

# 基于迭代信息传递和滑动窗口注意力的问题生成模型研究

陈千<sup>1,2</sup>, 高晓影<sup>1\*</sup>, 王素格<sup>1,2</sup>, 郭鑫<sup>1\*</sup>

<sup>1</sup>山西大学 计算机与信息技术学院, 山西 太原 030006

<sup>2</sup>山西大学 计算机智能与中文信息处理教育部重点实验室, 山西 太原 030006

{chenqian}@sxu.edu.cn, {870733176}@qq.com

{wsg}@sxu.edu.cn, {guoxinjsj}@sxu.edu.cn

## 摘要

知识图谱问题生成任务是从给定的知识图谱中生成与其相关的问题。目前, 知识图谱问题生成模型主要使用基于RNN或Transformer对知识图谱子图进行编码, 但这种方式丢失了显式的图结构化信息, 在解码器中忽视了局部信息对节点的重要性。本文提出迭代信息传递图编码器来编码子图, 获取子图显式的图结构化信息, 此外, 我们还使用滑动窗口注意力机制提高RNN解码器, 提升子图局部信息对节点的重要度。从WQ和PQ数据集上的实验结果看, 我们提出的模型比KTG模型在BLEU-4指标上分别高出2.16和15.44, 证明了该模型的有效性。

**关键词:** 知识图谱; 问题生成; 图神经网络

## Question Generation Model Based on Iterative Message Passing and Sliding Windows Hierarchical Attention

Qian Chen<sup>1,2</sup>, Xiaoying Gao<sup>1\*</sup>, Suge Wang<sup>1,2</sup>, Xin Guo<sup>1\*</sup>

<sup>1</sup>School of Computer and Information Technology, Shanxi University, Taiyuan, Shanxi 030006, China

<sup>2</sup>Key Laboratory Computational Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University, Taiyuan, Shanxi 030006, China

{chenqian}@sxu.edu.cn, {870733176}@qq.com

{wsg}@sxu.edu.cn, {guoxinjsj}@sxu.edu.cn

## Abstract

Knowledge graph question generation task is to generate related questions from a given knowledge graph. In recent years, the knowledge graph problem generation models mainly use RNN or Transformer to encode the knowledge graph subgraph, but those methods lose the explicit graph structure information, and the importance of local information to the node is ignored in the decoder. We propose an Iterative Message Passing graph encoder to encode sub-graphs and obtain the explicit structured information of the subgraphs. In addition, we also use the Sliding Windows Hierarchical Attention mechanism to improve the RNN decoder and increase the importance of the local information of the sub-graphs to nodes. We conduct experiments on WQ and PQ data sets, and the performance of the model is significantly higher than that of the baseline model.

**Keywords:** Knowledge graph, Question generation, Graph neural network

©2021 中国计算语言学大会

**基金项目:** 山西省国际合作重点研发计划项目(NO.201803D421024); 山西省应用基础研究计划项目(NO.201901D111032); 国家自然科学基金项目(NO.61502288, NO.61403238) This work was supported by Applied Basic Research Project of Shanxi Province (No. 201901D111032) and the Natural Science Foundation of China (No. 61502288, NO.61403238).

## 1 引言

问题生成任务指在给定形式的数 据（如文本 (Du et al., 2017; Song et al., 2018a)、图像 (Li et al., 2018)、表格 (Bao et al., 2018)或知识图谱 (Seyler et al., 2017; Kumar et al., 2019; Bi et al., 2020)）生成对应问题的任务。本文主要研究在给定的知识图谱上生成对应问题的任务，该研究具有广泛的应用前景，并且越来越受到学术界和工业界的关注。如为了训练一个基于知识图谱的问答系统神经网络，需要大规模的问答对训练样本，但人工标注训练样本的成本很高，且规模较小，问题生成任务尤其是在知识图谱上生成高质量的问题对（样本）成为解决该问题的有效手段；基于知识图谱的问题生成技术在聊天机器人、智能客服、阅读理解系统等诸多下游应用中具有重要应用价值。

目前，知识图谱问题生成任务普遍使用编码器-解码器框架将知识图谱子图作为输入生成对应的问题 (Serban et al., 2016; Elsahar et al., 2018)。其中，使用知识图谱中一个三元组生成问题的研究中，Elsahar et al. (2018)用额外的上下文信息丰富编码器，并为解码器提供注意力机制和复制机制提升生成问题的质量。Liu et al. (2019)提出一个包含多种上下文和可感知答案的损失函数的编码-解码器来表达给定的谓词和答案。虽然这些模型能够生成高质量的问答对，它们没有解决多跳推理问答任务中问答对短缺的问题。Kumar et al. (2019)提出用基于Transformer的模型从知识图谱子图（即多个三元组）中生成难度可控的多跳问题。Bi et al. (2020)提出在知识图谱子图上使用额外知识库生成语义丰富且多样性的问题。然而他们只是将知识图谱子图看做是多个三元组的集合，没有利用子图本身显式图形结构信息。为了解决这个问题，Chen et al. (2020)提出利用双向图编码器获取知识图谱子图的显示结构信息、生成有目标答案的问题的Graph2Seq模型。但他忽视了在编码器中知识图谱子图中独特的二部图结构以及在解码器中来自窗口大小相邻节点的局部信息对节点的重要程度。

为了应对上面提及到的挑战，首先，我们利用知识图谱子图独特的二部结构进行信息传递扩展图神经网络编码器，学习具有多关系边的知识图谱子图信息，用来解决第一个挑战。此外，我们将Graph2Seq思想应用到知识图谱问题生成中，用来解决第二个挑战。最后我们提出滑动窗口分层注意力机制提升知识图谱子图窗口大小相邻节点带来的局部信息对节点的重要程度，用来解决第三个挑战。

本文的主要贡献如下：

a.本文提出了用于从知识图谱子图中生成问题的Graph2Seq模型，使用一种沿着知识图谱子图的拓扑结构迭代传递节点和边信息完成图表示的方法，提高具有多关系边的知识图谱子图表示。

b.本文使用滑动窗口分层注意力机制扩展RNN解码器，能够生成问题词时，提升知识图谱子图窗口大小相邻节点带来的局部信息对节点的重要程度。

c.本文方法在WebQuestions和PathQuestions两个数据集上BLEU-4指标均有明显提升。

## 2 相关工作

传统问题生成技术主要使用基于模板的方法在知识图谱上生成问题 (Seyler et al., 2015; Song and Zhao, 2016; Seyler et al., 2017)，但模板的设计需要大量的人力，且当任务领域不同时，系统泛化性较低，基本没有可扩展性。随着神经网络的发展，编码器-解码器框架被用于知识图谱问题生成任务中。Serban et al. (2016)首次提出将知识图谱三元组用于回答自然语言问题的神经网络模型，并创建了30M Factoid Question-Answer语料库。然而，他们的方法需要大量的客观问题对作为训练数据。为了解决这个问题，Song et al. (2016)提出一种不需要人工设计模板的无监督系统，解决从特定领域的知识图谱中生成问题。Indurthi et al. (2017)提出一种基于RNN问题生成模型，用于从知识图谱中生成问答对。然而，这些方法 (Serban et al., 2016; Elsahar et al., 2018; Indurthi et al., 2017)仅仅使用单个三元组生成简单的问题，且采用的RNN编码器不能有效处理具有图形结构的数据。Kumar et al. (2019)提出基于Transformer的编码-解码器用于编码知识图谱子图，同时生成多跳且难度可控的问题，这是首次使用神经网络从知识图谱子图（即多个三元组）生成问题的模型。为了生成多样性且更流畅的问题，Bi et al. (2020)提出使用额外知识库获取更多信息生成高质量的问题，但以上的方法只是将知识图谱子图看做成一个三元组集合，没有充分利用知识图谱子图显式的结构特征，也没有充分利用三元组之间相互传递的信息。受场景图生成任务Xu et al. (2017)的启发，本文提出一种沿着知识图谱子图拓扑结构迭代传递节点和边信息的图编码器，有效获取子图的显式结构信息，提高具

有多关系边知识图谱子图表示。Chen et al. (2020)提出利用双向图编码器获取知识图谱子图的显示结构信息、生成有目标答案的问题的Graph2Seq模型，但他们解码器中的注意力机制是普通的全局注意力，不能提高来自窗口大小相邻节点的局部信息对节点的重要程度。本文使用滑动窗口分层注意力机制来关注来自窗口大小相邻节点的局部信息，准确得到节点注意力权重分布和词典中每个词的分布，进一步完善了解码器中对知识图谱子图节点丰富信息的获取。

### 3 问题生成模型

#### 3.1 模型框架

知识图谱问题生成任务是指给定知识图谱子图，生成与子图相对应的问题。

给定知识图谱子图 $G = \{v, e\}$ 和子图中答案节点 $V^a$ ，生成由单词序列 $\hat{Y} = \{y_1, y_2, \dots, y_p\}$ 组成的与答案最相关的问题，如式 (1) 所示。

$$\hat{Y} = \arg \max_Y \{Y | G, V^a\} \tag{1}$$

具体示例如表1所示。

知识图谱	(i_duke_of_savoy,parents,ix_duke_of_savoy), (ix_duke_of_savoy,parents,louis_duke_of_savoy), (louis_duke_of_savoy,place_of_death,lyon)
答案	lyon
标准答案句	where is the grandplace of death of i duck of savoy's mom ?

Table 1: 知识图谱问题生成的相关示例

本文提出一种基于迭代信息传递和滑动窗口注意力的问题生成模型 (Iterative Message Passing and Sliding Windows Hierachical based Question Generative Model, ImpSWH-QG)，如图1所示。该模型由节点和边编码模块、迭代信息传递 (Iterative Message Passing, Imp) 图编码器模块和基于滑动窗口分层 (Sliding Windows Hierachical, SWH) 注意力的RNN解码器模块构成。该模型首先将知识图谱子图和答案输入到迭代信息传递图编码器中，得到子图和节点的向量化表示；然后利用SWH注意力机制关注知识图谱子图中的每一个节点，得到节点注意力分布 $p_v(v)$ ，利用循环神经网络 (Recurrent Neural Network, RNN) 的输出状态和上下文向量得到词表词预测分布 $p_{vocab}(w)$ 和生成概率 $p_{gen}$ ；最后利用生成概率 $p_{gen}$ ，结合节点注意力分布 $p_v(v)$ 和词表词预测分布 $p_{vocab}(w)$ 来生成问题，本文接下来将对该模型进行具体介绍。

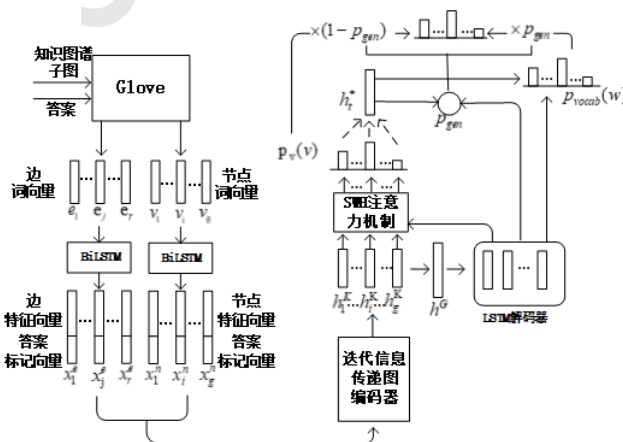


Figure 1: 基于迭代信息传递和滑动窗口注意力的问题生成模型图

### 3.2 编码器

RNN能对序列数据上进行有效建模，但它不能有效处理图结构数据，如知识图谱。在对知识图谱进行编码的时候，一种解决方案是将子图线性化为一个序列，进而使用基于RNN的编码器对序列进行编码，但这种编码方式丢失了获取子图显式结构的完整信息。

本文的编码器由节点和边编码模块以及Imp图编码器模块构成。其中，节点和边编码模块对输入知识图谱中节点和边使用预训练模型实现向量初始化，Imp图编码器模块在知识图谱子图上利用信息传递的基本思想来学习具有显式图结构信息的子图和节点表示，具体细节如下所述。

#### 3.2.1 节点和边编码

我们用 $V$ 表示知识图谱子图中节点的集合 $\{v_1, v_2, \dots, v_g\}$ ，其中每个节点都有自己的属性，如节点名或节点ID；同样，使用 $E$ 表示知识图谱子图中边的集合 $\{e_1, e_2, \dots, e_r\}$ ，其中每个边都有自己的属性，例如边名或边ID。

首先,本文通过Glove词向量 (Pennington et al., 2014)来初始化知识图谱子图中每个节点和边 (及其属性) 向量, 构建额外的词典 (Chen et al., 2020); 接着, 将每个节点和边向量分别输入到BiLSTMs (Hochreiter and Schmidhuber, 1997)中对节点和边 (及其属性) 进行编码, 并用答案标记向量 $x_a$ 表示子图中节点或边是否为生成问题的答案; 最终, 由BiLSTMs中前向和后向最后一个隐藏状态以及答案标记向量拼接得到节点和边的初始向量表示 $X^n = \{x_1^n, x_2^n, \dots, x_g^n\}$ 和 $X^e = \{x_1^e, x_2^e, \dots, x_r^e\}$ , 其中 $g$ 代表节点的个数,  $r$ 代表边的个数, 节点、边的向量维度均为 $d$ , 因此,  $X^n$ 和 $X^e$ 的维度分别为 $d * g$ 和 $d * r$ , 得到的节点和边的初始向量表示作为后续Imp图编码器的输入。

#### 3.2.2 迭代信息传递图编码器

为了将知识图谱子图显式的图结构信息融入到子图表示中以及对节点或边的双向信息进行建模, 本文引入Imp图编码器, 如图2所示, 获取具有显式图结构信息的子图表示和节点表示。

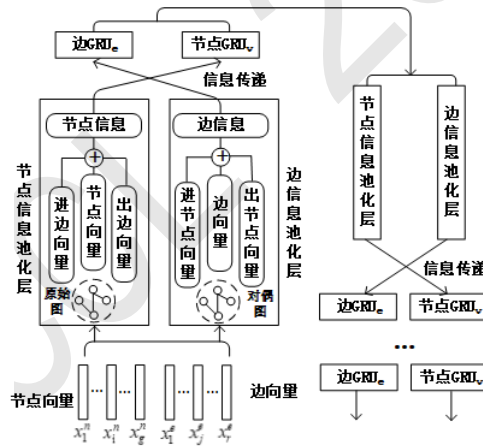


Figure 2: 迭代信息传递图编码器

Imp图编码器主要包括信息池化和信息传递两个模块。其中，信息池化模块构建以节点为中心的原始图作为节点信息池化层的输入，以边为中心的对偶图作为边信息池化层的输入，节点信息池化层和边信息池化层分别从进、出边两个方向（融合）得到节点信息（向量），从进、出节点两个方向（融合）得到边信息（向量）；信息传递模块接收节点信息（向量）和边信息（向量）作为输入，并分别输入到节点GRU<sub>v</sub>层和边GRU<sub>e</sub>层，完成一次知识图谱子图中节点和边的信息传递，具体细节如下：

节点 $v$ 、边 $e$ 的向量表示 $h_v^0$ 、 $h_e^0$ 分别用 $x_i^n$ 、 $x_j^e$ 初始化，输入到信息池化模块中的节点信息池化层和边信息池化层。具体而言，分别对节点 $v$ 的进边、出边向量进行聚合得到进边聚合向量 $h_{N-(v)}^k$ 、出边聚合向量 $h_{N-(v)}^k$ ，如公式 (2) 和 (3) 所示；同样，分别对边 $e$ 的进节点、出节点向量进行聚合得到进节点聚合向量 $h_{N-(e)}^k$ 、出节点聚合向量 $h_{N-(e)}^k$ ，如公式 (4) 和 (5)

所示。

$$h_{N|(v)}^k = \sum_{u:u \rightarrow v} \sigma \left( c_2^T \left[ h_v^{k-1}, h_{u \rightarrow v} \right] \right) h_{u \rightarrow v} \quad (2)$$

$$h_{N|(v)}^k = \sum_{u:v \rightarrow u} \sigma \left( c_1^T \left[ h_v^{k-1}, h_{v \rightarrow u} \right] \right) h_{v \rightarrow u} \quad (3)$$

$$h_{N|(e)}^k = \sigma \left( w_1^T \left[ h_v^{k-1}, h_{v \rightarrow u} \right] \right) h_v^{k-1} \quad (4)$$

$$h_{N|(e)}^k = \sigma \left( w_2^T \left[ h_u^{k-1}, h_{v \rightarrow u} \right] \right) h_u^{k-1} \quad (5)$$

接着，每次迭代周期内，本文对进、出两个方向上的节点聚合向量和边聚合向量进行融合，得到信息池化模块的节点聚合信息 $h_{N(v)}^k$ 、边聚合信息 $h_{N(e)}^k$ ，如公式（6）和（7）所示。

$$h_{N(v)}^k = \text{Relu} \left( h_{N|(v)}^k; h_{N|(v)}^k \right) \quad (6)$$

$$h_{N(e)}^k = \text{Relu} \left( h_{N|(e)}^k; h_{N|(e)}^k \right) \quad (7)$$

最后，信息传递模块分别将节点聚合信息和边聚合信息输入到节点 $GRU_v$ 和边 $GRU_e$ 中，得到节点和边一次编码更新 $h_v^k$ 和 $h_e^k$ ，如公式（8）和（9）所示。

$$h_v^k = GRU_v \left( h_v^{k-1}, h_{N(v)}^k \right) \quad (8)$$

$$h_e^k = GRU_e \left( h_e^{k-1}, h_{N(e)}^k \right) \quad (9)$$

经过K次信息池化模块和信息传递模块的迭代计算，得到节点v的最终向量表示 $h_v^K$ 。

为了计算子图的向量表示，首先将所有的节点向量表示经过一个全连接层，再对它们进行最大池化，最终得到一个d维的向量 $h^G$ 表示子图的向量表示，子图中节点向量表示和子图向量表示用作后续的解码器工作。

### 3.3 解码器

在解码阶段，本文引入基于SWH注意力机制的RNN解码器，提升对知识图谱子图中来自窗口大小相邻节点的局部信息对节点的重要程度，由SWH注意力机制得到的节点注意力分布和已解码的词表词预测分布共同得到问题词预测分布。

#### 3.3.1 滑动窗口分层注意力

为了生成更加流畅相关的问题，解码器的每一步会生成一个单词，当前步单词的选择依赖于节点注意力分布和已解码的词表词预测分布。为了准确得到节点注意力分布，本文提出SWH注意力机制，如图3所示。其中，滑动窗口的大小与获取节点局部信息的多少有关，滑动窗口越大，能够获取的节点的局部信息越多。

具体来讲，对于输入子图中节点v来讲，首先将当前解码器的输出状态 $s_t$ 作为查询单元，对节点v及其节点v左右两边各v个窗口大小的节点进行局部注意力，并得到局部上下文向量 $L_v$ ，如公式（10）和（11）所示。

$$\alpha_{t,v+w'} = \text{softmax} \left( \tanh \left( s_t^T Z h_v^K \right) \right) \quad (10)$$

$$L_v = \sum_{w'=-w}^w h_{v+w'} \alpha_{t,v+w'} \quad (11)$$

其中，Z为参数矩阵。

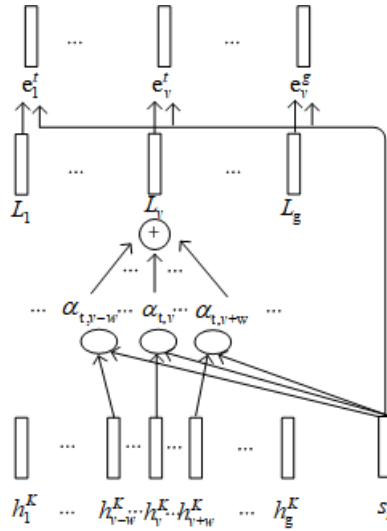


Figure 3: 滑动窗口分层注意力机制图

然后将当前解码器的输出状态 $s_t$ 作为查询单元，对上一步得到的局部上下文向量集合 $L = \{L_1, \dots, L_v, \dots, L_g\}$ 进行全局注意力，得到节点注意力分布 $p_v(v)$ 以及最终的上下文向量 $h_t^*$ ，如公式(12)~(14)所示。节点获取到的局部注意力和全局注意力共同构成了SWH注意力机制中的分层注意力，因此注意力机制融入到了分层中。

$$e_{t,v} = \tanh(w_s s_t + w_h h_v) \quad (12)$$

$$p_v(v) = \text{softmax}(e_{t,v}) \quad (13)$$

$$h_t^* = \sum_{v=1}^g p_v(v) L_v \quad (14)$$

其中，节点注意力分布表示模型在解码过程中对知识图谱子图中不同节点状态的比重。

### 3.3.2 基于SWH注意力机制的RNN解码器

在解码阶段，本文将子图嵌入 $h^G$ 分别经过两个全连接层得到解码器LSTM的初始隐藏状态( $c_0$ 和 $s_0$ )，对于第 $t$ 个时间步，将 $t-1$ 步的隐藏状态 $s_{t-1}$ 和前一个单词 $q_{t-1}$ 作为解码器的输入，获得该时间步解码器隐藏状态 $s_t$ ，如式(15)所示。

$$s_t = LSTM(s_{t-1}, q_{t-1}) \quad (15)$$

由SWH注意力机制得到上下文向量 $h_t^*$ 表示知识图谱子图中节点与解码器隐藏状态之间的关注度。用隐藏状态 $s_t$ 和滑动窗口分层注意力机制得到的上下文向量 $h_t^*$ 计算得到词表词预测分布 $p_{vocab}$ ，如式(16)所示。

$$p_{vocab} = \text{softmax}(k'(k[s_t, h_t^*] + b) + b') \quad (16)$$

通过采用Chen et al. (2020)提出的节点级的复制机制从输入子图中直接复制子图中的节点属性，解决了词级别复制机制的问题。由于由SWH注意力机制得到的节点注意力分布 $p_v(v)$ 已经度量了知识图谱子图中节点与部分解码隐藏状态的相关性，结合词表词预测分布 $p_{vocab}(w)$ 及对应的生成概率 $p_{gen}$ ，得到问题词预测分布 $p(w)$ ，如式(17)所示。

$$p(w) = p_{gen} p_{vocab}(w) + (1 - p_{gen}) p_v(v) \quad (17)$$

其中,  $p_{gen}$ 由上下文向量 $h_t^*$ 、隐藏状态 $s_t$ 和解码器 $t-1$ 步的解码单词 $y_{t-1}$ 计算得到, 如式(18)所示。

$$p_{gen} = \sigma(w_h^T h_t^* + w_s^T s_t + w_x^T y_{t-1} + b_{ptr}) \quad (18)$$

此外, 本文使用交叉熵损失函数作为目标函数, 并用Adam优化器进行优化, 如公式(19)所示。

$$L_{lm} = \sum_t -\log P(y_t^* | X, y_{<t}^*) \quad (19)$$

其中 $y_t^*$ 表示正确输出序列中的第 $t$ 个词。

## 4 实验结果与分析

本文使用ImpSWH-QG模型分别在WebQuestions(WQ)和PathQuestions(PQ)两个数据集上进行实验。本章节我们首先介绍了要比较的基线模型; 然后给出了任务的数据集与评测指标; 接着分析了ImpSWH-QG模型与基线模型的实验结果; 最后我们用两组实验证明了本文模型的有效性。

### 4.1 基线模型

本文将以下模型作为问题生成任务的基线模型:

①L2A模型 (Du et al., 2017), 该模型是一个具有注意力机制基于LSTM的Seq2Seq模型, 最初用在文本生成问题的任务中。

②Transformer(w/copy)模型 (Vaswani et al., 2017), 该模型使用Transformer编码器-解码器架构, 将线性化的知识图谱子图作为输入, 并使用Klein et al. (2017)提出的具有复制机制的开源Transformer实验设置。

③MHQG+AE模型 (Kumar et al., 2019), 是首次使用神经网络从知识图谱子图(即多个三元组)生成问题的模型, 该模型使用基于Transformer编码器对知识图谱子图进行编码, 并使用基于Transformer解码器生成问题。

④KTG模型 (Bi et al., 2020), 该模型使用额外知识库获取更多信息生成语义丰富且多样的问题。

### 4.2 数据与指标

本文使用WebQuestions(WQ)和PathQuestions(PQ)两组数据集来对比评估模型, 这两组数据集都是以Freebase (Google, 2018)作为基础知识图谱, 其中WQ数据集包含了来自WebQuestionsSP (Yih et al., 2016)和ComplexWebQuestions (Talmor and Berant, 2018)的问答数据集, 每一条问答数据集都包含自然语言问题、问题对应的SPARQL语句和答案实体。为了构建知识图谱子图, Kumar et al. (2019)将WQ中的每一条数据中的SPARQL查询转换成对应的子图, 我们的实验数据集使用Kumar et al. (2019)转换后的数据。WQ数据集包含18,989/2,000/2000(训练/验证/测试)条数据。与WQ数据集不同的是, PQ数据集 (Zhou et al., 2018)中的知识图谱子图是跨越2到3跳两个实体之间的路径, 包含9,793/1,000/1,000(训练/验证/测试)条数据。数据集的简要统计信息见表2。

数据集	实体数	边数	三元组数	样例数	问题长度
WQ	25703	672	2/99/5.8	22989	5/36/15
PQ	7250	378	2/3/2.7	9731	8/25/14

Table 2: 数据统计

表2描述了WQ和PQ两个数据集中知识图谱中最小/最大/平均三元组数, 问题长度的最小/最大/平均值。

本文使用BLEU-4 (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005)和ROUGE-L (Lin, 2004)3个指标评价实验结果。其中BLEU-4统计了问题生成模型生成的问题句中4元词组

与标准答案匹配的个数，验证问题生成模型生成的问题句的流畅性；METEOR综合考虑生成问题句的准确率和召回率，准确把握同义词对语义的影响；ROUGE-L用来预测标准答案中的单词在问题生成模型生成的问题句中出现的数量。

### 4.3 实验结果

本节展示了我们的模型与其它最新基线模型在WQ和PQ数据集上的评估结果，如表3所示。

方法	WQ			PQ		
	BLEU-4	METEOR	ROUGE-L	BLEU-4	METEOR	ROUGE-L
L2A	7.21	26.58	27.32	19.35	21.62	51.60
Transformer	9.48	14.92	33.82	57.52	44.25	74.45
MHQG+AE	13.70	29.92	37.62	27.21	34.62	60.21
KTG	25.62	29.01	50.62	46.81	52.15	73.55
ImpSWH-QG	<b>27.78</b>	<b>30.25</b>	<b>53.91</b>	<b>62.25</b>	<b>53.17</b>	<b>76.85</b>

Table 3: 问题生成结果

从实验结果可以发现，在WQ和PQ两个数据集上，ImpSWH-QG模型与其它4个基线模型相比，BLEU-4、METEOR和ROUGE-L值都明显提高，说明基于图神经网络的编码器比基于RNN和Transformer的编码器更适合处理知识图谱子图，表明获取知识图谱子图显式图结构信息的重要性。与WQ数据集上的结果相比，基于Transformer的编码器在PQ数据集上的结果要高于其它3个基线模型，说明基于Transformer的编码器在处理序列模型上比其它3个基线模型更有效，因为PQ数据集中的子图结构偏向路径结构，而WQ数据集中的子图结构偏向树结构。但在PQ数据集上，ImpSWH-QG模型的BLEU-4和ROUGE-L值明显高于Transformer模型，这得益于解码器中SWH注意力机制的使用，准确得到节点注意力分布。

### 4.4 模型分析

#### 4.4.1 迭代信息传递图编码器跳数的研究

本文在WQ数据集上探究了跳数对Imp图编码器编码的影响，如图4所示。实验结果表明，图编码器跳数在一定的阈值内，能够提高模型的性能。具体而言，当跳数小于阈值4时，随着图编码器跳数的增加，图编码器有助于知识图谱子图的编码效果，但当跳数大于4时，图编码器降低了知识图谱子图的编码效果。

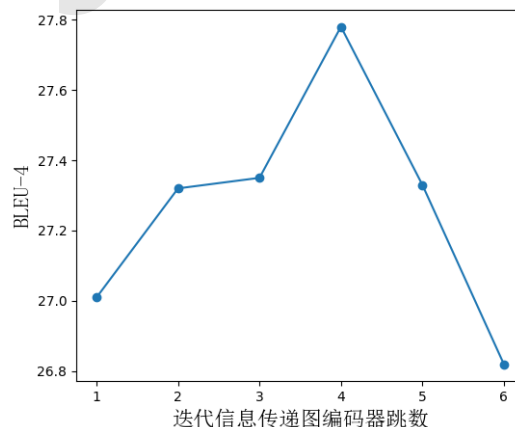


Figure 4: 迭代信息传递图编码器跳数的探究



#### 4.4.2 SWH注意力机制的有效性

本文在WQ数据集上对ImpSWH-QG模型进行消融实验证明模型中SWH注意力机制的有效性，如表4所示。

模型	BLEU-4	METEOR	ROUGE-L
Imp-QG	26.66	29.61	52.01
ImpSWH-QG	27.78	30.25	53.91

Table 4: 问题生成结果

其中，Imp-QG模型是用普通注意力机制代替滑动窗口分层注意力的模型。实验结果表明，与普通的注意力机制相比，模型中使用滑动窗口分层注意力机制后，三项评测指标都有明显的提升，准确得到节点注意力权重分布和词表词预测分布。

本文使用注意力可视化的方式对SWH注意力机制进行分析，如图5所示。

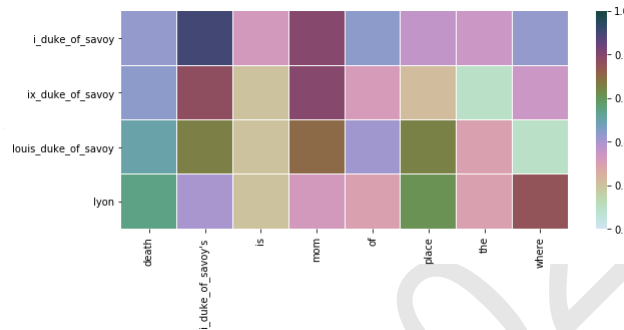


Figure 5: SWH注意力机制的可视化分析

其中，纵坐标表示生成目标问题对应的知识图谱子图中的节点以及词典中相关的词，横坐标表示生成的目标问题中的各个单词；图中颜色的深度表示权重的大小。由图5可知，目标问题中的各个单词通过SWH注意力机制可以有效注意到节点词或词典词，提高了生成的目标问题的质量。

#### 4.4.3 实例分析

本文通过实例分析验证是否使用SWH注意力机制生成问题的质量，如表5所示。表5中分别展示了由未加SWH注意力机制的Imp-QG模型和含有SWH注意力机制的ImpSWH-QG模型生成的问题。

知识图谱	(i_duke_of_savoy,parents,ix_duke_of_savoy), (ix_duke_of_savoy,parents,louis_duke_of_savoy), (louis_duke_of_savoy,place_of_death,lyon)
答案	lyon
标准答案句	where is the grandplace of death of i duck of savoy's mom ?
Imp-QG	who is the place of death of savoy's mom ?
ImpSWH-QG	where is the place of death of i duke of savoy's mom ?

Table 5: 问题生成结果

通过两个模型生成的问题来看，SWH注意力机制的添加，使问题生成模型在一定程度上改善了知识图谱子图和答案与问题不匹配的现象，ImpSWH-QG模型在问题生成时，滑动窗口技术能有效将知识图谱子图中节点名“i-duck-of-savoy”看做成一个整体，并通过对知识图谱子图中相邻节点局部信息的关注，该模型成功预测了生成问题中的疑问词“Where”。

## 5 结论与展望

本文提出的ImpSWH-QG模型，用Imp图编码器编码知识图谱子图，用SWH注意力机制得到的节点注意力分布和词表词预测分布确定问题词预测分布来得到问题。实验结果表明，该模型在两个数据集三项指标上取得的结果都比基线高。

虽然本文提出的模型有助于在知识图谱子图上生成高质量的问题，由于仅仅依靠从输入的知识图谱子图中获取信息，导致生成的问题多样性低。因此，未来我们将添加额外的知识作为输入，对生成问题的质量做进一步的提高。

## 参考文献

- Alon Talmor and Jonathan Berant. 2018. *The web as a knowledge-base for answering complex questions*. arXiv preprint arXiv:1803.06643.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. *Attention is all you need*. In *Advances in neural information processing systems*, pages 5998–6008.
- Cao. Liu, Kang. Liu, Shizhu. He, Zaiqing. Nie, and Jun. Zhao. 2019. *Generating questions for knowledge bases via incorporating diversified contexts and answer-aware loss*. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2431–2441.
- Chin-Yew Lin. 2004. *Rouge: A package for automatic evaluation of summaries*. *Text Summarization Branches Out*.
- Danfei Xu, Yuke Zhu, Christopher B. Choy, Li Fei-Fei. 2017. *Scene Graph Generation by Iterative Message Passing*. arXiv preprint arXiv:1701.02426v2.
- Dominic. Seyler, Mohamed. Yahya, and Klaus. Berberich. 2015. *Generating quiz questions from knowledge graphs*. In *Proceedings of the 24th International Conference on World Wide Web*, pages 113–114. ACM.
- Dominic. Seyler, Mohamed. Yahya, and Klaus. Berberich. 2017. *Knowledge questions from knowledge graphs*. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*, pages 11–18. ACM.
- Google. 2018. <https://developers.google.com/freebase>.
- Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M Rush. 2017. *Opennmt: Opensource toolkit for neural machine translation..* arXiv preprint arXiv:1701.02810.
- Hady. Elshakar, Christophe. Gravier, and Frederique. Laforest. 2018. *Zero-shot question generation from knowledge graphs for unseen predicates and entity types*. arXiv preprint arXiv:1603.06807.
- Iulian. Vlad. Serban, Alberto. Garcia-Durán, Caglar. Gulcehre, Sungjin. Ahn, Sarath. Chandar, Aaron. Courville, and Yoshua. Bengio. 2016. *Generating factoid questions with recurrent neural networks: The 30m factoid question-answer corpus*. arXiv preprint arXiv:1603.06807.
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. *Glove: Global vectors for word representation*. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.
- Junwei. Bao, Duyu. Tang, Nan. Duan, Zhao. Yan, Yuanhua. Lv, Ming. Zhou, and Tiejun. Zhao. 2018. *Table-to-text: Describing table region with natural language*. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Kishore Papineni, Salim Roukos, Todd Ward, and WeiJing Zhu. 2002. *Bleu: a method for automatic evaluation of machine translation*. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.
- Linfeng. Song and Lin. Zhao. 2016. *Question generation from a knowledge base with web exploration*. arXiv preprint arXiv:1610.03807.

- Linfeng. Song and Lin. Zhao. 2016. *Domain-specific question generation from a knowledge base*. CoRR.
- Linfeng. Song, Zhiguo. Wang, Wael. Hamza, Yue. Zhang, and Daniel. Gildea. 2018a. *Leveraging context information for natural question generation*. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pages 569–574.
- Mantong Zhou, Minlie Huang, and Xiaoyan Zhu. 2018. *An interpretable reasoning network for multi-relation question answering*. arXiv preprint arXiv:1801.04726.
- Satanjeev Banerjee and Alon Lavie. 2005. *Meteor: An automatic metric for mt evaluation with improved correlation with human judgments*. In Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization, pages 65–72.
- Sathish Reddy Indurthi, Dinesh Raghu, Mitesh M Khapra, and Sachindra Joshi. 2017. *Generating natural language question-answer pairs from a knowledge graph using a rnn based question generation mode*. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, pages 376–385.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. *Long short-term memory*. Neural computation, 9(8):1735–1780.
- Sheng Bi, Xiya Cheng, Yuan-Fang Li, Yongzhen Wang and Guilin Qi. 2020. *Knowledge-enriched, Type-constrained and Grammar-guided Question Generation over Knowledge Bases*. arXiv preprint arXiv:2010.03157v3
- Vishwajeet. Kumar, Yuncheng. Hua, Ganesh. Ramakrishnan, Guilin. Qi, Lianli. Gao, and Yuan-Fang. Li. 2019. *Difficulty-controllable multi-hop question generation from knowledge graphs*. In International Semantic Web Conference, pages 382–398. Springer.
- Wen-tau Yih, Matthew Richardson, Chris Meek, MingWei Chang, and Jina Suh. 2016. *The value of semantic parse labeling for knowledge base question answering*. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 201–206.
- Xinya. Du, Junru. Shao, and Claire. Cardie. 2017. *Learning to ask: Neural question generation for reading comprehension*. arXiv preprint arXiv:1705.00106
- Yikang. Li, Nan. Duan, Bolei. Zhou, Xiao. Chu, WanliOuyang, Xiaogang. Wang, and Ming. Zhou. 2018. *Visual question generation as dual task of visual question answering*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6116–6124.
- Yu. Chen, Lingfei. Wu, and Mohammed. JZaki. 2020. *Toward subgraph guided knowledge graph question generation with graph neural networks*. arXiv preprint arXiv:2004.06015.