Computer Science and Application 计算机科学与应用, 2020, 10(3), 564-574
Published Online March 2020 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2020.103058

Hans汉斯

A Method Incorporated Syntax Attention for Sentence Compression

Zhifeng Hao[1,2], Cheng Chen[1], Ruichu Cai[1], Wen Wen[1], Lijuan Wang[1]

[1]School of Computers, Guangdong University, Guangzhou Guangdong
[2]School of Mathematics & Big Data, Foshan University, Foshan Guangdong
Email: chenchengsmail@163.com

Received: Mar. 5th, 2020; accepted: Mar. 20th, 2020; published: Mar. 27th, 2020

Abstract

The size of dictionary in English Sentence Compression is limited, so using deep learning methods to compress sentences are prone to delete the keywords by mistake, then affect the meaning of the sentences after compression. To address this problem, this paper proposes a method incorporated syntax attention for sentence compression. Firstly, using two sets of encoder-decoder to encode and decode words and syntax, in the decoder stage, the Syntax-LSTM using syntax gates generates a syntax attention mechanism to lead a more grammatical output. The experimental results show that the F1 value reaches 0.7742 on the same domain dataset and 0.4186 on the cross-domain data, which proves that its results are more readable and more robustness compared with the existing methods.

Keywords

Sentence Compression, Syntax Attention Mechanism, Long Short-Term Memory, Robustness

一种融合语法信息的句子压缩方法

郝志峰[1,2]，陈　诚[1]，蔡瑞初[1]，温　雯[1]，王丽娟[1]

[1]广东工业大学计算机学院，广东 广州
[2]佛山科学技术学院数据与大数据学院，广东 佛山
Email: chenchengsmail@163.com

收稿日期：2020年3月5日；录用日期：2020年3月20日；发布日期：2020年3月27日

摘　要

英文句子压缩任务由于词典容量等限制，使用深度学习方法容易造成压缩后的句意与原句不同并一定程度影响语法逻辑。针对这一问题，文中提出一种融合语法信息的句子压缩方法。首先通过两组编解码器

文章引用: 郝志峰, 陈诚, 蔡瑞初, 温雯, 王丽娟. 一种融合语法信息的句子压缩方法[J]. 计算机科学与应用, 2020, 10(3): 564-574. DOI: 10.12677/csa.2020.103058

来对单词和词性分别进行编解码，在解码阶段通过带有语法注意力机制的长短期记忆网络(Syntax-LSTM)融合单词和词性信息产生语法注意力机制进而引导输出结果。与现有方法相比，实验结果表明该算法的F1值在领域数据集上达到了0.7742，在跨领域数据集上达到了0.4186，证明了其输出具有更好的可读性和鲁棒性。

关键词

句子压缩，语法注意力机制，长短期以及网络，鲁棒性

1. 引言

近年来，随着计算机处理技术的高速发展，互联网中的信息呈爆炸形式的发展，人们希望能够更精炼快速地捕获重要信息。句子压缩就是一种将冗长句子转换成精炼简洁句子的信息提炼方法。该项技术广泛地用于主题自动获取、摘要生成、问答系统等技术中，压缩后的句子需要保证原本句意不变，并且还需要一定程度上保证句子的可读性和语法上的逻辑性。

经典的句子压缩工作主要依赖于语法信息来进行判断，其主要思路是使用传统的句法信息，如基于选区的解析树(parse trees)的方法，通过使用语法树解析句子，以此进一步裁剪句子中的单词并重写句子[1] [2]。近年随着神经网络的发展，使用深度学习解决这个问题越来越受到人们的关注。Fillippova [3]首次将循环神经网络[4] [5] (Long Short-Term Memory)引入英文句子压缩中去，将句子压缩视为一种删除句子中单词的分类任务，相比传统方法有了较大的提升。神经网络模型是一种基于数据驱动的模型，拥有强大的特征提取能力，可以节约大量的人力物力，通过在大量的数据集上对模型进行训练，可以得到优于传统方法的效果。

尽管神经网络具有强大的特征提取能力，但网络端到端是不可控的。从句子压缩层面来看，在一些重要的名词或动词等被误删之后，往往会导致句意歪曲或压缩句阅读不通顺的问题。其次，由于不同领域下的常用词汇一般有较大的区别，在使用单一训练数据集得到的模型往往难以应对不同领域下得到的句子。对于不同风格和来源的句子，都需要使用到特定领域的训练数据集来进行训练才能达到理想效果，而数据集的获取是非常困难的，这一点极大地增加了应用难度。

针对以上问题，本文提出了一个既会使用句子中的单词信息，还会对词性信息进行获取的模型。通过提出一种带有语法注意力机制的长短期记忆网络，使用词性信息通过语法门控得到语法注意力，最终结合单词信息进行输出。使用词性序列来作为辅助序列是因为词性属于更为一般的信息，在句子中组成的单词不同的情况下，其所对应的词性序列往往有一定的相似性。使用词性序列进行辅助输出能够促进输出结果的语法正确性，进而保证可读性。本文的主要贡献如下：

1) 将句子的词性序列单独作为一条辅助序列，使用完整的词性序列并通过计算注意力机制来引导进行句子压缩。

2) 在模型中显式加入语法序列信息，使用语法注意力机制让输出的结果具有更强的可读性和鲁棒性，一定程度上也具有更好的跨领域应用能力。

3) 通过使用单词的语法信息，即使在词向量词典中找不到该词，也可以通过单词的词性信息一定程度上协助进行协助判断。一定程度上解决了词典非常见单词的 OOV (out of vocabulary)问题。

2. 相关工作

传统的句子压缩方法主要基于语法解析树(Parse Tree)。例如，Cohn [6]使用语法解析树来决定怎样重构句子，Filippova [7]通过剪枝依赖树删除冗余词进而达到压缩句子的目的。而随着神经网络技术的发展，越来越多学者考虑使用神经网络来解决这个问题。Fillippova 将编码器－解码器框架应用于删除式句子压缩任务，其核心组成是一个三层单向 LSTM 结构构成的编码解码模型，输出端使用 Softmax 作为激活函数来对每个单词进行二分类判断是否进行保留来组成压缩句，在大规模训练数据集上训练得到的效果相比较传统方法有较大的提升。后续 Lai [8]使用双向 LSTM 对句子双向信息进行捕捉，并在输出端使用条件随机场提高了效果；Tran [9]。提出适应于句子压缩任务的注意力机制即 t-attention 模型，通过对编码器序列每个节点的输出引入到对应的解码器中去，有了显著的提升。紧接着鹿忠磊[10]等人通过尝试增大 t-attention 的捕获范围来扩大信息捕获范围。

上述工作的输入为句中单词的词向量，改进的主要思路为通过增强编解码器的特征捕获能力来提高模型效果。而从语言学的角度来看，语法相关信息是一个同为通用的信息，Wilks [11]研究了通过对英文句子进行词性标注从而达到一定程度上消歧。而近期刘杰等[12]又通过对比发现使用循环神经网络能够更好判别出作文句间的逻辑性。Wang [13]等人开始尝试将语法信息显性地引入句子压缩任务中，输入到网络的词向量中加入词性和单词之间的依赖信息，在跨领域数据上验证了通过显式地加入相关语法信息能够增强模型的效果和迁移能力。

3. 本文方法

3.1. 问题定义

删除式句子压缩，即通过在原本长度较长的句子上，使用删除句子中单词的方式得到较短的句子，通过去除冗余词并保留核心词的方式重新组成压缩句。假设输入的原句为：

$$s=\left(w_1,w_2,\cdots,w_n\right) \tag{1}$$

w_i 表示组成句子的单词。通过删除原句中的一些单词并使用剩余单词共同组成新的句子。即希望得到这么一系列的标签 $y=\left(y_1,y_2,\cdots,y_n\right)$，其中 $y\in\{0,1\}$。其中标签 0 表示删除该单词，标签 1 表示保留该单词。示例如表 1 数据示例所示：

Table 1. Data example
表 1. 数据示例

原句	压缩句	标签
Eurozone business activity slowed in October, coming off a 27-month high in September to highlight concerns the economy is recovering only slowly from recession, a survey showed on Thursday.	Eurozone business activity slowed, coming off a high in September economy is recovering slowly.	1 1 1 1 0 0 1 1 1 0 1 1 1 0 0 0 0 1 1 1 0 1 0 0 0 0 0

3.2. 基本模型

本文借鉴了 Filippova 所提出的 3 层 LSTM 结构，提出了 LSTM-Res 网络。同样使用 3 层的 LSTM 结构作为编码器和解码器，多层的网络结构可以在不同捕获不同映射维度下的语义信息。此外，受到残差网络[14]的启发，在本次实验中将第一层和第二层的网络捕获到的低维和高维的语义叠加作为第三层的

LSTM 的输入。通过结合第一层和第二层的输出结果，能够捕获得到更多层次的语义信息，从而更好地得到输出的结果。其网络结构如图 1 LSTM-Res 结构所示。

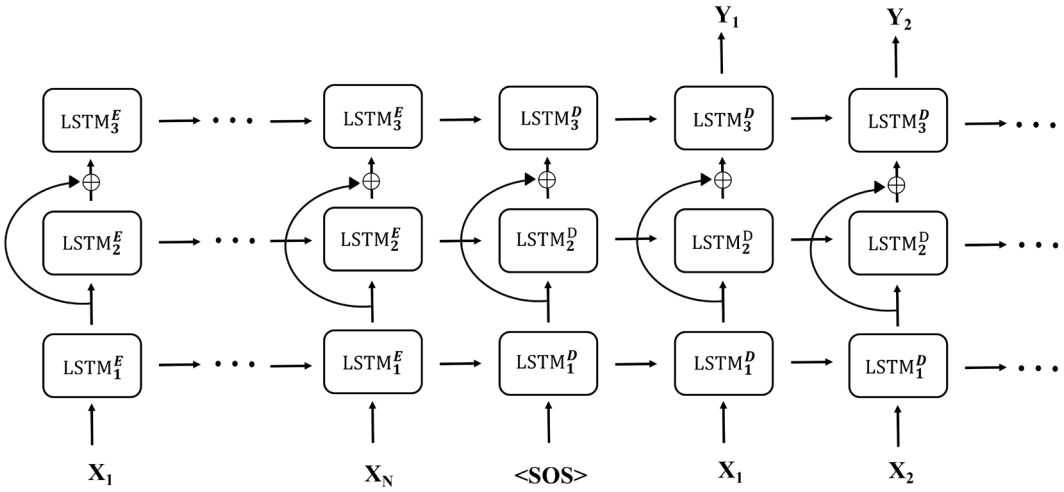

Figure 1. LSTM-Res structure
图 1. LSTM-Res 结构

图中的三层网络其中 $LSTM^E$ 代表编码器，$LSTM^D$ 表示解码器，其下标数字表示所在层数。作为外部输入的 X_1 到 X_N 表示句子中的 N 个单词的词向量，将句子完整投入到编码器之后将隐含层信息投入到解码器中；使用<SOS>作为解码器解码输出的标志，再将原句中的单词投入到解码器中，即对于编码器和解码器而言，输入单词是保持一致的。此时每个单词对应的输出 Y 则为每个单词对应是否删除的标志。

3.3. 语法特征

句子压缩任务中，不同句子包含的单词往往不近相同，但却可能拥有相似的语法逻辑，因此通过加入单词的词性信息，能够协助更好地理解句意。对于词性特征，在本次论文实验中使用 spacy 来进行分词和词性捕获，主要将词性分为 17 种类型如表 2 词性符号表示。

Table 2. Symbol of property of word
表 2. 词性符号表示

标注方式	词性	标注方式	词性
ADJ	形容词	DET	限定词
ADV	副词	NUM	数词
INTJ	感叹词	PART	粒子
NOUN	名词	PRON	代词
PROPN	专有名词	SCONJ	从属连词
VERB	动词	PUNCT	标点
ADP	介词	SYM	符号
AUX	助动词	X	其他
CCONJ	连接词		

除了本身词性序列排序上上具有相似性，由于词向量[15]。所需占用空间巨大[16]，词向量的词典只会对常用高频词进行保留，对于词表中没有的词则会统一使用<OOV>进行标记。然而，特殊的人名地名

等往往是句子中的重要组成成分，统一使用<OOV>进行替代会损失大量的词义信息。通过使用词性信息能够一定程度缓解这种状况，具体示例如表 3 数据标志所示。

Table 3. Data flag example
表 3. 数据标志示例

原句	词向量标志	词性标志	压缩句
Serge Ibaka - the Oklahoma City Thunder forward who was born in the Congo but played in Spain - has been granted Spanish citizenship and will play for the country in EuroBasket this summer, the event where spots in the 2012 Olympics will be decided.	<OOV> <OOV> <OOV> the <OOV> City <OOV> forward who was born in the <OOV>but played in <OOV> <OOV> has been granted <OOV>citizenship and will play for the country in <OOV>this summer, the event where <OOV>in the 2012 <OOV> will be decided.	<PROPN> <PROPN> <PUNCT> <DET> <PROPN> <PROPN> <PROPN> <ADV> <NOUN> <VERB> <VERB> <ADP> <DET> <PROPN> <CCONJ> <VERB> <DET> <PROPN> <PUNCT> <VERB> <VERB> <VERB> <ADJ> <NOUN> <CCONJ> <VERB> <VERB> <ADP> <DET> <NOUN> <ADP> <PROPN> <DET> <NOUN> <PUNCT> <DET> <NOUN> <ADV> <NOUN> <ADP> <DET> <NUM> <PROPN> <VERB> <VERB> <VERB> <PUNCT>	Serge Ibaka has been granted Spanish citizenship and will play in EuroBasket.

通过上述示例可以观察到，当句子中出现特殊的人名地名或动作时会被判定为<OOV>字符，此时通过额外补充的词性序列在一定程度上也可以更好地为句子压缩提供方向。如上例中的人名 Serge Ibaka 和形容词 Spanish 都使用<OOV>标志，但通过加入了专有名词<PROPN>和形容词<ADJ>作为标记；而一些动作如例子中的 has been granted 则被记为动词<VERB>。对于大多数陈述性的句子而言，其核心组成部分往往需要包括一个主语和一个谓语，而对于本次使用的新闻数据集而言主语往往是人或物，谓语则是动作，通过这样显性地添加词性信息到网络中，能够一定程度上正确地引导输出结果。

通过结合词性信息的网络整体处理框架如图 2 编码器解码器基本结构所示：

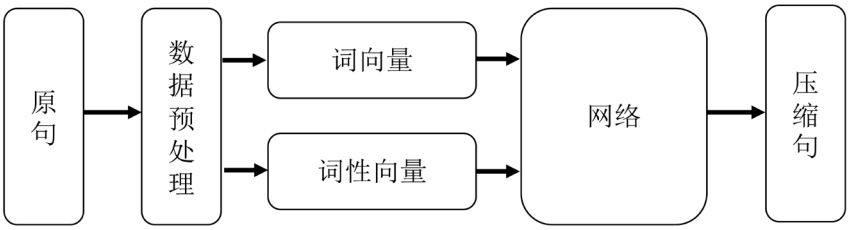

Figure 2. Encoder and decoder basic structure
图 2. 编码器解码器基本结构

先对原句子进行预处理，包括分词以及对词性信息进行捕获等。得到句子中的词向量和词性向量。将词向量和词性向量输入到网络中，网络通过结合词性序列和词向量序列最终得到压缩后的句子。

3.4. 语法注意力机制

为了更好地将语法特征融入网络，本文提出一种带有语法注意力机制的长短期记忆网络(Syntax-LSTM)，使用句子的语法信息控制门控信号，得到语法注意力来引导最终的输出判断。网络整体框架如图 3 带有语法注意力机制的编码解码结构所示：

输入到网络中的单词序列和词性序列分别使用两个独立的编码器和解码器进行编解码。编码解码器的结构主要基于上文提出的 LSTM-Res。对于单词序列，使用循环神经网络结构来进行信息捕获；同样地，词性信息也是使用循环神经网络来构建序列上的关联。词性序列相比单词序列，排列上更具有规则性，通过捕捉序列上的排序规律，使用循环神经网络结构能够更好地捕获词性信息，从而对最终输出进

行引导。本文通过一种带有语法门控的 LSTM (Syntax-LSTM)结构作为单词序列的解码器，使用这个语法门控来生成语法注意力机制，进而对最终的输出结果进行限制。改进之后的解码器部分的结构如图 4 编码解码结构：

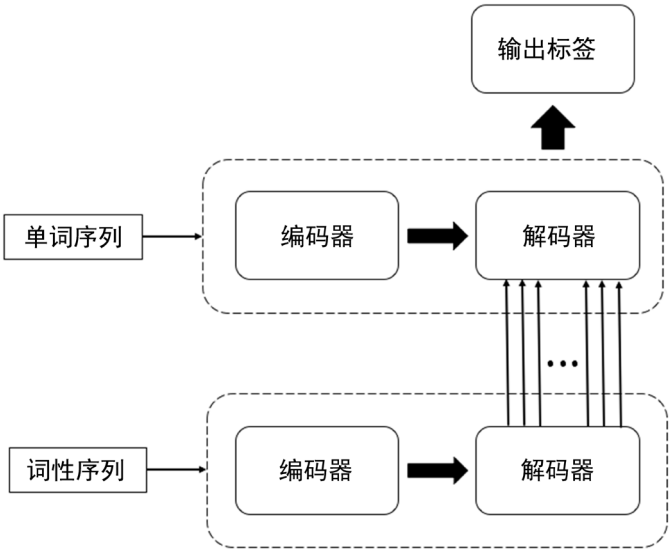

Figure 3. Encoding and decoding structure with syntax attention mechanism
图 3. 带有语法注意力机制的编码解码结构

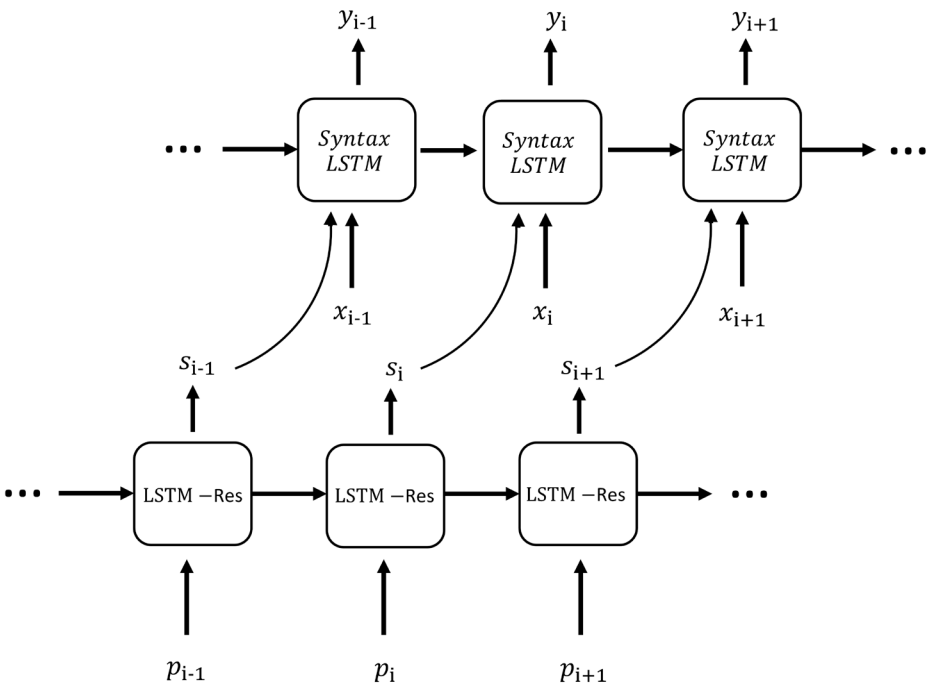

Figure 4. Decoder structure
图 4. 解码器结构

P_i 为句子中第 i 时刻单词对应的词性，投入到解码器之后得到输出词性信息 S_i。再将词性信息 S_i 作为引导信息和第 i 时刻的单词 x_i 一并投入到 Syntax-LSTM，输出对应标签 y_i。

对于 Syntax-LSTM，在原本的 LSTM 结构中增加一个额外结合语法信息的语法门控，并且由词性解码器的输出进行控制。具体结构如图 5 Syntax-LSTM 内部结构所示：

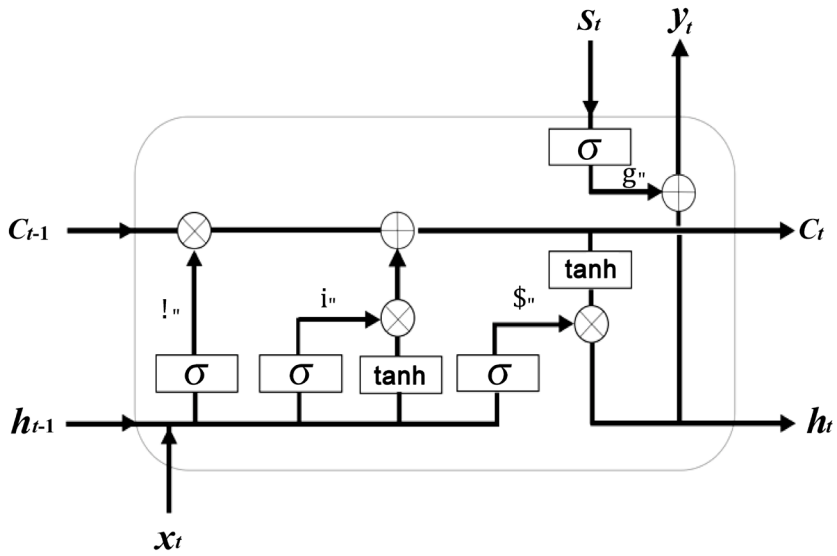

Figure 5. Internal structure of Syntax-LSTM
图 5. Syntax-LSTM 内部结构

Syntax-LSTM 的工作原理如下：

$$f_t = \sigma\left(W_i \cdot x_t + U_i \cdot h_{t-1} + b_i\right) \tag{2}$$

$$i_t = \sigma\left(W_f \cdot x_t + U_f \cdot h_{t-1} + b_f\right) \tag{3}$$

$$o_t = \sigma\left(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o\right) \tag{4}$$

$$g_t = \sigma\left(W_g \cdot s_t + U_g \cdot h_t + b_o\right) \tag{5}$$

$$\widetilde{C_t} = \tanh\left(W_c \cdot x_t + U_c \cdot h_{t-1} + b_c\right) \tag{6}$$

$$C_t = f_t \otimes C_{t-1} + i_t \otimes \widetilde{C_t} \tag{7}$$

$$h_t = o_t \otimes \tanh C_t \tag{8}$$

$$y_t = g_t \otimes h_t \tag{9}$$

其中 x_t 表示当前节点的输出信息 C_{t-1} 和 h_{t-1} 分别表示从上一节点传递过来的细胞状态(cell state)，和隐含状态(hidden state)。 f_t 表示遗忘门(forget gate)，控制从上一时刻传递过来状态是否进行丢弃；i_t 表示输入门(input gate)，控制当前时刻信息是否添加到隐含状态中；o_t 表示输出门(output gate)，控制选择作为当前节点输出的信息。此外，使用额外的信息 s_t，即对应于文本中词性解码器的输出来对最终输出结果 y_t 进行控制，通过这种方式让最终的输出结果更符合语法逻辑。

4. 数据与实验

4.1. 数据与预处理

对于有监督训练尤其是深度学习模型而言，需要大量对应带标注语料数据。本文的训练数据集使用 Filippova [17]公开的 4 万条数据集，这批数据来自谷歌新闻(Google News)中收集到的“标题－新闻”句

子对，进而通过参考标题对原句的依赖树进行剪枝，使用这种方式能够构建为原句子序列的压缩句，进而得到删除式句子压缩所需要的“原句－压缩句”平行语料。数据分为三部分：36,000 条句子对组成训练集用于模型训练，2000 条验证集用于训练过程中的交叉验证，2000 多条测试集用于最终测试模型效果。

实验采用了开源的自然语言处理工具 spacy 对句子进行了分词等预处理工作。对于单词的词向量，采用了开源 word2vec 中的 skip-gram 训练方式基于本次数据集进行训练最终得到维度为 97 的词向量，使用预训练好的词向量能够一定程度上提高实验效果。由于词向量的训练依赖于词典，通过统计数据集选出词频最高的 8000 个词构成词典训练他们的词向量。

超出单词表的单词或字符均使用符号<OOV>作为标识，在句子开头加入<SOS>作为句子开始的标志。由于句子长短不一，本次实验中通过统计选取数据集句子长度，选择了 120 个单词的句长，即对于小于 120 词的句子，使用<PAD>进行补齐，而对于超出 120 个单词的句子，则对超出的单词直接进行截断丢弃。

此外，由于本次任务为删除式的句子压缩，因而需要使用原句子和压缩后的句子构造对应的标签，标签为 0 表示句子中的该单词应该被删除，而标签为 1 则表示单词应该被保留，而<PAD>和<SOS>部分则使用标签 2 进行区别表示。

4.2. 实验设置

本次实验中的词性编码解码器和单词编码解码器的输入维度和隐含层维度大小统一设置均为 100。其中输入的词性向量和词向量均为 97 维度，而末尾的 3 个维度为标志位。在编码阶段，标志位不会进行标记均设为 0；而在解码阶段，当前词的标志位由上一个词的标签输出结果决定，标签输出 1、2、3 则对应的标志位上的第 1、2、3 位置为 1，其他位置为 0。通过使用标志位的方式，能够将上一个节点的输出的结果传递给下一个节点，当前节点结合上一输出标签引导当前节点的输出。

根据本次数据集的特点对模型采用了如表 4 模型基本参数的设定方式。并且采用了提前停止的策略[18] (early stop)，当验证集的 F1 数值超过 5 轮不增加之后，停止训练。

Table 4. Model basic parameters
表 4. 模型基本参数

参数	设定值
最大训练轮数	50
批大小	2000
学习率	0.001
最大句长	120
词典大小	8000
词向量维度	97
隐含层大小	100

4.3. 实验结果

本文对实验结果的评价指标主要通过 F1 分数、以及压缩率 CR(Compression Rate)二者进行衡量。F1 指标是一种综合召回率和精确率的指标。其表示方法如下：

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{10}$$

上述公式中的precision表示精确率，recall表示召回率。压缩率是一种用来描述句子压缩前后单词占比的指标，其计算公式如下：

$$\mathrm{CR} = \frac{N_{\text{压缩后句子单词数}}}{N_{\text{原句子单词数}}} \tag{11}$$

对于这一指标的目标是压缩率能够与原本句子中的基本保持一致，结合 F1 指标，CR 能够一定程度说明压缩后的句子充分压缩了并且整体语义保留完整。

主要对比的网络主要包括 Filippova 提出的三层 LSTM 结构，一种使用双向 LSTM 更好捕获句子前后语义并进行解码的网络结构以及在神经机器翻译系统中提出来在双向 LSTM 的基础上加入注意力机制(Attention)，一种针对于句子压缩任务特点提出来的 T-Attention 结构，也是目前的 state-of-the-art。实验测试集结果如表 5 Google News 数据集上的表现所示。

Table 5. Performance on the dataset of Google News
表 5. Google News 数据集上的表现

Approach	F1	CR (Ground Truth: 0.401)
3LSTM	0.7577	0.3749
Bi-LSTM	0.7510	0.4026
Attention	0.7631	0.3711
Bi-LSTM-TA	0.7698	0.3783
LSTM-Res	0.7706	0.3742
LSTM-Res-Syntax	0.7742	0.4025

增加残差结构的 LSTM-Res 相比 Filippova 的三层 LSTM 结构 F1 数值上高出 0.0129，并且是与当前达到 state-of-the-art 的 Bi-LSTM-TA 相当。而在通过加入语法信息，又在 LSTM-Res 又在原有的基础上有所提升，并且可以观察到压缩率 CR 和测试集的标准压缩率最为接近。对于同一来源的训练集的测试样本，加入语法门控的效果与当前 state-of-the-art 相当。

为了进一步比较模型的跨领域泛化能力，下面使用了与训练数据集不同来源且写作风格不同的 NBC News 数据集来做进一步的验证。

Table 6. Performance on the dataset of NBC News
表 6. NBC News 数据集上的表现

Approach	F1	CR (Ground Truth: 0.6947)
3LSTM	0.4054	0.3177
Bi-LSTM	0.4007	0.3243
Attention	0.3963	0.3042
Bi-LSTM-TA	0.3918	0.3000
LSTM-Res	0.3958	0.3024
LSTM-Res-Syntax	0.4186	0.3274

通过表 6 NBC News 数据集上的表现可以看到，在 Google News 的测试集中表现一般的 3LSTM 和 Bi-LSTM 在 NBC News 上表现良好，这一定程度上说明了简单的网络尽管对于单一数据上的拟合不够好，

却能够在不同来源的数据集中变现出了良好的泛化能力。相比之下，原本表现良好的 Bi-LSTM-tA 在跨领域数据集中表现反而为最差。这反应了拟合能力强的模型往往在跨领域或者是在处理差别较大的数据时效果大打折扣。而本文提出的加入了语法信息的 LSTM-Res-Syntax 则通过捕捉句子中通用的语法信息，增强对于未见过句型的泛化能力。

5. 结论

本文提出一种融合语法信息的句子压缩方法。在该方法中，文本使用了词性序列来作为语法信息，然后提出了一种通过语法门来形成语法注意力的 Syntax-LSTM，通过这一门控生成注意力，最终进行单词序列标签的判断。词性相比单词而言，是一种更为一般的信息，对于单词不同的句子其词性序列却有着一定的相似性，通过捕捉这类的相似结构，能够得到更好的输出并具有更强的泛化能力。最终实验取得了目前的 state-of-the-art 相当的分数，并在跨领域数据集 NBC News 上的性能均优于其他模型，证明了该方法具有更好的鲁棒性和可迁移性。下一步将对语法门控引入到文本摘要任务，尝试用删除式的方式做摘要的抽取。

基金项目

国家自然科学基金(61876043)；广东省自然科学基金(2014A030306004，2014A030308008)；广东特支计划(2015TQ01X140)；广州市珠江科技新星(201610010101)；广州市科技计划(201902010058)；NSFC-广东联合基金(U1501254)。

参考文献

[1] Jing, H. (2000) Sentence Reduction for Automatic Text Summarization. *Conference on Applied Natural Language Processing*, Seattle, 29 April-4 May 2000, 310-315. https://doi.org/10.3115/974147.974190

[2] Knight, K. and Marcu, D. (2000) Statistics-Based Summarization—Step One: Sentence Compression. *National Conference on Artificial Intelligence*, Austin, 703-710.

[3] Filippova, K., Alfonseca, E., Colmenares, C.A., *et al*. (2015) Sentence Compression by Deletion with LSTMs. *Empirical Methods in Natural Language Processing*, Lisbon, September 2015, 360-368. https://doi.org/10.18653/v1/D15-1042

[4] Hochreiter, S. and Schmidhuber, J. (1996) LSTM Can Solve Hard Long Time Lag Problems. *Neural Information Processing Systems*, Denver, 473-479.

[5] Gers, F.A., Schmidhuber, J., Cummins, F., *et al*. (2000) Learning to Forget: Continual Prediction with LSTM. *Neural Computation*, **12**, 2451-2471. https://doi.org/10.1162/089976600300015015

[6] Cohn, T. and Lapata, M. (2009) Sentence Compression as Tree Transduction. *Journal of Artificial Intelligence Research*, **34**, 637-674. https://doi.org/10.1613/jair.2655

[7] Filippova, K. and Strube, M. (2008) Dependency Tree Based Sentence Compression. *International Conference on Natural Language Generation*, Salt Fork, 12-14 June 2008, 25-32. https://doi.org/10.3115/1708322.1708329

[8] Lai, D.V., Truong, N. and Minh, N.L. (2017) Deletion-Based Sentence Compression Using Bi-enc-dec LSTM, 2017. In: *Conference of the Pacific Association for Computational Linguistics*, Springer, Singapore, 249-260. https://doi.org/10.1007/978-981-10-8438-6_20

[9] Tran, N., Luong, V., Nguyen, N.L., *et al*. (2016) Effective Attention-Based Neural Architectures for Sentence Compression with Bidirectional Long Short-Term Memory. *Symposium on Information and Communication Technology*, Ho Chi Minh City, 8-9 December 2016, 123-130. https://doi.org/10.1145/3011077.3011111

[10] 鹿忠磊, 刘文芬, 周艳芳, 等. 基于预读及简单注意力机制的句子压缩方法[J]. 计算机应用研究, 2019, 36(2): 57-61+80.

[11] Wilks, Y. and Stevenson, M. (1998) The Grammar of Sense: Using Part-of-Speech Tags as a First Step in Semantic Disambiguation. *Natural Language Engineering*, **4**, 135-143. https://doi.org/10.1017/S1351324998001946

[12] 刘杰, 孙娜, 袁克柔, 等. 中文作文句间逻辑合理性智能判别方法研究[J]. 计算机应用与软件, 2019, 36(1): 77-83.

[13] Wang, L., Jiang, J., Chieu, H.L., *et al*. (2017) Can Syntax Help? Improving an LSTM-Based Sentence Compression Model for New Domains. *Meeting of the Association for Computational Linguistics*, Vancouver, July 2017, 1385-1393. https://doi.org/10.18653/v1/P17-1127

[14] He, K., Zhang, X., Ren, S., *et al*. (2016) Deep Residual Learning for Image Recognition. *Computer Vision and Pattern Recognition*, Las Vegas, 27-30 June 2016, 770-778. https://doi.org/10.1109/CVPR.2016.90

[15] Mikolov, T., Sutskever, I., Chen, K., *et al*. (2013) Distributed Representations of Words and Phrases and Their Compositionality. *Advances in Neural Information Processing Systems*, Lake Tahoe, December 2013, 3111-3119.

[16] Mikolov, T., Chen, K., Corrado, G., *et al*. (2013) Efficient Estimation of Word Representations in Vector Space. *International Conference on Learning Representations*. arXiv preprint arXiv:1301.3781

[17] Filippova, K. and Altun, Y. (2013) Overcoming the Lack of Parallel Data in Sentence Compression. *Empirical Methods in Natural Language Processing*, Seattle, 1481-1491.

[18] Prechelt, L. (1998) Early Stopping—But When? In: Orr, G.B. and Müller, K.-R., Eds., *Neural Networks*: *Tricks of the Trade*, Vol. 1524, Springer, Berlin, Heidelberg, 55-69. https://doi.org/10.1007/3-540-49430-8_3