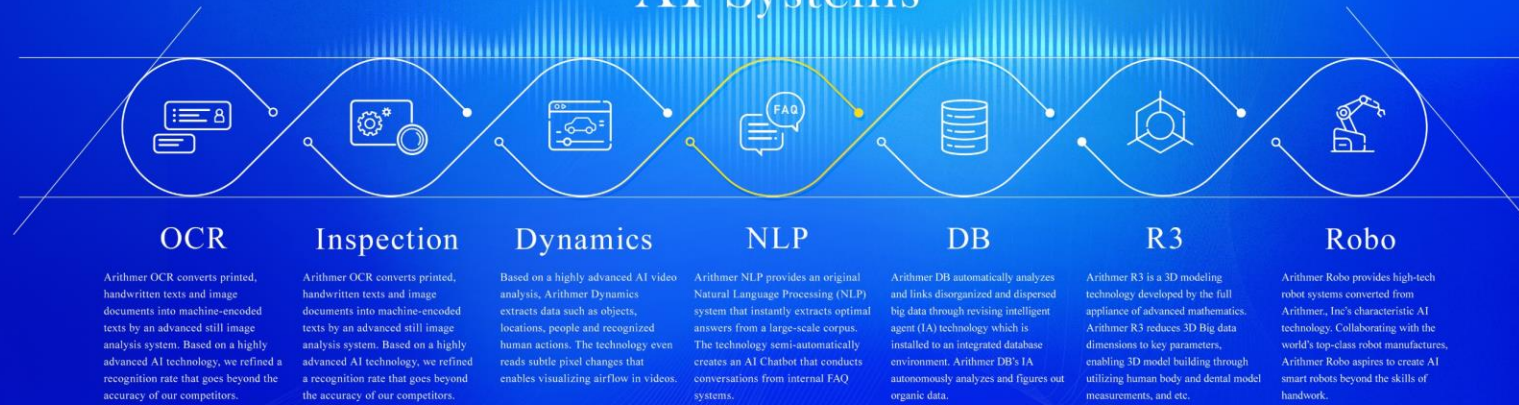


# Arithmer NLP

## AI Systems



# ACL2020

## Task-oriented Dialogue System 関連論文紹介

Arithmer NLP Div. Haruya Umemoto

2020/08/21

## 梅本 晴弥 (Umemoto Haruya)

### キャリア

- 2018, 青山学院大学 (情報テクノロジー学科) 卒業 学士 (工学)
- 2020, 青山学院大学大学院 (知能情報コース) 卒業 修士 (工学)
- 2020, Arithmer 株式会社

専門分野：強化学習，機械学習，自然言語処理（対話システム）

### 現職

- 機械学習エンジニア（NLPチーム）
- 自然言語処理システムの実装，対話システムの研究開発を担当

ポートフォリオサイト：<https://umeco.tokyo>

## 目次

1. Dialogue State Trackingタスクの評価方法について
2. Transferable Multi-Domain State Generator for Task-Oriented Dialogue Systems, 2019
3. SAS: Dialogue State Tracking via Slot Attention and Slot Information Sharing, 2020
4. A Contextual Hierarchical Attention Network with Adaptive Objective for Dialogue State Tracking, 2020

## 内容

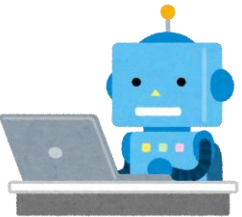
- 最新の対話システム（特にタスク指向）研究について発表
- 事前に関連技術に関しての知識がない方でも分かりやすいように説明多め
- 最近のモデルのスタンダードになっているACL2019の論文1本, ACL2020の論文を2本紹介

## Task-Oriented Dialogueの例

最終的な目標が存在する対話

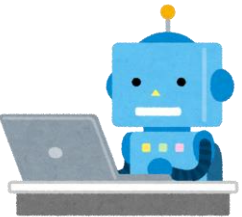
I am feeling hungry so I would like to find a place to eat

Do you have a specific which you want the eating place to be located at?



I would like for it to be in San Jose

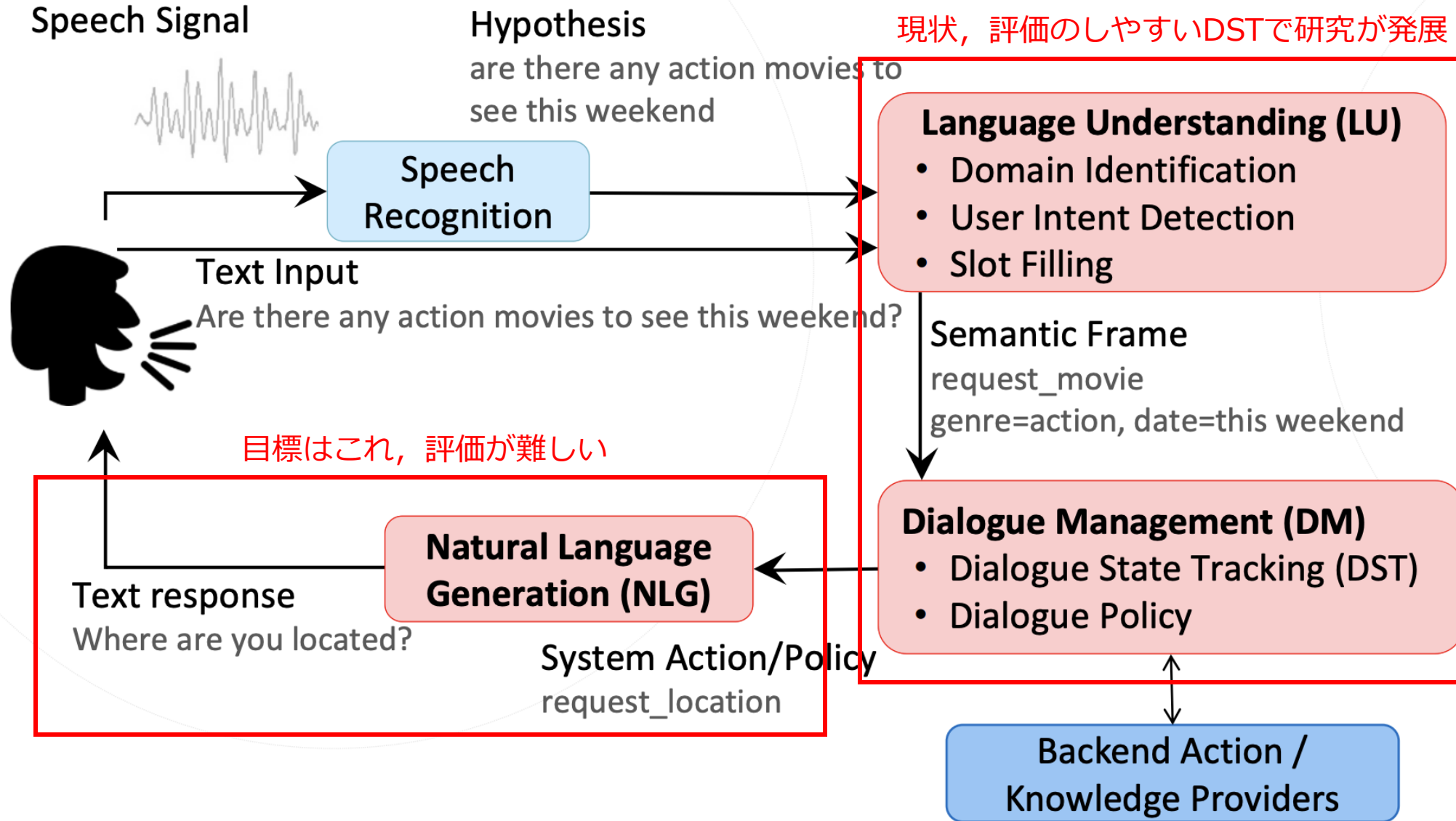
Is there a specific cuisine type you enjoy, such as Mexican, Italian or something else?



I usually like eating the American type of food

→レストランの予約がしたい

## Task-oriented dialogueにおけるPipeline手法[1]



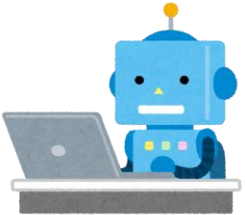
# Dialogue State Tracking (DST)

対話をする上で記憶すべき単語を抜き出すタスク



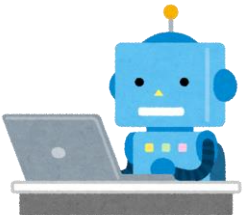
I am feeling hungry so I would like to find a place to eat

Do you have a specific which you want the eating place to be located at?



I would like for it to be in **San Jose**

Is there a specific cuisine type you enjoy, such as Mexican, Italian or something else?



I usually like eating the **American type** of food

## MultiDomain Wizard-of-Oz dataset (MultiWOZ)[2]

人と人の対話を人手でアノテーションしたDST評価データセット

## example

Agent: I have two restaurants. They are Pizza Hut Cherry Hinton and Restaurant Alimentum.

User: What type of food do each of them serve?

**restaurant.name:** *Pizza Hut Cherry Hinton, Restaurant Alimentum* (other slots: none)

User: I would like to visit a museum or a nice nightclub in the north.

**attraction.type:** *museum, nightclub*

User: I would also like a reservation at a Jamaican restaurant in that area for seven people at 12:45, if there is none Chinese would also be good.

**restaurant.food:** *Jamaican (preferred), Chinese*

User: I would prefer one in the cheap range, a moderately priced one is fine if a cheap one isn't there.

**restaurant.pricerange:** *cheap (preferred), moderate*

## vocab. size of slots

Slot Name	2.0	2.1
taxi-leaveAt	119	108
taxi-destination	277	252
taxi-departure	261	254
taxi-arriveBy	101	97
restaurant-people	9	9
restaurant-day	10	10
restaurant-time	61	72
restaurant-food	104	109
restaurant-pricerange	11	5
restaurant-name	183	190
<u>restaurant-area</u>	19	7

domain

## versions of WOZ

WOZ 2.0	ACL 2018
WOZ 2.1	LREC 2020
WOZ 2.2	ACL 2020

WOZでは1inputにつき全てのslotの予測をする

## 概要

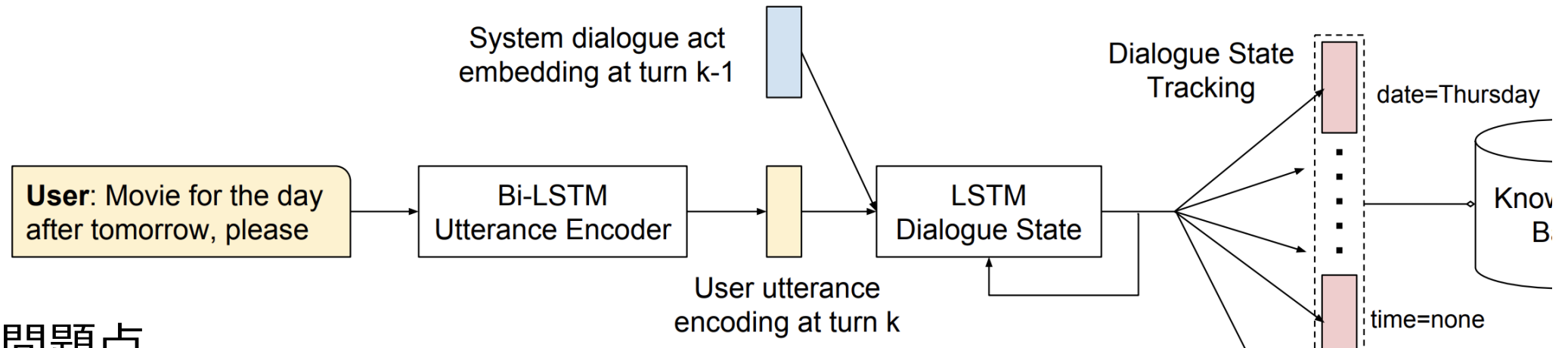
DST予測モデルにおいて、Attention機構やCopy機構を利用し、長文の入力から効率良く情報を抽出できる手法を提案  
Zero-Shot, Few-Shotの問題に対しても予測できることを示した

[ACL2019 Outstanding paper!](#)

## 新規性

1. ドメイン名・スロット名を入力として扱う
2. copy機構を利用した語彙出力でスロット値を予測
3. スロット値の状態も予測し、スロット値予測と同時に最適化
4. few-shot, zero-shotの問題に対する精度も調査





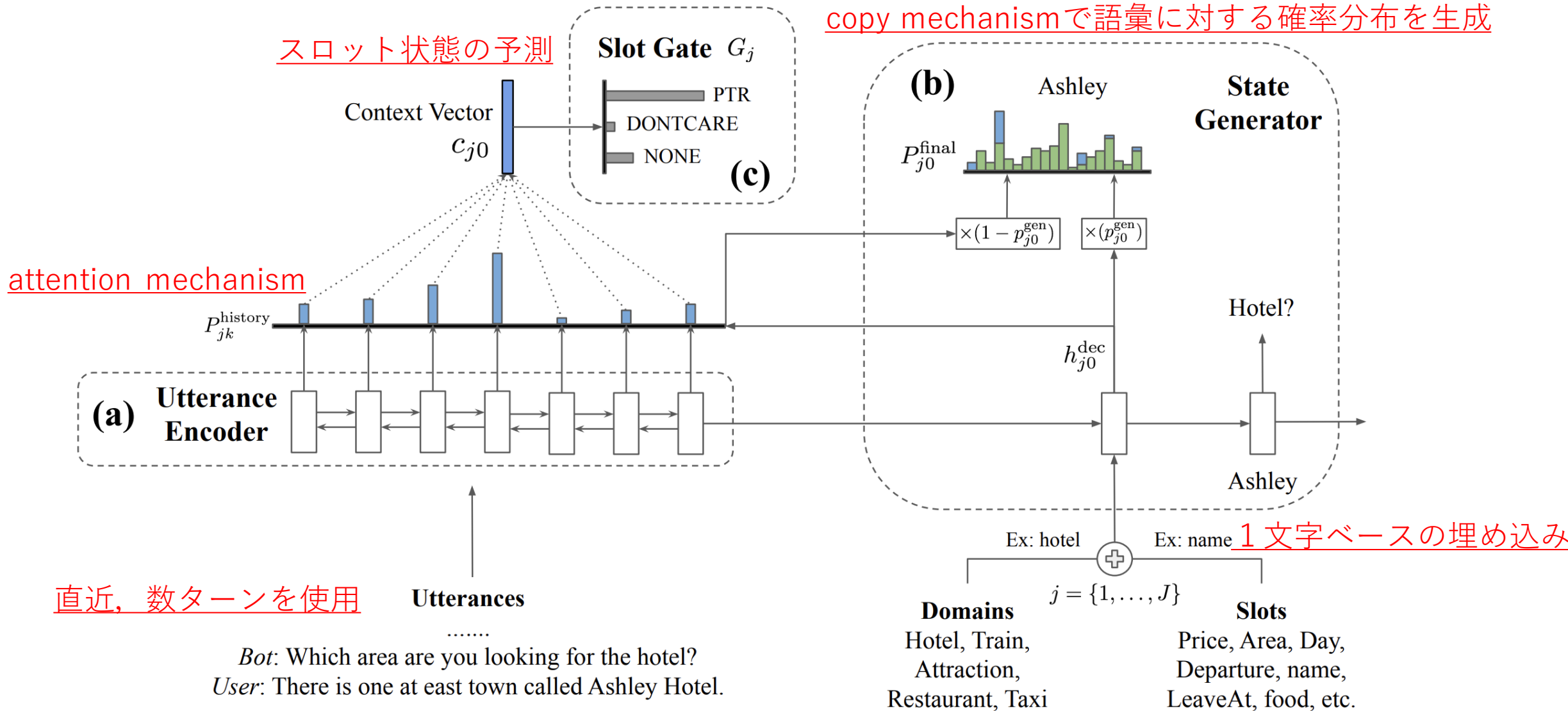
## 問題点

1. 長文の入力をすると学習がうまくいかない

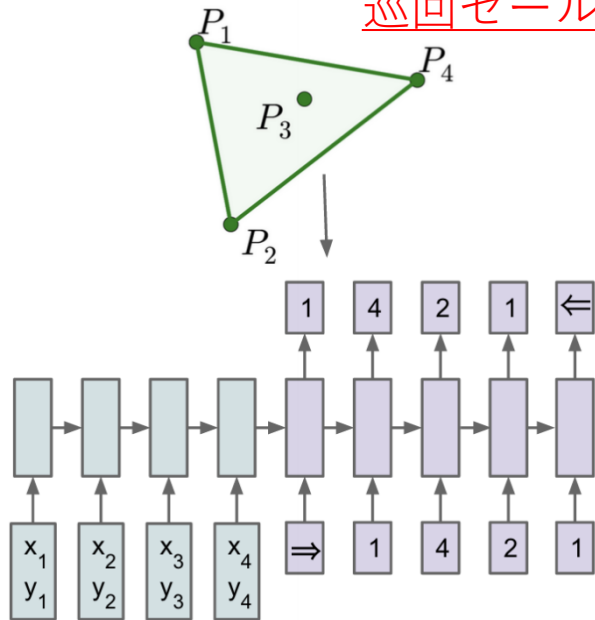
過剰な情報をうまく処理することができない

2. モデルがスロット数の増大に対応できない

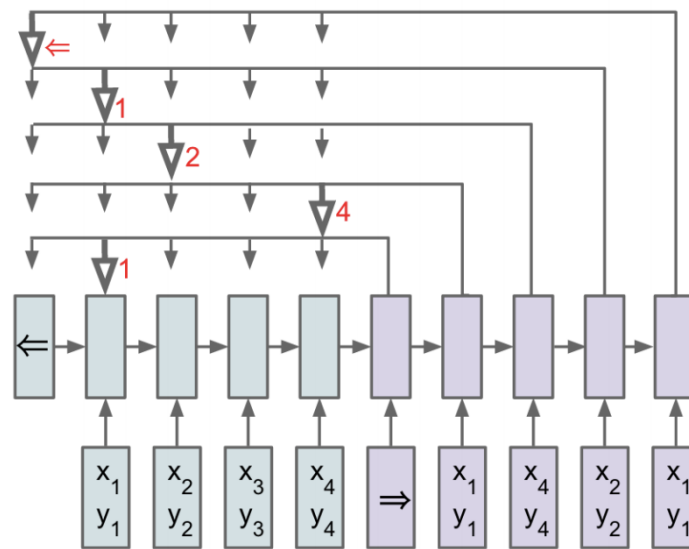
スロット値の予測をするモデルの構造上対応できない



巡回セールスマン問題を仮定



(a) Sequence-to-Sequence



(b) Ptr-Net

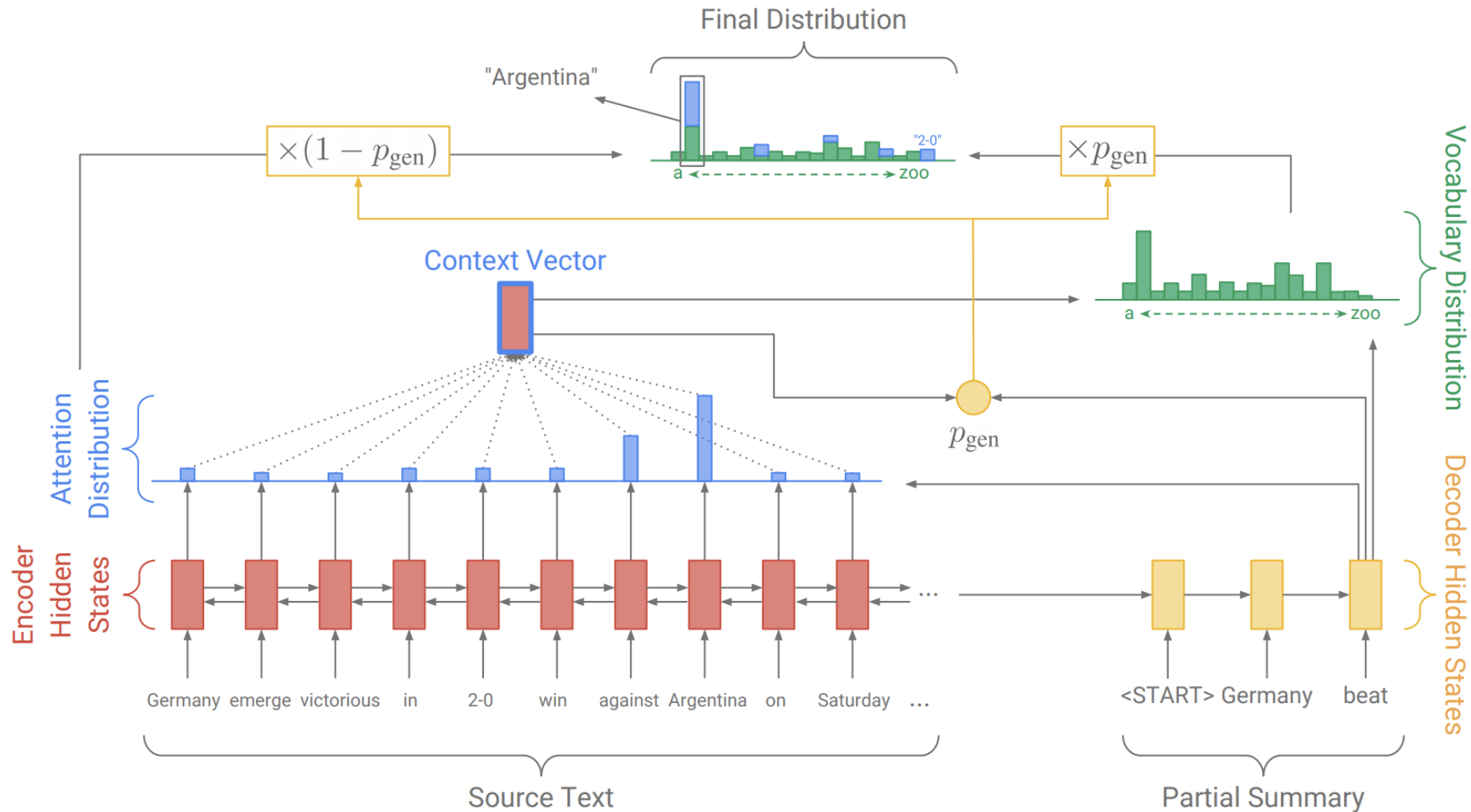
$$u_j^i = v^T \tanh(W_1 e_j + W_2 d_i) \quad j \in (1, \dots, n)$$

$$a_j^i = \text{softmax}(u_j^i) \quad j \in (1, \dots, n)$$

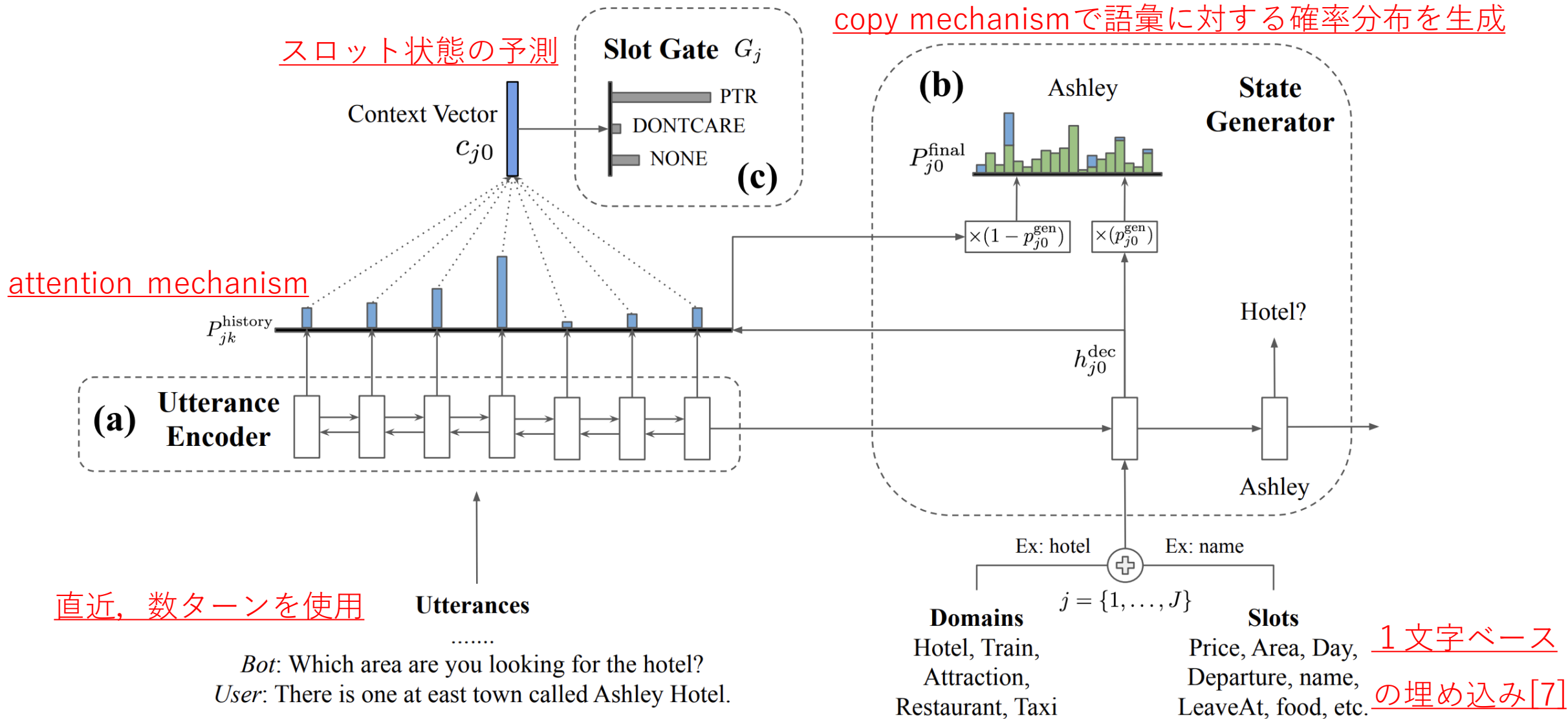
$e_i$  :  $i$ 番目のエンコーダの隠れ状態  
 $d_i$  :  $i$ 番目のデコーダの隠れ状態  
 $v, W$  : モデルパラメータ

attentionを利用して，入力に対する分布を出力

NLPの応用ではOut-Of-Vocablary問題の解決に使える



入力から選択する分布と語彙から選択する分布を合成する



通常の評価結果

未知のドメインの評価結果  
(Zero-Shot)

	MultiWOZ		MultiWOZ (Only Restaurant)	
	<i>Joint</i>	<i>Slot</i>	<i>Joint</i>	<i>Slot</i>
<i>MDBT</i>	15.57	89.53	17.98	54.99
<i>GLAD</i>	35.57	95.44	53.23	96.54
<i>GCE</i>	36.27	98.42	60.93	95.85
<i>SpanPtr</i>	30.28	93.85	49.12	87.89
<i>TRADE</i>	<b>48.62</b>	96.92	<b>65.35</b>	93.28

	Trained Single		Zero-Shot	
	<i>Joint</i>	<i>Slot</i>	<i>Joint</i>	<i>Slot</i>
<i>Hotel</i>	55.52	92.66	13.70	65.32
<i>Train</i>	77.71	95.30	22.37	49.31
<i>Attraction</i>	71.64	88.97	19.87	55.53
<i>Restaurant</i>	65.35	93.28	11.52	53.43
<i>Taxi</i>	76.13	89.53	<b>60.58</b>	73.92

Joint : 1対話においてスロット予測が全て正解だった確率

Slot : 1つ1つのスロット予測の精度

従来手法の精度を大幅に上回る精度を達成

継続学習 (continuous learning)の技術を利用

## Elastic Weight Consolization (EWC)

$$L_{ewc}(\Theta) = L(\Theta) + \sum_i \frac{\lambda}{2} F_i (\Theta_i - \Theta_{S,i})^2$$

$\Theta$  : 現在のパラメータ

$\Theta_S$  : 事前学習後のパラメータ

## Gradient Episodic Memory (GEM)

Minimize $_{\Theta}$   $L(\Theta)$

Subject to  $L(\Theta, K) \leq L(\Theta_S, K)$

$K$  : 事前学習のデータ (サンプル)

1. 豊富にデータがあるドメインで事前学習
2. データ数が少ないドメインに対して継続学習の手法で最適化

Evaluation on 4 Domains		Joint Slot <i>Except Hotel</i>	Joint Slot <i>Except Train</i>	Joint Slot <i>Except Attraction</i>	Joint Slot <i>Except Restaurant</i>	Joint Slot <i>Except Taxi</i>
Base Model (BM) training on 4 domains		58.98 96.75	55.26 96.76	55.02 97.03	54.69 96.64	49.87 96.77
Fine-tuning BM on 1% new domain	<i>Naive</i>	36.08 93.48	23.25 90.32	40.05 95.54	32.85 91.69	46.10 96.34
	<i>EWC</i>	40.82 94.16	28.02 91.49	45.37 84.94	34.45 92.53	<b>46.88</b> 96.44
	<i>GEM</i>	<b>53.54</b> <b>96.27</b>	<b>50.69</b> <b>96.42</b>	<b>50.51</b> <b>96.66</b>	<b>45.91</b> <b>95.58</b>	46.43 <b>96.45</b>
Evaluation on New Domain		<i>Hotel</i>	<i>Train</i>	<i>Attraction</i>	<i>Restaurant</i>	<i>Taxi</i>
Training 1% New Domain		19.53 77.33	44.24 85.66	<b>35.88</b> <b>68.60</b>	32.72 82.39	60.38 72.82
Fine-tuning BM on 1% new domain	<i>Naive</i>	19.13 75.22	<b>59.83</b> <b>90.63</b>	29.39 60.73	<b>42.42</b> <b>86.82</b>	<b>63.81</b> <b>79.81</b>
	<i>EWC</i>	19.35 76.25	58.10 90.33	32.28 62.43	40.93 85.80	63.61 79.65
	<i>GEM</i>	<b>19.73</b> <b>77.92</b>	54.31 89.55	34.73 64.37	39.24 86.05	63.16 79.27

1行目：学習するドメインの種類（全5ドメインの内1ドメインをFew-shotと仮定）

2行目：全データが普通に入手できる時の精度（予測精度の上限）

3行目：1%ドメインで学習後の、事前学習した4ドメインの精度

5行目：1%ドメインのみで学習した精度



- Attention機構, Copy機構を利用した, 長文の入力から効率良く情報を抽出できるDSTモデルを提案
- ドメイン名・スロット名を入力として扱うことで, 従来のスロット数増加問題を解決
- MultiWOZタスクにおいて従来手法の結果を大幅に超えるDST精度を示した
- データが入手できない, 少量しか入手できない場合の問題設定を提起し, 提案手法を用いた予測精度ベンチマークを示した

## 概要

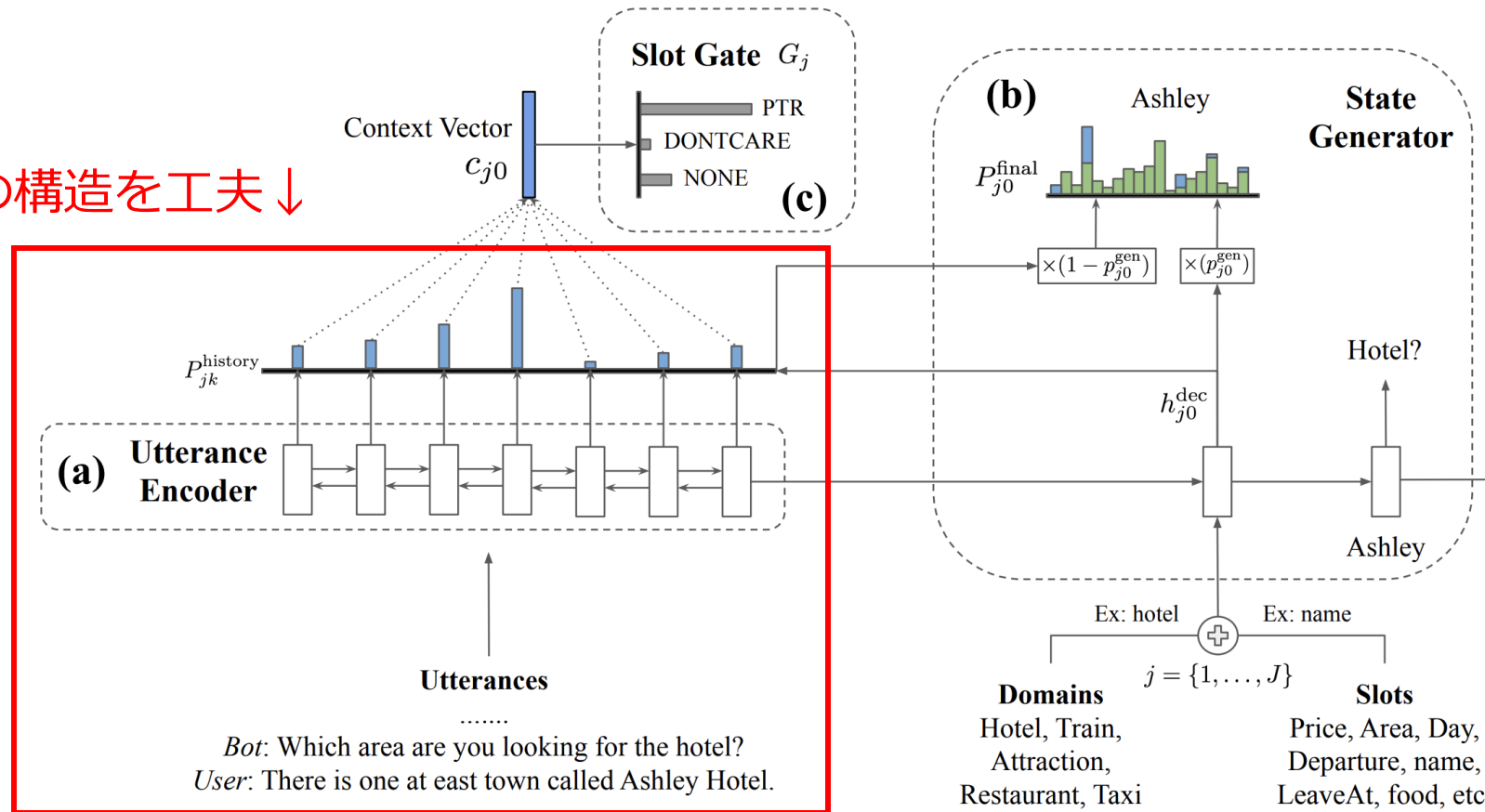
長大な入力をスロット名を利用して選択的に変換するSlot Attentionと、スロット間の情報を共有するSlot Information Sharingを提案  
SoTAであるTRADEの性能を上回る精度を示した

[TRADE + Slot attention + Slot information sharingのモデル](#)

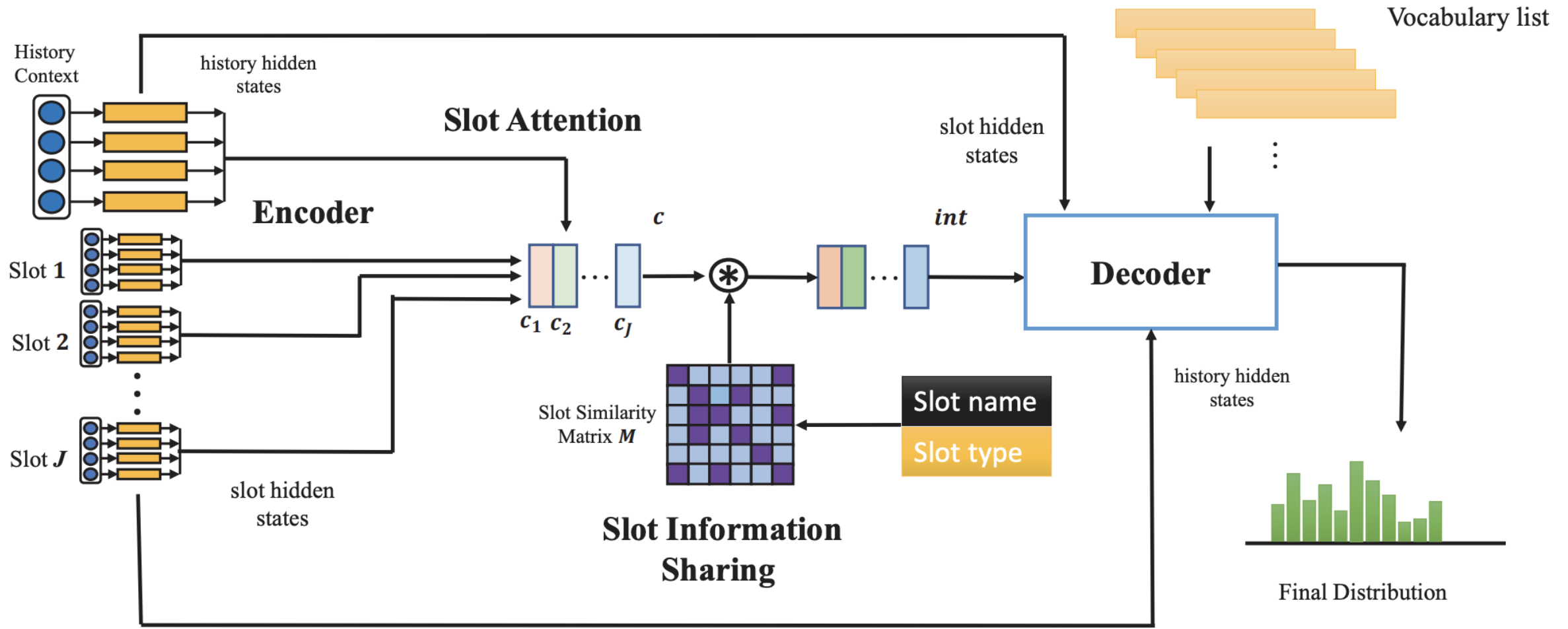
## 新規性

1. 入力の各スロットに必要な情報をSlot Attentionで重み付け
2. 事前に作成したスロット間の類似行列を用いてスロット間の情報を共有

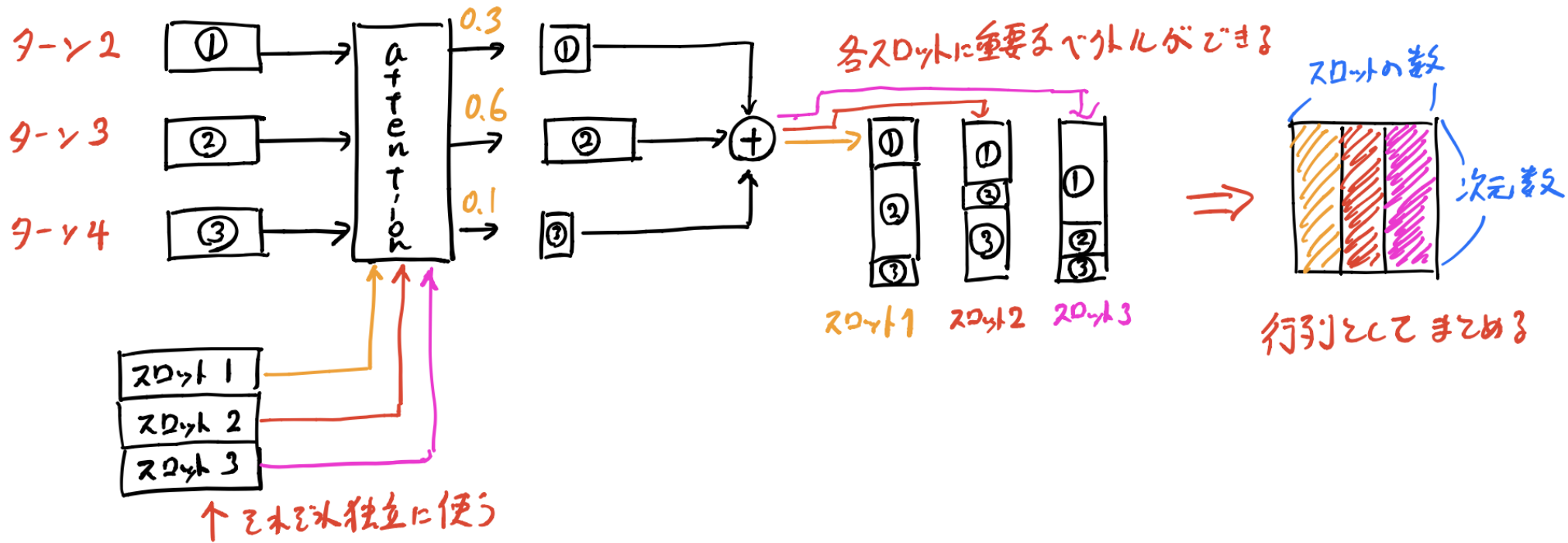
SASではこの構造を工夫↓



attentionを使用しているものの, 大量の情報をうまく扱えていないのでは?



文章 (ターンごとに入力、一気に全部入力しない!)



大量の文章を入力せず、各ターンの入力に注意することで効果的な処理が可能

類似度行列... スロット間の類似度を表す

例: スロット1とスロット3が似ていると3つ.

	1	2	3
1	1	0	1
2	0	1	0
3	1	0	1

これを利用して  
⇒

1	0	1
0	1	0
1	0	1

Slot attentionで作った行列

X

1	0	1
0	1	0
1	0	1

=

1	0	1
0	1	0
1	0	1

他のスロットが重視する情報をまぎれ込む

## 類似度行列の作り方

1. ドメイン名, スロット値の型名のcosine sim.を計算 (RNNを使用)
2. 閾値法, K-means法で行列を作成

$$sim = \alpha \cdot sname + (1 - \alpha) \cdot vtype$$

$$M_{ij} = \begin{cases} 1 & \text{if } sim_{ij} \geq \beta \\ 0 & \text{if } sim_{ij} < \beta \end{cases}$$

従来手法との比較

Model	MultiWOZ		MultiWOZ(res)	
	Joint	Slot	Joint	Slot
MDBT	15.57	89.53	17.98	54.99
GLAD	35.57	95.44	53.23	<b>96.54</b>
GCE	36.27	<b>98.42</b>	60.93	95.85
SpanPtr	30.28	93.85	49.12	87.89
TRADE	48.62	96.92	65.35	93.28
SAS	<b>51.03</b>	97.20	<b>67.34</b>	93.83

joint精度をTRADEから2.41%向上

ablation study

Model	MultiWOZ		MultiWOZ(res)	
	Joint	Slot	Joint	Slot
SAS-att-shr	48.62	96.92	65.35	93.28
SAS-shr	49.99	97.10	66.89	93.62
SAS(RT shr)	<b>51.03</b>	<b>97.20</b>	<b>67.34</b>	<b>93.83</b>
SAS(KM shr)	50.46	97.15	66.65	93.78
SAS(HM shr)	50.27	97.13	66.89	93.62

Slot Attentionで1.37%向上

Slot Information Sharingで1.04%向上

## まとめ

- 情報過多な入力をうまく扱うために、対話ターン毎にAttentionするSlot Attentionと、スロット間で情報を共有するSlot Information Sharingを提案
- MultiWOZにおいて、TRADEをjoint精度で2.43%上回る精度を示した

## 感想

- Slot Information Sharingは結局のところAttentionなのでは？  
(Attentionとして扱った方がパラメータ自動推定されるし…)
- 明示的にSlot Information Sharingするのか、スロット名をembeddingして上手く情報を共有してくれるのを期待した方がいいのかは、もう少し慎重に調べる必要がある

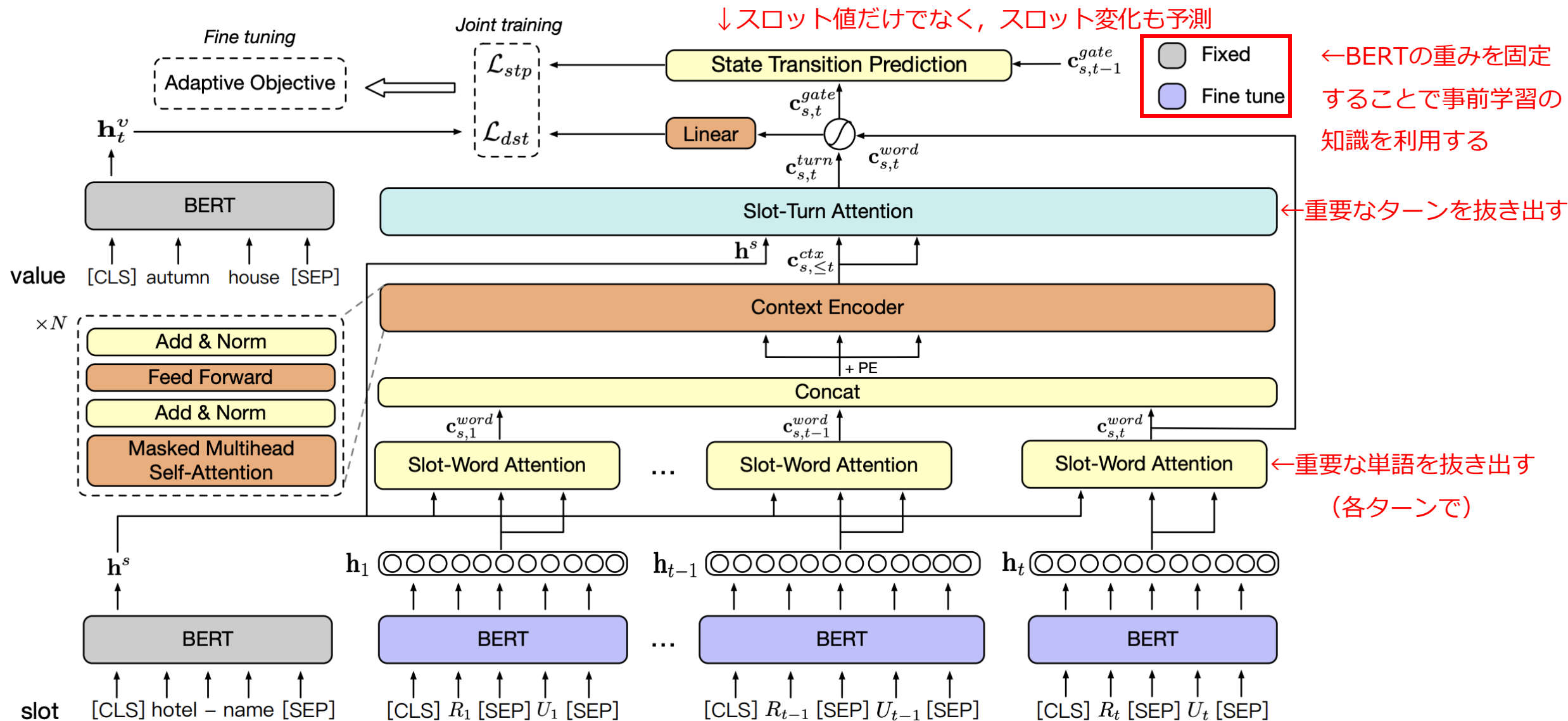


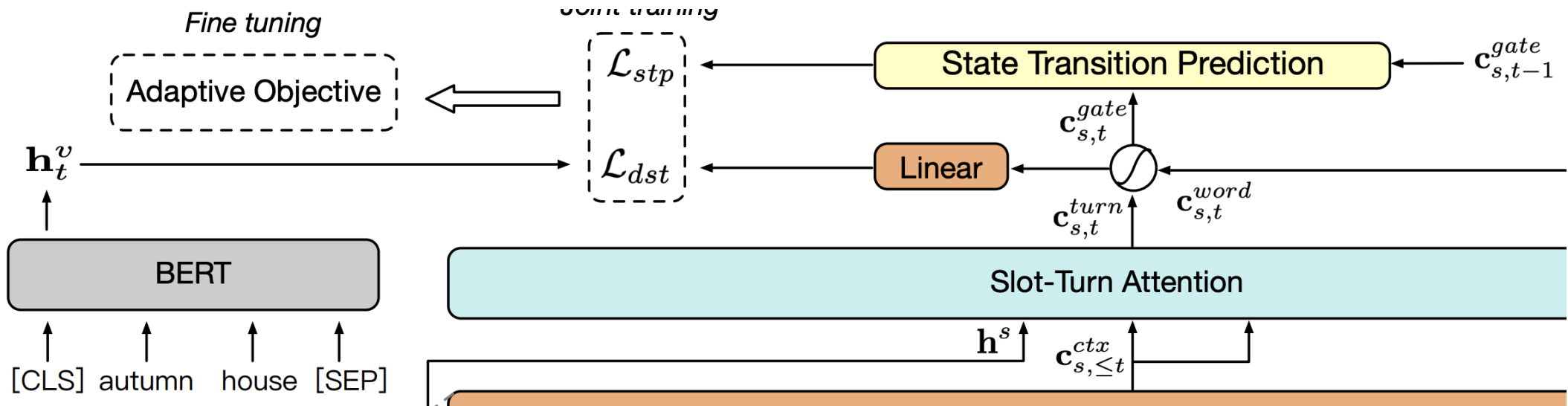
## 概要

単語レベルとターンレベルのAttentionと事前学習モデルBERTを利用したDSTモデルと、スロットの予測難易度に基づいた適応的誤差を提案  
ACL2020のWOZタスクではトップクラスの精度を示した

## 新規性

1. 階層的な単語レベル, ターンレベルのAttention構造で効率的に入力を変換
2. 重みをfixした事前学習BERTを利用することで, データセット外の知識を利用
3. 適応的誤差を利用することで, 予測が難しいスロットの学習を促進





↓出力単語の分散表現をBERTの表現に近づける

$$g_{s,t} = \sigma(\mathbf{W}_g \odot [\mathbf{c}_{s,t}^{word}; \mathbf{c}_{s,t}^{turn}])$$

$$p(v_t | U_{\leq t}, R_{\leq t}, s) = \frac{\exp(-\|o_{s,t} - h_t^v\|_2)}{\sum_{v' \in \mathcal{V}_s} \exp(-\|o_{s,t} - h_t^{v'}\|_2)}$$

$$\mathbf{c}_{s,t}^{gate} = g_{s,t} \otimes \mathbf{c}_{s,t}^{word} + (1 - g_{s,t}) \otimes \mathbf{c}_{s,t}^{turn}$$

$$\mathcal{L}_{dst} = \sum_{s \in \mathcal{S}} \sum_{t=1}^T -\log(p(\hat{v}_t | U_{\leq t}, R_{\leq t}, s))$$

$$\mathbf{o}_{s,t} = \text{LayerNorm}(\text{Linear}(\text{Dropout}(\mathbf{c}_{s,t}^{gate})))$$

(ラベルが複数あると思うが、このモデルは複数出力できるのか??)

## スロットのクラス間の不均衡(imbalance)に対して動的にlossを調整する手法

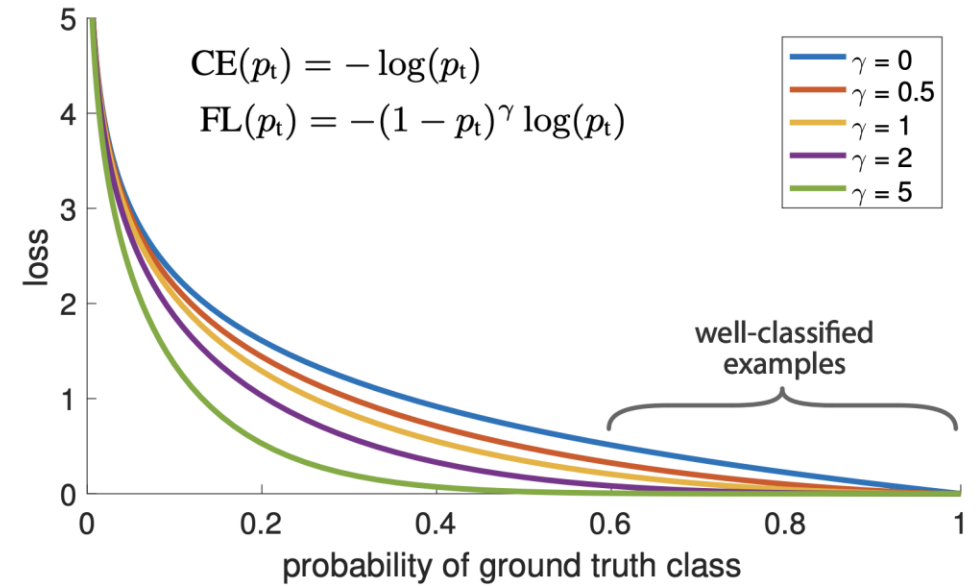
### Focal Loss[10]

$$p_t = \begin{cases} p & \text{if } y = 1 \\ 1 - p & \text{otherwise,} \end{cases}$$

$$\text{CE}(p_t) = -\log(p_t).$$

$$\text{FL}(p_t) = -\underline{(1 - p_t)^\gamma} \log(p_t).$$

よく予測できてる誤差を小さくする



### CHAN's adaptive objective

$$\mathcal{L}_{adapt}(s, v_t) = -\underline{\alpha_s \beta(s, v_t)} \log(p(s, v_t))$$

$$\beta(s, v_t) = (1 - p(s, v_t))^\gamma \quad \alpha_s = \frac{1 - acc_s^{val}}{\sum_{s' \in \mathcal{S}} 1 - acc_{s'}^{val}} \cdot |\mathcal{S}|$$

Focal lossと一緒に

↑ val. datasetを用いてスロット予測難易度を計算

Model	Ontology	MultiWOZ 2.0		MultiWOZ 2.1	
		Joint (%)	Slot (%)	Joint (%)	Slot (%)
DSTreader (Gao et al., 2019b)	×	39.41	-	36.40*	-
GLAD-RCFS (Sharma et al., 2019)	✓	46.31	-	-	-
HyST (Goel et al., 2019)	✓	42.33	-	38.10*	-
TRADE (Wu et al., 2019)	×	48.60	96.92	45.60*	-
DST-QA (Zhou and Small, 2019)	×	51.44	97.24	51.17	97.21
SOM-DST (Kim et al., 2019)	×	51.38	-	52.57	-
SUMBT (Lee et al., 2019)	✓	48.81 <sup>†</sup>	97.33 <sup>†</sup>	52.75 <sup>‡</sup>	97.56 <sup>‡</sup>
DST-picklist (Zhang et al., 2019)	✓	-	-	53.30	-
Our Model	✓	<b>52.68</b>	<b>97.69</b>	<b>58.55</b>	<b>98.14</b>

MultiWOZ 2.1においてjoint精度をTRADEから12.95%向上  
(ただしスロット値候補を事前定義しているのでfairではない)

## 学習方法の比較

Model	MultiWOZ 2.1
Our Model	<b>58.55</b>
- state transition prediction	57.86 (-0.69)
- adaptive objective fine-tuning	57.45 (-1.10)
- above two (only CHAN) <sup>†</sup>	57.00 (-1.55)
Our Model (FL ( $\alpha=1, \gamma=2$ )) <sup>‡</sup>	58.10 (-0.45)

1. 不必要な予測をすることで精度が向上
2. Focal Lossよりも提案誤差の方が良い

## モデル構造自体の比較

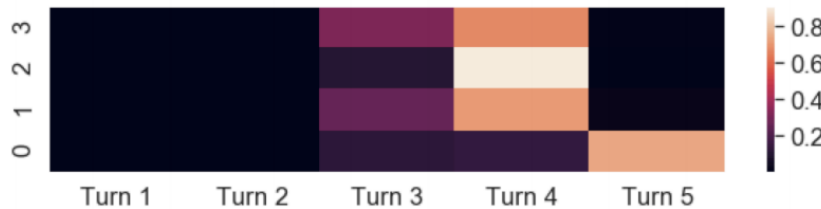
Model	MultiWOZ 2.1
CHAN	<b>57.00</b>
- global-local fusion gate	56.76 (-0.24)
- slot-turn attention	56.85 (-0.15)
- context encoder	55.28 (-1.72)
- above three <sup>†</sup>	50.28 (-6.72)

1. Transformer型のcontext encoderは影響が大
2. 3つとも抜くと大幅に精度が低下

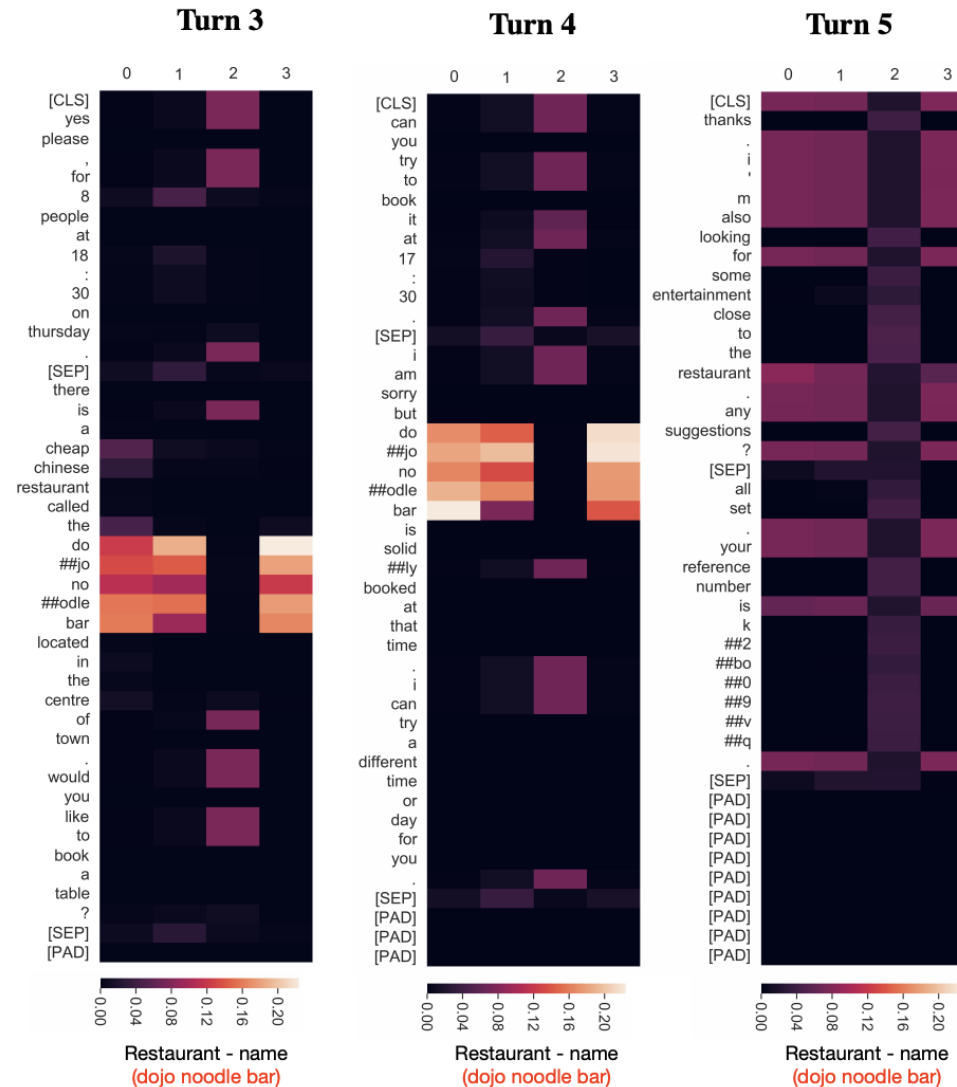
## Dialogue Example

- Turn 1,** U: i am looking for a cheap restaurant in the center of the city.
- Turn 2,** R: do you have any specific type of food you would like?  
U: no, i ' m not picky as long as the prices are low.
- Turn 3,** R: there is a cheap chinese restaurant called the **dojo noodle bar** located in the centre of town. would you like to book a table?  
U: yes please, for 8 people at 18 : 30 on thursday.
- Turn 4,** R: i am sorry but **dojo noodle bar** is solidly booked at that time. i can try a different time or day for you.  
U: can you try to book it at 17 : 30 .
- Turn 5,** R: all set. your reference number is k2bo09vq.  
U: thanks. i ' m also looking for some entertainment close to the restaurant. any suggestions?

## Slot-turn Attention



## Slot-word Attention



- 単語とターンレベルのAttention, Transformer型の中間層, そして事前学習モデルのBERTを利用したDST予測モデルを提案
- また, 学習を促進させるためにスロット変化を予測する副タスクも同時に最適化し, スロット間の不均衡問題を解消するために適応的誤差を利用
- MultiWOZにおいて, 従来手法を大きく上回る結果を示した
  1. MultiWOZ2.0 : TRADE + 4.08%
  2. MultiWOZ2.1 : TRADE + 12.95%



- [1] Chen, Yun-Nung, Asli Celikyilmaz, and Dilek Hakkani-Tur. "Deep learning for dialogue systems." *Proceedings of the 27th International Conference on Computational Linguistics: Tutorial Abstracts*. 2018.
- [2] ERIC, Mihail, et al. **MultiWOZ 2.1: A Consolidated Multi-Domain Dialogue Dataset with State Corrections and State Tracking Baselines.** *arXiv preprint arXiv:1907.01669*, 2019.
- [3] WU, Chien-Sheng, et al. **Transferable multi-domain state generator for task-oriented dialogue systems.** *arXiv preprint arXiv:1905.08743*, 2019.
- [4] LIU, Bing, et al. **Dialogue learning with human teaching and feedback in end-to-end trainable task-oriented dialogue systems.** *arXiv preprint arXiv:1804.06512*, 2018.
- [5] Vinyals, Oriol, Meire Fortunato, and Navdeep Jaitly. "Pointer networks." *Advances in neural information processing systems*. 2015.
- [6] SEE, Abigail; LIU, Peter J.; MANNING, Christopher D. **Get to the point: Summarization with pointer-generator networks.** *arXiv preprint arXiv:1704.04368*, 2017.
- [7] HASHIMOTO, Kazuma, et al. **A joint many-task model: Growing a neural network for multiple nlp tasks.** *arXiv preprint arXiv:1611.01587*, 2016.
- [8] HU, Jiaying, et al. **SAS: Dialogue State Tracking via Slot Attention and Slot Information Sharing.** In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2020.
- [9] SHAN, Yong, et al. **A Contextual Hierarchical Attention Network with Adaptive Objective for Dialogue State Tracking.** *arXiv preprint arXiv:2006.01554*, 2020.
- [10] LIN, Tsung-Yi, et al. **Focal loss for dense object detection.** In: *Proceedings of the IEEE international conference on computer vision*. 2017.

人間に、愛を。  
未来に、AIを。

Arithmer 株式会社

〒106-6040

東京都港区六本木一丁目6番1号 泉ガーデンタワー 38/40F(受付)

03-5579-6683

<https://arithmer.co.jp/>

Arithmer

