자연어 이해 모델의 성능 향상을 위한 교차 게이트 메커니즘 방법

김성주○, 김원우, 설용수, 강인호
네이버
{sungju.kim, wonwoo.k, yongsoo.seol, once.ihkang}@navercorp.com

Cross Gated Mechanism to Improve Natural Language Understanding

Sung-Ju Kim○, Won-Woo Kim, Yong-Soo Seol, In-Ho Kang
Naver Corporation

요 약

자연어 이해 모델은 대화 시스템의 핵심적인 구성 요소로서 자연어 문장에 대해 그 의도와 정보를 파악하여 의도(intent)와 슬롯(slot)의 형태로 분석하는 모델이다. 최근 연구에서 의도와 슬롯의 추정을 단일 합동 모델(joint model)을 이용하여 합동 학습(joint training)을 하는 연구들이 진행되고 있다. 합동 모델을 이용한 합동 학습은 의도와 슬롯의 추정 정보가 모델 내에서 암시적으로 교류 되도록 하여 의도와 슬롯 추정 성능이 향상된다. 본 논문에서는 기존 합동 모델이 암시적으로 추정 정보를 교류하는 데서 더 나아가 모델 내의 의도와 슬롯 추정 정보를 명시적으로 교류하도록 모델링하여 의도와 슬롯 추정 성능을 높일 수 있는 교차 게이트 메커니즘(Cross Gated Mechanism)을 제안한다.

주제어: 대화 시스템, 자연어 이해, BERT, Joint training

1. 서론

자연어 이해 모델은 대화 시스템의 핵심적인 구성 요소이다. 자연어 이해 모델은 텍스트 형태로 입력된 사용자의 질의에 대해 의도를 파악하고 사용자의 질의 속에 있는 정보들을 추출하여 대화 시스템이 이해할 수 있는 형태의 정보로 출력한다. 이 단계는 사용자의 질의에서 의도와 슬롯을 추정하는 문제로 정의 할 수 있다. 대화 시스템의 앞 단계인 자연어 이해 단계에서 발생하는 의도와 슬롯 추정 오류들은 뒤로 전파되고 누적되어 전체 대화의 실패로 이어지게 된다. 그렇기 때문에 대화 시스템의 성능에 큰 영향을 끼치는 자연어 이해 단계에서의 오류를 줄이려는 연구들이 활발히 진행 되고 있다[1, 2, 4, 5, 6].

고전적인 자연어 이해에 대한 연구는 의도와 슬롯 추정에 대한 문제를 각각 별개의 모델로 모델링하여 해결하는 연구들이 많이 진행 되어 왔다. 하지만 최근 자연어 이해를 위한 의도와 슬롯에 대한 추정 모델을 단일 합동 모델로 모델링하는 연구들이 주류를 이루고 있다[1, 2, 3, 4]. 의도와 슬롯 추정을 위한 합동 모델(joint model)은 질의를 질의 벡터 표상(query representation)으로 인코딩하고, 이 질의 벡터 표상을 이용해 의도와 슬롯을 동시에 추정하도록 합동 학습(joint training)한다. 이러한 합동 모델은 학습한 질의 벡터 표상을 통해 의도와 슬롯을 동시에 추정하고 의도와 슬롯 간의 암시적인 정보 교류를 통해 의도와 슬롯 추정 성능이 모두 향상된다. 하지만 기존의 자연어 이해를 위한 합동 모델들은 의도 추정과 슬롯 추정의 과업의 구분에 관계없이 모델 내에서 동일한 구조로 정보가 교류된다.

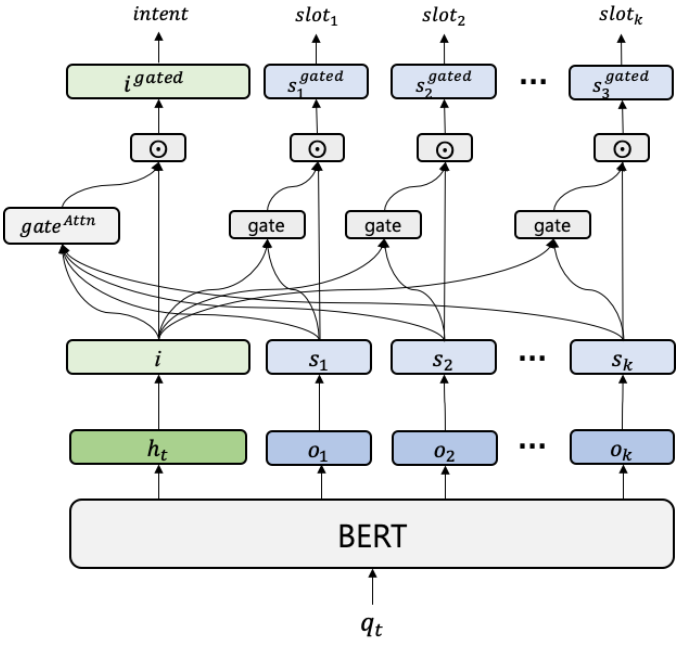

그림 1. 교차 게이트 메커니즘을 적용한 모델

본 논문에서는 의도 추정과 슬롯 추정의 각 과업에 맞게 명시적으로 의도와 슬롯 추정 정보를 교류가 가능한 형태로 모델링하여 의도와 슬롯 추정 성능을 높일 수 있는 교차 게이트 메커니즘을 제안한다.

2. 관련 연구

고전적인 자연어 이해 연구에서는 의도 추정과 슬롯 추정을 별개의 문제로 두고 해결하는 연구들이 있었다. 의도 추정을 SVM 또는 MLP 모델 등으로 활용하고 시퀀스 모델링이 필요한 슬롯 추정 문제에서는 CRF 또는 RNN 기반의 모델을 이용하여 해결하는 연구들이 진행 되었다 [5, 6].

자연어 이해 분야의 최근 연구는 의도 추정과 슬롯 추정을 합동 모델로 모델링하고 학습하는 연구들이 활발히 연구되고 있다 [1, 2, 3, 4]. Bing의 연구 [2]에서 RNN 기반의 모델을 활용해 질의의 벡터화 표상으로 활용되는 잠재 상태(hidden state)를 이용해 의도를 예측하고 seq2seq RNN decoder를 사용해 슬롯을 추정하는 합동 모델을 제안하였다. 최근 ELMO, BERT, XLNet 등의 언어 이해를 위한 pre-train 된 모델을 활용하여 자연어 이해 문제를 푸는 연구들도 진행되고 있다 [7, 8].

이러한 합동 모델들은 의도 추정 정보와 슬롯에 추정 정보가 암시적으로 교류를 되도록 하여 성능을 향상시키는데, 의도 추정과 슬롯 추정의 각 과업에 관계없이 모델 내부에서 동일한 구조로 정보가 교류 되도록 모델링 되어 있다.

본 논문에서는 그림 1과 같이 의도 추정과 슬롯 추정에 대해 각자의 과업에 맞게 의도 추정 정보와 슬롯 추정 정보를 명시적으로 참조하도록 모델링하여 의도와 슬롯 추정 성능을 모두 향상시키는 교차 게이트 메커니즘을 제안한다.

3. 교차 게이트 메커니즘을 이용한 자연어 이해 모델

이 장에서는 BERT[9] 기반의 문장 인코딩 모델에 제안하는 교차 게이트 메커니즘 방법을 적용하여 만든 자연어 이해 모델을 소개한다. 3.1절과 3.2절에 문장 표현 방법과 문장을 인코딩하여 문장 표상 벡터와 토큰 표상 벡터로 변환하는 과정을 설명한다. 교차 게이트 메커니즘을 구성하는 의도 게이트(Intent gate)와 슬롯 게이트(Slot gate)에 대해 각각 3.3절과 3.4절에 나누어 설명한다. 3.5절에서 의도와 슬롯의 합동 학습 방법에 대해 설명한다.

3.1. 문장 입력

본 논문에서 제안하는 모델은 음절 단위로 토큰화된 문장 $\mathbf{w} = (w_1, w_2, \dots, w_T)$을 입력으로 받는다. 토큰화된 문장 $\mathbf{w}$의 각 토큰 w는 토큰 embedding matrix를 참조하여 벡터화된 토큰 입력 $\mathbf{x} = (x_1, x_2, \dots, x_T)$으로 변환된다. 그리고 각 토큰의 순서 정보를 표현하기 위해 각 토큰 $\mathbf{x}$는 각 토큰의 위치에 대응하는 positional embedding $\mathbf{p} = (p_1, p_2, \dots, p_T)$ 을 가진다. 이때, $x_i \in \mathbb{R}^d$, $d =$ 임베딩 크기 ,$T =$ 시퀀스의 길이다.

3.2. BERT Transformer

문장 내의 문맥을 고려한 문장과 토큰의 벡터 표상을 만들기 위해 BERT 모델을 사용한다[9]. 벡터화된 입력 $\mathbf{x}$와 $\mathbf{p}$는 BERT 블록에 입력되어 문장 표상 벡터 h와 토큰 표상 벡터 $\mathbf{o} = (o_1, o_2, \dots, o_T)$ 를 출력한다. 이때, h는 BERT 의 특수 토큰인 [CLS] 위치의 임베딩을 뜻하고 $\mathbf{o}$는 동일 입력 토큰과 동일한 위치에 출력되는 토큰 임베딩을 뜻한다.

$$(h, \mathbf{o}) = BERT(\mathbf{x}, \mathbf{p}) \quad (1)$$

3.3. 슬롯 게이트 메커니즘

BERT로 임베딩된 토큰 벡터 표상 o_j를 W^o와 곱하고 바이어스 b^o를 더하여 슬롯 정보 벡터 s_j로 변환한다. s_j와 h를 결합(concatenate)한 후 $W^{slotfusion}$와 곱하고 바이어스를 더하여 슬롯과 의도의 정보를 함께 표현하는 벡터 s^{fusion}를 만든다. 이때, $W^{slotfusion} \in \mathbb{R}^{2d \times d}$이다.

$$\begin{aligned} s_j &= W^o o_j + b^o \\ s_j^{fusion} &= W^{slotfusion}[s_j, h] + b^{fusion} \end{aligned} \quad (2)$$

토큰 벡터 표상 o_j를 W'^o와 곱하고 바이어스를 더한 s'을 h와 결합하고 $W^{slotgate}$와 곱한 후 바이어스를 더한 후 sigmoid를 취해 0~1 사이의 실수 값으로 한정시킨 슬롯 게이트 벡터 g_j^{slot}를 구한다. g_j^{slot}는 슬롯 추정 정보 s_j와 슬롯과 의도 정보가 함께 표현된 s_j^{fusion} 사이의 가중치를 결정하는 가중치 게이트 역할을 한다. 이때, $g_j^{slot} \in \mathbb{R}^d$이다.

$$\begin{aligned} s'_j &= W'^o o_j + b'^o \\ g_j^{slot} &= sigmoid(W^{slotgate}[s'_j, h] + b^{slotgate}) \end{aligned} \quad (3)$$

최종적으로 s_j와 g_j^{slot} 에 대해 아다마르 곱(hadamard product) 연산 $\odot$ 하고 s_j^{fusion} 와 $(1 - g_j^{slot})$에 대해 $\odot$ 연산하고 더하여 슬롯 추정 벡터 s_j^{gated}를 구한 후 슬롯 클래스 개수를 맞춰 주도록 투영 연산 후 softmax를 취하여 슬롯 확률 벡터 $\hat{s_j}$를 구한다.

$$\begin{aligned} s_j^{gated} &= g_j^{slot} \odot s_j + (1 - g_j^{slot}) \odot s_j^{fusion} \\ \hat{y}_{j,l}^{slot} &= \frac{e^{W^{slotcls} s_{j,l}^{gated}}}{\sum_k e^{W^{slotcls} s_{j,k}^{gated}}} \end{aligned} \quad (4)$$

3.4 의도 게이트 메커니즘

의도 게이트를 구성하기 위해서는 전체 슬롯 표상 $\mathbf{s} = (s_1, s_2, \dots, s_T)$를 고려해야 한다. 현재 의도 정보를 표현하는 i와 전체 슬롯 표상 $\mathbf{s}$간의 attention score α_j을 계산하고 가중합하여 전체 슬롯 표상을 현재 의도 정보 관점으로 요약하는 벡터 $s^{summary}$를 구한다.

$$
\begin{aligned}
i &= W^h h + b^h \\
b_j &= o^T \cdot i \\
\alpha_j &= \frac{e^{b_j}}{\sum_k e^{b_j}} \\
s^{summary} &= \sum_{k=1}^{T} \alpha_k s_j
\end{aligned} \tag{5}
$$

획득한 전체 슬롯 표상 요약 벡터인 $s^{summary}$와 의도 정보 벡터 i를 결합하고 슬롯 게이트와 마찬가지로 수식 (6)와 같이 연산하여 i^{fusion}와 g^{int}를 구한다. g^{int}는 슬롯 게이트와 마찬가지로 의도 추정 정보 i'와 의도와 슬롯 요약 퓨전 정보 i^{fusion}간의 가중치 게이트 역할을 한다. 이 때, $g^{int} \in \mathbb{R}^d, W^{intfusion} \in \mathbb{R}^{2d \times d}$ 이다.

$$
\begin{aligned}
i^{concat} &= [s^{summary}, i] \\
i^{fusion} &= W^{intfusion} i^{concat} + b^{intfusion} \\
g^{int} &= sigmoid(W^{intgate} i^{concat} + b^{intgate})
\end{aligned} \tag{6}
$$

마지막으로 수식(7) 에서 g^{int}, i', i^{fusion}에 대해 슬롯 게이트 메커니즘과 동일하게 게이팅 연산을 진행 후 의도 추정 확률 벡터 $\hat{y}^{intent}$ 를 구한다.

$$
\begin{aligned}
i' &= W'^h h + b'^h \\
i^{gated} &= g^{int} \odot i' + (1 - g_j^{int}) \odot i^{fusion} \\
\hat{y}_j^{intent} &= \frac{e^{W^{intcls} i_j^{gated}}}{\sum_k e^{W^{intcls} i_k^{gated}}}
\end{aligned} \tag{7}
$$

3.5 의도, 슬롯 합동 학습

본 논문에서 제안하는 모델은 의도와 슬롯을 동시에 함께 학습하고 추정하는 합동 모델이다 본 논문에서 제안 하는 모델의 의도와 슬롯을 동시에 학습하기 위해 학습 데이터 $(y^{intent}, y^{slot}, \mathrm{w})$ 쌍에 대해 모델 $p(y^{intent}, y^{slot}|\mathrm{w})$를 최대화 하도록 학습한다. 실제 구현에서는 $p(y^{intent}, y^{slot}|\mathrm{w})$에 대해 negative log likelihood를 적용하여 목적 함수 $L(\theta)$를 정의 후 최소화 하도록 학습한다.

$$
\begin{aligned}
p_\theta(y^{intent}, y^{slot}|\mathrm{w}) &= p_\theta(y^{intent}|w) \prod_{k=1}^{T} p_\theta(y_k^{slot}|\mathrm{w}) \\
L(\theta) &= -\sum_{n=1}^{N} \log p_\theta(y_n^{intent}|w) \prod_{k=1}^{T} p_\theta(y_{n,k}^{slot}|\mathrm{w}) \\
\hat{\theta} &= \underset{\theta}{\mathrm{argmin}}\ L(\theta)
\end{aligned} \tag{8}
$$

4. 실험 및 결과

4.1 데이터 구성

모델을 평가하기 위해 실험 데이터는 ATIS(Airline Travel Information System) 데이터와 클로바 서비스[1] 발화 데이터로 자체 구축한 한국어 대화 데이터를 사용하여 모델을 학습하고 평가하였다. 두 데이터 셋 모두 의도와 슬롯 정보를 포함하고 있고 슬롯은 IOB(In-Out-Begin) 포맷으로 태깅되어 있다. ATIS 데이터 셋에 대한 IOB 태깅은 공백으로 토큰화한 단어 단위의 IOB 태깅으로 라벨링 되어 있다. 학습 데이터, 검증 데이터, 테스트 데이터의 수는 각각 4,478개, 500개, 893개를 사용하였다. ATIS 데이터의 의도 라벨의 개수는 18개이며 슬롯 라벨의 개수는 130개이다. 한국어 대화 데이터의 IOB 태깅 정보는 각 슬롯에 대한 음절 단위 시작과 끝 위치 정보로 표현되고 해당 정보를 이용해 표 1과 같이 음절 단위의 IOB 태깅 데이터로 변환하여 사용하였다. 학습 데이터, 검증 데이터, 테스트 데이터의 수는 각각 191,698개, 10,000개, 1,000개이다. 의도 라벨의 개수는 12개이고 슬롯 라벨의 개수는 28개를 사용하였다.

표 1. 모델 학습에 입력되는 데이터 포맷

문장	하	#나	언	#니	동	#화	틀	#어	#줘
슬롯	B-epi	I-epi	I-epi	I-epi	I-epi	I-epi	O	O	O
의도	PlayEpisode								

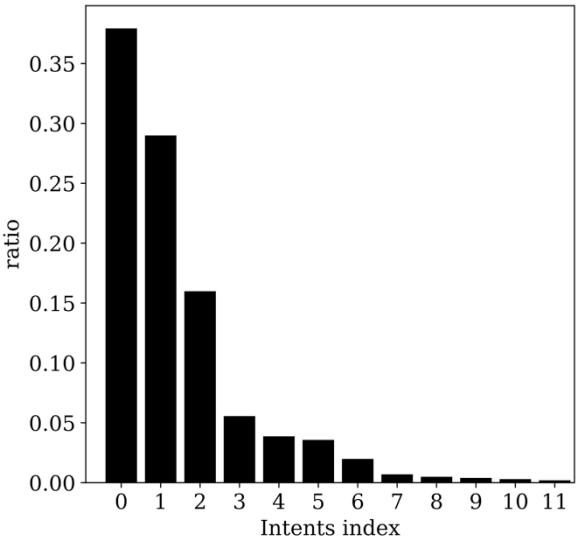

그림 2. 한국어 대화 데이터 의도 비율

4.2 모델 파라미터 및 학습

앞서 3.2절에 서술한 것과 같이 본 논문에서는 pre-train 된 BERT를 이용하여 문장과 토큰에 대한 벡터 표

[1] 네이버 앱의 음성 검색 서비스와 프렌즈 스피커 서비스

표 2. 모델 성능 측정

모델	BERT 레이어	ATIS			한국어 대화 데이터		
		의도 정확도	슬롯 F1	문장 정확도	의도 정확도	슬롯 F1	문장 정확도
Joint BERT [8]	1	95.97	93.59	82.42	99.80	99.60	92.45
	12	96.98(97.50)	95.73(96.10)	87.57(88.20)	99.80	99.67	93.55
Cross Gated (제안 모델)	1	96.75	94.65	84.66	99.90	99.63	93.54
	12	**97.31**	**96.15**	**88.02**	**100.0**	**99.68**	**94.04**

상을 생성하였다. BERT에 사용된 파라미터는 BERT를 제안한 [9]의 파라미터와 같이 Multi head의 수 12개, 잠재 상태의 크기 768, dropout 0.1, 위치 임베딩 크기 512를 사용하였다. 본 논문에 나타낸 명시적으로 그 weight의 크기를 서술하지 않은 모든 weight에 대해 $\boldsymbol{W}^* \in \mathbb{R}^{d\times d}$ 와 모든 정보 벡터 $i^*, \boldsymbol{s}_*^* \in \mathbb{R}^d$ 에 사용된 잠재 상태 크기 d는 768을 사용하였다. ATIS 데이터 셋에 대하여 BERT 레이어 12개인 모델을 실험하였고 배치 크기 128, Epoch는 최대 100으로 설정하여 학습하였다. 한국어 대화 데이터의 경우 BERT 레이어 1개, 12개인 모델에 대해 실험하였고 배치 크기는 각각 512, 32를 사용하였다. Epoch는 최대 20으로 설정하였다. 한국어 대화 데이터에 대해서는 그림 2와 같이 학습 데이터의 불균형이 심하여 미니 배치 학습 과정에서 임의의 한 배치에 각 의도가 균등 비율로 샘플링 되도록 하였다. 그림 2의 의도명은 서비스와 밀접한 관계가 있기 때문에 인덱스로만 표현하였다. 의도에 대해 모델 최적화 알고리즘은 Adam 최적화 알고리즘[10]을 사용하였고 learning rate 은 5.0×10^{-5}, $\beta_1 = 0.9, \beta_2 = 0.999$ 로 설정하여 학습을 진행하였다.

4.3 실험 결과

교차 게이트 메커니즘을 이용한 의도와 슬롯 성능 향상을 확인하기 위해 Joint BERT[2][8]모델을 Baseline으로 두고 실험하였다. Joint BERT를 재현하여 실험한 결과가 [8]에서 기록한 결과와 일치 하지 않아 자체 실험 환경에서 재현한 결과와 [8]에서 기록한 결과를 표 2에 함께 기록하였다. [8]에서 기록한 점수는 괄호 안에 기록하였다.

모델 비교를 위해 의도 정확도, 슬롯 F1 점수, 문장 정확도를 지표로 설정하였다. 문장 정확도는 임의의 한 발화 샘플에 대해 의도와 슬롯에 대한 추정 결과가 모두 정답일 때를 correct, 의도와 슬롯들 중 하나라도 틀렸을 경우를 wrong으로 두고 correct/(correct+ wrong)을 계산하여 문장 정확도로 하였다.

표 2에서 제안한 모델이 자체 환경에서 실험한 Baseline 에 비해 의도 정확도, 슬롯 F1 점수, 문장 정확도에서 모두 더 높은 성능을 보였다. 슬롯 F1 점수의 경우 [8]에서 보고한 슬롯 F1 점수보다 높은 성능을 보였다.

표 3. 모델 속도 측정, 단위: 발화/초

모델	BERT 레이어	속도 평균	속도 분산
Joint BERT [8]	1	0.0621	0.0025
	12	0.7167	0.0166
Cross Gated (제안 모델)	1	0.0914	0.0003
	12	0.789	0.0025

표 4. 제거 실험 (ATIS)

모델	의도 정확도	슬롯 F1	문장 정확도
w/o Intent gate	96.98	96.11	87.68
w/o Slot gate	**97.31**	95.59	87.79
Cross Gated (제안 모델)	**97.31**	**96.15**	**88.02**

의도와 슬롯 추정 성능 향상에 반한 속도 성능 감소를 알아보기 위해 Baseline과 제안 모델의 속도 측정 실험을 진행 하였다. 속도 측정은 단일 발화에 대한 싱글 CPU 수행 속도를 측정하였다. CPU 재원은 Xeon CPU ES-2630, 2.20 GHz 모델을 사용하였다. 표 3에서 제안 모델은 동일한 1 레이어 모델, 12 레이어 모델에서 각각 0.029 발화/초, 0.073 발화/초 가량의 속도가 하락하였다. 하지만 표 2와 표 3에서 제안 모델의 1 레이어 모델이 Joint BERT 12 레이어 모델에 비해 의도 정확도, 슬롯 F1 점수, 문장 정확도, 속도의 모든 지표에서 성능이 모두 높음을 알 수 있다.

표 4에서 의도 게이트 메커니즘과 슬롯 게이트 메커니즘 방법 각각의 성능 향상을 측정하기 위해 ATIS 데이

[2] https://github.com/sz128/slot_filling_and_intent_detection_of_SLU

터 셋에서 제안 모델에 대해 의도 게이트 메커니즘과 슬롯 게이트 메커니즘을 제거하며 제거 실험을 진행 하였다. 의도 게이트 메커니즘과 슬롯 게이트 메커니즘를 각각 추가함에 따라 성능이 향상하였고 두 메커니즘을 모두 적용한 제안 모델이 의도 정확도, 슬롯 F1 점수, 문장 정확도의 모든 지표에서 가장 높은 성능을 보였다.

5. 결론

본 연구에서는 의도 및 슬롯 추정의 자연어 이해 문제에서 BERT 기반의 의도와 슬롯 정보를 명시적으로 교류를 모델링하는 교차 게이트 메커니즘을 제안하였다. 실험 결과 자체 실험 환경에서 기존 BERT 기반의 의도, 슬롯 추정 모델[8]보다 더 높은 성능을 보였다. 또한 제안한 교차 게이트 메커니즘은 자연어 이해 성능 향상에 합동 모델의 의도 추정과 슬롯 추정의 각 과업에 맞는 명시적인 정보 교류 모델링 방법론이 성능 향상에 도움이 된다는 것을 보였다.

향후 연구로는 본 논문에서 제안한 교차 게이팅 메커니즘을 개선하고 모델 상위 임베딩 공간에서의 명시적 정보 교류 메커니즘에 대한 해석 및 분석 방법을 연구할 것이다.

참고문헌

[1] Guo, Daniel, et al. "Joint semantic utterance classification and slot filling with recursive neural networks." 2014 IEEE Spoken Language Technology Workshop (SLT). IEEE, pp. 554-559, 2014.

[2] Liu, Bing, and Ian Lane. "Attention-based recurrent neural network models for joint intent detection and slot filling." arXiv preprint arXiv:1609.01454, 2016.

[3] Hakkani-Tür, Dilek, et al. "Multi-domain joint semantic frame parsing using bi-directional rnn-lstm." Interspeech, pp. 715-719, 2016.

[4] Yao, Kaisheng, et al. "Recurrent neural networks for language understanding." Interspeech, pp. 2524-2528, 2013.

[5] Mesnil, Grégoire, et al. "Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding." Interspeech, pp. 3771-3775, 2013.

[6] Raymond, Christian, and Giuseppe Riccardi. "Generative and discriminative algorithms for spoken language understanding." Eighth Annual Conference of the International Speech Communication Association, 2007.

[7] Siddhant, Aditya, Anuj Goyal, and Angeliki Metallinou. "Unsupervised transfer learning for spoken language understanding in intelligent agents." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 33, pp. 4959-4966, 2019.

[8] Chen, Qian, Zhu Zhuo, and Wen Wang. "BERT for Joint Intent Classification and Slot Filling." arXiv preprint arXiv:1902.10909, 2019.

[9] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805, 2018.

[10] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980, 2014.