文章编号：2095-2163(2022)09-0050-06 中图分类号：TP391.1 文献标志码：A

基于任务型对话系统的电子病历结构化录入系统设计

程路易，王志军
（东华大学 计算机科学与技术学院，上海 201620）

摘 要：电子病历包括多个业务领域的临床信息，涉及到大量信息录入场景。然而，部分场景中存在使用键盘录入数据不便的情况。针对于此，本文设计了一种基于任务型对话系统的电子病历结构化录入系统，可使用户通过人机对话的方式，完成数据的录入并自动进行结构化。针对当前的语音识别与对话系统技术存在错误级联的问题，增加了语音纠错模块，让用户能够通过语音，修改结构化抽取结果。针对当前系统可扩展性差，难以适应新的抽取需求的问题，提出了一种基于可扩展对话状态追踪模型，能在不修改模型结构的情况下，训练新增的结构化内容。实验结果证明了系统的有效性和可扩展性。

关键词：任务型对话系统；语音交互；电子病历；信息抽取

The electronic medical record entry system based on task-oriented dialogue system

CHENG Luyi, WANG Zhijun
(School of Computer Science and Technology, Donghua University, Shanghai 201620, China)

【Abstract】 The electronic medical record contains clinical information in multiple domains. There are large amount of scenarios with data entry within these domains. However, using the keyboard to input data in some scenarios is inconvenient, including recording pathological examination records and hospitalization records. In response to this situation, the paper designs a task-oriented dialogue system to complete both the data entry task and the automatic structural task through human-machine dialogue. Current speech recognition and dialogue system technology suffer from cascading errors. Consequently, a modification stage is added after the initial input, allowing the users to change the attribute with the voice. Due to the poor scalability of the current system, it is hard to meet the demand of extracting new attributes without changing the structure of the model. The scalable dialogue state tracking model is designed to train newly added attributes without changing the structure of the model. Experiments are conducted on two different scenarios. The result shows the system is effective and scalable.

【Key words】 task-oriented dialogue system; voice interaction; electronic medical record; information extraction

0 引 言

随着信息化建设的持续推进，电子病历在医院中得到了广泛的应用。由于电子病历包括多个业务领域临床信息，如病历概要、门(急)诊诊疗记录、住院诊疗记录等，因此涉及到大量信息录入场景。目前主要是由医生通过键盘手工输入信息，但在部分场景下存在不便用键盘录入数据的情况。例如：在病理检查场景下，医生手持器械检查时双手被占用，无法再使用键盘进行录入；在住院诊疗场景下，医生需要与患者近距离交流，没有时间和空间使用键盘进行记录等。

为此，本文提出使用语音交互的方式来进行电子病历录入。已有的工作是将任务分为 2 步。第一步是语音识别，将语音转换为文字；第二步是信息抽取，将文本进行结构化，获得包含样本、属性、属性值的三元组。然而，这类方法存在一些具有挑战性的问题：

(1)录入文本过长。不同于普通的语音查询场景，如询问天气[1]，餐厅搜索[2]等，电子病历文本相对较长。传统的基于键盘的输入方式影响并不大，但切换到基于语音的输入方式将成为一个挑战。

(2)错误级联。当前的语音识别和信息抽取方法的效果都还不够好，使用 pipeline 的方法会产生错误级联问题，导致最终录入效果不佳。

(3)可扩展性差。由于当前的方法需要先对属性进行充分训练后才可以用于属性抽取，难以适用新属性的抽取。

针对上述挑战，本文设计了一个基于任务型对话系统[3]的电子病历结构化录入系统。提出了基于任务型对话系统的语音交互方式，能够让用户进行分段式语音录入；在当前语音识别和分段式语音录入的基础上，增加了语音纠错阶段，通过基于语音对话的方式，对结构化抽取错误的结果进行纠正；提出了一种

作者简介：程路易(1993-)，男，硕士研究生，主要研究方向：自然语言处理-对话系统；王志军(1973-)，男，博士，副教授，硕士生导师，主要研究方向：物联网信息服务、数据库、信息检索。
通讯作者：王志军 Email：zjwang@ dhu.edu.cn

收稿日期：2022-03-02

基于可扩展对话状态追踪的模型，仅需要属性的一段描述信息，即可进行信息抽取。本文在病理检查记录录入、住院诊疗记录录入两个场景中进行了实验，实验结果证明了系统的有效性和可扩展性。

1　系统架构

系统由语音识别、语音合成以及对话系统三部分组成，总体架构如图 1 所示。首先在系统前端录制用户语音，使用语音识别模块将语音转换为文本；将文本输入后端的任务型对话系统，系统将从文本中抽取出属性值并产生系统语段，一起发送至前端；再使用语音合成模块将系统语段转换成语音，并将抽取出的属性值在前端进行展示，同时等待用户下一轮的输入。以上流程确保了系统前后端之间传输的数据都是文本，减少了前后端间传输的数据量。

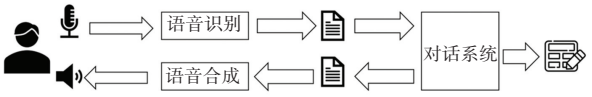

图 1　系统总体架构

Fig. 1　Overall architecture of the system

对话系统模块包含 2 个阶段：录入阶段、纠错阶段。其中，录入阶段的任务是让用户通过多轮对话的方式，完成电子病历结构化录入。输入是当前用户语段、对话历史、属性描述等内容；输出是属性（槽）、属性值对以及系统语段。当用户告知系统“录入完毕”时，系统自动进入纠错阶段。纠错阶段的任务是直接使用语音指令，修改录入有误的属性值。当用户表示没有需要修改的属性后，整个流程结束。

1.1　录入阶段架构

录入阶段架构如图 2 所示。图 2 中，录入阶段主要由 3 个模块组成，包括：对话状态追踪、对话策略以及自然语言生成。其中，对话状态追踪模块的输入包括当前用户语段、对话历史语段以及属性的描述，输出是对话状态，具体包括一组槽、值对。此模块包含 2 个子模块：可分类槽模块与不可分类槽模块，分别用于预测有预设候选值、没有预设候选值的属性的值，最终合并成对话状态。对话策略模块基于规则，输入是对话状态，输出是系统动作。自然语言生成模块同样基于规则实现，输入对话动作，输出系统回复。系统不断重复整个过程，直到用户表示录入完毕，进入纠错阶段。

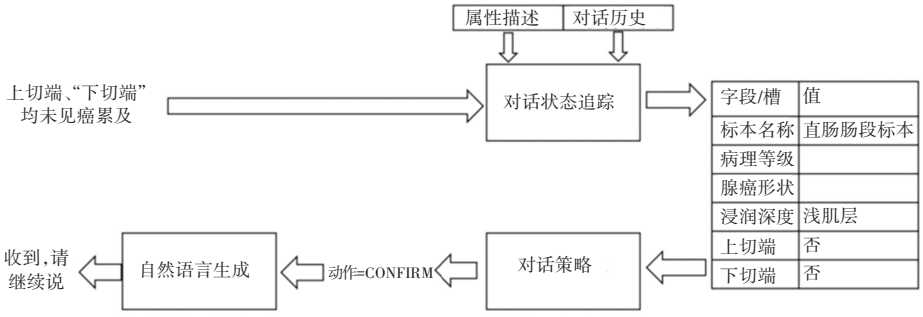

图 2　录入阶段架构

Fig. 2　Architecture of the data entry stage

1.2　纠错阶段架构

纠错阶段架构如图 3 所示。图 3 中，纠错阶段主要由自然语言理解、对话策略以及自然语言生成模块组成。其中，自然语言理解模块的输入是当前用户语段，输出是从语段中抽取出的一个槽的名称及其值；纠错阶段剩余 2 个模块功能与录入阶段中的相应模块类似，当用户表示没有需要修改的内容后，整个流程结束。

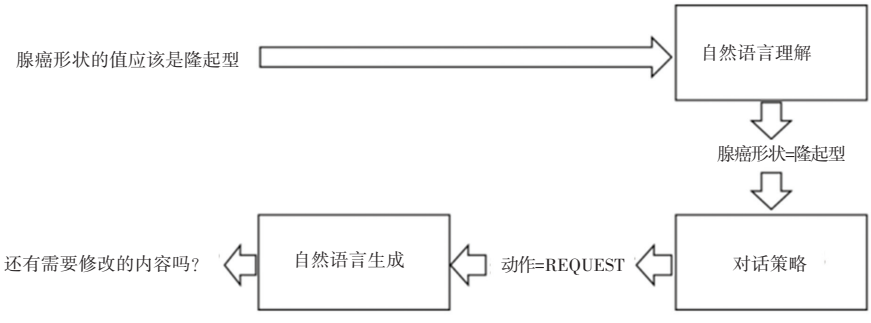

图 3　纠错阶段架构

Fig. 3　Architecture of the modification stage

2 系统实现

系统后端使用 Python 语言开发。其中,深度学习部分主要使用基于 Pytorch[4] 的 Huggingface Transformers[5] 框架,使用 bert-base-chinese 预训练语言模型;Web 服务器部分使用 Flask 框架实现。前端使用 JavaScript 语言开发,界面部分主要使用了基于 React 的 Ant Design 组件库。

2.1 后端对话系统的实现

2.1.1 数据集的构建

对于录入阶段,需要解决原始数据集中的文本与实际场景不匹配的问题。原始数据集中,每条数据是一段约 100~300 字的文本,而实际系统需要用户使用多轮对话的方式进行交互。因此,需对原始数据集的文本进行如下处理:

首先,将一整段文本依据逗号、句号、分号等标点符号进行切分,形成多个短句,再对这些短句进行合并,形成用户语段,确保每个语段中包含 1~3 个短句。接下来,需要在每个用户语段之后生成系统语段。本文使用了构建模板的方式,构建了一些简单的回复,如“收到,请继续说”、“好的,请说下一句”等。在整个对话结束时,则插入表示结束录入的语段,如“录入完毕”。最后,使用直至当前用户轮的所有对话历史作为模型语段部分的输入。

在纠错阶段,本文构建了一个符合使用自然语言修改属性值场景的数据集。方法如下:

首先,对 2 个数据集中的每个属性,获取可能值的集合。然后定义一组模板,预设一些用户需要修改属性值时可能说的话,如“把 KEY 的值修改为 VALUE”等。最后,将属性及其所有可能的值填入模板中相应位置生成样本,并使用 BIO 形式[6] 进行标注,见表 1。

表 1 纠错阶段的标注形式

Tab. 1 Annotation method of modification stage

字符	腺	癌	形	状	的	值	应	该	是	隆	起	型
标注	B-K	I-K	I-K	I-K	O	O	O	O	O	B-V	I-V	I-V

2.1.2 录入阶段对话状态追踪

为了使系统具有一定的可扩展性,本模块将可分类、不可分类槽的预测,分解为可分类槽填充与不可分类槽填充两个子任务,并建模为机器阅读理解问题。录入阶段对话状态追踪模块架构如图 4 所示。把对话历史、当前用户语段作为阅读理解中的篇章,并在各属性标注规范的基础上总结出描述,作为阅读理解中的问题。对于可分类槽填充模块,需要完成多选阅读理解任务,将描述与属性的候选值进行拼接,使每个候选值都成为一个候选项,模型将从中选出一个选项作为槽的值,具体标注形式,见表 2。对于不可分类槽填充模块,需要完成抽取式阅读理解任务,直接将描述作为问题,模型将预测出属性值在语段中的起始位置与结束位置,具体标注形式见表 3。

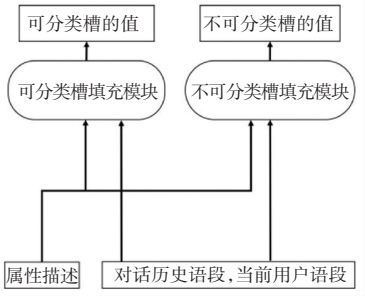

图 4 录入阶段对话状态追踪模块架构

Fig. 4 Architecture of the dialogue state tracking module

表 2 可分类槽的标注形式

Tab. 2 Annotation method of categorical slots

对话历史/篇章	浸润至浆膜外纤维脂肪组织,脉管内见癌栓。
描述、选项 1	是否侵犯脉管,见癌栓:是
描述、选项 2	是否侵犯脉管,见癌栓:否
描述、选项 3	是否侵犯脉管,见癌栓:无
正确选项	选项 1

表 3 不可分类槽的标注形式

Tab. 3 Annotation method of non-categorical slots

对话历史/篇章	浸润至浆膜外纤维脂肪组织,脉管内见癌栓。
描述/问题	相邻最表浅处上皮与间质交界处到肿瘤浸润最深处的距离
起始位置	3
结束位置	12

模块输入包括 3 部分:属性描述、对话历史、当前用户语段。其中,当前用户语段由用户从前端以 json 格式发送至后端,属性描述可使用配置文件的形式,存储在后端。对话历史信息存储在后端,在自然语言生成模块输出系统语段之后,系统将当前用户语段与当前系统语段存入对话历史中。

预测结果将直接写入对话状态中,覆盖原有的值。系统中的对话状态同样使用 json 格式存储,具体结构为:{“槽 1 的名称”:“值”,“槽 2 的名称”:“值”……,“结束录入”:“值”},以便于发送至前端

进行展示。对所有槽都完成预测后，将对话状态传入对话策略模块的同时，将其发送至前端，使得用户能够在说完每个语段后，看到系统的反馈。此外，当用户需要录入一条全新的病历时，对话状态追踪模块会清空对话状态以及对话历史。

2.1.3　纠错阶段自然语言理解

在纠错阶段，用户通过一个语段说明需要修改的属性与属性值，对相应属性进行修改。相比录入阶段对话状态的追踪任务，本阶段无需考虑对话历史，只需从当前轮中抽取出修改所需的属性与属性值。因此，使用传统任务型对话系统中广泛运用的自然语言理解模块，通过序列标注的方式，识别出语段中的属性名称与属性值。

模块所使用的模型结构与文献[7]中提出的单句标注任务模型基本相同。模型将按照 2.1.1 节所标注的标签，为每个字符进行分类。将 Outside（*O*）标签之外所有标签的 Begin（*B*）与 Inside（*I*）部分所对应的单词进行合并，产生实体。经合并后，共有 *K*、*V*、*O* 三种标签形式。以表 1 为例，标签为 *K* 的实体为“腺癌形状”，标签为 *V* 的实体为“隆起型”。本模块将找到标签为 *K* 的实体与所有槽的名称进行匹配。由于实际用户输入是通过语音识别产生的，很可能无法完全匹配。因此，使用计算编辑距离的方式，计算标签为 *K* 的实体与槽的名称相似度，并选取分数最高的作为需要修改的属性。如：语音识别结果为“癌形状”，该结果与“腺癌形状”的相似度为 1−1/(3+4)= 0.86，而与“标本大小”的相似度为 1−(3+4)/(3+4)= 0。因此，需要修改的属性为“腺癌形状”。同时，设置一个阈值，若相似度最高的属性分数小于阈值，则不修改任何内容。此外，模块同样使用计算编辑距离的方式将输入语段与表示结束纠错的一组语段进行匹配，若相似度大于阈值，则认为用户纠错完毕。模块的输出依然使用 json 格式，若存在需要修改的值，输出的用户动作为｛需要修改的属性名：值｝。如果用户表示结束纠错，则输出为｛结束修改：是｝。

2.1.4　对话策略与自然语言生成

系统对话策略模块的主要作用，是根据录入阶段对话追踪输出的对话状态，或是纠错阶段自然语言理解模块输出的槽、值对，产生对应的系统动作，控制自然语言生成模块的输出。自然语言生成模块的作用是将系统动作转化为自然语言。

需要产生的系统动作主要用于系统处理完当前的用户输入后，给用户一个反馈，引导用户进行下一轮对话。因此，采用基于规则的对话策略模块，具体规则如下：

（1）录入阶段。如果输入的对话状态中，“结束录入”槽的值为“否”，则输出系统动作｛动作：CONFIRM｝；否则，输出系统动作｛动作：REQUEST｝。同时，向前端发送 json，内容为｛inputend：True｝，使得前端进入纠错阶段。

（2）纠错阶段。若输入的用户动作中包含｛需要修改的属性名：值｝，则以 json 的形式向前端发送此内容，同时输出系统动作｛动作：REQUEST｝。直到输入中包含｛结束修改：是｝时，模块会向前端发送 json，内容为｛modend：True｝。通知前端整个录入流程已完全结束，模块不再输出任何系统动作，结束后端的流程。

系统采用基于模板的自然语言生成模块，当输入的系统动作为 CONFIRM 时，系统准备了多种表示确认的短句，如：“收到，请继续说”、“好的，请说下一句”等。当输入动作为 REQUEST 时，系统同样预置了多种询问用户是否需要继续进行修改的语句。本模块在收到系统动作后，会从对应的模板中随机选择一个语句进行回复，使得系统产生的回复从用户的角度看，具有一定的多样性。同时，有效地避免了模块可能产生不确定、不安全回复的可能。

2.2　前端语音交互

为了节省前后端之间数据传输所需带宽，系统完全在前端实现与用户的语音交互。由于系统整体采用 B/S 架构，需要在浏览器中完成录音与播放、调用第三方语音识别、语音合成 API。系统采用发送事件的方式，完成这些任务之间的同步。各任务具体实现细节如下：

（1）录音任务需要完成的内容主要包括：获取录音权限、开始录音、停止录音。系统直接调用浏览器内置的 WebAudio 接口，完成麦克风权限的获取以及音频录制。经过调研发现，不同类型的麦克风所需的冷启动时长有所不同。冷启动时长是指从系统初次获取到录音权限起，直至录音设备能够录制用户的语音为止。如果在取得录音权限后立即开始录制音频，会造成录音开头部分的音频丢失。因此，在用户授权使用麦克风后，系统播放一段音频，引导用户开始语音录入。在开始播放引导音频的同时让麦克风开始录音，在音频播放结束的前一刻停止录音，使麦克风获得足够的冷启动时间，能够完整录制用户的声音。在用户说完当前语段后，系统将自动停止录音。尽管一些第三方的语音识别 API 本身

提供此功能,但效果并不理想,无法及时停止录音,导致上传的音频存在大段空白,不利于语音识别。本系统利用 WebAudio API,以一定的时间间隔检测当前录音的音量,当出现连续三个时间点的音量低于一定阈值时停止录音。为了缓解录音中包含大段空白或是录音提前终止的情况,在录入阶段与纠错阶段,设置了不同的时间间隔,一定程度上改善了用户体验。

(2)录制的音频流将被发送至第三方语音识别 API 进行语音识别。本系统使用讯飞语音识别 API,由于该 API 使用 Websocket 协议通信,被浏览器允许直接进行“跨域”请求,系统将直接音频发送至讯飞服务器,减少了前后端间数据的传输。在收到了后端对话系统返回的文本后,发送至讯飞文字转语音 API 进行语音合成。其中,主要调节的参数是播报的语速,让用户感到系统能够及时对每个输入做出响应。系统最后将合成的音频流使用 WebAudio API 进行播放。

3 实例验证与分析

3.1 实验环境

本文实验环境设置见表 4。

表 4 实验环境

Tab. 4 Experimental setting

环境属性	环境属性值
Python	3.9
Pytorch	1.7
Huggingface Transformers	4.2.2
操作系统	Gentoo Linux
内存	64 G
显卡	Nvidia Geforce RTX 3090

3.2 数据集

本次实验获取到 2 个不同录入场景下的数据集。对此拟做阐释分述如下。

场景 *A* 的数据来源于上海市某大型三甲医院的肠癌病理检查数据,包含原始诊断文本以及 12 个属性。其中,淋巴结转移情况、基底切端、上切端、下切端、神经侵犯、脉管侵犯为有预设候选值:是、否、无的属性;标本名称、肿瘤大小、浸润深度、病理等级、分化程度、腺癌形状为无预设候选值。训练集、测试集中包含的数据量分别为 1 672、634。

场景 *B* 的数据选取 2019 全国知识图谱与语义计算大会(CCKS2019)面向中文电子病历的医疗实体识别任务,以及属性抽取任务中的部分数据。其中包含原始文本以及疾病和诊断、影像检查、实验室检验、手术、药物、解剖部位等 6 个无预设候选值的属性。训练集、测试集中数据量分别为 998、378 条数据。

为了能够模拟语音录入、语音纠错的实际情况,将按 2.1.1 节方法中生成的对话文本,输入到讯飞语音合成中产生语音,再将语音传入讯飞语音识别模块,最终生成语音识别后的文本。针对用于语音修改的文本,本文穷举了所有属性的所有值,配合一些预定的模板,生成了用于语音修改的语段,再经过语音合成与语音识别,得到最终的纠错指令。

由于数据集中的每条文本记录包含多个属性,本文分属性统计属性值预测正确记录的条数,并将其与总记录条数的比值作为该属性的准确率。

3.3 实验结果与分析

实验结果见表 5、表 6。

表 5 场景 *A*

Tab. 5 Results of scenario *A*

输入	属性											
	标本名称	肿瘤大小	浸润深度	病理等级	分化程度	腺癌形状	淋巴转移	基底切端	上切端	下切端	神经侵犯	脉管侵犯
录入	71.92	95.89	81.38	55.04	98.10	95.89	92.90	94.47	91.00	92.74	99.21	99.21
纠错	91.48	98.10	96.84	100.00	98.26	100.00	99.52	99.84	99.68	99.52	100.00	100.00

表 6 场景 *B*

Tab. 6 Results of scenario *B*

输入	属性					
	疾病和诊断	影像检查	实验室检验	手术	药物	解剖部位
录入	53.43	83.86	74.07	84.39	86.77	38.62
纠错	84.65	95.76	88.09	89.15	95.76	87.83

从录入阶段的结果可以观察到，场景 A、B 中部分属性准确率较低，如病理等级、解剖部位等，导致录入效果不佳。其中，有预设候选值的属性准确率都在 90%以上，而无预设候选值的准确率都相对较低，原因是预设候选值为“是”、“否”、“无”，语音识别的正确率更高。其次，从纠错阶段的结果能够看到，纠错阶段大幅提升了属性的准确率，场景 A 中所有属性的准确率均超过 90%。实验证明，本系统使用同一模型，能够在 2 个场景中完成录入任务，说明系统具有一定的可扩展性。

4　结束语

本文设计了基于任务型对话系统的电子病历结构化录入系统，能够让用户以人机对话的形式完成结构化录入。并且增加了纠错阶段，让用户能够通过语音，修改录入有误的属性值，还通过实验验证了纠错阶段的有效性。同时，该系统能够使用一个模型完成 2 个不同场景的录入任务，表明系统具有一定的可扩展性。

参考文献

[1] ZUE V, SENEFF S, GLASS J, et al.Juplter: a telephone-based conversational interface for weather information [J]. IEEE Transactions on Speech and Audio Processing, 2000, 8(1): 85-96.

[2] HENDERSON M, THOMSON B, WILLIAMS J D. The second dialog state tracking challenge [C]//Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL). Philadelphia, PA, U.S.A.: Association for Computational Linguistics, 2014: 263-272.

[3] ZHANG Z, TAKANOBU R, ZHU Q, et al. Recent advances and challenges in task-oriented dialog systems [J]. Science China Technological Sciences, 2020, 63(10): 2011-2027.

[4] PASZKE A, GROSS S, MASSA F, et al. Pytorch: An imperative style, high-performance deep learning library[C]//WALLACH H, LAROCHELLE H, BEYGELZIMER A, et al. Advances in Neural Information Processing Systems. Vancouver, Canada: Curran Associates, Inc.,2019, 32:8026-8037.

[5] WOLF T, DEBUT L, SANH V, et al. Transformers: State-of-the-art natural language processing[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Online: Association for Computational Linguistics, 2020: 38-45.

[6] RAMSHAW L A, MARCUS M P. Text chunking using transformation-based learning [M]// ARMSTRONG S, CHURCH K, ISABELLE P, et al. Natural language processing using very large corpora. text, speech and language technology. Dordrecht: Springer, 1999,11: 157-176.

[7] DEVLIN J, CHANG M W, LEE K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding[C]// Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Minneapolis, Minnesota: Association for Computational Linguistics, 2019: 4171-4186.

(上接第 49 页)

[7] AZAR S M, ATIGH M G, NICKABADI A, et al. Convolutional relational machine for group activity recognition [C]// 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, USA:IEEE, 2019:7884-7893.

[8] QI Mengshi, WANG Yunhong, QIN Jie, et al. stagNet: An attentive semantic RNN for group activity and individual action recognition[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2019,30(2):549 - 565.

[9] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks [J]. arXiv preprint arXiv: 1609.02907, 2016.

[10] LU Lihua, LU Yao, YU Ruizhe, et al. GAIM: Graph attention interaction model for collective activity recognition [J]. IEEE Transactions on Multimedia, 2020 ,22(2): 524-539.

[11] SHU Tianmin , TODOROVIC S, ZHU Songchun. CERN: Confidence-energy recurrent network for group activity recognition [C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, USA:IEEE, 2017:4255-4263.

[12] SZEGEDY C, VANHOUCKE V, IOFFE S, et al. Rethinking the inception architecture for computer vision [C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA:IEEE, 2016:2818-2826.

[13] HE K, GKIOXARI G, DOLLÁR P, et al. Mask R-CNN[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(2):386-397 .

[14] YUAN Hangjie, NI Dong. Learning visual context for group activity recognition [C]// 35th AAAI Conference on Artificial Conference. AAAI ,2021: 3261-3269.

[15] GAVRILYUK K, SANFORD R, JAVAN M, et al. Actor-transformers for group activity recognition[C]// 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2020:839-848.

[16] SHU Xiangbo, ZHANG Liyan, SUN Yunlian, et al. Host-parasite: Graph LSTM-in-LSTM for group activity recognition [J]. IEEE Transactions on Neural Networks and Learning Systems,2021,32(2):663-674.

[17] LU Lihua, LU Yao, WANG Shunzhou. Learning multi-level interaction relations and feature representations for group activity recognition[M]// MultiMedia Modeling. MMM 2021. Lecture Notes in Computer Science( ). Cham: Springer, 2021, 12572: 617-628.