中国科学: 信息科学 2021 年 第 51 卷 第 11 期: 1853–1866
SCIENTIA SINICA Informationis
论文

基于隐式网络和显式网络相似性学习的零样本意图识别

孙鹏飞, 欧阳亚文, 戴新宇*, 张文明

计算机软件新技术国家重点实验室 (南京大学), 南京 210023
* 通信作者. E-mail: daixinyu@nju.edu.cn

收稿日期: 2020–08–25; 接受日期: 2020–10–08; 网络出版日期: 2021–11–10

国家自然科学基金 (批准号: 61772261, 61672277) 和江苏省自然科学基金 (优秀青年基金项目) (批准号: BK20170074) 资助项目

摘要 意图识别是对话系统的一个重要组成部分. 现有的工作主要集中在使用充足的标记数据进行意图识别. 然而, 这些方法不能识别训练数据中不存在的意图. 为了解决这个问题, 我们提出了一种基于隐式网络和显式网络的相似性学习模型, 用于零样本意图识别, 该模型能够从词级和句子级学习用户话术和意图描述之间的相似性. 为了增强意图的表示, 我们引入槽位类型作为意图描述. 并依据表达方式的不同将意图分为显式意图和隐式意图, 分别从词级和句子级构建显式网络和隐式网络. 同时, 为了更好地结合这两部分信息, 我们还设计了关系层来融合不同层级的信息. 在两个基准数据集上的实验结果表明, 我们的模型明显优于现有的最先进的模型, 并展示了从词级和句子级同时学习相似性的有效性.

关键词 零样本意图识别, 隐式网络, 显式网络, 关系层, 选择门

1 引言

意图识别是对话系统中的一个重要组成部分, 旨在将用户的话术分到预先定义的意图类别中 (例如, 询问天气和预订餐馆等). 尽管当前的深度学习方法在意图识别任务中取得了显著的成果 [1,2]. 但是, 它们的表现过分依赖于训练数据的质量和规模. 而在实践中, 为所有意图标注足够的训练数据集是一项非常困难且劳动密集的任务. 此外, 新兴意图每天都会出现, 对于不在训练数据集中的新兴意图, 现有的意图识别模型难以作出正确的预测.

为了解决这些问题, 人们经常使用零样本学习 (zero-shot learning) 的方法. 零样本意图识别旨在识别那些没有训练样本的新兴意图. 一些研究主要集中在借助外部资源将知识从现有意图转移到新兴意图, 以实现对新兴意图的推断和预测. 其中最常见的方法: 人工定义意图属性 [3]、人工定义的领域

引用格式: 孙鹏飞, 欧阳亚文, 戴新宇, 等. 基于隐式网络和显式网络相似性学习的零样本意图识别. 中国科学: 信息科学, 2021, 51: 1853–1866, doi: 10.1360/SSI-2020-0266
Sun P F, Ouyang Y W, Dai X Y, et al. Similarity learning with implicit-network and explicit-network for zero-shot intent detection (in Chinese). Sci Sin Inform, 2021, 51: 1853–1866, doi: 10.1360/SSI-2020-0266

表 1 用户的意图表达与相对应的槽位类型 a)

Table 1 The user's expression of intent and the corresponding slot types

Intent	Slot types	Utterance with explicit intent	Utterance with implicit intent
AddToPlaylist	Music_item, playlist...	Add this tune to the duetos playlist.	Include the album by arthur rhames in urban poet.
BookRestaurant	Cuisine, timeRange...	Book a table at a top-rated brasserie in Pakistan.	Please get seating at bouchon in tonga for jimmie and chasity howard.
GetWeather	Country, city...	What is the forecast for temperate weather in bellechester?	Is it nice now in madawaska?

a) The two utterance expressions of each row are distinct but have the same intent. Explicit intent means that users clearly indicate their intent requirements in utterances, and include intent related words. Implicit intent means that users do not express their intent and need to analyze their potential intent to infer their real intent.

本体 [4]、标签本体 [5,6]. 然而, 为每一个新兴意图注释额外的信息既费时又费力. 相反, 研究人员探索了一些基于相似性学习的零样本意图识别方法 [7,8]. 他们利用神经网络将意图标签和话术映射到潜在的语义空间, 然后度量它们的相似性. 但是, 词的语义在不同语境中是动态变化的, 使得这些方法经常会遇到语义漂移问题. 这样导致学习到的词的语义表征与词的原始语义表征有一定的偏差, 致使模型缺乏泛化能力.

近年来, 有研究指出槽位填充任务和意图识别任务的联合建模有助于提高意图识别效果 [9,10]. 这说明了槽位类型可以为意图识别提供线索. 因此, 使用槽位类型来增强意图表示是非常自然和可信的. 如表 1 所示, 不同的意图与不同的槽位类型相关联. 例如, 意图 AddToPlaylist 与槽位类型 "music_item, playlist,..." 相关联. 同样, 意图 BookRestaurant 和 GetWeather 也有相对应的槽位类型. 因此, 槽位类型被用作意图描述是可行的, 且有助于意图识别.

除了引入槽位类型作为意图描述之外, 我们还根据表达方式的不同将意图分为显式意图和隐式意图 [11]. 显式意图是指用户在话术中明确指出他们的意图需求, 并包含与意图相关的词语. 隐式意图意味着用户没有明确表达他们的意图, 需要分析他们的潜在意图, 来推断他们的真实意图. 为了直观地说明, 在表 1 中列举一些例子, 表格中的每一行是同一意图对应的两种不同的表达. 例如, 显式话术 "add this tune to the duetos playlist" 包含与意图相关的单词 (如 "add", "playlist"), 明确地表达了用户想要在播放列表中添加内容的意图. 而在另一个隐式意图话术 "include the album by arthur rhames in urban poet" 中, 用户没有明确地表达想要把专辑添加到播放列表的意图, 但我们可以推断出用户可能想要将专辑添加到播放列表中. 从上面的例子可以看出, 同一意图有两种不同的表达方式. 因此, 如何设计一个模型来识别这两种不同表达的意图就成为一个挑战.

针对上述问题, 本文提出了一种基于隐式网络和显式网络相似性学习的模型 (implicit-network and explicit-network, INTENT), 用于零样本意图识别. 为了避免注释额外信息的负担, 我们首先尝试使用槽位类型作为意图描述, 这可以提供更完整的意图语义并增强意图的表示. 同时, 为了更好地对显式意图和隐式意图进行建模, 我们分别构建了显式网络和隐式网络. 对于显式网络, 我们主要使用基于交互建模的方法来捕获词级信息. 通过构建交互矩阵提取语义焦点, 可以合理地对上下文信息进行建模. 而对于隐式网络, 我们使用编码器 – 比较框架 (encoder-compare framework) 来学习单词的重要性, 并捕捉句子级别的信息, 这样可以降低不重要词带来的影响. 为了更好地融合多层级的信息, 我们提出了关系层, 它可以从多个角度自动提取有意义的信息并动态地进行集成. 具体来说, 我们探索了两种关系层, 包括: (1) 选择门 (switch gate), 使用门控机制来决定是使用隐式网络还是显式网络来进

行最终的预测. 此外, 值得注意的是, 选择门可以作为二进制开关进行决策, 也可以作为概率混合开关进行融合. (2) 多层感知器 (multilayer perceptron, MLP), 这里使用 MLP 来学习包含词级和句子级信息的丰富表示. 最后, 我们在两个公开的数据集 (SNIPS [12] 和 ATIS [13]) 上进行了实验. 实验结果表明, INTENT 模型在新兴意图识别方面取得了不错的效果.

综上所述, 我们的主要贡献如下:

- 我们提出了一种用于零样本意图识别的新模型 INTENT, 该模型可以对显式意图和隐式意图进行建模.
- 据我们所知, 我们的模型是第 1 个使用槽位类型作为意图描述的模型, 以提供更完整的意图语义并增强意图的表示.
- 我们深入研究了不同的关系层, 并进行了大量的实验来表明关系层的有效性. 同时, 在两个基准数据集上的实验验证了 INTENT 模型的有效性和优越性.

2 相关工作

对话系统受到越来越多的关注. 意图识别和槽位填充是自然语言理解的重要环节, 是面向任务对话系统的核心模块. 有些工作将意图识别和槽位填充分开处理, 把意图识别作为文本分类问题. 因此, 无需任何特征工程, 就可以使用神经网络来学习低维和连续的话术表示. 例如, 文献 [14] 设计了一个基于卷积神经网络 (convolutional neural network, CNN) 的句子分类模型. 文献 [15] 成功地将循环神经网络 (recurrent neural network, RNN) 和长短期记忆模型 (long short-term memory, LSTM) 应用于话术分类. 文献 [16] 采用基于注意力的卷积神经网络对上下文信息进行有效编码, 从而实现话术分类. 这些工作在意图识别任务中取得了不错的效果, 但是在零样本意图识别任务中效果不佳.

最近, 有些工作集中于将两个子任务意图识别和槽位填充联合优化, 以达到协同的效果 [17,18]. 文献 [10] 提出了一种基于胶囊的神经网络模型, 该模型通过一种动态路由机制完成槽位填充和意图识别. 文献 [2] 引入了 SF-ID 网络来建立意图识别和槽位填充的直接连接, 以帮助它们相互促进. 尽管这些模型表现良好, 但它们仍取决于标注数据的数量和质量, 这限制了模型对新兴意图识别的可伸缩性.

为了解决新兴意图识别的挑战, 许多研究者探索零样本意图识别的方法. 文献 [8] 通过语义空间揭示了类别和话术之间的联系. 而该语义空间是通过在大量搜索引擎查询日志数据上训练的深度神经网络学习的. 文献 [3] 使用人工定义的意图属性作为一种先验知识, 借助先验知识将其从可见类别转移到新兴类别, 以实现对新兴类别的预测. 文献 [4] 利用领域的属性信息增强领域的表示, 并将话术和领域投射到相同的语义空间中, 来实现新兴类别的识别. 然而, 获取这些意图和领域对应的属性是非常困难的, 且不现实的. 文献 [6] 提出了一种基于词嵌入和目标域本体描述的零样本学习方法, 可以实现零样本语义解析器. 文献 [5] 扩展了这项工作, 提出了一个在线自适应策略, 该策略只需要轻量级的监督信息即可. 文献 [7] 提出基于卷积深度语义匹配模型 (convolutional deep structured semantic model, CDSSM) 进行零样本意图分类. 该模型联合学习了意图和相关话术的表征, 通过这些表征建立可见和不可见意图之间的语义关系, 从而获得了更为有力的结果. 文献 [19] 使用胶囊神经网络, 将已知类别的知识转移到未知类别中, 从而进行零样本意图识别. 虽然, 这些方法取得了不错的效果, 但是它们忽略了词嵌入只能提供有限的语义信息, 这很容易导致语义漂移问题. 我们的工作通过引入额外的槽位类型增强意图的语义表示, 并在话术和意图描述之间建立更细粒度的交互以提高性能, 从而避免上述问题.

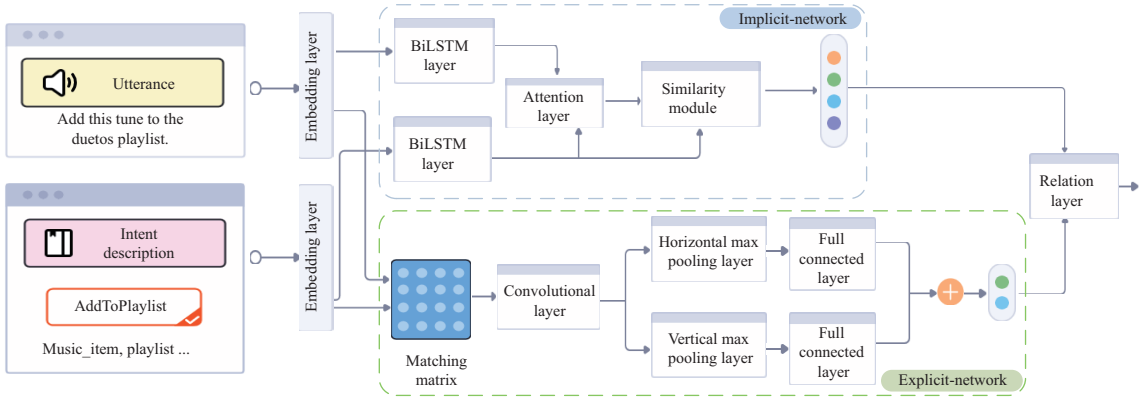


图 1 (网络版彩图) INTENT 模型由隐式网络 (上) 和显式网络 (下) 组成. 上面核心模块是注意力双向 LSTM, 下面核心模块是匹配矩阵上的 CNN. 在训练过程中, 两个模块的参数联合优化

Figure 1 (Color online) INTENT model is composed of the implicit-network (top) and the explicit-network (bottom). The top core module is attention BiLSTM, and the bottom core module is CNN on matching matrix. The parameters of the two modules are jointly optimized during the training process

3 方法

本节将详细介绍用于零样本意图识别的方法. 图 1 提供了 INTENT 模型架构的概述. 该模型由 3 个部分组成: (1) 隐式网络, 它利用基于注意力机制的 RNN 作为编码器 – 比较框架, 以便从意图 – 话术对中捕获隐式表示 (句子级). (2) 显式网络, 它在词匹配矩阵上使用 CNN 从意图 – 话术对中提取显式表示 (词级). (3) 关系层, 它融合显式表示和隐式表示, 以建模两个网络之间的关系. 下面的章节给出了框架的细节.

3.1 隐式网络

为了从意图 – 话术对中获取隐式表示 (句子级), 我们构建了一个基于注意力机制的 RNN 作为编码器 – 比较框架. 这是因为意图描述由多个槽位类型组成, 不同的槽位类型对应话术中不同的词, 这使得意图描述中的每个词对话术编码的贡献不同. 基于这一观察结果, 我们用双向长短期记忆网络 [20] (bi-directional long short-term memory, BiLSTM) 对话术和意图描述进行编码. 然后, 运用注意机制来捕捉话术中重要的信息. 最后, 在学到的语义向量空间中计算它们之间的相似性. 更多细节解释如下.

3.1.1 话术编码

给一个话术 $u = \{w_1^u, w_2^u, \ldots, w_i^u, \ldots, w_T^u\}$ 包含 T 个词, 经过映射得到对应的词向量表示 $e = \{\boldsymbol{e}_1^u, \boldsymbol{e}_2^u, \ldots, \boldsymbol{e}_t^u, \ldots, \boldsymbol{e}_T^u\}$. 随后, 我们使用 BiLSTM 来学习话术中每个词的表示:

$$\overrightarrow{\boldsymbol{h}_t^u} = \overrightarrow{\mathrm{LSTM}}(\boldsymbol{e}_t^u, \overrightarrow{\boldsymbol{h}_{t-1}^u}), \tag{1}$$

$$\overleftarrow{\boldsymbol{h}_t^u} = \overleftarrow{\mathrm{LSTM}}(\boldsymbol{e}_t^u, \overleftarrow{\boldsymbol{h}_{t-1}^u}), \tag{2}$$

$$\boldsymbol{h}_t^u = [\overrightarrow{\boldsymbol{h}_t^u}; \overleftarrow{\boldsymbol{h}_t^u}]. \tag{3}$$

对于每个词, 我们将前向隐藏状态 $\overrightarrow{\boldsymbol{h}_t^u}$ 与后向隐藏状态 $\overleftarrow{\boldsymbol{h}_t^u}$ 拼接得到隐藏状态 $\boldsymbol{h}_t^u$. 这样, 对于包含 T 个词的话术, 得到整个话术的隐藏表示 $\boldsymbol{h}^u = (\boldsymbol{h}_1^u, \boldsymbol{h}_2^u, \ldots, \boldsymbol{h}_t^u, \ldots, \boldsymbol{h}_T^u)$ 作为话术的表示.

3.1.2 意图描述编码

给定一个意图 y 及其对应的意图描述 $d=\{s_1,s_2,\ldots,s_i,\ldots,s_{N_s}\}$, 其中 N_s 是意图描述中的槽位类型的数量. 第 i 个槽位类型 s_i 可表示为 $s_i=\{w_1^{s_i},w_2^{s_i},\ldots,w_t^{s_i},\ldots,w_{N_w}^{s_i}\}$, 其中 N_w 是槽位类型的长度, $w_t^{s_i}$ 为槽位类型 s_i 的第 t 个词, $w_t^{s_i}$ 对应的词向量是 $\boldsymbol{e}_t^{s_i}$, 我们使用它作为网络的输入.

$$\overrightarrow{\boldsymbol{h}_t^{s_i}}=\overrightarrow{\mathrm{LSTM}}(\boldsymbol{e}_t^{s_i},\overrightarrow{\boldsymbol{h}_{t-1}^{s_i}}),\tag{4}$$

$$\overleftarrow{\boldsymbol{h}_t^{s_i}}=\overleftarrow{\mathrm{LSTM}}(\boldsymbol{e}_t^{s_i},\overleftarrow{\boldsymbol{h}_{t-1}^{s_i}}),\tag{5}$$

$$\boldsymbol{h}^{s_i}=[\overrightarrow{\boldsymbol{h}_{N_w}^{s_i}};\overleftarrow{\boldsymbol{h}_1^{s_i}}],\tag{6}$$

其中, LSTM 是共享的. 我们将最后一个前向隐藏状态 $\overrightarrow{\boldsymbol{h}_{N_w}^{s_i}}$ 和第一个后向隐藏状态 $\overleftarrow{\boldsymbol{h}_1^{s_i}}$ 拼接起来, 作为槽位类型的语义表示 $\boldsymbol{h}^{s_i}$. 由此, 我们得到意图描述的表示 $\boldsymbol{h}^d=\{\boldsymbol{h}^{s_1},\boldsymbol{h}^{s_2},\ldots,\boldsymbol{h}^{s_i},\ldots,\boldsymbol{h}^{s_{N_s}}\}$.

3.1.3 注意力层

对于意图描述的不同部分, 话术中的每个词通常具有不同的信息量和意义. 因此, 我们提出一个点积注意力层, 该层能够根据意图描述的不同部分来表示话术. 因此, 对于意图描述的每一部分 $\boldsymbol{h}^{s_i}$, 其对应的话术表示 $\boldsymbol{u}_i$ 为

$$\boldsymbol{u}_i=\sum_{j=1}^{T}a_{ij}\boldsymbol{h}_j^u,\tag{7}$$

$$a_{ij}=\frac{\exp(\mathrm{score}(\boldsymbol{h}^{s_i},\boldsymbol{h}_j^u))}{\sum_{k=1}^{T}\exp(\mathrm{score}(\boldsymbol{h}^{s_i},\boldsymbol{h}_k^u))},\tag{8}$$

$$\mathrm{score}(\boldsymbol{h}^{s_i},\boldsymbol{h}_j^u)={\boldsymbol{h}^{s_i}}^{\mathrm{T}}\boldsymbol{h}_j^u,\tag{9}$$

这里, score$(\cdot)$ 为点积运算. a_{ij} 表示话术中的第 j 个词在意图描述 $\boldsymbol{h}^{s_i}$ 的注意力权重.

3.1.4 相似性模块

为了计算话术和意图描述之间的相似性, 我们将话术表示 $\boldsymbol{u}_i$ 与意图描述的每个元素 $\boldsymbol{h}^{s_i}$ 进行点积运算:

$$z_i=\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{h}^{s_i},\quad 1\leqslant i\leqslant N_s,\tag{10}$$

这样, 便从句子级得到了隐式网络的相似性表示: $\boldsymbol{z}_{\mathrm{implicit}}=[z_1,z_2,\ldots,z_{N_s}]$.

3.2 显式网络

在考虑词级相似性时, 我们将其视为一个文本匹配问题. 这是由于基于编码器 – 比较的模型是无法捕获词级的交互信息的. 因此, 受文献 [21] 的启发, 我们将匹配矩阵视为二维图像, 然后利用 CNN 对其进行处理, 以获取不同表达的词级信息. 详情如下.

3.2.1 匹配矩阵

匹配矩阵反映了话术与意图描述之间的词相似度. 因此, 我们计算话术中每个单词 $\boldsymbol{e}_i^u$ 与意图描述中每个单词 $\boldsymbol{e}_j^s$ 之间的余弦相似度, 将其作为匹配矩阵中的每一项:

$$m_{ij}=\mathrm{cosine}(\boldsymbol{e}_i^u,\boldsymbol{e}_j^s).\tag{11}$$

这样便得到匹配矩阵 $\boldsymbol{M}$, 其中每个元素 m_{ij} 代表话术中第 i 个词与意图描述中第 j 个词之间的余弦相似度.

3.2.2 卷积层

为了提取不同的匹配模式, 我们在匹配矩阵上应用卷积操作. 具体来说, 通过第 k 个卷积核 $\boldsymbol{w}^k$ 对整个匹配矩阵 $\boldsymbol{M}$ 进行扫描, 以生成特征图 $\boldsymbol{f}^k$, 其中每个元素的计算如下:

$$\boldsymbol{f}^k_{i,j} = \sigma(\boldsymbol{w}^k \times \boldsymbol{M}_{[i-r_k:i+r_k, j-r_k:j+r_k]} + b_k), \tag{12}$$

其中 $\times$ 表示卷积运算, r_k 表示第 k 个卷积核的大小, 步长为 1. 本文使用线性整流函数 [22] (rectified linear unit, ReLU) 作为激活函数 σ.

3.2.3 池化层

为了保留最大匹配特征, 我们从水平和垂直方向构造了不同的池化操作, 即对于话术中的每个单词, 保留意图描述中最相似的词; 以及对意图描述中的每个单词, 保留话术中的最相似的词:

$$\boldsymbol{a}^k_h = [\max(\boldsymbol{f}^k_{1,\cdot}), \max(\boldsymbol{f}^k_{2,\cdot}), \ldots, \max(\boldsymbol{f}^k_{d_2,\cdot})], \tag{13}$$

$$\boldsymbol{a}^k_v = [\max(\boldsymbol{f}^k_{\cdot,1}), \max(\boldsymbol{f}^k_{\cdot,2}), \ldots, \max(\boldsymbol{f}^k_{\cdot,d_1})], \tag{14}$$

其中 d_1 和 d_2 表示特征图的宽度和长度.

3.2.4 全连接层

经过卷积和最大池化操作后, 再通过全连接层得到最终的水平特征和垂直特征, 一般公式如下:

$$z_h = \boldsymbol{W}_2\sigma(\boldsymbol{W}_1\boldsymbol{a}_h + \boldsymbol{b}_1) + \boldsymbol{b}_2, \tag{15}$$

$$z_v = \boldsymbol{W}_2\sigma(\boldsymbol{W}_1\boldsymbol{a}_v + \boldsymbol{b}_1) + \boldsymbol{b}_2, \tag{16}$$

其中, $\boldsymbol{a}_h, \boldsymbol{a}_v$ 分别为水平池化层的输出和垂直池化层的输出, σ 为激活函数, 这里为 ReLU. 这样便从词级获得显式网络的相似性表示: $\boldsymbol{z}_{\text{explicit}} = [z_h, z_v]$.

3.3 关系层

我们使用关系层对隐式网络和显式网络之间的交互进行建模. 它以隐式网络获得的句子级相关特征 $\boldsymbol{z}_{\text{implicit}}$ 和显式网络得到的词级匹配特征 $\boldsymbol{z}_{\text{explicit}}$ 为输入, 然后输出融合后的信息并进行预测. 我们的框架探索了两种类型的关系层.

3.3.1 多层感知器

多层感知器能够自动提取信息. 因此, 我们拼接显式网络的输出 $\boldsymbol{z}_{\text{explicit}}$ 和隐式网络的输出 $\boldsymbol{z}_{\text{implicit}}$, 将其作为 MLP 层的输入, 进一步学习整体匹配分数, 从而融合多级信息, 其公式如下:

$$S(u, y) = \text{MLP}([\boldsymbol{z}_{\text{implicit}}; \boldsymbol{z}_{\text{explicit}}]). \tag{17}$$

3.3.2 选择门

我们设计选择门来融合显式网络的输出 $\boldsymbol{z}_{\text{explicit}}$ 和隐式网络的输出 $\boldsymbol{z}_{\text{implicit}}$, 并通过选择门决定是使用隐式网络还是显式网络进行最终预测. 选择门定义如下:

$$g = \sigma(\text{FC}([\boldsymbol{z}_{\text{implicit}}; \boldsymbol{z}_{\text{explicit}}])), \tag{18}$$

其中, FC$(\cdot)$ 是全连接层, 而 σ 是 S 型激活函数. 因此, 给定话术 u 和每个意图 y 之间的相似度得分 $S(u, y)$ 定义为

$$S(u, y) = g\text{MLP}(\boldsymbol{z}_{\text{implicit}}) + (1 - g)\text{MLP}(\boldsymbol{z}_{\text{explicit}}). \tag{19}$$

我们注意到选择门既可以作为一个二进制开关在 $\boldsymbol{z}_{\text{implicit}}$ 和 $\boldsymbol{z}_{\text{explicit}}$ 之间进行决策, 也可以作为概率混合开关进行概率融合.

3.4 损失函数

合页损失函数 (hinge loss) 是检索和排序任务中最流行的, 非常适合我们的任务. 这是因为我们希望正样本的相似性得分越高越好, 负样本的相似性得分越低越好. 故我们的合页损失函数定义为

$$\mathcal{L} = \sum_{(u, y^+) \in \mathcal{D}} \left(\sum_{y^- \in \mathcal{Y}^-} [\gamma + S(u, y^-) - S(u, y^+)]_+ \right), \tag{20}$$

其中, y^+ 是正样本的意图, y^- 是负样本的意图. γ 是边际值, 默认值为 1. $[x]_+$ 意味着 $\max(0, x)$.

4 实验

本节设计相关实验来评估 INTENT 模型. 同时, 我们还描述用于评估的数据集、基线方法, 以及实验设置的更多细节.

4.1 数据集

我们主要使用两个公开的基准数据集: SNIPS[1)] 和 ATIS[2)]. SNIPS 是个人语音助手数据集, 它以一种众包方式收集. ATIS 是单域 (航空旅行) 数据集, 包含预订航班的记录. 上述数据集具体的统计数据参见表 2.

对于意图识别任务, 我们参照文献 [23] 对数据集进行划分. 对于零样本意图识别任务, 我们遵循文献 [19] 中的划分原则, 将 SNIPS 数据集中的 RateBook 和 AddToPlaylist 视为新兴意图. 对于 ATIS 数据集, 我们选择 Aircraft, Airport, Capacity, Distance, GroundFare 和 Quantity 作为新兴意图. 在不重叠意图标签的情况下, 将样本随机划分为训练集和测试集, 更多细节如表 2 所示. 为了进行广义零样本意图识别, 我们参照文献 [24] 随机将每个可见意图的 70% 样本作为训练集, 然后将剩下的 30% 可见意图的样本和所有新兴意图的样本作为测试集.

4.2 基线方法

我们将 INTENT 模型和已有零样本意图识别方法进行比较.

1) https://github.com/snipsco/nlu-benchmark/.
2) https://github.com/MiuLab/SlotGated-SLU/tree/master/data/atis.

表 2　对于零样本意图识别任务, SNIPS 和 ATIS 数据集的统计

Table 2　For zero-shot intent detection task, statistics of SNIPS and ATIS datasets

Task	SNIPS	ATIS
Vocab size	11641	950
Average sentence length	9.05	11.13
Slot types	53	390
Train samples	9888	5597
Test samples	3914	274
Existing intents	5	20
Emerging intents	2	6

• Zero-Shot SVM [3], 它构建一个表征学习分类器, 该分类器学习如何构建话术表示和意图表示, 并判断两者是否兼容.

• Zero-Shot CDSSM [7], 它通过词的 n-gram 和卷积池化操作来捕获意图 – 话术对中的上下文关系, 从而获得意图和话术的向量表示, 预测新兴意图.

• Zero-shot DNN [4], 它在 CDSSM 的基础上, 引入一组描述意图的属性, 并对话术和意图使用单独的编码器, 获得相应的表示. 为了公平对比, 在我们的任务中, 提供槽位类型作为意图的属性.

• INTENTCAPSNET-ZSL [19], 它提出了一种基于胶囊网络的零样本意图识别模型, 通过分层的方式从话术中提取和聚合语义.

• ReCapsNet-ZS [24], 它重构了用于零样本意图识别的胶囊网络, 并通过转换矩阵将先验知识从已知意图转移到新兴意图, 以实现零样本意图识别.

4.3　实验细节

我们通过 GloVe [25] 预训练的 300 维向量对嵌入层进行初始化. 对于 BiLSTM, 我们将隐藏层的维度设置为 128. CNN 通道数为 8, 内核大小设为 3×3. 在损失函数 hinge loss 中, γ 设为 1. 批量大小为 128. 采用 Adam [26] 进行优化, SNIPS 和 ATIS 的初始学习率分别为 0.01 和 0.05. 对于句子的最大长度, SNIPS 和 ATIS 分别设置为 35 和 30. 为了避免过拟合, dropout 设置为 0.5. 我们以 1:4 的比例对正实例和负实例进行采样 [27]. 但是, 在选择负样本时, 采用不同的采样策略并没有观察到显著性差异.

所有的模型都使用 pytorch[3)] 实现, 运行在 NVIDIA Tesla P4 GPU 上. 除非另有说明, 否则所有基线模型的参数都从其论文中继承而来. 在训练过程中, 我们随机打乱了每个批次的所有例子. 我们在每次迭代后评估模型表现, 当验证集上的错误开始增加时停止训练. 对于 SNIPS 和 ATIS, 训练一个 epoch 的平均时间分别为 300 s 和 180 s 左右.

5　结果

5.1　对比分析

在进行零样本意图识别之前, 首先评估模型在意图识别任务上的表现. 将 INTENT 模型与其他文本分类方法进行比较, 包括 CNN [14], Bi-LSTM [20], Self-Attention Bi-LSTM [28]. 然后, 将 INTENT

3) https://pytorch.org/.

表 3 意图识别在两个数据集上准确率的对比

Table 3 Comparison of accuracy for intent detection on two datasets

Model	SNIPS	ATIS
CNN [14]	0.9595	0.9124
Bi-LSTM [20]	0.9501	0.9241
Self-Attention BiLSTM [28]	0.9524	0.9264
INETENTCAPSNET [19]	0.9621	0.9480
ReCapsNet-ZS [24]	0.9664	0.9515
INTENT+MLP	0.9726	0.9549
INTENT+Switch-Gate	**0.9785**	**0.9619**

表 4 广义零样本意图识别和零样本意图识别在两个数据集上平均准确率的对比

Table 4 Comparison of mean accuracy for generalized zero-shot and zero-shot intent detection on two datasets

Model	Generalized zero-shot		Zero-shot	
	SNIPS	ATIS	SNIPS	ATIS
Zero-shot SVM [3]	0.0271	0.0116	0.6847	0.5847
Zero-shot CDSSM [7]	0.0111	0.0263	0.7588	0.6429
Zero-shot DNN [4]	0.0682	0.0523	0.7165	0.6294
INTENTCAPSNET-ZSL [19]	–	–	0.7752	0.6743
ReCapsNet-ZS [24]	**0.1121**	**0.0923**	0.7996	0.6821
INTENT + MLP	0.0914	0.0762	0.8387	0.6892
INTENT + switch gate	0.1109	0.0803	**0.8612**	**0.7029**

模型与 INETENTCAPSNET [19] 和 ReCapsNet-ZS [24] 等一些最新的零样本意图识别模型进行比较. 表 3 [14, 19, 20, 24, 28] 给出了 5 次运行的平均结果. 在 SNIPS 数据集中, 与 ReCapsNet-ZS 和 CNN 相比, 我们使用选择门模型的准确率分别提高了 1.21% 和 1.9%. 在 ATIS 数据集中, 与 ReCapsNet-ZS 和 Self-Attention Bi-LSTM 相比, 准确率分别提高了 1.04% 和 3.55%. 这些结果表明, 当有大量的训练数据可用时, 我们的模型也能获得有竞争力的结果.

同样, 我们按照文献 [24] 的设置来评估所提出模型在广义零样本意图识别上的表现. 从表 4 的左侧, 我们可以看到广义零样本意图识别的结果整体偏低. 这是因为已出现的标签集成到测试集中, 使问题变得更加棘手. 但是, 可以看到 INTENT 模型有一定的竞争力. 这是由于我们的模型引入了槽位类型作为意图描述来增强意图表示, 并从词级和句子级进行了联合建模, 这使得模型具有更好的泛化能力.

此外, 我们在表 4 的右侧部分呈现了零样本意图识别的结果. 从实验结果上看, 我们的模型在两个数据集上的表现优于其他基线模型, 这说明 INTENT 模型可以有效地解决零样本意图识别问题. 特别是关系层中的选择门性能优于 MLP, 表明选择门的操作能够自动检测出多层级的重要信息, 并能更好地与网络进行交互.

5.2 消融实验

除非另有说明, 否则我们在以下实验中应用选择门作为关系层. 我们进一步分析了不同模块对模型的贡献, 从表 5 中的实验结果可以看出, 我们模型的所有模块对最终的结果都有影响. 具体分析

表 5　零样本意图识别在两个数据集上进行消融研究

Table 5　Ablation study for zero-shot intent detection on two datasets

Model	SNIPS	ATIS
INTENT	0.8612	0.7029
Implicit-network w/o attention	0.7891	0.6617
Implicit-network	0.8072	0.6742
Explicit-network	0.8306	0.6551
INTENT w/o description	0.8366	0.6825

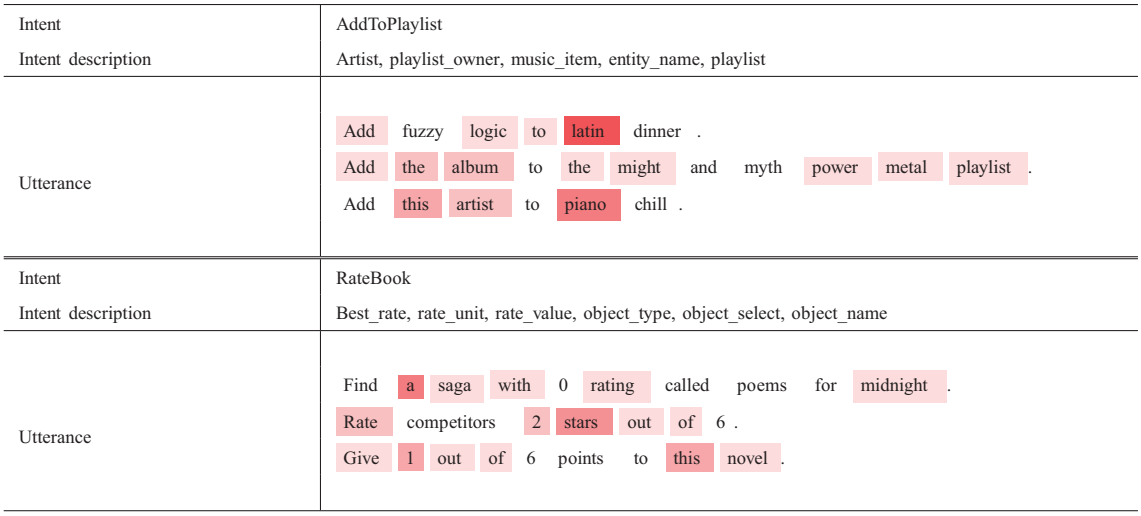


图 2　(网络版彩图) 在 SNIPS 中新兴意图的注意力可视化示例

Figure 2　(Color online) Examples of the visualization of attention with emerging intents in SNIPS

如下:

• 在隐式网络上采用注意力机制可以提高性能 (约 1%∼2%). 在 5.3 小节中进一步验证, 注意力机制可以关注话术中相关的描述, 并捕获更重要的信息, 这有助于获得匹配信息.

• 与 ATIS 相比, INTENT 模型在 SNIPS 上表现更好. 我们认为这是由于 ATIS 的意图都来自同一个领域, 意图间的描述更有可能重叠, 使得增强效果不明显. 例如, "flight_time" 和 "flight_no" 是两个不同的意图描述, 但都有单词 "flight". 这些重叠词可能会导致网络的性能下降.

• 在引入意图描述后, 准确率提高了 2%∼3%, 这进一步说明了增强的意图表示是有效的.

• 最后, 我们看到所有模块都对该模型有所贡献. 实验结果表明, ATIS 和 SNIPS 的准确性分别为 70.29% 和 86.12%, 超过已有的方法, 更进一步说明显式网络和隐式网络具有互补的优势.

5.3　注意力机制有效性分析

消融实验的结果表明, 注意力机制对获得最佳表现起着至关重要的作用. 为了进一步理解语义特征提取的能力, 我们在 SNIPS 数据集上可视化了话术中词的注意力分布 (见图 2). 我们发现注意力机制具有语义特征提取的能力, 并且能够聚焦于重要的词. 对于新兴意图 AddToPlaylist, 注意力机制不仅关注意图描述中的 "playlist" 和 "artist", 还关注一些语义相关的表达, 如 "piano" 和 "album". 这可能是由于训练数据中存在相似的可见意图 PlayMusic, 使得注意力机制可以从可见意图中转移相似

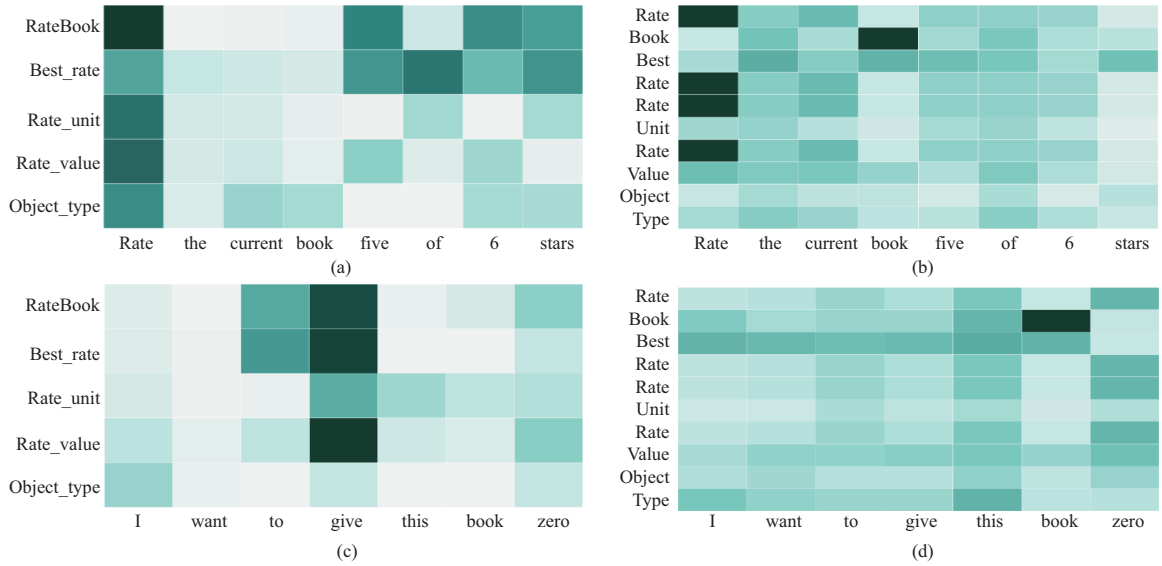


图 3 (网络版彩图) 模型可视化. (a) 显式意图; (b) 隐式意图. 颜色越深的区域表示值越大.

Figure 3 (Color online) Model visualization. (a) Explicit intent; (b) implicit intent. The darker the area, the greater the value

性知识. 而对于新兴意图 RateBook, 我们可以看到 INTENT 模型仅关注某些信息, 如 “rate”, “give”, “rating” 等. 这可能是因为在训练数据中没有类似的可见意图, 影响了模型的特征提取能力. 但由于 INTENT 模型通过引入槽位类型增强了意图表示, 使得模型关注话术中不同的信息. 这样, 即使模型无法关注某些意图相关词 (如 “0”, “6” or “6 point”), 却仍然可以关注其他重要信息, 这使得模型整体效果还不错.

5.4 语义漂移分析

语义漂移问题产生的根本原因是词的语义动态变化, 使得词语的上下文重要性难以衡量, 造成句子失去语义焦点. 为了进一步说明显式网络和隐式网络是如何通过关注语义焦点和识别重要词避免语义漂移问题带来的影响的, 这里以 SNIPS 数据集为例, 对匹配矩阵和注意力矩阵进行可视化. 由于篇幅有限, 我们在图 3 中只列举了两个示例, 分别来自显式意图和隐式意图. 让我们看一个显式意图的示例, 如图 3(a) 所示, 话术中包含意图相关的词 “rate”, 这符合显式意图的定义. 从图 3(a) 的注意力矩阵可以看出, 我们使用了注意力机制对不同的词赋予不同的权重, 关注到意图相关的重要词 (“rate”). 这样可以减少不重要词的影响, 从而更好地捕捉语义焦点, 避免语义漂移. 而从图 3(a) 的匹配矩阵中可以看出, 如果两句话中的词越接近, 最终的匹配得分就越高, 这样显式网络便可以很好地从匹配矩阵中提取相似度最高的单词, 从而捕获语义焦点, 不用担心被不相关的词所误导, 避免语义漂移问题的发生. 类似地, 对于隐式意图, 我们可以在图 3(b) 中看到相似的结果. 从图中的注意力矩阵和匹配矩阵也可以看出, INTENT 模型能够很好地捕捉语义焦点, 有效地避免语义漂移. 综上实验结果表明, 该模型能够正确地捕捉重要信息, 有效地避免语义漂移.

5.5 深入分析

为了进一步评估模型, 我们构建了一个实验来识别数据集中的显式意图和隐式意图, 并对比分析模型的效果. 我们从 SNIPS 数据集中选择意图 RateBook 作为新兴意图. 依据话术中是否包含与意图

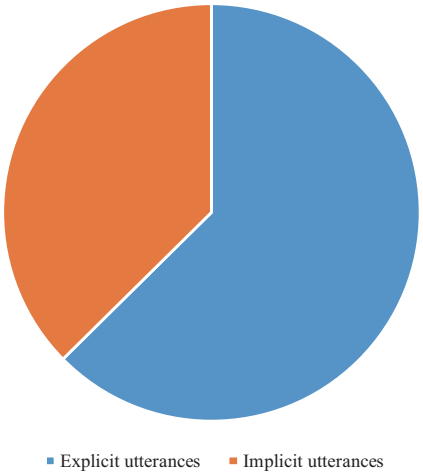


图 4 (网络版彩图) 显式话术和隐式话术的比例

Figure 4 (Color online) The proportion of explicit utterances and implicit utterances

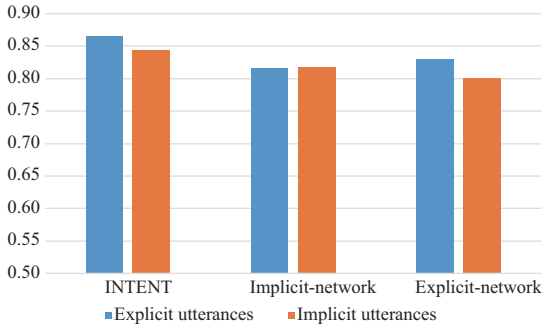


图 5 (网络版彩图) 显式/隐式话术在不同模型中的准确率

Figure 5 (Color online) The accuracy of explicit/implicit utterances in different models

相关的词 (例如 “rate”, “rating”), 我们将其分为显式意图和隐式意图, 统计结果如图 4 所示. 实验结果表明, 显式意图的样本占比更高, 这是由 SNIPS 数据集本身决定的. 在此基础上, 我们评估模型的表现, 具体结果如图 5 所示. 深入分析后发现, INTENT 模型对显式意图的整体效果要优于隐式意图. 这是因为显式意图的话术中清楚地包含意图信息, 所以模型更易于识别用户的意图. 同时, 我们进一步分析了各个模型的效果. 而对于隐式意图的样本, 隐式网络的效果要略优于显式网络的效果. 同样, 对于显式意图的样本, 显式网络的效果要优于隐式网络的效果. 这表明显式网络更倾向于捕获单词级信息, 而隐式网络更倾向于捕获句子级信息. 但是, 无论是显式还是隐式, INTENT 模型的效果都要优于独立的模型. 这表明词级和句子级信息的融合可以有效地提高模型的性能.

6 结语

本文提出了一种新的零样本意图识别方法, 该方法由两个独立的深层神经网络组成: 隐式网络 (用于从意图 – 话术对中捕获隐式信息表示 —— 句子级) 和显式网络 (用于提取意图 – 话术对中的显式信息表示 —— 词级). 此外, 我们探讨了两个不同的关系层, 并对它们的影响进行了深入的研究. 在两个基准数据集上进行的大量实验表明了 INTENT 模型的有效性. 在未来的工作中, 我们计划扩展我们的方法, 以解决少样本意图识别问题.

参考文献

1 Yang Z, Yang D, Dyer C, et al. Hierarchical attention networks for document classification. In: Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, 2016. 1480–1489

2 Haihong E, Niu P, Chen Z, et al. A novel bi-directional interrelated model for joint intent detection and slot filling. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, 2019. 5467–5471

3 Yazdani M, Henderson J. A model of zero-shot learning of spoken language understanding. In: Proceedings of Conference on Empirical Methods in Natural Language Processing, Lisbon, 2015. 244–249

4 Kumar A, Muddireddy P R, Dreyer M, et al. Zero-shot learning across heterogeneous overlapping domains. In: Proceedings of the 18th Annual Conference of the International Speech Communication Association, Stockholm, 2017. 2914–2918

5 Ferreira E, Jabaian B, Lefevre F. Online adaptative zero-shot learning spoken language understanding using word-embedding. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Brisbane, 2015. 5321–5325

6 Ferreira E, Jabaian B, Lefevre F. Zero-shot semantic parser for spoken language understanding. In: Proceedings of the 16th Annual Conference of the International Speech Communication Association, Dresden, 2015

7 Chen Y N, Hakkani-Tür D, He X. Zero-shot learning of intent embeddings for expansion by convolutional deep structured semantic models. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Shanghai, 2016. 6045–6049

8 Dauphin Y N, Tur G, Hakkani-Tür D, et al. Zero-shot learning for semantic utterance classification. 2013. ArXiv:1401.0509

9 Xu P, Sarikaya R. Convolutional neural network based triangular crf for joint intent detection and slot filling. In: Proceedings of Automatic Speech Recognition and Understanding Workshop, Olomouc, 2013. 78–83

10 Zhang C, Li Y, Du N, et al. Joint slot filling and intent detection via capsule neural networks. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, 2019. 5259–5267

11 Chen Z, Liu B, Hsu M, et al. Identifying intention posts in discussion forums. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Atlanta, 2013. 1041–1050

12 Coucke A, Saade A, Ball A, et al. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. 2017. ArXiv:1805.10190

13 Tur G, Hakkani-Tür D, Heck L. What is left to be understood in ATIS. In: Proceedings of IEEE Spoken Language Technology Workshop, 2010. 19–24

14 Kim Y. Convolutional neural networks for sentence classification. In: Proceedings of Conference on Empirical Methods in Natural Language Processing, Doha, 2014. 1746–1751

15 Ravuri S, Stolcke A. Recurrent neural network and LSTM models for lexical utterance classification. In: Proceedings of the 16th Annual Conference of the International Speech Communication Association, Dresden, 2015

16 Zhao Z, Wu Y. Attention-based convolutional neural networks for sentence classification. In: Proceedings of the 17th Annual Conference of the International Speech Communication Association, San Francisco, 2016. 705–709

17 Guo D, Tur G, Yih W, et al. Joint semantic utterance classification and slot filling with recursive neural networks. In: Proceedings of IEEE Spoken Language Technology Workshop, California, 2014. 554–559

18 Liu B, Lane I. Attention-based recurrent neural network models for joint intent detection and slot filling. In: Proceedings of the 17th Annual Conference of the International Speech Communication Association, San Francisco, 2016. 685–689

19 Xia C, Zhang C, Yan X, et al. Zero-shot user intent detection via capsule neural networks. In: Proceedings of Conference on Empirical Methods in Natural Language Processing, 2018. 3090–3099

20 Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput, 1997, 9: 1735–1780

21 Pang L, Lan Y, Guo J, et al. Text matching as image recognition. In: Proceedings of the 30th AAAI Conference on Artificial Intelligence, 2016. 2793–2799

22 Dahl G E, Sainath T N, Hinton G E. Improving deep neural networks for LVCSR using rectified linear units and dropout. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, 2013. 8609–8613

23 Goo C W, Gao G, Hsu Y K, et al. Slot-gated modeling for joint slot filling and intent prediction. In: Proceedings of the 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, 2018. 753–757

24 Liu H, Zhang X, Fan L, et al. Reconstructing capsule networks for zero-shot intent classification. In: Proceedings of Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on

Natural Language Processing, Hong Kong, 2019. 4801–4811
25 Pennington J, Socher R, Manning C D. Glove: global vectors for word representation. In: Proceedings of Conference on Empirical Methods in Natural Language Processing, Doha, 2014. 1532–1543
26 Kingma D P, Ba J. ADAM: a method for stochastic optimization. 2014. ArXiv:1412.6980
27 Huang P S, He X, Gao J, et al. Learning deep structured semantic models for web search using clickthrough data. In: Proceedings of the 22nd ACM international conference on Information & Knowledge Management, 2013. 2333–2338
28 Lin Z, Feng M, Santos C N, et al. A structured self-attentive sentence embedding. 2017. ArXiv:1703.03130

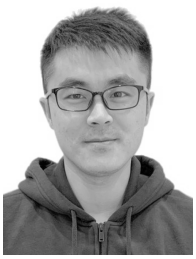
Similarity learning with implicit-network and explicit-network for zero-shot intent detection

Pengfei SUN, Yawen OUYANG, Xinyu DAI* & Wenming ZHANG

State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023*, China*
* Corresponding author. E-mail: daixinyu@nju.edu.cn

Abstract Intent detection is an important component in dialog systems. Existing studies mainly focus on intent detection with sufficient labeled data. However, these methods are unable to detect intents that do not exist in training data. To tackle this problem, we propose an implicit-network and explicit-network model for zero-shot intent detection, which is capable of learning similarities between utterances and intent description from word level and sentence level. To enhance the representation of the intent, we introduce slot types as the intent description. We divide intent into explicit intent and implicit intent according to different expression ways, and construct explicit-network and implicit-network from word level and sentence level respectively. Meanwhile, in order to better combine these two parts of information, we also design a relation layer to fuse different levels of information. Experiments on two benchmark datasets show that our model significantly outperforms existing state-of-the-art models and demonstrate the effectiveness of learning similarity from word level and sentence level simultaneously.

Keywords zero-shot intent detection, implicit network, explicit network, relation layer, switch gate



Pengfei SUN was born in 1987. He received his M.E. degree from Chongqing University of Posts and Telecommunications, Chongqing, China, in 2013. Currently, he is working towards a Ph.D. degree with State Key Laboratory for Novel Software Technology, Department of Computer Science & Technology at Nanjing University, China. His research interest lies primarily in natural language processing.

Yawen OUYANG was born in 1996. He received his B.E. degree from Dalian University of Technology, Liaoning, China, in 2018. Currently, he is working towards a Ph.D. degree with State Key Laboratory for Novel Software Technology, Department of Computer Science & Technology at Nanjing University, China. His research interest lies primarily in natural language processing.

Xinyu DAI was born in 1979. He received my Ph.D. in computer science at Nanjing University, China, in 2005. He is currently a professor in Department of Computer Science and Technology at Nanjing University. His research interests majorly include natural language processing and recommender systems.

Wenming ZHANG was born in 1996. He received his B.S. degree from Nanjing University, Nanjing, China, in 2018. Currently, he is working towards a M.S. degree with State Key Laboratory for Novel Software Technology, Department of Computer Science & Technology at Nanjing University, China. His research interests include natural language processing and recommender systems.