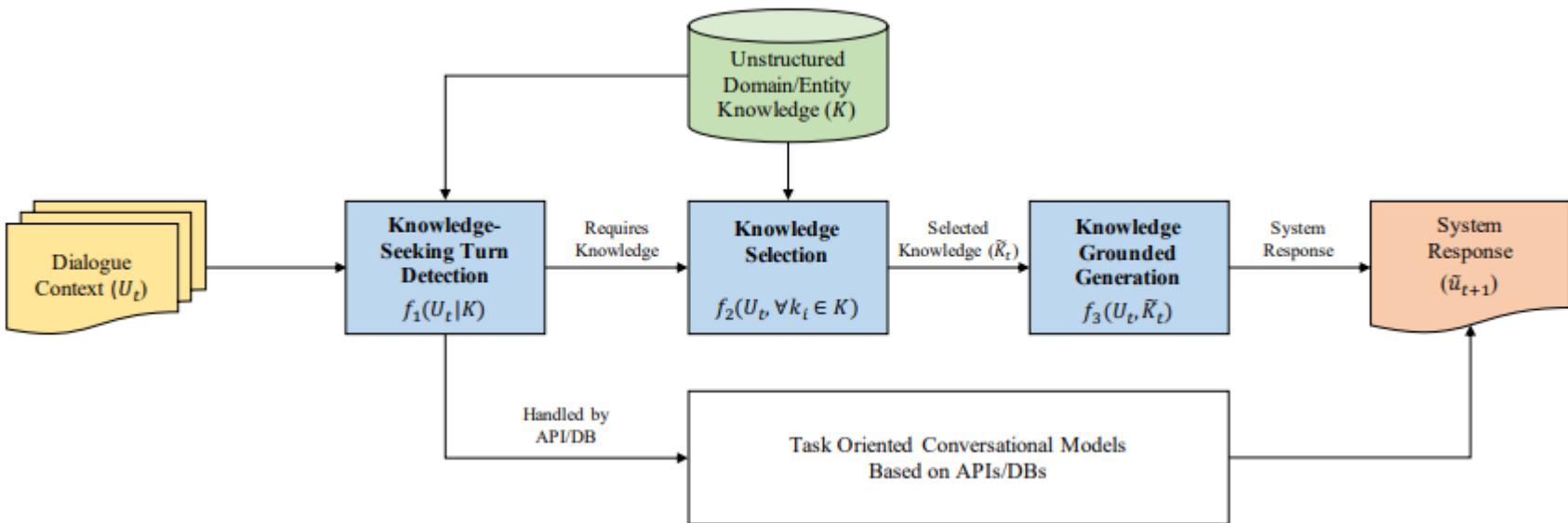
- Full Task: Response Generation
  - Final evaluation of the proposed model

Method	Predicted Knowledge				Gold Knowledge			
	Test Seen		Test Unseen		Test Seen		Test Unseen	
	PPL	F1	PPL	F1	PPL	F1	PPL	F1
Repeat last utterance	-	13.8	-	13.7	-	13.8	-	13.7
Transformer (no knowledge)	-	-	-	-	41.8	17.8	87.0	14.0
E2E Transformer MemNet (no auxiliary loss)	66.5	15.9	103.6	14.3	24.2	33.6	35.5	29.5
E2E Transformer MemNet (w/ auxiliary loss)	63.5	16.9	97.3	14.4	<b>23.1</b>	<b>35.5</b>	<b>32.8</b>	<b>32.2</b>
Two-Stage Transformer MemNet	54.8	18.6	88.5	<b>17.4</b>	30.0	30.7	42.7	28.6
Two-Stage Transformer MemNet (w/ K.D.)	<b>46.5</b>	<b>18.9</b>	<b>84.8</b>	17.3	28.6	30.6	43.7	28.0

\* PPL: perplexity (PPL) of the gold response and unigram F1



# SOTA in DSTC9



출처: Kim et al., Beyond Domain APIs: Task-oriented Conversational Modeling with Unstructured Knowledge Access Track in DSTC9, 2021



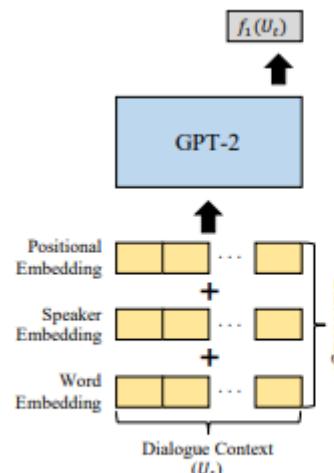
# SOTA in DSTC9

Based on Large-Scaled Pre-trained Language Model

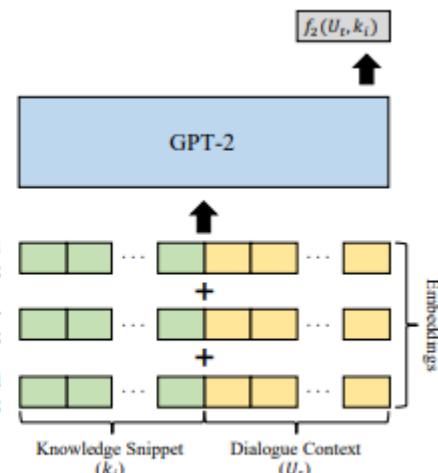
$$f_1(U_t|K) = \begin{cases} 1 & \text{if } \exists k \in K \text{ satisfies } u_t, \\ 0 & \text{otherwise,} \end{cases}$$

$$f_2(U_t, k_i) = \begin{cases} 1 & \text{if } k_i \in K \text{ is relevant to } U_t, \\ 0 & \text{otherwise.} \end{cases}$$

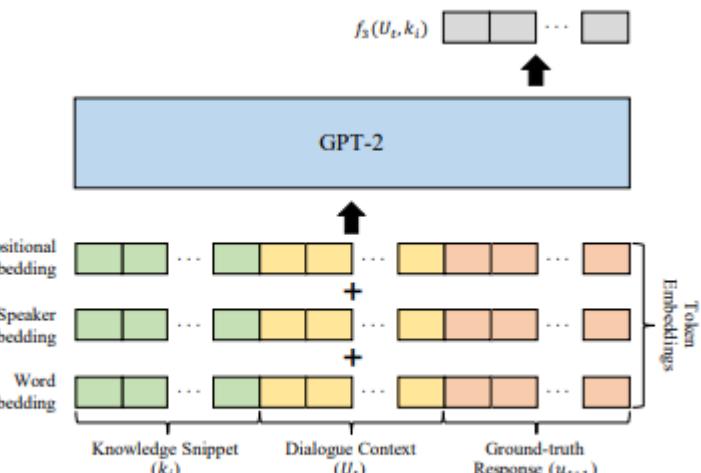
$$f_3(U_t, \tilde{K}_t) = \tilde{u}_{t+1}$$



(a) Detection baseline



(b) Selection baseline



(c) Generation baseline

출처: Kim et al., Beyond Domain APIs: Task-oriented Conversational Modeling with Unstructured Knowledge Access Track in DSTC9, 2021



# SOTA in DialDoc Shared Task

## Dialogue

User: Will you keep me informed about when my insurance is .....

Agent: Yes. The New York state requires every motorist to main ...

User: Is it needed that my address is always up to date with .....

Agent: Yes, It is required. If not we won't be able to keep you .....

User: Can you tell me more about the traffic points and its cost? ..

⋮

[1]

[1], [2]

[3]

[1], [2], [3]

[3], [4], [5]

⋮

Agent:

Response generation – subtask2

?

Knowledge identification– subtask1

## Document

[1] Clothing allowance

[2] you may be able to get money each year to help buy new clothes. [3] This is a disability compensation benefit known as an annual clothing allowance. [4] Find out if you can get this benefit. Can I get disability ...

⋮

input

Knowledge  
identification  
model

input

output

[2] you may be able to get  
money each year to help....



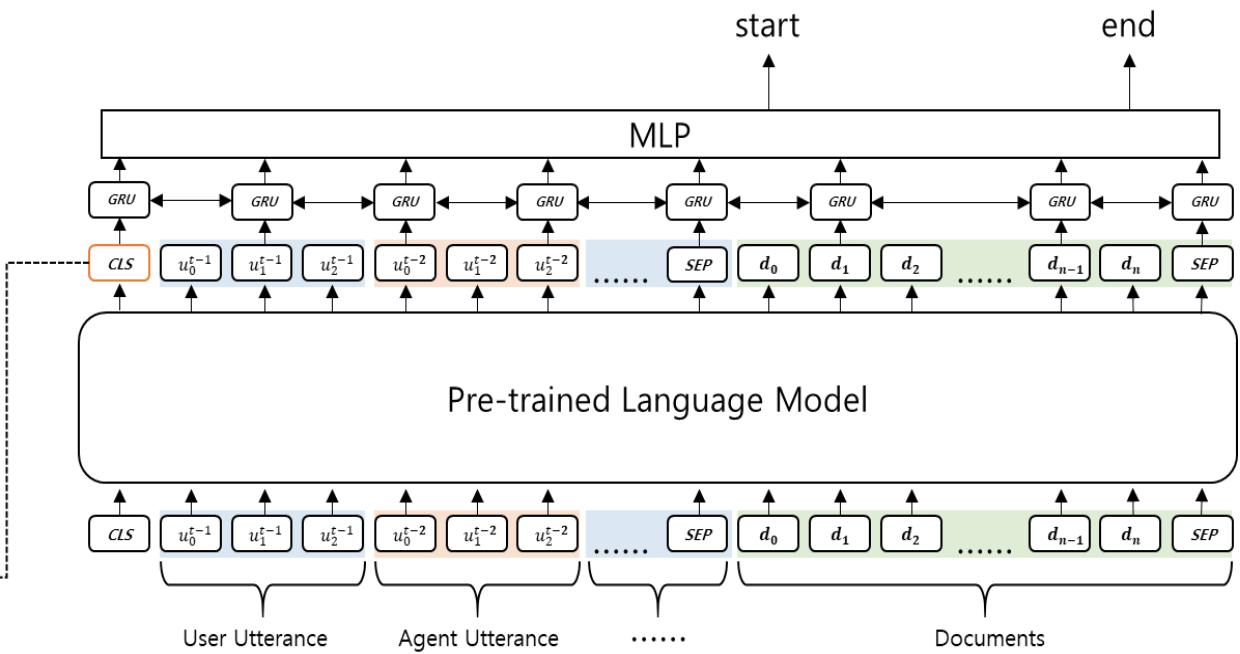
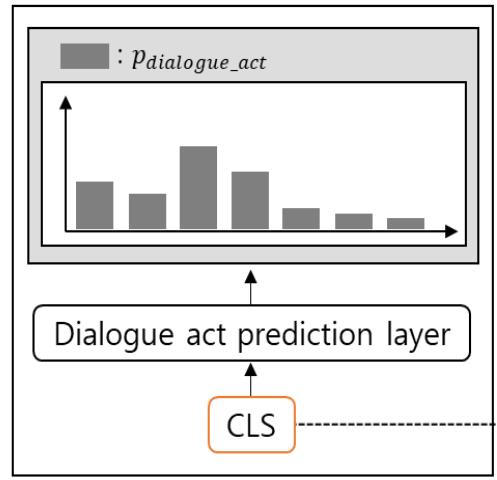
# 문서 구조

Title	Section ID	Span ID	Text
For Your Surviving Divorced Spouse	8	31	For Your Surviving Divorced Spouse
	9	32	If you have a surviving divorced spouse
		33	they could get the same benefits as your widow or widower provided that your marriage lasted 10 years or more.
	10	34	Benefits paid to a surviving divorced spouse won't affect the benefit amounts your other survivors will receive based on your earnings record.
	11	35	If your former spouse is caring for your child who is under age 16 or disabled and gets benefits on your record,
		36	they will not have to meet the length - of - marriage rule.
		37	The child must be your natural or legally adopted child.



# KU-DialMRC

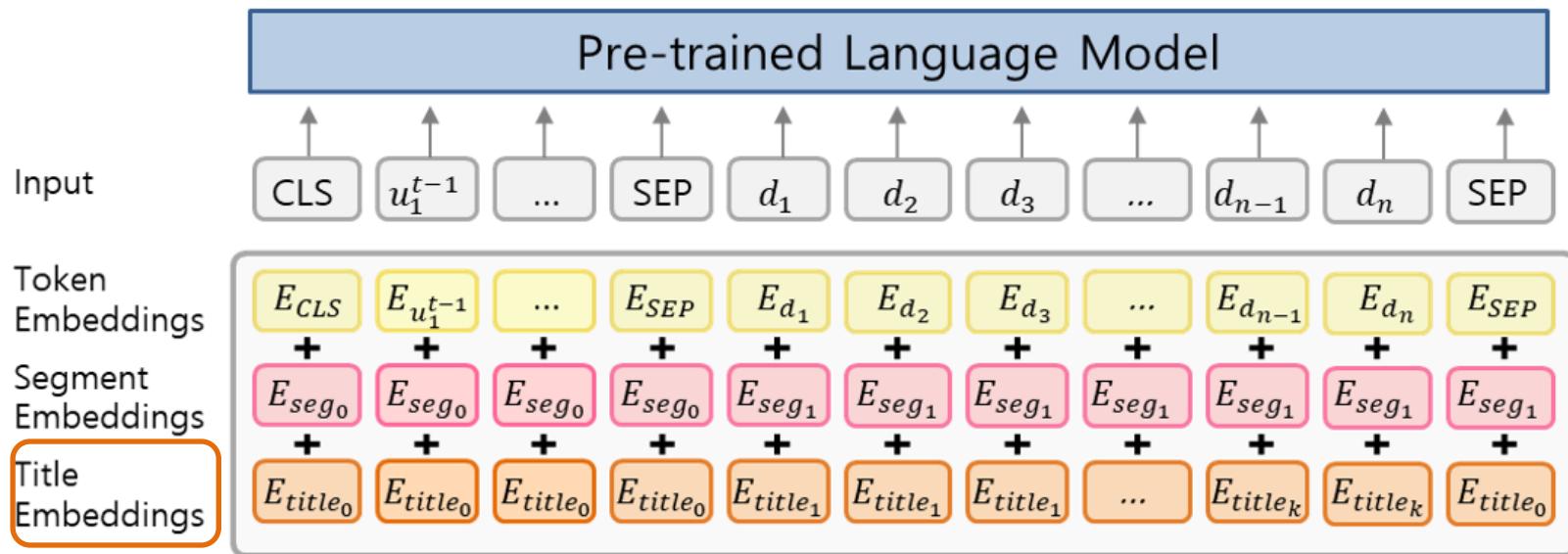
Dialogue act prediction



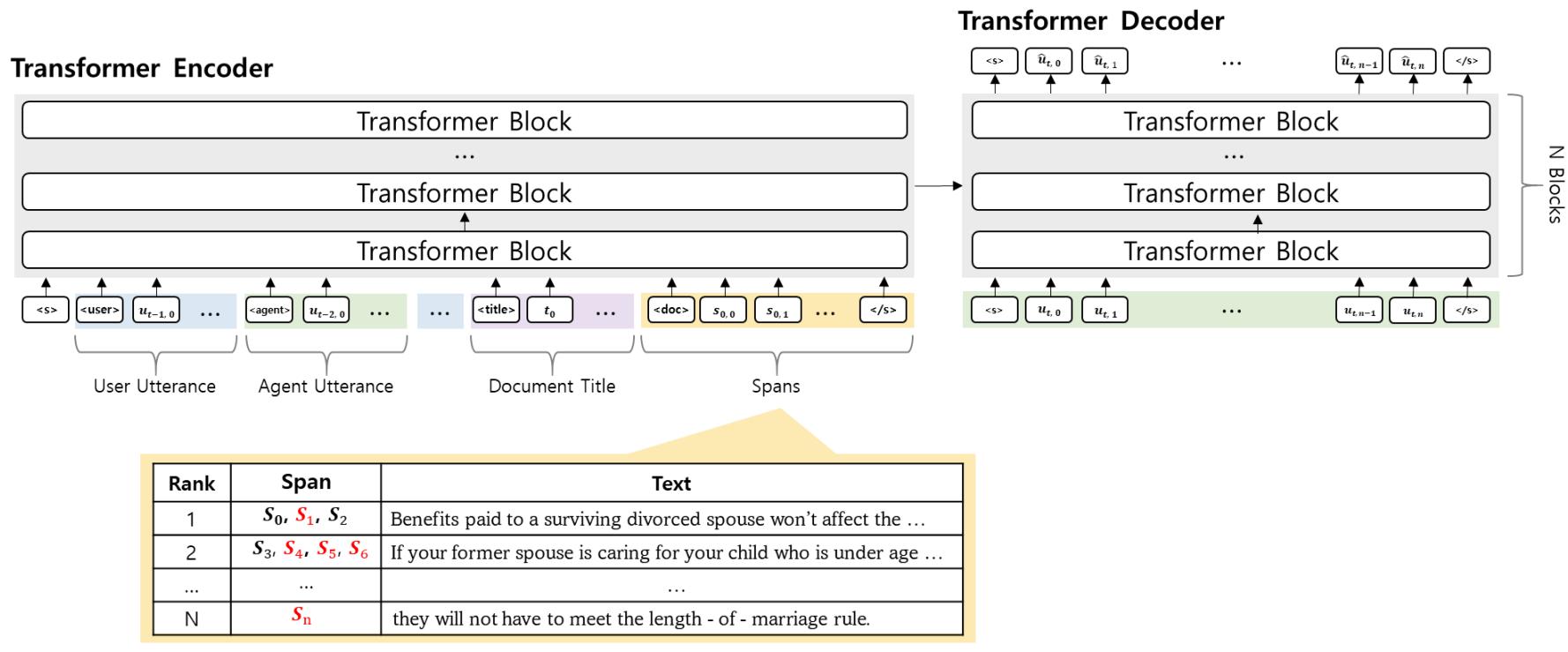
# Input Embedding for KU-DialMRC

- Title embedding

- 타이틀(title) 임베딩을 통해서 각 토큰(token)이 어느 타이틀에 속해 있는지에 대한 정보를 추가

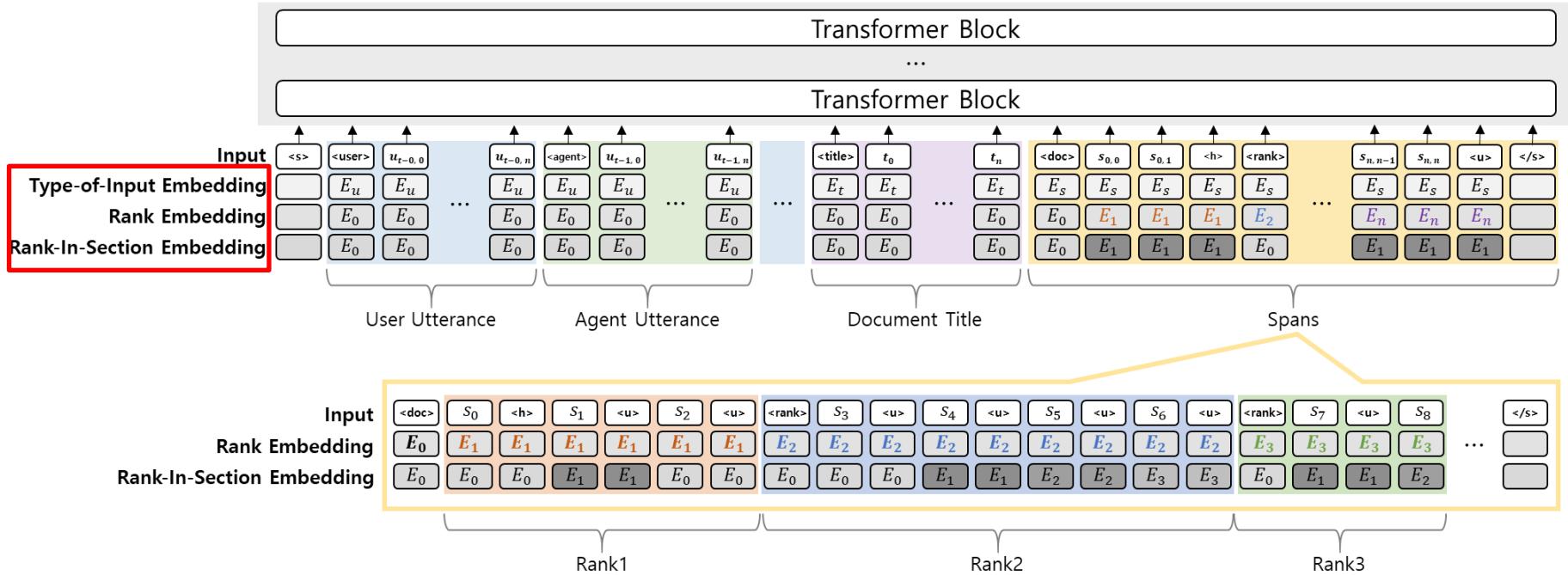


# KU-DirRG



# Input Embedding for KU-DirRG

## Transformer Encoder



# 실험 및 평가

- Document data
  - Seen domain : ssa, va, dmv, studentaid
  - Unseen domain : covid19

도메인	대화 수	문서 수	문서 당 평균			
			tokens	spans	paragraphs	sections
ssa	1,192	109	795	70	17	5
va	1,330	138	818	70	20	9
dmv	1,305	149	944	77	18	10
studentaid	966	91	1,007	75	20	9
all	4,793	487	888	73	18	8



# 실험 및 평가

- Dialogue data
  - Test-dev : Test dataset의 30%, seen domain만 포함
  - Test : unseen domain 추가

대화 데이터 종류	대화 수
Train	3,474
Validation	661
Test-dev	198
Test	787



# 실험 및 평가

Rank	Participant team	sacrebleu	Last submission at
1	KU_NLP	41.06	2 months ago
2	A-NLU (RWTH)	40.41	2 months ago
3	CAiRE (CAiRE)	37.72	2 months ago
4	nlp_opt_center	36.19	2 months ago
5	SCIR-DT	30.73	2 months ago
6	Schlussstein	30.68	2 months ago
7	CUCG	28.11	2 months ago



# 질의응답

---

Q & A

Homepage: <http://nlp.konkuk.ac.kr>  
E-mail: [nlpdrkim@konkuk.ac.kr](mailto:nlpdrkim@konkuk.ac.kr)

---

