·人工智能与模式识别·　　文章编号：1000-3428(2023)03-0073-07　　文献标志码：A　　中图分类号：TP18

利用知识强化语言模型的口语理解方法

刘高军[1,2,3]，王　岳[1,2,3]，段建勇[1,2,3]，何　丽[1,2,3]，王　昊[1,2,3]

（1.北方工业大学 信息学院，北京 100144；2.CNONIX国家标准应用与推广实验室，北京 100144；
3.富媒体数字出版内容组织与知识服务重点实验室，北京 100144）

摘　要：基于预训练的语言模型在口语理解（SLU）任务中具有优异的性能表现。然而，与人类理解语言的方式相比，单纯的语言模型只能建立文本层级的上下文关联，缺少丰富的外部知识来支持其完成更为复杂的推理。提出一种针对SLU任务的基于Transformer的双向编码器表示（BERT）的联合模型。引入单词级别的意图特征并使用注意力机制为BERT融合外部知识。此外，由于SLU包含意图检测和槽填充2个相互关联的子任务，模型通过联合训练捕捉2个子任务间的关联性，充分运用这种关联性增强外部知识对于SLU任务的性能提升效果，并将外部知识转化为可用于特定子任务的特征信息。在ATIS和Snips 2个公开数据集上的实验结果表明，该模型句子级别的语义准确率分别为89.1%和93.3%，与BERT模型相比，分别提升了0.9和0.4个百分点，能够有效利用外部知识提升自身性能，在SLU任务中拥有比BERT更为优秀的性能表现。

关键词：口语理解；外部知识；语言模型；意图检测；槽填充；联合训练

开放科学（资源服务）标志码（OSID）：

中文引用格式：刘高军，王岳，段建勇，等.利用知识强化语言模型的口语理解方法[J].计算机工程，2023，49(3)：73-79.

英文引用格式：LIU G J，WANG Y，DUAN J Y，et al.Methods of spoken language understanding using knowledge reinforcement language model[J].Computer Engineering，2023，49(3)：73-79.

Methods of Spoken Language Understanding Using Knowledge Reinforcement Language Model

LIU Gaojun[1,2,3]，WANG Yue[1,2,3]，DUAN Jianyong[1,2,3]，HE Li[1,2,3]，WANG Hao[1,2,3]

（1.School of Information Science and Technology，North China University of Technology，Beijing 100144，China；
2.CNONIX National Standard Application and Promotion Laboratory，Beijing 100144，China；
3.Rich Media Digital Publishing Content Organization and Knowledge Service Key Laboratory，Beijing 100144，China）

【Abstract】Pretrained language representations have shown excellent performance in Spoken Language Understanding（SLU）.However，compared with the way humans understand language，language representations can only establish the contextual relationship of an input sequence.Additionally，they lack the external knowledge required to complete more complex reasoning. This paper proposes a joint model based on the Bidirectional Encoder Representations from Transformer（BERT）for SLU.The model uses the attention mechanism to fuse external knowledge.In addition，SLUs contain two interrelated subtasks，namely intention detection and slot filling. Therefore，the model captures the correlation between the two subtasks through joint training.The model makes full use of this correlation to further enhance the performance improvement effect of the external knowledge on SLU tasks. Additionally，the external knowledge is converted into characteristic information that can be used for specific subtasks.The experimental results on the ATIS and Snips datasets show that the semantic accuracy of the sentence level of this model is increased by 89.1% and 93.3%，respectively.This is 0.9 and 0.4 percentage points higher than that of the BERT model.Additionally，the model can effectively use external knowledge to improve its own performance.Therefore，the model exhibits better performance in SLU missions than BERT.

【Key words】Spoken Language Understanding（SLU）；external knowledge；language model；intention detection；slot filling；joint training

DOI：10. 19678/j. issn. 1000-3428. 0062149

基金项目：国家自然科学基金面上项目“面向新闻事件的查询时效性计算模型研究”（61972003）；富媒体数字出版内容组织与知识服务重点实验室项目（ZD2021-11/05）。

作者简介：刘高军（1962—），男，教授，主研方向为软件工程与服务；王　岳，硕士研究生；段建勇，教授、博士；何　丽，副教授、硕士；王　昊，副教授、博士。

收稿日期：2021-07-21　　**修回日期：**2021-12-23　**E-mail：**liugj@ncut.edu.cn

0 概述

口语理解(Spoken Language Understanding,SLU)是任务型对话系统的关键组成部分,通常包含意图检测和槽填充2个相互关联的子任务。其中,意图检测需要完成对口语语句整体意图的分类,槽填充则标注句子中的关键语义成分。

近年来,以基于Transformer的双向编码器表示(Bidirectional Encoder Representations from Transformer,BERT)[1]为代表的基于预训练的语言模型在各领域中都得到了广泛的应用,并在多项任务中表现出优异的性能。文献[2]验证了BERT模型在SLU任务中极其优异的性能表现。然而,以BERT为代表的语言模型只能建立文本的上下文关联,而缺少丰富的外部知识来支持其完成更为复杂的推理。基于这一背景,研究人员开始尝试通过为语言模型融合知识进行迁移训练来提升其在特定任务中的性能表现。文献[3]通过使用注意力机制为BERT融入外部知识来提升其在机器阅读理解任务中的性能表现。而在口语理解领域,则较少有研究人员进行该方面的研究。与文献[3]相同,研究人员使用2个知识库为BERT提供外部知识,记录单词间语义关系的WordNet[4]以及保存了实体概念的NELL(Never-Ending Language Learning)[5],它们分别为模型提供了语言学知识和真实世界知识,使用KB-LSTM[6]提供的100维预训练嵌入来对知识进行表示,通过这些外部知识来提升模型对于模糊、以及可能存在错误语句的理解能力,并赋予模型更加优秀的复杂推理能力。

由于口语理解任务包含了意图检测和槽填充这2个相关联的子任务,近年来出现了大量的联合模型,旨在通过此原理,利用各种联合训练机制提升模型在口语理解任务中的性能表现[7-9]。例如:文献[10]将编码器-解码器结构和注意力机制引入到口语理解任务中,使2个子模块使用共同的编码器计算损失函数来实现联合训练;文献[11]通过门控机制为槽填充引入意图信息来提升模型的性能表现;文献[12]提出了循环运行的子网络结构,使2个子模块互相利用对方产生的特征信息提升各自的性能表现;文献[13]通过引入单词级别的意图检测和堆栈传播机制[14]来提升模型的性能表现。

本文提出一种针对口语理解任务的基于BERT的联合模型。该模型引入单词级别的意图特征,使用注意力机制为BERT融入外部知识,通过外部知识的联合训练机制提升其在口语理解任务中的性能表现。最终在ATIS[15]和Snips[16]2个公开数据集上进行了实验。

1 本文方法

下文通过4个模块对本文模型进行介绍:

1)BERT编码器和意图编码器:为每个口语语句输出其对应的单词级别的语义槽值特征和意图特征。

2)知识整合器:通过注意力机制将每个单词所对应的多个知识嵌入进行加权运算并得到相应的知识向量。

3)知识注意力层:使用前2个模块的输出来计算单词级别的知识上下文向量。

4)知识解码器:以单词级别的意图特征、语义槽值特征和知识上下文向量作为输入,经过计算给出单词级别的意图输出和语义槽值输出。

在上述4个模块中,知识整合器和知识解码器是模型中最重要的2个模块,它们分别完成了融合知识和联合训练这2个重要的步骤。

1.1 模型构成

1.1.1 BERT编码器和意图编码器

如图1所示,BERT编码器和意图编码器会为每个口语语句输出其对应的单词级别的语义槽值特征和意图特征。

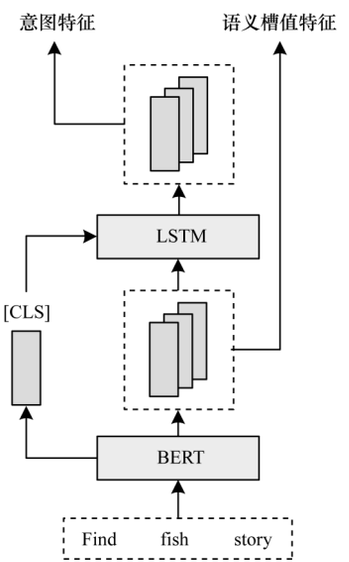

图1 BERT编码器和意图编码器的结构

Fig.1 Structure of BERT encoder and intent encoder

对于一个输入语句$X=(x_1,x_2,\cdots,x_T)$,T是输入语句的长度,BERT编码器和意图编码器会映射出与之相对应的d维单词级别语义槽值特征$\boldsymbol{Y}^S=(\boldsymbol{y}^S_1,\boldsymbol{y}^S_2,\cdots,\boldsymbol{y}^S_T)$以及$d$维单词级别意图特征$\boldsymbol{Y}^I=(\boldsymbol{y}^I_1,\boldsymbol{y}^I_2,\cdots,\boldsymbol{y}^I_T)$。BERT编码器会输出$T$个维度为$d$的隐藏层状态向量和一个可以用于分类任务的$d$维特殊向量[CLS]。与文献[2]相同,将隐藏层状态向量记为$\boldsymbol{Y}^S=(\boldsymbol{y}^S_1,\boldsymbol{y}^S_2,\cdots,\boldsymbol{y}^S_T)$,用以表示单词级别的语义槽值特征。另外,将特殊向量[CLS]看作句子级别的意图特征。

在BERT编码器之后,意图编码器将为整个模型引入单词级别的意图特征。使用一个隐藏层大小为d的单向长短期记忆(Long Short-Term Memory,LSTM)[17]作为意图编码器,对于一个输入语句$X=(x_1,x_2,\cdots,x_T)$,意图编码器接收这句话所对应的语义槽值特征序列$\boldsymbol{Y}^S=(\boldsymbol{y}^S_1,\boldsymbol{y}^S_2,\cdots,\boldsymbol{y}^S_T)$作为输入,并映射出与之相对应的$T$个维度为$d$的单词级别意图特征向量$\boldsymbol{Y}^I=(\boldsymbol{y}^I_1,\boldsymbol{y}^I_2,\cdots,\boldsymbol{y}^I_T)$。对于时间步$i$,意图编码器的隐藏层状态$\boldsymbol{h}_i$由前一个隐藏层状态$\boldsymbol{h}_{i-1}$、前一个意图特征$\boldsymbol{y}^I_{i-1}$和当前时间步的语义槽值特征输入$\boldsymbol{y}^S_i$共同计算得出,如式(1)所示:

$$\boldsymbol{h}_i = f\left(\boldsymbol{h}_{i-1}, \boldsymbol{y}^I_{i-1}, \boldsymbol{y}^S_i\right) \tag{1}$$

另外，由于BERT输出的特殊向量[CLS]中蕴含了整个句子的意图特征信息，因此使用其作为单向LSTM的初始隐藏层状态输入$\boldsymbol{h}_0$。

1.1.2　知识整合器

如图2所示，知识整合器会通过注意力机制将每个单词所对应的多个知识嵌入进行加权运算并得到相应的知识向量。对于一个输入语句$X=(x_1,x_2,\cdots,x_T)$，T是输入语句的长度，知识整合器接收BERT编码器和意图编码器所输出的单词级别语义槽值特征$\boldsymbol{Y}^S=(\boldsymbol{y}^S_1,\boldsymbol{y}^S_2,\cdots,\boldsymbol{y}^S_T)$以及单词级别的意图特征$\boldsymbol{Y}^I=(\boldsymbol{y}^I_1,\boldsymbol{y}^I_2,\cdots,\boldsymbol{y}^I_T)$作为输入，并结合相关的知识嵌入计算得到维度为100的单词级别知识特征$\boldsymbol{K}=(\boldsymbol{k}_1,\boldsymbol{k}_2,\cdots,\boldsymbol{k}_T)$。

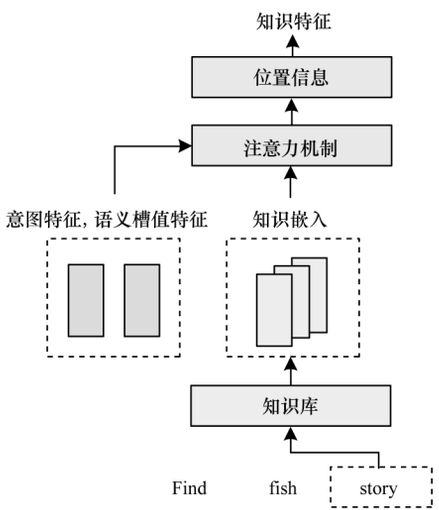

图2　知识整合器的结构

Fig.2　Structure of knowledge integrator

具体地，对于输入语句$X=(x_1,x_2,\cdots,x_T)$中的单词x_i，BERT编码器和意图编码器会给出单词x_i所对应的单词级别意图特征$\boldsymbol{y}^I_i$和单词级别的语义槽值特征$\boldsymbol{y}^S_i$。此外，从知识库中检索单词x_i所对应的n个知识概念$C(x_i)$，并得到这些知识概念的100维嵌入$\boldsymbol{C}_i=(\boldsymbol{c}_{i,1},\boldsymbol{c}_{i,2},\cdots,\boldsymbol{c}_{i,n})$。

然后使用注意力机制[18]来计算概念嵌入$\boldsymbol{c}_{i,j}$和单词x_i的相关性系数$\alpha_{i,j}$，并通过加权计算得到单词x_i的100维知识特征$\boldsymbol{k}_i$。计算概念嵌入$\boldsymbol{c}_{i,j}$和单词x_i之间相关性系数$\alpha_{i,j}$的具体方法如式(2)所示：

$$\alpha_{i,j} \propto \exp\left(\left(\boldsymbol{y}^I_i \oplus \boldsymbol{y}^S_i\right)^{\mathrm{T}} \boldsymbol{W}_c \boldsymbol{c}_{i,j}\right) \tag{2}$$

其中：$\boldsymbol{W}_c$是经过训练所得到的$2d\times100$维权重参数。在得到相关性系数$\alpha_{i,j}$后，通过式(3)进行加权计算得到单词x_i的100维知识特征$\boldsymbol{k}_i$：

$$\boldsymbol{k}_i = \sum_j \alpha_{i,j} \boldsymbol{c}_{i,j} \tag{3}$$

为单词级别的知识特征$\boldsymbol{K}=(\boldsymbol{k}_1,\boldsymbol{k}_2,\cdots,\boldsymbol{k}_T)$添加位置信息，位置信息的计算方法与文献[19]相同。具体地，对于单词x_i，使用式(4)和式(5)计算与之对应的100维位置嵌入$\boldsymbol{p}_i$：

$$\boldsymbol{p}_{(i,2j)} = \sin\left(i/10\ 000^{2j/100}\right) \tag{4}$$

$$\boldsymbol{p}_{(i,2j+1)} = \cos\left(i/10\ 000^{2j/100}\right) \tag{5}$$

其中：i表示单词在句子中的位置；j代表知识特征向量的下标位置。通过将单词x_i所对应的100维单词级别知识特征$\boldsymbol{k}_i$与100维位置嵌入$\boldsymbol{p}_i$按位相加，得到包含位置信息的知识特征$\boldsymbol{k}_i=\boldsymbol{k}_i+\boldsymbol{p}_i$。

1.1.3　知识注意力层

知识注意力层使用前2个模块的输出来计算单词级别的知识上下文向量。

对于任意一个输入语句$X=(x_1,x_2,\cdots,x_T)$中的某一个具体的单词x_i，前述模块将输出单词x_i所对应的意图特征$\boldsymbol{y}^I_i$、语义槽值特征$\boldsymbol{y}^S_i$和知识特征$\boldsymbol{k}_i$。知识注意力层将使用注意力机制来计算单词x_i和知识特征$\boldsymbol{k}_j$的相关性系数$\beta_{i,j}$，并通过式(6)加权计算得到单词x_i的100维知识上下文向量$\boldsymbol{e}_i$。单词x_i和知识特征$\boldsymbol{k}_j$之间相关性系数$\beta_{i,j}$的计算公式如式(6)所示：

$$\beta_{i,j} \propto \exp\left(\left(\boldsymbol{y}^I_i \oplus \boldsymbol{y}^S_i\right)\boldsymbol{W}_k \boldsymbol{k}_j\right) \tag{6}$$

其中：$\boldsymbol{W}_k$是经过训练所得到的$d\times100$维权重参数。然后，通过式(7)进行加权计算得到单词x_i的100维知识上下文向量$\boldsymbol{e}_i$：

$$\boldsymbol{e}_i = \sum_j \left(\beta_{i,j} \boldsymbol{k}_j\right) \tag{7}$$

对知识上下文向量进行层标准化[20]处理，得到与输入语句$X=(x_1,x_2,\cdots,x_T)$所对应的知识上下文向量$\boldsymbol{E}=(\boldsymbol{e}_1,\boldsymbol{e}_2,\cdots,\boldsymbol{e}_T)$。

1.1.4　知识解码器

知识解码器为整个模型提供了一套可以融合知识特征的联合训练机制。该模块以单词级别的语义槽值特征$\boldsymbol{Y}^S=(\boldsymbol{y}^S_1,\boldsymbol{y}^S_2,\cdots,\boldsymbol{y}^S_T)$、意图特征$\boldsymbol{Y}^I=(\boldsymbol{y}^I_1,\boldsymbol{y}^I_2,\cdots,\boldsymbol{y}^I_T)$和知识上下文向量$\boldsymbol{E}=(\boldsymbol{e}_1,\boldsymbol{e}_2,\cdots,\boldsymbol{e}_T)$作为输入，并输出单词级别的意图预测结果$\boldsymbol{O}^I=(\boldsymbol{o}^I_1,\boldsymbol{o}^I_2,\cdots,\boldsymbol{o}^I_T)$和语义槽值检测结果$\boldsymbol{O}^S=(\boldsymbol{o}^S_1,\boldsymbol{o}^S_2,\cdots,\boldsymbol{o}^S_T)$。知识解码器结构如图3所示。

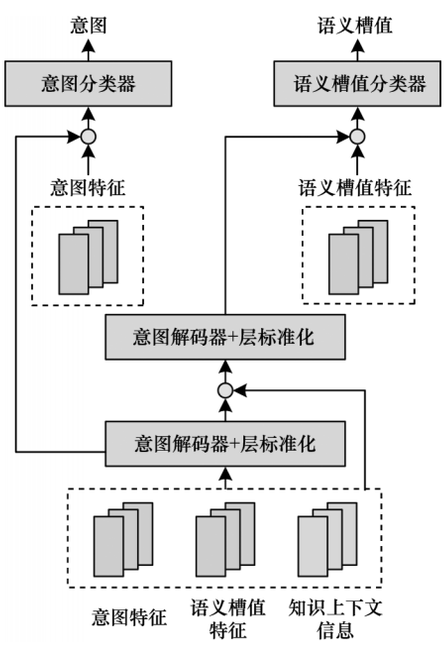

图3　知识解码器的结构

Fig.3　Structure of knowledge decoder

知识解码器包含2个由隐藏层大小为100的双向LSTM实现的解码器，且这2个解码器连接同一个层标准化对输出进行归一化处理。此外，语义槽值解码器会接收意图解码器的输出来提升槽填充子任务的性能表现。对于一个输入语句$X=(x_1,x_2,\cdots,x_T)$，2个编码器接收前几个模块输出的单词级别的语义槽值特征$\boldsymbol{Y}^S=(\boldsymbol{y}^S_1,\boldsymbol{y}^S_2,\cdots,\boldsymbol{y}^S_T)$、意图特征$\boldsymbol{Y}^I=(\boldsymbol{y}^I_1,\boldsymbol{y}^I_2,\cdots,\boldsymbol{y}^I_T)$和知识上下文向量$\boldsymbol{E}=(\boldsymbol{e}_1,\boldsymbol{e}_2,\cdots,\boldsymbol{e}_T)$作为输入，映射出与输入语句所对应的200维单词级别的意图特征$\boldsymbol{U}^I=(\boldsymbol{u}^I_1,\boldsymbol{u}^I_2,\cdots,\boldsymbol{u}^I_T)$和语义槽值特征$\boldsymbol{U}^S=(\boldsymbol{u}^S_1,\boldsymbol{u}^S_2,\cdots,\boldsymbol{u}^S_T)$，经过同一个层标准化计算处理后得到意图特征$\boldsymbol{V}^I=(\boldsymbol{v}^I_1,\boldsymbol{v}^I_2,\cdots,\boldsymbol{v}^I_T)$和语义槽值特征$\boldsymbol{V}^S=(\boldsymbol{v}^S_1,\boldsymbol{v}^S_2,\cdots,\boldsymbol{v}^S_T)$。对于时间步$i$，意图解码器的隐藏层状态$\boldsymbol{h}^I_i$和语义槽值解码器的隐藏层状态$\boldsymbol{h}^S_i$按式(8)和式(9)计算得到：

$$\boldsymbol{h}^I_i=f\left(\boldsymbol{h}^I_{i-1},\boldsymbol{u}^I_{i-1},\left(\boldsymbol{y}^I_i\oplus\boldsymbol{y}^S_i\oplus\boldsymbol{e}_i\right)\right) \tag{8}$$

$$\boldsymbol{h}^S_i=f\left(\boldsymbol{h}^S_{i-1},\boldsymbol{u}^S_{i-1},\left(\boldsymbol{y}^I_i\oplus\boldsymbol{y}^S_i\oplus\boldsymbol{e}_i\oplus\boldsymbol{v}^I_i\right)\right) \tag{9}$$

其中：$\boldsymbol{h}^I_{i-1}$和$\boldsymbol{h}^S_{i-1}$分别为意图解码器和语义槽值解码器的前一个隐藏层状态；$\boldsymbol{u}^I_{i-1}$和$\boldsymbol{u}^S_{i-1}$分别是意图解码器和语义槽值解码器的前一个输出特征；$\boldsymbol{y}^I_i$、$\boldsymbol{y}^S_i$、$\boldsymbol{e}_i$是当前时间步所对应单词x_i的意图特征、语义槽值特征和知识上下文向量；$\boldsymbol{v}^I_i$是标准化后的单词级别意图特征向量。

在得到解码器隐藏层状态后，通过2个由线性层实现的分类器完成单词级别的意图检测和槽填充。其中，对于输入语句$X=(x_1,x_2,\cdots,x_T)$、意图分类器接收单词级别的意图特征$\boldsymbol{Y}^I=(\boldsymbol{y}^I_1,\boldsymbol{y}^I_2,\cdots,\boldsymbol{y}^I_T)$和$\boldsymbol{V}^I=(\boldsymbol{v}^I_1,\boldsymbol{v}^I_2,\cdots,\boldsymbol{v}^I_T)$，通过计算得到单词级别的意图检测结果$\boldsymbol{O}^I=(\boldsymbol{o}^I_1,\boldsymbol{o}^I_2,\cdots,\boldsymbol{o}^I_T)$。对于单词$x_i$，计算过程如式(10)和式(11)所示：

$$\boldsymbol{z}^I_i=\mathrm{softmax}\left(\boldsymbol{W}^I_y\left(\boldsymbol{y}^I_i\oplus\boldsymbol{v}^I_i\right)\right) \tag{10}$$

$$\boldsymbol{o}^I_i=\mathrm{argmax}\left(\boldsymbol{z}^I_i\right) \tag{11}$$

其中：$\boldsymbol{W}^I_y$是可训练参数；$\boldsymbol{z}^I_i$是单词x_i所对应的单词级别意图输出分布；$\boldsymbol{o}^I_i$是该单词所对应的意图。语义槽值分类器的计算过程与意图分类器类似，计算过程如式(12)和式(13)所示：

$$\boldsymbol{z}^S_i=\mathrm{softmax}\left(\boldsymbol{W}^S_y\left(\boldsymbol{y}^S_i\oplus\boldsymbol{v}^S_i\right)\right) \tag{12}$$

$$\boldsymbol{o}^S_i=\mathrm{argmax}\left(\boldsymbol{z}^S_i\right) \tag{13}$$

1.2 意图的合成

对于输入语句$X=(x_1,x_2,\cdots,x_T)$，还需要根据它的单词级别意图检测结果$\boldsymbol{O}^I=(\boldsymbol{o}^I_1,\boldsymbol{o}^I_2,\cdots,\boldsymbol{o}^I_T)$计算得出句子级别的意图预测结果$\boldsymbol{o}^I$。计算方法如式(14)所示：

$$\boldsymbol{o}^I=\mathrm{argmax}\sum_{i=1}^{m}\sum_{j=1}^{n_I}\boldsymbol{\alpha}_j\boldsymbol{l}\left[\boldsymbol{o}^I_i=j\right] \tag{14}$$

其中：m是句子的长度；n_I是意图标签的数量；$\boldsymbol{\alpha}_j$表示一个第j位为1且其他位为0的n_I维0-1向量；argmax表示返回α中最大值的索引操作。

1.3 联合训练

由于模型引入了单词级别的意图特征并进行了单词级别的意图检测和槽填充，为了最大限度地利用这些特征，使用式(15)~式(17)来计算模型的损失值L：

$$L^I\triangleq-\sum_{j=1}^{m}\sum_{i=1}^{n_I}\hat{\boldsymbol{z}}^{i,I}_j\log_a\left(\boldsymbol{z}^{i,I}_j\right) \tag{15}$$

$$L^S\triangleq-\sum_{j=1}^{m}\sum_{i=1}^{n_S}\hat{\boldsymbol{z}}^{i,S}_j\log_a\left(\boldsymbol{z}^{i,S}_j\right) \tag{16}$$

$$L=L^I+L^S \tag{17}$$

其中：L^I和L^S是意图检测和槽填充2个子任务的损失值；$\hat{\boldsymbol{z}}^{i,I}_j$和$\hat{\boldsymbol{z}}^{i,S}_j$是正确的意图标签和语义槽值标签分布；$n_I$和$n_S$是意图标签和语义槽值标签的数量；$L$代表整体损失值。

2 实验

2.1 知识表示和检索

本文选择WordNet和NELL(Never-Ending Language Learner)2个知识库构成模型最终使用的知识库。WordNet是一个英语语义数据库，它根据词汇的细分词义概念对其进行分组，将有相同词义的词汇归为一个同义词集合。WordNet为每一个词义提供了简短的定义，并记录不同的词义之间的语义关系，如felidae和cat的关系是hypernym_of。NELL通过互联网挖掘的方法自动抽取并保存了大量的三元组实体概念知识，如(Google，is a，company)。之所以选择这2个知识库是因为两者存储的知识构成了互补关系，WordNet为模型提供语言学知识，NELL为模型提供真实世界知识，从而帮助模型获取比文本更细粒度的特征信息。

2.2 数据集

通过在ATIS和Snips公开数据集上进行的一系列实验验证了模型出色的性能表现。2个数据集均采用与文献[11]相同的数据划分。ATIS数据集收录了用户预定、查询航班信息时的录音数据，在训练集、开发集和测试集中分别含有4 478、500和893条语句。训练集中还含有21个意图标签和120个语义槽值标签。Snips数据集收录了个人语音助手Snips的用户数据，在训练集、开发集和测试集中分别包含13 084、700和700条语句。训练集中含有7个意图标签和72个语义槽值标签。

2.3 实验设置

使用基于英语的无大小写BERT-Base模型进行微调训练，该模型含有12层、768个隐藏层状态和12个注意力头数。在微调训练过程中，根据模型在开发集上的效果来选择所有超参数的取值。语句最大长度为50，每个单词的最大知识数为80。对于ATIS和Snips数据集，批大小分别被设置为16和64，分类器的遗忘率分别为0.2和0.1。此外，使用初始学习率为5e-5的Adam[21]作为优化器。

2.4 实验结果

模型在2个数据集上的性能表现如表1所示。使用准确率和F1得分来评估模型在意图检测和槽填充2个子任务中的效果，用句子级别的语义准确

率来评估模型的整体性能表现。

表1　本文模型在2个数据集上的性能表现

Table 1　Performance of the model in this paper on two datasets　%

模型	ATIS			Snips		
	Intent	Slot	Sent	Intent	Slot	Sent
BERT	97.5	96.1	88.2	98.6	97.0	92.8
BERT+堆栈	97.5	96.1	88.6	99.0	97.0	92.9
BERT+知识	97.9	96.1	89.1	98.9	97.2	93.3

表1的第1部分是基准模型，由2个基于BERT的联合模型BERT SLU[2]和BERT+Stack-Prop[13]组成。这2个模型均在2019年被提出，且相比基于循环神经网络（Recurrent Neural Network，RNN）[22]、卷积神经网络（Convolutional Neural Network，CNN）[23]、条件随机场（Conditional Random Fields，CRF）[24]及支持向量机（Support Vector Machine，SVN）[25]的传统口语理解模型具有更加优秀的性能表现。其中，BERT SLU仅使用BERT完成口语理解任务，BERT+Stack-Prop则引入堆栈传播机制来提升BERT在口语理解中的性能表现，本文分别使用BERT和BERT+堆栈作为这2个模型的简称。

第2部分则展示了BERT在融合外部知识之后的性能表现，本文简称为BERT+知识。可以看出，融合外部知识后，BERT模型在2个数据集上均有着超过基准模型的性能表现。在ATIS数据集上，与仅使用BERT的BERT SLU相比，在意图准确率中获得了0.4个百分点的提升，且在句子级别的语义准确率这一指标上获得了0.9个百分点的提升。在Snips数据集上，模型在槽填充F1得分中获得了0.2个百分点的提升，且在句子级别的语义准确率这一指标上获得了0.5个百分点的提升。

通过以上分析可以看出：

1）引入外部知识可以提升BERT模型在口语理解任务中的性能表现。

2）引入单词级别的意图特征提高了知识融合的效果。

3）联合训练机制能够利用外部知识计算，得到可以被特定子任务使用的特征信息。

通过设置多组消融实验，从不同的角度验证了上述方法的正确性。

2.5　引入知识的效果

使用以下3个模型进行消融实验来验证引入外部知识对模型性能的提升作用：

1）不引入任何知识，仅让知识解码器以单词级别的意图特征和语义槽值特征作为输入进行联合训练。

2）仅为模型引入WordNet所提供的语言学知识。

3）仅为模型引入NELL所提供的真实世界知识。

可以从表2的第1部分看到3个参与消融实验的模型性能表现。从实验结果可以看出：与融合2种知识模型相比，模型在2组数据集上的性能表现均出现了下滑，且仅融合WordNet知识和仅融合NELL知识的模型在ATIS数据集上的性能表现均优于无任何知识的模型，而在Snips数据集上，仅融合NELL知识的模型虽然在整体的性能表现上略低于不融合外部知识的模型，但在意图检测和槽填充这2个子任务上均优于后者。

表2　融合不同外部知识对模型性能的影响

Table 2　Effect of integrate different external knowledge on model performance　%

模型	ATIS			Snips		
	Intent	Slot	Sent	Intent	Slot	Sent
无知识	97.3	95.9	87.9	98.6	96.4	92.3
仅融合WordNet	97.8	95.8	88.2	98.4	96.8	92.7
仅融合NELL	98.0	96.0	88.6	98.7	96.6	92.0
BERT+知识	97.9	96.1	89.1	98.9	97.2	93.3

从表2可以看出：融入知识模型可以有效提升模型的性能表现，并且在一定程度上体现了WordNet和NELL这2个知识库可以有效互补。

2.6　引入单词级别的意图特征效果

引入一个仅使用句子级别意图特征的模型作为对照，验证引入单词级别的意图特征对模型效果的影响。该模型不使用LSTM对输入语句的每个单词映射出与之对应的单词级别意图特征，而使用BERT编码器输出的特殊向量作为每个单词的意图特征。

从表3所示的实验结果可以看出：引入单词级别的意图特征可以进一步提高模型的性能表现，这可能是因为句子级别的意图特征在融合知识和联合训练的过程中无法有效涵盖每个单词的语义特性。此外，使用句子级别的意图特征也不利于通过注意力机制来计算2个单词间的相关性。

表3　单词级别意图特征对模型性能的影响

Table 3　Effect of word level intent feature on model performance　%

模型	ATIS			Snips		
	Intent	Slot	Sent	Intent	Slot	Sent
句子级意图	97.5	96.1	88.5	98.4	96.7	92.1
BERT+知识	97.9	96.1	89.1	98.9	97.2	93.3

2.7　引入位置信息的效果

在知识融合器中，通过注意力机制来对知识嵌入做加权求和得到单词的知识向量。如果不为知识嵌入加入位置信息，那么这些向量就难以在接下来的注意力计算中反映彼此的位置关系。为了验证上述观点，设置一个不会在知识融合过程中为知识向量添加位置信息的模型作为对照组。

从表4可以看出：引入位置信息可以进一步提升模型的性能表现，也同时验证了上述猜想的合理性。

表4 位置信息对模型性能的影响

Table 4 Effect of location information on model performance %

模型	ATIS			Snips		
	Intent	Slot	Sent	Intent	Slot	Sent
无位置信息	98.0	95.8	88.4	98.6	96.5	91.9
BERT+知识	97.9	96.1	89.1	98.9	97.2	93.3

2.8 引入联合训练机制的效果

引入联合训练机制的消融实验验证了联合训练对模型性能的提升。实验中使用2个模型作为对照：

1)将意图解码器和语义槽值解码器从模型中删除，用知识上下文向量来替代这2个解码器所输出的特征信息。

2)使语义槽值解码器不再以意图解码器输出的特征作为输入，让2个解码器独立运行。

如表5所示，在2个数据集上使用2个独立解码器的模型在性能上优于不含解码器的模型，且低于使用2个相关联解码器进行联合训练的情况。这表明在2个子任务所对应的解码器之间引入联合训练机制可以进一步提升模型的性能表现。

表5 联合训练机制对模型性能的影响

Table 5 Effect of joint training mechanism on model performance %

模型	ATIS			Snips		
	Intent	Slot	Sent	Intent	Slot	Sent
无解码器	97.8	95.7	87.9	98.7	95.8	91.3
独立解码器	97.4	96.1	88.6	98.9	96.5	92.0
BERT+知识	97.9	96.1	89.1	98.9	97.2	93.3

2.9 案例分析

从ATIS数据集中选择2个案例来体现模型在引入外部知识后相比BERT SLU的性能提升。

从表6可以看出：虽然BERT使用大量的文本数据集来进行训练，它仍然无法正确地标记“michigan”的语义槽值。而本文提出的模型却因为融合了NELL提供的真实世界知识而正确地判断了密歇根是美国的一个州而非一座城市。

表6 NELL知识对模型性能的影响

Table 6 Effect of NELL knowledge on model performance

句子	本文模型所预测的正确结果		BERT SLU所预测的错误结果	
	意图	槽值	意图	槽值
please find a flight from las vegas to michigan	atis_flight	O,O,O,O,O,B-fromloc.city_name, I-fromloc.city_name,O,B-toloc.state_name	atis_flight	O,O,O,O,O,B-fromloc.city_name, I-fromloc.city_name,O,B-toloc.city_name

表7所示的案例则体现了本文模型在应对逻辑复杂不清晰的口语语句时具有更强的性能表现。在表7的复杂语句中，本文模型识别出了2个“milwaukee”之间隐含的逗号分隔符，成功地构建出了这条语句中的内部逻辑关系。这是因为WordNet所提供的语言学知识使模型性能得到了提升。

表7 WordNet知识对模型性能的影响

Table 7 Effect of WordNet knowledge on model performance

句子	本文模型所预测的正确结果		BERT SLU所预测的错误结果	
	意图	槽值	意图	槽值
is there one airline that flies from burbank to milwaukee milwaukee to st. louis and from st.louis to burbank	atis_airline	O, O, O, O, O, O, O, B-fromloc. city_name, O, B-toloc.city_name,B-fromloc.city_name,O,B-toloc. city_name, I-toloc. city_name, O, O, B-fromloc. city_name, I-fromloc. city_name, O, B-toloc. city_name	atis_airline	O, O, O, O, O, O, O, B-fromloc. city_name, O, B-toloc.city_name,B-stoploc.city_name,O,B-toloc. city_name, I-toloc. city_name, O, O, B-fromloc. city_name, I-fromloc. city_name, O, B-toloc. city_name

3 结束语

本文提出一种基于BERT的联合模型，旨在验证预训练语言模型融入知识进行迁移学习这一方法在口语理解领域的可行性。通过引入单词级别的意图特征并使用注意力机制为模型融入外部知识，并使用联合训练机制来提升模型在口语理解任务中的性能表现。在ATIS和Snips这2个公开数据集上进行的实验结果表明，该模型相比BERT模型具有更好的性能表现。后续将继续寻找更为高效的可融合知识的联合学习机制，并探索跨语言知识融合方法，通过融合外部知识提升口语理解模型的性能。

参考文献

[1] DEVLIN J, CHANG M W, LEE K, et al. BERT: pre-training of deep bidirectional transformers for language understanding[EB/OL]. [2021-06-20]. https://arxiv.org/abs/1810.04805.

[2] CHEN Q, ZHUO Z, WANG W. BERT for joint intent classification and slot filling[EB/OL]. [2021-06-20]. https://arxiv.org/abs/1902.10909.

[3] YANG A, WANG Q, LIU J, et al. Enhancing pre-trained language representations with rich knowledge for machine reading comprehension[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, USA: Association for Computational Linguistics, 2019: 218-229.

[4] MILLER G A. WordNet[J]. Communications of the ACM,

1995,38(11):39-41.
[5] CARLSON A, BETTERIDGE J, KISIEL B. Toward an architecture for never-ending language learning[C]// Proceedings of the 24th AAAI Conference on Artificial Intelligence. [S. 1.]: AAAI Press, 2010: 365-379.
[6] YANG B S, MITCHELL T. Leveraging knowledge bases in LSTMs for improving machine reading[C]//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Washington D. C., USA: IEEE Press, 2017: 332-346.
[7] GUO D, TUR G, YIH W T, et al. Joint semantic utterance classification and slot filling with recursive neural networks[C]//Proceedings of IEEE Spoken Language Technology Workshop. Washington D. C., USA: IEEE Press, 2015: 554-559.
[8] XU P Y, SARIKAYA R. Convolutional neural network based triangular CRF for joint intent detection and slot filling[C]//Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding. Washington D. C., USA: IEEE Press, 2014: 78-83.
[9] LIU B, LANE I. Attention-based recurrent neural network models for joint intent detection and slot filling[EB/OL]. [2021-06-20]. https://arxiv.org/abs/1609.01454.
[10] ZHANG X D, WANG H F. A joint model of intent determination and slot filling for spoken language understanding[C]//Proceedings of the 25th International Joint Conference on Artificial Intelligence. New York, USA: ACM Press, 2016: 2993-2999.
[11] GOO C W, GAO G, HSU Y K, et al. Slot-gated modeling for joint slot filling and intent prediction[C]//Proceedings of 2018 Conference of the North American Chapter of the Association for Computational Linguistic. Washington D. C., USA: IEEE Press, 2018: 1257-1266.
[12] HAIHONG E, NIU P Q, CHEN Z F, et al. A novel bi-directional interrelated model for joint intent detection and slot filling[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Washington D. C., USA: IEEE Press, 2019: 457-471.
[13] QIN L B, CHE W X, LI Y M, et al. A stack-propagation framework with token-level intent detection for spoken language understanding[C]//Processing of the 9th International Joint Conference on Natural Language Processing. Washington D. C., USA: IEEE Press, 2019: 753-766.
[14] ZHANG Y, WEISS D. Stack-propagation: improved representation learning for syntax[EB/OL]. [2021-06-20]. https://arxiv.org/abs/1603.06598.
[15] HEMPHILL C T, GODFREY J J, DODDINGTON G R. The atis spoken language systems pilot corpus[EB/OL]. [2021-06-20]. https://aclanthology.org/H90-1021/.
[16] COUCKE A, SAADE A, BALL A, et al. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces[EB/OL]. [2021-06-20]. https://arxiv.org/abs/1805.10190.
[17] HOCHREITER S, SCHMIDHUBER J. Long short-term memory[J]. Neural Computation, 1997, 9(8): 1735-1780.
[18] ZHONG V, XIONG C, SOCHER R. Global-locally self-attentive dialogue state tracker[EB/OL]. [2021-06-20]. https://arxiv.org/abs/1805.09655.
[19] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Proceedings of the 31th Annual Conference on Neural Information Processing Systems. Cambridge, USA: MIT Press, 2017: 367-378.
[20] BA J L, KIROS J R, HINTON G E. Layer normalization[EB/OL]. [2021-06-20]. https://arxiv.org/abs/1607.06450.
[21] KINGMA D P, BA J. Adam: a method for stochastic optimization[EB/OL]. [2021-06-20]. https://arxiv.org/abs/1412.6980.
[22] ZAREMBA W, SUTSKEVER I, VINYALS O. Recurrent neural network regularization[EB/OL]. [2021-06-20]. https://arxiv.org/abs/1409.2329.
[23] KIM Y. Convolutional neural networks for sentence classification[EB/OL]. [2021-06-20]. https://arxiv.org/abs/1408.5882.
[24] LAFFERTY J, MCCALLUM A, PEREIRA F C N. Conditional random fields: probabilistic models for segmenting and labeling sequence data[EB/OL]. [2021-06-20]. https://repository.upenn.edu/cgi/viewcontent.cgi?article=1162&context=cis_papers.
[25] BURGES C J, BURGES C. A tutorial on support vector machines for pattern recognition[EB/OL]. [2021-06-20]. https://www.microsoft.com/en-us/research/publication/a-tutorial-on-support-vector-machines-for-pattern-recognition/?from=http%3A%2F%2Fresearch.microsoft.com%2Fpubs%2F67119%2Fsvmtutorial.pdf.

编辑 索书志

(上接第66页)

[18] NEWMAN M E J, GIRVAN M. Fast algorithm for detecting community structure in networks[J]. Physical Review E: Statistical Nonlinear & Soft Matter Physics, 2004, 69(6): 066133.
[19] SHEN M E, PENG M F, LI S J, et al. Optimal island partition of ADN based on complex network community structure [C]//Proceedings of the 9th International Conference on Power Science and Engineering(ICPSE). Washington D. C., USA: IEEE Press, 2020: 17-21.
[20] SUKHWANI H, MARTÍNEZ J M, CHANG X L, et al. Performance modeling of PBFT consensus process for permissioned blockchain network(Hyperledger Fabric)[C]//Proceedings of the 36th Symposium on Reliable Distributed Systems. Washington D. C., USA: IEEE Press, 2017: 253-255.
[21] YU Y, LIU S M, YEOH P L, et al. LayerChain: a hierarchical edge-cloud blockchain for large-scale low-delay industrial Internet of Things applications[J]. IEEE Transactions on Industrial Informatics, 2021, 17(7): 5077-5086.
[22] CASTRO M, LISKOV B. Practical Byzantine fault tolerance [C]//Proceedings of the 3rd Symposium on Operating Systems Design and Implementation. New York, USA: ACM Press, 1999: 173-186.
[23] SALAMA H F. Multicast routing for real-time communication of high-speed networks[D]. Raleigh, USA: North Carolina State University, 1996.
[24] LI W Y, FENG C L, ZHANG L, et al. A scalable multi-layer PBFT consensus for blockchain [J]. IEEE Transactions on Parallel and Distributed Systems, 2020, 32(5): 1146-1160.
[25] WANG G, SHI Z J, NIXON M, et al. SMChain: a scalable blockchain protocol for secure metering systems in distributed industrial plants[C]//Proceedings of International Conference on Internet of Things Design and Implementation. New York, USA: ACM Press, 2019: 249-254.

编辑 陆燕菲