基于门控化上下文感知网络的词语释义生成方法*

张海同[1,2] 孔存良[2,3] 何姗[4] 杨麟儿[2,3] 杜永萍[1] 杨尔弘[2,3]

（1.北京工业大学 信息学部，北京 100124；2.北京语言大学 语言资源高精尖创新中心，北京 100083；3.北京语言大学 信息科学学院，北京 100083；4.云南师范大学华文学院 国际汉语教育学院，昆明 650500）

摘要：传统的词典编纂工作主要采用人工编纂的方式，效率较低且耗费大量的资源。为减少人工编纂的时间和经济成本，本文提出一种基于门控化上下文感知网络的词语释义生成方法，利用门控循环神经网络（GRU）对词语释义生成过程进行建模，自动为目标词生成词语释义。该模型基于编码器-解码器架构。编码器首先利用双向 GRU 对目标词的上下文进行编码，并采用不同的匹配策略进行目标词与上下文的交互，结合注意力机制分别从粗粒度和细粒度两个层次将上下文信息融合到目标词的向量表示中，最终获得目标词在特定语境中的编码向量。解码器则同时基于目标词的语境与语义信息为目标词生成上下文相关的词语释义。此外，通过向模型提供目标词字符级特征信息，进一步提高了生成释义的质量。在英文牛津词典数据集上进行的实验表明，本文提出的方法能够生成易于阅读和理解的词语释义，在释义建模的困惑度和生成释义的 BLEU 值上分别超出此前模型 4.45 和 2.19，具有显著提升。

关键词：释义生成；GRU；编码器-解码器；注意力机制

中图分类号：TP391　　**文献标识码：**A

Gated Context-Aware Network for Definition Generation

ZHANG Haitong[1,2], KONG Cunliang[2,3], HE Shan[4], YANG Liner[2,3],

DU Yongping[1], YANG Erhong[2,3]

(1. Beijing University of Technology, Faculty of Information Technology, Beijing, 100124, China; 2. Beijing Advanced Innovation Center for Language Resources, Beijing Language and Culture University, Beijing,100083, China; 3. School of Information Science, Beijing Language and Culture University, Beijing,100083, China; 4. Yunnan Normal University, School of international Studies, Kunming, 650500, China)

Abstract: The traditional lexicography mainly adopts manual compilation, which is inefficient and consumes a lot of resources. In order to reduce the cost of time and economy from manual compilation, this paper proposes a gated context-aware network for definition generation. It utilizes GRU to model the definitions of words and generates the textual definition for the target word automatically. The model is based on the encoder-decoder architecture. Firstly, the context of the target word is encoded by bidirectional GRU. Then, different matching strategies are used to interact the target word with context and the context information is incorporated into the target word embedding from two aspects of coarse-grained and fine-grained by the attention mechanism to obtain the meaning of the target word in a specific context. The decoding process based on the contextual and semantic information to generate context-dependent definition of the target word. In addition, the

* **收稿日期：**　　**定稿日期：**

基金项目：语言资源高精尖创新中心项目(TYZ19005)；国家重点研发计划项目（No.2018YFC1900804）；国家语委信息化项目（No.YB135-89）

作者简介：张海同（1995-），男，硕士研究生，主要研究方向为自然语言处理；孔存良（1995-），男，硕士研究生，主要研究方向为自然语言处理；何姗（1988-），女，博士，主要研究方向为词典学、汉语国际教育；杨麟儿（1983-），男，通讯作者，博士，主要研究方向为句法分析；杜永萍（1977-），女，教授，主要研究方向为自然语言处理；杨尔弘（1965-），女，教授，主要研究方向为自然语言处理。

quality of generated definitions is further improved by providing the character level information of target words. The experimental results show that the proposed model improves the perplexity of definition modeling and the BLEU score of definition generation on the English Oxford dictionary dataset by 4.45 and 2.19 respectively, and can generate readable and understandable definitions.

Key words: Definition Generation; GRU; Encoder-Decoder; Attention

1 引言

随着全球化进程的不断发展，英语作为世界上最通用的语言正在被广泛普及，越来越多的非英语母语者选择英语作为第二语言(English as a Second Language, ESL)进行学习[1]。在语言学习的过程中，难免会面临单词不认识，意义不理解的情况，因此查词典的工作在学习外语的过程中十分重要。英英词典指的是用英语解释英语的词典，相比于英汉词典、英汉双解词典，它能够帮助我们更深刻地理解英语词汇本身的含义，也有助于我们学习英文中更地道的表达方式，掌握多个同义词在不同语境中的选择和使用。目前市面上的五大主流英英词典包括：牛津、朗文、柯林斯、剑桥和麦克米伦系列词典[2]。

随着时代的发展，有些词语的释义发生了改变并且不断的涌现出一些新生词汇。比如，由于社交软件“推特”的火热，“follow”已经在《牛津英语词典》中扩充了“关注”的含义。“vape”最初只是作为“vapour”或者“vaporize”的缩写，有蒸汽或汽化的意思。近年来电子烟逐渐普及，“vape”越来越多的被用做“电子烟”的意义。随着“vape”使用频率的逐渐增加，牛津在线词典 2014 年将其正式收录，并定义为“电子烟，吸电子烟”。同时，类似“tweet”(推文，发推文)，“taikonaut”(中国宇航员)，“geekery”(极客范儿)等反映当下语言应用趋势的网络热词也逐渐被收录到英文词典中，引起了广大词典编辑人员的关注。

传统的词典编纂工作主要采用人工编纂的方式，依赖语言学等专项的领域专家，往往会耗费大量的财力物力。因此本文研究利用深度学习的方法设计神经网络模型对英文词典释义的生成进行建模，自动生成词语释义，从而缩减人工编纂词典的时间和经济成本，尝试生成更易于阅读和理解的释义。图 1 是释义生成的一个示例，输入一个目标词与其特定的上下文，模型根据目标词的上下文获得在特定语境下目标词的相应含义，并完成“目标词→释义”的映射。释义生成是自然语言处理中的一项文本生成任务，其模型不仅需要理解释义的语义和

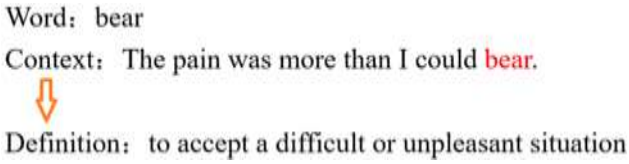

图 1：释义生成示例

结构信息，还需要生成人类可读的自然语言文本。

本文在现有研究的基础上提出了一种门控化上下文感知的词语释义生成模型。我们采用一种多级别的目标词与上下文的交互方式，分别通过 1)信息匹配与门控感知机制粗粒度地将句子级别的，2）注意力（Attention）机制细粒度地将词级别的上下文语义感知信息融合到目标词的向量表示中。在解码过程中，模型同时考虑目标词的语义与语境信息来生成词语释义。在牛津英文词典数据集上的实验表明了本文提出的模型较之前的模型具有显著提升，验证了本文方法的有效性。

本文的论文结构如下：第二节介绍了词典释义生成的相关工作；第三节介绍我们提出的门控化上下文感知网络及其各个组成模块的功能与原理；第四节介绍了我们的实验设置以及对实验结果的详细分析；第五节是本文的结论。

2 相关工作

近年来，随着深度学习的发展，分布式的词表示（词向量）[3]已经成为神经网络模型的基础组件，在许多自然语言处理任务上取得了良好的表现，词向量也被认为可以捕捉词语在语言系统中的语义信息。释义建模是由 Noraset 等人[4]提出的用于评估词向量所捕捉的语义信息的一种方式。作者将目标词作为种子信息[5]置于释义序列的开头，通过基于长短时记忆网络(Long Short-Term Memory, LSTM)[6]语言模型的释义生成模型对词典释义进行建模，利用预训练的目标词词向量生成该词语的自然语言释义。此外，作者利用一个字符级别的卷积神经网络(Convolutional Neural Networks, CNN)为被定义词提供字符级信息，并提供目标词的上位词关系来近一步提高生成释义的质量。

然而，Noraset 等人[4]并没有考虑多义词的现象，因此带来了词歧义的问题。Gadetsky 等人[7]在生成目标词释义时考虑了目标词的上下文信息，提出了 Adaptive Skip Gram Model 和 Attention based Model 来生成上下文相关的词语释义。Adaptive Skip Gram Model 利用 Adaptive Skip-gram 向量表示[8]在不同的上下文中为目标词提供不同的词向量；而 Attention based Model 则使用注意力机制[9]提取与特定含义相关的目标词词向量组件。此外，国内 Yang 等人[10]研究将义原引入中文释义建模任务中来进行中文词语释义的生成。

3 门控化上下文感知网络

3.1 模型架构

本文提出了一种门控化上下文感知的词语释义生成模型(Gated Context-Aware Network, GCA)。如图 2 所示，模型基于编码器-解码器[11]架构，主要由上下文编码模块、上下文交互模块、门控感知模块、门控注意力模块和解码器模块组成。其中，上下文编码模块和解码器模块均由循环神经网络（RNN）构成。考虑到门控循环单元（GRU）[12]可以有效的解决传统 RNN 由于序列过长而引起的梯度弥散问题[13]，并且相比于长短期记忆网络（LSTM）结

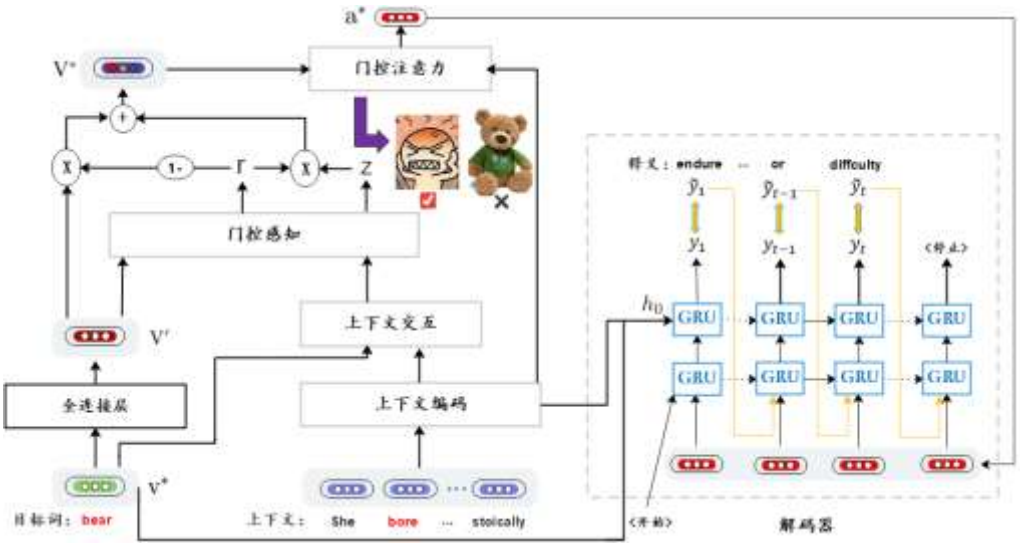

图 2：门控化上下文感知网络

构更简单、参数更少，本文主要采用 GRU 进行模型构建。

在模型中，上下文编码模块首先将目标词的上下文转化成固定维度大小的向量表示；然后，分别通过上下文交互模块、门控感知模块、门控注意力模块将特定的上下文信息融合到目标词词向量的表示中，以获得特定含义的目标词信息；最后，解码器模块根据编码器获得的信息为目标词生成上下文相关的词语释义。本文将对这些模块分别进行详细介绍。

3.2 上下文编码模块

给定含有 M 个词$\{\square_{\square}\}_{\square=1}^{\square}$的上下文，上下文编码模块首先通过嵌入层[14]利用预训练的词向量将目标词的上下文中的每一个词转化成向量表示$\{\square_{\square}\}_{\square=1}^{\square}$。随后，利用 BiGRU 分别通过前向和反向计算得到两组不同的隐藏表示，并通过向量拼接得到每个词最终的表示，如公式（1）所示：

$$\begin{aligned}\overrightarrow{f_t} &= \overrightarrow{\square\square\square_{\square}}(\square_1,\dots,\square_{\square})\\ \overleftarrow{f_t} &= \overleftarrow{\square\square\square_{\square}}(\square_1,\dots,\square_{\square})\\ f_t &= [\overrightarrow{f_t},\overleftarrow{f_t}]\end{aligned} \tag{1}$$

其中$f_t \in \mathbb{R}^{2d}$代表上下文中第 t 个词语的表示。我们采用一个最大池化（max-pooling）层[15]将得到的一组单词表示$\{f_t\}_{t=1}^{M}$结合成上下文的句子嵌入$\mathrm{V_c}$。实验结果证明了最大池化的合并方法可以在使用相对较少参数的同时性能要优于其他句子编码方法。

3.3 上下文交互模块

被定义的目标词w^*的词向量v^*同样通过预训练的词向量进行初始化。将目标词的词向量v^*和上下文编码模块得到的上下文表示$\mathrm{V_c}$输入到上下文交互模块中，以便将特定上下文的句子级别信息融合到被定义词的向量表示中。我们采用多种方式计算目标词与其上下文之间的匹配程度，包括三种不同的交互策略：1)目标词词向量与上下文表示向量的拼接$(\mathrm{v}^*;\mathrm{V_c}) \in \mathbb{R}^{4d}$；2)目标词词向量与上下文表示向量的点积$\mathrm{v}^* \odot \mathrm{V_c}$；3)目标词词向量与上下文表示向量的绝对元素差异$|\mathrm{v}^* - \mathrm{V_c}|$。将三种交互策略的结果进行拼接，如公式（2）：

$$m^* = [\mathrm{v}^*;\mathrm{V_c};\mathrm{v}^* \odot \mathrm{V_c};|\,\mathrm{v}^* - \mathrm{V_c}|]\ , \tag{2}$$

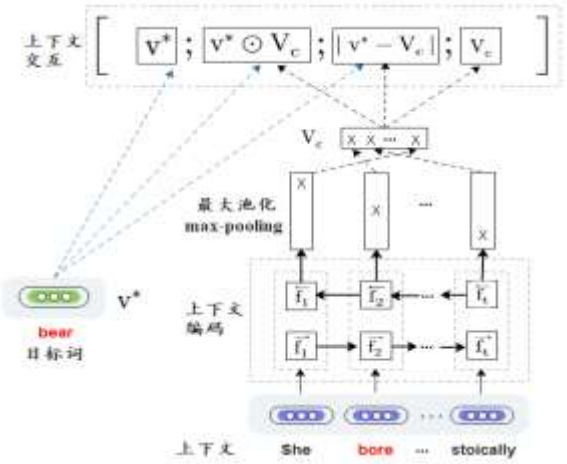

图 3：上下文编码模块与上下文交互模块

得到目标词与上下文的交互信息$m^* \in \mathbb{R}^{8d}$。该向量从不同方面捕获了目标词与其上下文之

间句子级别的匹配关系。上下文编码模块和上下文交互模块的详细示意图如图 3 所示。

3.4 门控感知模块

模型利用门控感知模块来度量将句子级别的交互信息m^*融合到目标词词向量表示中的程度。首先，使用全连接层对目标词词向量进行线性变换，得到向量表示V'：

$$\mathrm{V}' = W_v \mathrm{v}^* + b_v \ . \tag{3}$$

其次，门控感知模块接收目标词的向量表示V'及其对应的上下文交互信息m^*，生成融合了上下文信息后新的目标词表示$z \in \mathbb{R}^{2d}$以及它的门控向量$r \in \mathbb{R}^{2d}$，分别如公式（4）（5）：

$$z = \tanh(W_z^{(1)} m^* + W_z^{(2)} \mathrm{V}' + b_z) \ , \tag{4}$$

$$r = \mathrm{sigmoid}(W_r^{(1)} m^* + W_r^{(2)} \mathrm{V}' + b_r) \ , \tag{5}$$

其中$W_z^{(1)}, W_r^{(1)} \in \mathbb{R}^{2d \times 8d}$，$W_v, W_z^{(2)}$和$W_r^{(2)} \in \mathbb{R}^{2d \times 2d}$与$b_v, b_z$和$b_r \in \mathbb{R}^{2d}$是训练参数，$r$中的每个元素决定要添加多少上下文的信息到目标词的词向量表示中。最终门控化上下文感知的目标词词表示$\mathrm{V}^* \in \mathbb{R}^{2d}$为$z$和$r$的线性组合，如公式（6）所示：

$$\mathrm{V}^* = (1 - r) \odot \mathrm{V}' + r \odot z \ . \tag{6}$$

3.5 门控注意力模块

在目标词的上下文中，每一个词对目标词含义的影响程度也是不同的。例如“park”既有“公园”也有“停车”的意思。对于“停车”这一义项来说，它的上下文为“He parked his car outside her house”。显然，相比于上下文中的其他词，“car”更能帮助我们理解“park”在这句话中的含义。因此，我们利用门控注意力模块，通过 Attention 机制使得模型更加关注上下文中的局部重要信息。

门控注意力模块利用 Attention 机制得到特定 token 的上下文表示$\widetilde{c_t}$，并将上下文表示与门控感知模块计算得到的上下文感知的目标词表示V^*进行逐元素相乘，得到结合注意力的目标词表示a^*：

$$\propto_t = softmax(\mathrm{C}^{\mathrm{T}} \mathrm{V}^*) \ , \tag{7}$$

$$\widetilde{c_t} = \mathrm{C} \propto_t \ , \tag{8}$$

$$\mathrm{a}^* = \mathrm{V}^* \odot \widetilde{c_t} \ , \tag{9}$$

其中，$\mathrm{C} = \{f_t\}_{t=1}^{M}$为上下文编码模块在未进行最大池化前的上下文句子中所有单词的隐藏表示。

3.6 解码器模块

解码器模块在给定目标词及其上下文的情况下为目标词生成文本释义。该模块同样采用 GRU 作为解码单元。为了使得解码器在解码过程中可以获得目标词与其上下文的显式信息以生成通顺、一致的词语释义，我们利用目标词预训练的词向量v^*与其上下文嵌入表示$\mathrm{V_c}$的拼接作为解码器 GRU 的初始隐藏状态：

$$h_0 = [\mathrm{v}^*; \mathrm{V_c}] \ , \tag{10}$$

在每一个时间步 t，GRU 单元接收释义中 t 时刻的答案词表示与结合注意力的目标词表示的拼接，根据前一个时间步的隐藏状态，GRU 单元更新该时刻的隐藏状态：

$$h_t = g(x_t, h_{t-1}) \ , \tag{11}$$

$$x_t = [\mathrm{v_t}; \mathrm{a}^*] \ , \tag{12}$$

其中，v_t是 t 时刻释义中的答案词w_t的词向量，g为 GRU 单元的递归非线性函数。随后，通过线性层对隐藏状态进行线性变换，得到输出：

$$O_t = W_o \cdot h_t \quad , \tag{13}$$

其中，$W_o \in \mathbb{R}^{V_{voc} \times 2d}$（$V_{voc}$为词表大小）。

最后，模型通过 softmax 函数得到在词表大小上的概率分布y_t，如公式（14-15）所示：

$$p_{t,i} = \frac{\exp(O_{t,i})}{\sum_j \exp(O_{t,j})} \quad , \tag{14}$$

$$\square_{\square} = \text{argmax}\square_{\square,\square} \quad , \tag{15}$$

当解码器生成停止标识时，解码过程结束。

3.7 辅助特征

在英文中，许多单词都是由词根和词缀组成的。比如“boundless”由词根“bound”和后缀“less”组成。这样的词缀信息往往能在一定程度上体现词语的含义。我们希望模型可以学习这些特征来提高生成词语释义的质量。因此，我们使用字符级信息来为模型提供额外的辅助特征。与 Noraset 等人[16]不同，我们采用 Bi-GRU 分别从前向和反向对被定义词的字符序列进行建模，将最后的输出$\overrightarrow{k_n}$与$\overleftarrow{k_n}$进行拼接，并通过线性变换得到最终的字符嵌入表示CH_{gru}，如公式（16）所示：

$$k = [\overrightarrow{k_n}; \overleftarrow{k_n}]$$
$$CH_{gru} = W_c k + b_c \quad , \tag{16}$$

我们将字符嵌入CH_{gru}与目标词词向量v^*进行拼接，以将其融入到模型之中。

4 实验与结果

4.1 数据集及评价指标

我们在 Gadetsky 等人[7]收集的英文牛津词典数据集[1]上进行了实验，该数据集使用英文牛津词典语料[2]，每一个条目由目标词、上下文句子以及对应的词语释义三元组组成。该数据集的数据分布如表 1 所示。

本实验采用的评价指标是模型在测试集上的困惑度（PPL）以及生成词语释义的 BLEU 值[17]。困惑度可以衡量模型建模释义的效果，困惑度越低说明模型越能捕捉词语释义的语义和结构特征。BLEU 是机器翻译和文本生成常用的评价指标，它反映了生成结果与参考答案之间的 N 元文法准确率。

表 1 英文牛津词典数据集的统计数据

英文牛津词典数据	训练集	验证集	测试集
词语	33,128	8,867	8,850
条目	97,855	12,232	12,232
Tokens	1,078,828	134,486	133,987
释义平均长度	11.03	10.99	10.95

4.2 实验设置

本文提出的 GCA 模型的解码器使用两层 GRU，hidden size 为 300，嵌入层使用预训练

[1] https://github.com/agadetsky/pytorch-definitions
[2] https://developer.oxforddictionaries.com/

的 300 维 Word2Vec[3]进行初始化并在训练中固定。上下文编码模块采用单独的嵌入层，同样使用 300 维 Word2Vec 进行初始化但在训练过程中进行微调。编码器使用单层 BiGRU，hidden size 为 150。字符级嵌入的维度为 64，采用单层 BiGRU，hidden size 为 50，线性层大小为 150。

我们首先使用 wikitext-103 数据集[18]对模型的解码器部分进行预训练，并且将v^*和$\square_0$设置为 0 向量，使得解码器无条件进行学习。训练时解码器部分加载预训练的参数，采用 Adam[19]优化算法，学习率为 0.001，端到端地最小化模型的负对数似然损失。为了加快训练速度，合理利用训练资源，我们使用小批量（mini batch）梯度下降法，并将 batch size 大小设为 30。当模型在验证集上的损失连续 5 个 epoch 没有下降时停止训练。我们选择在验证集上 PPL 值最小的模型在测试集上进行测试，并使用 $\tau = 0.05$的 sample temperature 采样算法进行释义的生成。为了公平地进行比较，我们参考之前的工作[4][7]，使用 Moses 库[4]中的“sentence-bleu”脚本进行 BLEU 指标的计算，并报告测试集上所有数据的平均值。我们的代码会在随后进行开源[5]。

4.3 实验结果与分析

4.3.1 与其他模型性能的比较

本文对比的基线模型包括 Noraset 等人[4]最优的模型 **S+G+CH** 以及 Gadetsky 等人[7]提出的模型 **S+I-Adaptive** 和 **S+I-Attention**。其中 **S+G+CH** 模型并未考虑目标词的上下文信息。我们复现了上述模型，并将其与我们提出的 GCA 模型进行比较。实验结果如表 2 所示，其中“*”表示考虑上下文的模型。每行结果中，“/”之前为我们复现模型的结果，“/”之后为原始论文中的结果。从实验结果可以看出，我们复现的基线模型结果要优于论文中的结果。考虑上下文的基线模型 S+I-Attention 在 PPL 和 BLEU 两个指标上均优于其他基线模型。由于一词多义现象的存在，通过引入目标词的上下文信息进行词义消歧，使得模型可以为同一个词在不同的语境下生成不同的释义，从而提高生成释义的准确性。从实验结果可以看出，本文提出的模型 GCA 的性能优于所有基线模型，在释义建模的 PPL 值和生成释义的 BLEU 值上分别提升了 4.45 和 2.19，这充分说明了我们提出方法的有效性。在下面的章节中我们将深入分析模型每一个模块对模型性能的影响。

表 2 各模型在牛津英文词典测试集上的性能比较

模型	PPL	BLEU
S+G+CH(2017)[4]	41.74/45.62	12.71/11.62
S+I-Adaptive*(2018)[7]	48.32/46.08	12.52/11.53
S+I-Attention*(2018)[7]	40.89/43.54	12.76/12.08
GCA*	**36.44**	**14.95**

4.3.2 上下文编码模块编码方式对模型性能的影响

[3] https://code.google.com/archive/p/word2vec/
[4] http://www.statmt.org/moses/
[5] https://github.com/blcu-nlp/gcan-definition

采用神经网络将句子编码成固定大小向量表示的方式有很多种，我们比较上下文编码模块中使用的最大池化的编码方式（**Max-Pooling**）[15]和三种其他句子编码方法的性能：（1）**BiGRU-Last**：双向 GRU 最后一个隐藏状态进行拼接；（2）**Average pooling**：平均池化；（3）**Inner attention**[20]：在隐藏状态上应用注意力机制。实验结果如表 3 所示。相比于 Inner attention 这种复杂的层次化 Attention 交互方法，我们的方法在使用相对简单操作的同时取得了更好的性能。

表 3 不同句子编码方式的性能比较

编码方式	PPL	BLEU
BiGRU-Last	37.35	14.84
Average pooling	36.89	14.91
Inner attention	37.69	14.93
Max-Pooling	**36.44**	**14.95**

4.3.3 门控注意力模块与门控感知模块对模型性能的影响

我们通过消融分析来探究门控注意力模块和门控感知模块对模型的性能的影响。分别将门控注意力模块与门控感知模块从模型中去掉，使得解码器每一个时间步 t 上的输入x_t（公式 11）分别变为公式 17 和公式 18：

$$x_t = [\mathrm{v_t}; \mathrm{V}^*] \ , \tag{17}$$

$$x_t = [\mathrm{v_t}; m^*] \ . \tag{18}$$

实验结果如表 4，当去除两个模块之后，模型的性能均有所下降。实验结果表明，门控注意力模块和门控感知模块分别从细粒度和粗粒度两个层次，将目标词上下文的词级别信息和句子级信息融合到目标词的词向量表示中，这有助于获得目标词在特定语境中的含义信息，从而生成质量更高的词语释义。

表 4 消融分析

模型	PPL	BLEU
GCA	**36.44**	**14.95**
-Attention	36.88	14.61
-Gated	37.65	14.17

表 5 不同注意力函数的性能比较

模型	PPL	BLEU
Sum	36.96	14.45
Concatenate	37.45	14.33
Dot	**36.44**	**14.95**

此外，我们还分析了门控注意力模块注意力函数对模型性能的影响： Sum，**相加**($\mathrm{a}^* = \mathrm{V}^* + \tilde{c}_t$)；Concatenate，**拼接**($\mathrm{a}^* = [\mathrm{V}^*; \tilde{c}_t]$)；Dot，**点乘**（公式 9）。结果如表 5 所示。我们发现元素相乘的方式比其他两种方式的效果更好。这说明点乘操作更有助于目标词与上下文中每一个词的交互，从而使得模型关注局部更加重要的信息。

4.3.4 字符级辅助信息对模型性能的影响

我们研究了字符级信息对释义生成的影响，结果如表 6 所示。词语的字符级信息可以表示词语复杂的形态学特征，这可以作为传统预训练词向量一种有效的补充，从而进一步提升模型的性能。并且，我们使用 GRU 方式建模字符级信息的性能要优于 Noraset 等人[4]使用

表 6 字符级特征对模型的影响

模型	PPL	BLEU
GCA -CH	36.59	14.68
GCA +CH(CNN)	**36.39**	14.75
GCA +CH(GRU)	36.44	**14.95**

CNN 的方式，这说明了在字符级别建模中顺序关系的有效性。

4.3.5 样例分析

表 7 展示了不同模型为多义词“play”生成的释义示例，Ground Truth 代表词语在词典中的标准释义。S+G+CH 模型没有考虑目标词的上下文信息，所以对于不同上下文的同一目标词，该模型都会生成相同的释义。由于预训练的词向量是在大型语料上进行训练得到，单一的向量表示中会混合目标词所有的语义信息，模型往往倾向于对目标词生成它最常见的一个释义。S+I-Attention 模型通过利用上下文的软二进制掩码可以结合语境进行目标词释义的生成，但是它利用上下文信息的方式并不鲁棒，在“play”的第二个条目中生成了错误意义的释义。我们提出的 GCA 模型通过更加全面的交互方式将不同级别的上下文信息融合到目

表 7 不同模型对于目标词 play 生成的释义

Word	Context	Model	Definition
play	The band played all night long.	S+G+CH	a game or contest in which a player or team is awarded
		S+I-Attention	a piece of music or other material that can be played by a person
		GCA	**perform a musical performance**
		Ground Truth	perform on a musical instrument
play	Pele played for the Brazilian teams in many important matches.	S+G+CH	a game or contest in which a player or team is awarded
		S+I-Attention	a playing card that is played by a player
		GCA	**engage in sport**
		Ground Truth	participate in games or sport

标词的向量表示中，可以为目标词生成更准确、更通顺，上下文相关的词语释义。

5 结论

本文提出了一种基于门控化上下文感知网络的词语释义生成方法。通过双向 GRU 对目标词的上下文进行编码，采用多种匹配策略进行目标词与上下文的交互。我们通过门控感知机制和注意力机制分别从粗粒度和细粒度两个层次将上下文信息融合到目标词的向量表示中，可以自动的生成上下文相关的词语释义，从而减轻人工词典编纂工作的时间和经济成本。实验结果表明本文提出的方法优于已有的释义生成模型，可以生成更加准确、通顺的词语释义。在今后的工作中，我们将尝试使用 Transformer[21] 替代 GRU 解码器，利用 Seq2Seq+Attention 机制来实现对释义更好的建模。此外，我们会进一步探究中文独有的特

点并将我们的模型用于中文的释义生成工作中。

参考文献

[1] 贺芸, 庄成余. 论英语全球化传播的原因及其影响[D]. , 2004.

[2] 章宜华. 对外汉语学习词典释义问题探讨——国内外二语学习词典的对比研究[J]. 世界汉语教学, 2011, 1: 6-9.

[3] Turian J, Ratinov L, Bengio Y. Word representations: a simple and general method for semi-supervised learning[C] Proceedings of the 48th annual meeting of the association for computational linguistics. Association for Computational Linguistics, 2010: 384-394.

[4] Noraset T, Liang C, Birnbaum L, et al. Definition modeling: Learning to define word embeddings in natural language[C]. Thirty-First AAAI Conference on Artificial Intelligence. 2017.

[5] Sutskever I, Martens J, Hinton G E. Generating text with recurrent neural networks[C]. Proceedings of the 28th International Conference on Machine Learning (ICML-11). 2011: 1017-1024.

[6] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.

[7] Gadetsky A, Yakubovskiy I, Vetrov D. Conditional generators of words definitions[J]. arXiv preprint arXiv:1806.10090, 2018.

[8] Bartunov S, Kondrashkin D, Osokin A, et al. Breaking sticks and ambiguities with adaptive skip-gram[C]. Artificial Intelligence and Statistics. 2016: 130-138.

[9] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[J]. arXiv preprint arXiv:1409.0473, 2014.

[10] Yang L, Kong C, Chen Y, et al. Incorporating Sememes into Chinese Definition Modeling[J]. arXiv preprint arXiv:1905.06512, 2019.

[11] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[C]. Advances in neural information processing systems. 2014: 3104-3112.

[12] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.

[13] Hochreiter S, Bengio Y, Frasconi P, et al. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies[J]. 2001.

[14] Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality[C]. Advances in neural information processing systems. 2013: 3111-3119.

[15] Collobert R, Weston J, Bottou L, et al. Natural language processing (almost) from scratch[J]. Journal of machine learning research, 2011, 12(Aug): 2493-2537.

[16] Kim Y, Jernite Y, Sontag D, et al. Character-aware neural language models[C]. Thirtieth AAAI Conference on Artificial Intelligence. 2016.

[17] Papineni K, Roukos S, Ward T, et al. BLEU: a method for automatic evaluation of machine translation[C]. Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics, 2002: 311-318.

[18] Merity S, Xiong C, Bradbury J, et al. Pointer sentinel mixture models[J]. arXiv preprint arXiv:1609.07843, 2016.

[19] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.

[20] Yang Z, Yang D, Dyer C, et al. Hierarchical attention networks for document classification[C]. Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2016: 1480-1489.

[21] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]. Advances in neural information processing systems. 2017: 5998-6008.

作者联系方式:

张海同（1995-），男，北京工业大学硕士生，主要研究领域为自然语言处理。Email: 13073103@emails.bjut.edu.cn