

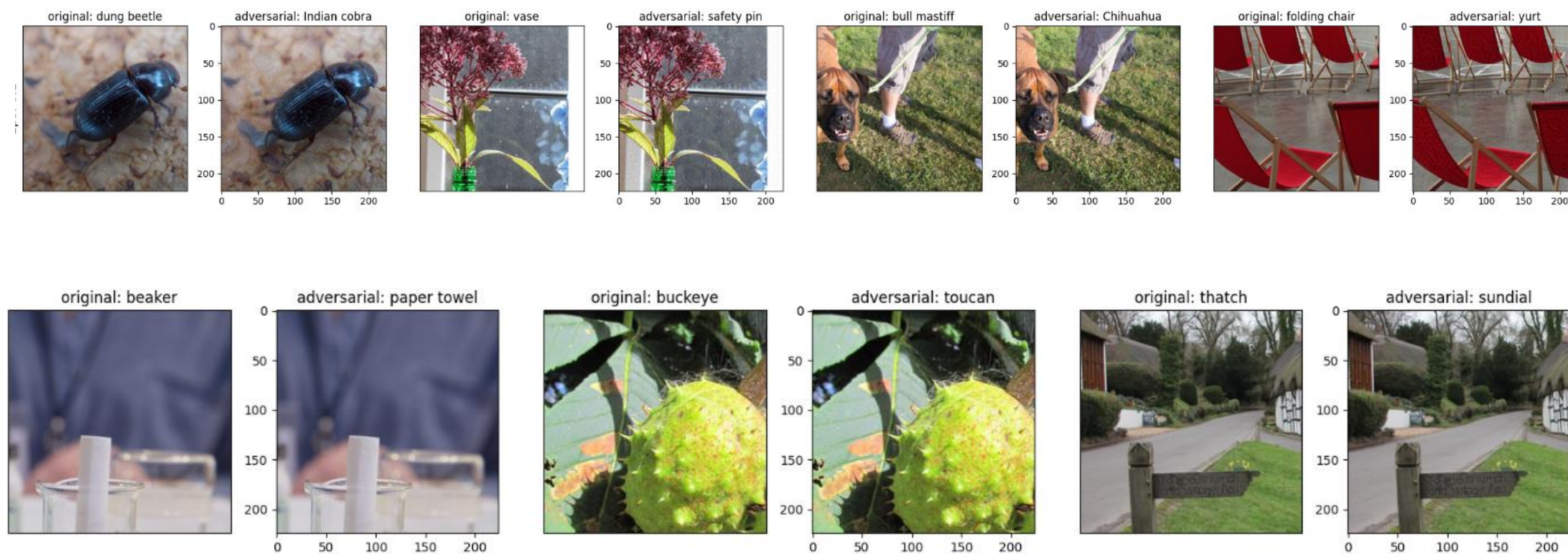
总结

黄飞虎

2021.1.27

学习情况

看了李宏毅的视频， 完成了实验六和实验七



```
rate 0.9500 epoch 0: train loss: 0.4837, acc 0.8658 valid loss: 1.1188, acc 0.7939
rate 0.9500 epoch 1: train loss: 0.4743, acc 0.8683 valid loss: 1.1312, acc 0.7933
rate 0.9500 epoch 2: train loss: 0.4829, acc 0.8688 valid loss: 1.1095, acc 0.7991
rate 0.9500 epoch 3: train loss: 0.4864, acc 0.8708 valid loss: 1.1138, acc 0.7939
rate 0.9500 epoch 4: train loss: 0.4948, acc 0.8678 valid loss: 1.0869, acc 0.7968
rate 0.9025 epoch 0: train loss: 0.5941, acc 0.8395 valid loss: 1.1754, acc 0.7790
rate 0.9025 epoch 1: train loss: 0.5712, acc 0.8436 valid loss: 1.1884, acc 0.7799
rate 0.9025 epoch 2: train loss: 0.5843, acc 0.8413 valid loss: 1.1644, acc 0.7802
rate 0.9025 epoch 3: train loss: 0.6091, acc 0.8415 valid loss: 1.2055, acc 0.7802
rate 0.9025 epoch 4: train loss: 0.5870, acc 0.8420 valid loss: 1.2056, acc 0.7776
rate 0.8574 epoch 0: train loss: 0.6915, acc 0.8070 valid loss: 1.2395, acc 0.7603
rate 0.8574 epoch 1: train loss: 0.6875, acc 0.8145 valid loss: 1.2149, acc 0.7662
rate 0.8574 epoch 2: train loss: 0.6969, acc 0.8090 valid loss: 1.2088, acc 0.7638
rate 0.8574 epoch 3: train loss: 0.7253, acc 0.8030 valid loss: 1.1934, acc 0.7638
rate 0.8574 epoch 4: train loss: 0.6902, acc 0.8107 valid loss: 1.2210, acc 0.7609
rate 0.8145 epoch 0: train loss: 0.8220, acc 0.7731 valid loss: 1.3012, acc 0.7344
rate 0.8145 epoch 1: train loss: 0.7905, acc 0.7788 valid loss: 1.2771, acc 0.7327
rate 0.8145 epoch 2: train loss: 0.8290, acc 0.7749 valid loss: 1.3147, acc 0.7294
rate 0.8145 epoch 3: train loss: 0.8192, acc 0.7734 valid loss: 1.2613, acc 0.7347
rate 0.8145 epoch 4: train loss: 0.8123, acc 0.7768 valid loss: 1.3132, acc 0.7344
```

```
? custom_small_rate_0.95.bin
? custom_small_rate_0.9025.bin
? custom_small_rate_0.7737809374999999.bin
? custom_small_rate_0.8145062499999999.bin
? custom_small_rate_0.8573749999999999.bin
```

论文情况

通常任务型对话系统都是基于pipeline的方式实现的，整个pipeline由五个模块组成：语音识别；自然语言理解；对话管理；自然语言生成；语音合成。现在越来越多的产品还融入了知识库，主要是在对话管理模块引入。在这里除了语音识别和语音合成模块不属于自然语言处理范畴且属于可选项之外，其他的三个模块都是必要的。

自然语言理解（NLU）：主要作用是对用户输入的句子或者语音识别的结果进行处理，提取用户的对话意图以及用户所传递的信息。

对话管理（DM）：对话管理分为两个子模块，对话状态追踪（DST）和对话策略学习（DPL），其主要作用是根据NLU的结果来更新系统的状态，并生成相应的系统动作。

自然语言生成（NLG）：将DM输出的系统动作文本化，用文本的形式将系统的动作表达出来。

DST的常用方法主要有三种：基于规则的方法，生成式模型，判别式模型，目前来看判别式模型的表现最好，也是当前研究的最多的方向。

Neural Belief Tracker: Data-Driven Dialogue State Tracking

下载链接: <https://arxiv.org/pdf/1606.03777.pdf>

本文的工作

belief tracker是现代口语对话系统的核心组件之一，它可以在对话的每个步骤中估算用户的目标，但是，大多数当前方法很难扩展到更大，更复杂的对话域。这是由于它们对以下方面的依赖：

1. 需要大量带注释的训练数据的口语理解模型
2. 手工制作的词典，用于捕获用户语言的某些语言变化

提出了一种新颖的Neural Belief Tracking (NBT) 框架，该框架通过基于表示学习的最新进展来克服这些问题。

介绍

下图中的示例显示了三轮对话中每个用户语句后的真实状态，从该示例可以看出，DST模型依赖于标识用户话语中的本体。

User: I'm looking for a cheaper restaurant
`inform(price=cheap)`
System: Sure. What kind - and where?
User: Thai food, somewhere downtown
`inform(price=cheap, food=Thai, area=centre)`
System: The House serves cheap Thai food
User: Where is it?
`inform(price=cheap, food=Thai, area=centre); request(address)`
System: The House is at 106 Regent Street

Figure 1: Annotated dialogue states in a sample dialogue. Underlined words show rephrasings which are typically handled using semantic dictionaries.

下图给出了一个针对三个槽值对的字典的示例（传统的做法是建语义词典），我们称其为**非词化**(delexicalisation)的这种方法显然无法扩展到更大，更复杂的对话域。如意大利语和德语这种词汇和形态丰富的语言。

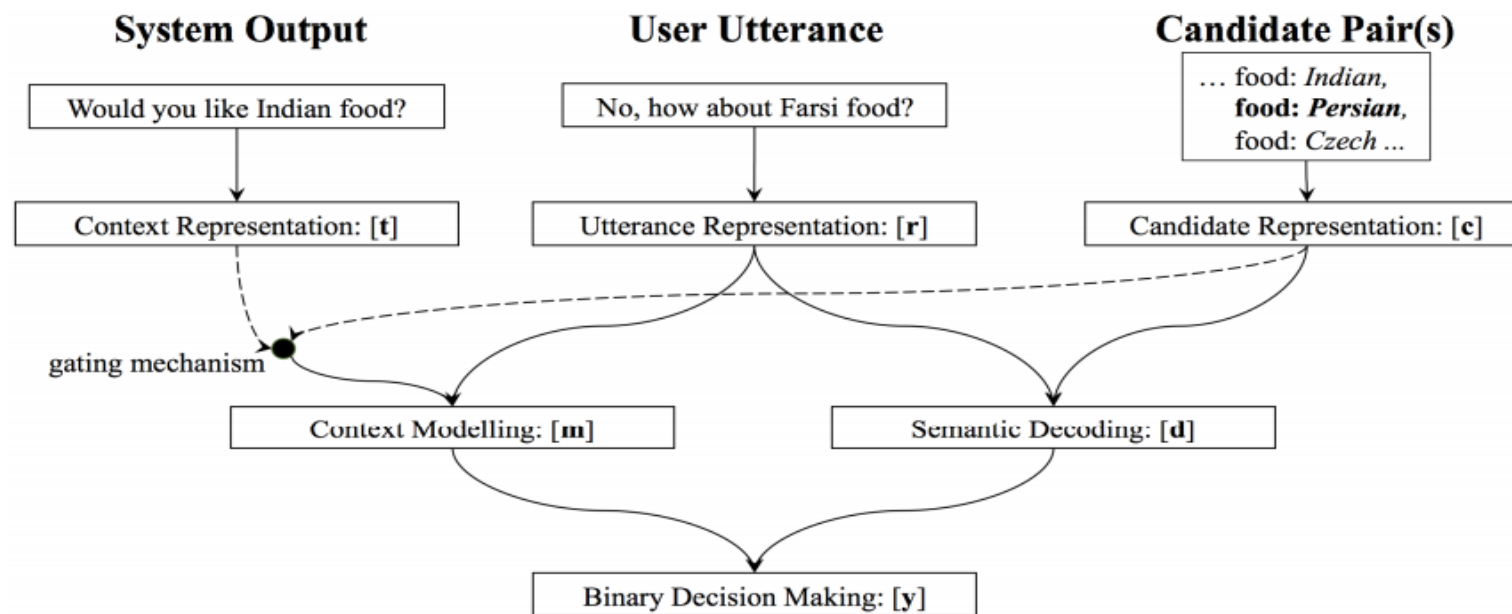
FOOD=CHEAP: [affordable, budget, low-cost, low-priced, inexpensive, cheaper, economic, ...]
RATING=HIGH: [best, high-rated, highly rated, top-rated, cool, chic, popular, trendy, ...]
AREA=CENTRE: [center, downtown, central, city centre, midtown, town centre, ...]

Figure 2: An example semantic dictionary with rephrasings for three ontology values in a *restaurant search* domain.

本文提出的工作的主要动机是克服影响以前的信念跟踪模型的限制。NBT模型通过以下方式有效地从可用数据中学习

- 利用预训练词向量中的语义信息来解决词汇/形态上的歧义
- 最大化本体值之间共享的参数数量
- 具有学习领域特定释义和其他变体的灵活性，这使得依靠精确匹配和去词缀化作为一种可靠的策略是不可行的

模型结构如右图所示



表征学习

这里分别使用了两个模型来得到文本的表征：NBT-DNN和NBT-CNN，所有的表征学习都是建立在词向量上。

模型的输入包括系统的前一个对话动作，用户的输入 u 和一个候选的slot-value对。输入 u 的词向量分别是 u_1, \dots, u_k 。 V_i^n 是 n 个词向量的拼接。

$$v_i^n = u_i \oplus \dots \oplus u_{i+n-1}$$

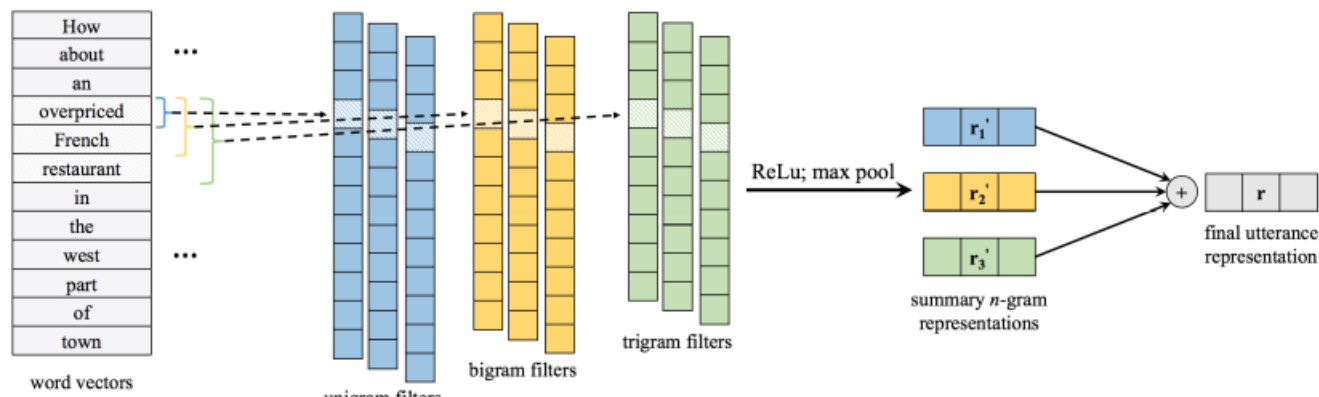
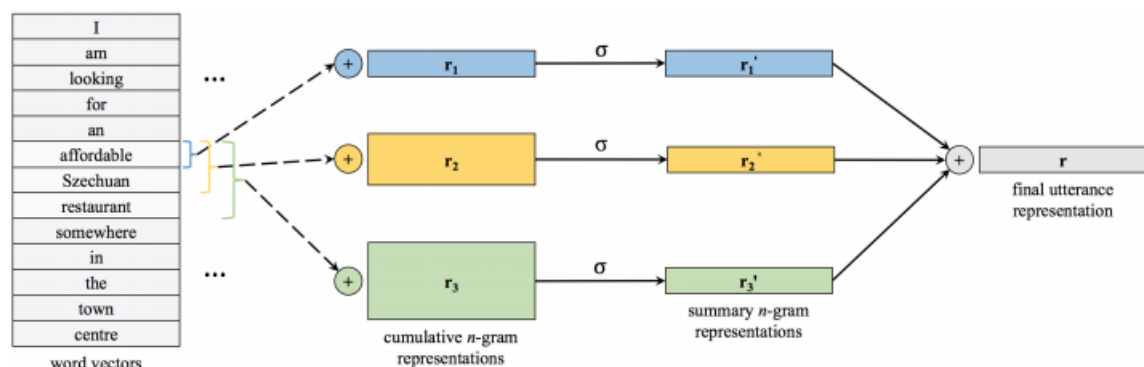
$$r_n = \sum_{i=1}^{k_u-n+1} v_i^n$$

$$r'_n = \sigma(W_n^s r_n + b_n^s)$$

$$r = r'_1 + r'_2 + r'_3$$

$$R_n = F_n^s m_n$$

$$r'_n = \text{maxpool}(\text{ReLU}(R_n + b_n^s))$$



Context Modelling

当用户询问时，仅从当前用户的输入还不足以抽取意图，belief tracker应该考虑对话的上下文，特别是上一句系统的动作。论文提出了两种动作：System Request和System Confirm。

第一种情景是，系统对一个特定的slot发出提问，用户一般会给出具体的value。第二种是系统询问用户，某个slot-value是否正确，用户一般只会回答对或错。这两个场景应分别计算。 t_q 是request的参数， (t_s, t_v) 是confirm的参数。 t_q, t_s, t_v 都是slot/value的词向量，多个词时直接求和得到。通过系统动作，候选对 (c_s, c_v) 作为一个门，控制输入表征r的信息输出（般情况下系统要么是请求，要么是确认，那么此时 $t(q)$ 为零向量或者 $(t(s), t(v))$ 是零向量。）

$$\begin{aligned}\mathbf{m}_r &= (\mathbf{c}_s \cdot \mathbf{t}_q)\mathbf{r} \\ \mathbf{m}_c &= (\mathbf{c}_s \cdot \mathbf{t}_s)(\mathbf{c}_v \cdot \mathbf{t}_v)\mathbf{r}\end{aligned}$$

Binary Decision Maker: 最后的二分类决策层。 $\boldsymbol{\varphi}_{\text{dim}}(\mathbf{x}) = \boldsymbol{\sigma}(\mathbf{W}\mathbf{x} + \mathbf{b})$ 将输入 \mathbf{x} 映射到维度为size的向量，softmax二分类，完成slot-value对的存在检测。

$$\mathbf{y} = \phi_2(\phi_{100}(\mathbf{d}) + \phi_{100}(\mathbf{m}_r) + \phi_{100}(\mathbf{m}_c))$$

Belief State Update Mechanism

论文提出了一种简单的belief state的更新机制：先估计当前轮对话的slot-value，再更新历史记录。

对于第 t 轮对话， sys_{t-1} 表示前一个系统动作， h^t 是ASR输出的结果假设， h_i^t 是N-best list中的第 i 个， s 是slot， v 是value，NBT模型需要估计 (s, v) 在用户的口语输入中的概率

$$\mathbb{P}(s, v | h^t, sys^{t-1}) = \sum_{i=1}^N p_i^t \mathbb{P}(s, v | h_i^t, sys^t)$$

对于当前和历史对话的belief state更新，引入一个权重系数 λ ：

$$\mathbb{P}(s, v | h^{1:t}, sys^{1:t-1}) = \lambda \mathbb{P}(s, v | h^t, sys^{t-1}) + (1 - \lambda) \mathbb{P}(s, v | h^{1:t-1}, sys^{1:t-2})$$

实验

实验仍然是task-oriented的对话任务，数据集有两个：DSTC2和WOZ 2.0。

评估

引入了两个关键的评估指标

- 1: 目标(“联合目标精度”):所有用户搜索目标约束都被正确识别的对话比例;
- 2: 请求:类似的，对话的比例是在用户对信息的请求被正确识别的地方。

Baseline Models

- 1.实现著名的竞争性非词汇化的基线系统的数据集的模型
- 2.用特定任务的语义字典补充的相同的基线模型(由基线系统创建者产生)

实验结果

DST Model	DSTC2		WOZ 2.0	
	Goals	Requests	Goals	Requests
Delexicalisation-Based Model	69.1	95.7	70.8	87.1
Delexicalisation-Based Model + Semantic Dictionary	72.9*	95.7	83.7*	87.6
NEURAL BELIEF TRACKER: NBT-DNN	72.6*	96.4	84.4*	91.2*
NEURAL BELIEF TRACKER: NBT-CNN	73.4*	96.5	84.2*	91.6*

可以看到，NBT-DNN和NBT-CNN都能超过基于语义词典的模型，当然NBT-CNN多了不同n-grams特征的权重学习，会更好一点。论文还做了不同词向量对结果影响的实验。

Word Vectors	DSTC2		WOZ 2.0	
	Goals	Requests	Goals	Requests
XAVIER (No Info.)	64.2	81.2	81.2	90.7
GloVe	69.0*	96.4*	80.1	91.4
Paragram-SL999	73.4*	96.5*	84.2*	91.6

Thank you