

基于注意力机制的句子排序方法

栾克鑫¹, 杜新凯², 孙承杰¹, 刘秉权¹, 王晓龙¹

(1. 哈尔滨工业大学, 黑龙江省哈尔滨市 150000; 2. 北京中科汇联科技股份有限公司, 北京市 100085)

摘要: 句子排序是多文档自动摘要和答案融合任务的关键技术, 其效果直接影响摘要或者答案融合结果的可读性。作为句子排序的重要依据, 语义逻辑关系的准确度对于排序结果的可读性有很大影响。为此, 本文提出了引入注意力机制的句子排序模型, 以增强句子语义逻辑关系的捕获能力, 进而获取句子的合理排序。实验结果表明, 在句子排序任务中, 本文引入注意力机制的句子排序模型明显优于基线方法。

关键词: 句子排序; 注意力机制; 语义逻辑关系

中图分类号: TP391

文献标识码: A

Sentence ordering method based on attention mechanism

LUAN Kexin¹, DU Xinkai², SUN Chengjie¹, LIU Bingquan¹, WANG Xiaolong¹

(1. Harbin Institute of Technology, Harbin, Heilongjiang 150000, China; 2. Huilan Technology, Beijing 100085, China)

Abstract: Sentence ordering is the key technology of automatic summarization for multi document and answer fusion, and its effect directly affects the readability of the result of automatic summarization or answer fusion. As an important basis for sentence ordering, the accuracy of semantic logic relation is very important for the readability of result. Therefore, this paper proposes a sentence ordering model with attention mechanism, which enhances the capture ability of sentence semantic logic relation, and then obtains the reasonable order of sentences. The experimental results show that the sentence ordering model with attention mechanism is superior to baseline method in sentence ordering task.

Key words: sentence ordering; attention mechanism; semantic logic relation

1 简介

语义连贯性是文本可读、准确地表达语义和传递信息的必要条件。文本语义连贯也是读者理解文本的基础。例如:

“华罗庚教授是一位自学成才的著名的数学家。他 14 岁开始自学数学,每天坚持自学 10 小时,从不间断。1932 年,22 岁的华罗庚应清华大学数学系主任熊庆来的邀请,到清华大学工作。”

这段话能够明确表达出华罗庚是一名数学家以及他的早期经历。而语意连贯性差的段落,影响语义表达,影响读者理解。例如:

“1932 年,22 岁的华罗庚应清华大学数学系主任熊庆来的邀请,到清华大学工作。他 14 岁开始自学数学,每天坚持自学 10 小时,从不间断。华罗庚教授是一位自学成才的著名的数学家。”

上面的段落语义不连贯,不能很好的表述华罗庚的经历,影响读者理解。

因此,语义连贯性是很多自然语言处理(Natural Language Processing, NLP)任务(如多文档自动文摘^[1-3]、答案摘要^[4])的核心问题,研究如何保持语义连贯性的方法对 NLP 是有

收稿日期:

定稿日期:

基金项目: 国家自然科学基金项目 (61572151、61602131); 国家 863 项目 (2015AA015405).

作者简介: 栾克鑫(1992-), 男, 硕士, 主要研究方向: 答案自动抽取与融合、自动文摘等; 杜新凯(1982-), 男, 硕士, 主要研究方向: 智能客服, 自然语言处理; 孙承杰(1980-), 男, 博士, 副教授, 主要研究方向: 信息推荐, 信息抽取; 刘秉权(1970-), 男, 博士, 副教授, 主要研究方向: 自然语言处理、信息检索、知识服务等; 王晓龙(1955-), 男, 博士, 教授, 主要研究方向: 网络信息处理、自然语言处理、人工智能等

重要意义的。

通常在一个文本段落中,这种语意的连贯性是通过句子的顺序来保证的,句子之间的关系通过语法特征、语义特征保证的,比如在一个顺畅的文本中,句子之间通过一些连词(如因为...所以...,虽然...但是...等)以及句子所表达的语义逻辑,来确定先后顺序。然而,在多文档自动文摘(或者答案摘要任务)中,由于句子都是来自于不同的文本段落,无法利用句子在原文中的位置信息,只能通过语义关系来确定句子顺序。因此通过确定语义关系确定句子顺序来解决多文档自动文摘及答案摘要任务中的语义连贯性问题是一种可行方法。

关于解决句子排序^[5]问题,徐永东^[6]提出了一种多文档摘要中基于时间信息的句子排序,利用基于规则的时间信息抽取,语义计算以及时序推理方法来解决句子排序问题。姚超^[7]提出了一种基于内聚度的多文档文摘的句子排序方法,通过将相同话题的句子聚合到一起,避免话题中断,改善文摘可读性。

随着深度学习的发展,很多深度学习的方法被引入到句子排序中,Chen^[8]尝试了基于卷积神经网络(Convolutional Neural Networks, CNN)^[9],长短期记忆网络(Long Short-Term Memory Network, LSTM)^[10]的句子排序方法,使用 CNN、LSTM 等模型判断句子的前后句关系,并利用集束搜索(Beam Search)算法求解句子的最优排序。Logeswaran^[11]提出了一种基于循环神经网络的句子排序方法,通过判断句子在每个位置的可能性,求得最优排序结果。Gong^[12]提出了一种基于端到端的指针网络的句子排序方法,通过端到端的指针网络判断每个位置上的句子的可能性,求得较优排序结果。康世泽^[13]提出了一种基于马尔科夫随机游走模型的句子排序方法。

句子排序任务输入的句子集合来自于多个不同文档,句子之间没有明显的连接词,只能通过句子间语义逻辑关系进行排序,为此本文将通过如下的两个步骤完成句子排序任务。

一、求解句子间的逻辑关系

二、利用集束搜索算法求解较优的句子排序

句子排序任务的核心任务是识别句子间的逻辑关系。本文首先将句子两两组合,形成句子对,例如,一个句子集合有 3 句话,(句 1,句 2,句 3),将 3 个句子两两组合,并排除自身组合,形成 6 个句子对,如(句 1,句 2),(句 1,句 3),(句 2,句 1)等。将构造的句子对输入到模型中,通过模型得到输出概率,判断句子间逻辑关系。

注意力机制在机器翻译和生成式短文本摘要领域有着广泛的应用,其能够有效的发掘句子间的内在联系。注意力机制的重点放在关键词上,忽略其他无关词汇,其能够有效发现前后两句中词与词之间的关系。而词与词之间的关系是句子之间关系的基础,因此注意力机制能够有利于捕捉句子间语义逻辑关系。本文将注意力机制引入句子排序模型,提升句子排序的效果。

本文做的主要贡献如下:

1、提出了一个基于句子匹配方法(sentence matching)^[14]的句子排序方法,结合句子匹配方法,寻找句子间的联系,提升模型捕捉句子相关性的能力。

2、提出了基于静态注意力机制(static attention)^[15]的句子排序模型,结合静态注意力机制,利用句子间词与词的联系,提升模型捕捉句子间语义逻辑关系的能力。

3、提出了基于词对齐注意力机制(word-by-word attention)^[15]的句子排序模型,结合词对齐注意力机制模型,动态更新句子间词与词之间权重,捕捉词与词之间的联系,提升捕捉句子间语义逻辑关系能力。

4、提出了一种基于句内注意力机制(inner-attention)^[16]的句子排序方法,结合句内注意力机制,利用句子内的最重要信息,寻找句子间联系,捕捉两句子间的语义逻辑关系。

2 基于注意力机制的句子排序模型

2.1 问题描述

在答案融合及多文档摘要任务中，候选句子是从多个不同文档中抽取的，句子的顺序无法利用其在文档中的位置来或者显式的连接词进行排列，只能通过句子间的逻辑关系进行排序。因此，解决句子排序问题的主要方法就是挖掘句子之间的语义逻辑关系。 s 中包含 n 个无序句子 $s_1, s_2, s_3, \dots, s_n$ ，句子排序的任务是将 n 个无序句子排序成有序的句子序列 $s_{o_1}, s_{o_2}, s_{o_3}, \dots, s_{o_n}$ 。 s_{o_n} 对应的是排序后的句子。

本文将句子排序任务分成句子逻辑关系判断和排序两个部分：

- 1、句子逻辑关系判断：句子排序任务中，句子逻辑关系判断可以转化为判断句子的前后关系。因此，本文建立模型计算句子 s_i 及 s_j 的前后句关系概率 $score(s, o, i, j)$ 。
- 2、排序：本文使用集束搜索算法寻找较优句子排序结果。句子排序任务的目标函数如式(1)：

$$Score(s, o) = \sum_{i=1}^k \sum_{j=i+1}^k score(s, o, i, j) \quad (1)$$

其中， $score(s, o, i, j)$ 是句子 s_i 及句子 s_j 的前后关系的概率打分， s 为句子集合， o 为句子顺序，目标函数求解使 $Score(s, o)$ 得分最大的句子顺序。

接下来三个小节将分别介绍句子语义逻辑关系判别模型，特征提取方法及如何根据句子逻辑关系利用集束搜索(Beam Search)算法实现句子排序。

2.2 句子语义逻辑关系判别模型

句子语义逻辑关系判别模型如图 1 所示：

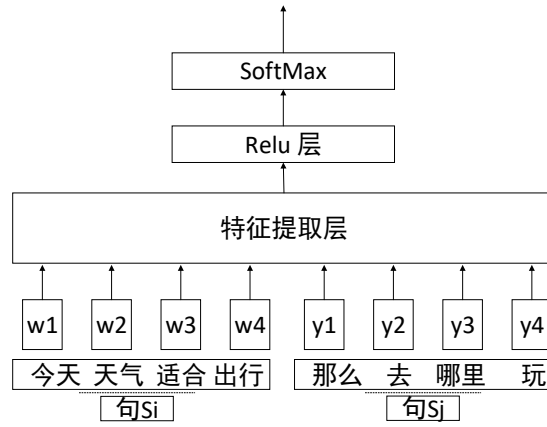


图 1 句子语义逻辑关系判别模型

对于两个句子的句子语义逻辑关系判别，本文将其看作一个二分类问题，即句子 s_i 与 s_j 是语义逻辑的前后关系。本文将句子 s_i 与 s_j 用预训练的词向量进行表示，然后通过特征提取层对两个句子进行特征提取，得到特征 $etra(s_i, s_j)$ ，然后输入到前馈神经网络中，得到两句话的前后关系置信度 p_{ij} 。如果 $p_{ij} > 0.5$ ，就说明句子 s_i 在句子 s_j 之前的更合理，否则，句子 s_i 放在句子 s_j 之后的更合理。

句子 s_i 与 s_j 的概率 p_{ij} 的计算公式如下所示：

$$h_{ij} = \phi(W_h^T etra(s_i, s_j)) + b_h \quad (2)$$

$$p_{ij} = \sigma(W_p^T h_{ij} + b_p) \quad (3)$$

其中， $W_h \in R^{F \times h}$, F 为特征提取层得到的特征的维度， h 为隐含层维度， $b_h \in R^h$ ， $W_p \in R^{h \times 2}$, $b_p \in R^2$ ， ϕ 为 Relu 激活函数， σ 为 Softmax 激活函数，损失函数采用交叉熵损失函数。模型的目标函数如(1)所示，其中 $score(s, o, i, j) = p_{ij}$ ，表示句子 s_i 及 s_j 的前后句关系概率，

$Score(s, o)$ 为句子顺序 o 下的总分数。

2.3 特征提取方法

如图 1 所示, 特征提取是句子语义逻辑关系判断的关键一步, 其表示语义逻辑关系的能力关系到句子逻辑关系判断的准确性。为此, 本文在本小结中从句子匹配和注意力机制两个角度进行特征提取, 并基于这些特征进行句子语义逻辑关系的判断。

2.3.1 基于句子匹配的特征提取

Mou^[4]提出了一种句子匹配方法(sentence matching), 通过对提取的两个句子特征进行拼接、按位乘和按位减操作, 得到组合特征。通过组合特征, 句子匹配方法能够有效地识别句子之前的相关性。而句子间的相关性是句子逻辑关系的一个很重要的影响因素, 因此本文引入了句子匹配方法提升句子逻辑关系的判别能力, 进而提高句子排序效果。

基于句子匹配的特征提取方法如图 2 所示:

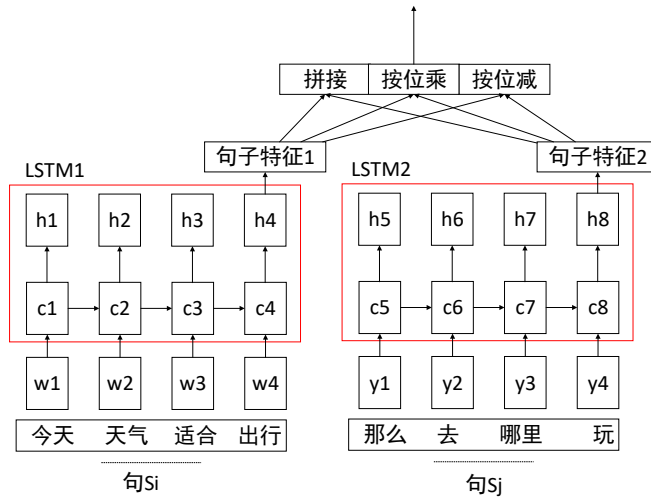


图 2 基于句子匹配方法的特征提取

本文实现了基于句子匹配的特征提取方法, 该方法利用 LSTM 对句子词向量特征进行特征提取, 并用句子匹配方法对得到的 LSTM 特征进行进一步操作。LSTM 全称是长短时记忆网络, 是一种用来处理时序问题的模型。通常的 LSTM 具有记忆块 $c, c \in R^L$, 其中 L 为 LSTM 的维度, LSTM 还具有三个门输入门 $i, i \in R^L$, 输出门 $o, o \in R^L$ 以及遗忘门 $f, f \in R^L$ 。LSTM 计算公式如式(4)(5)(6)所示:

$$\begin{bmatrix} i_t \\ o_t \\ f_t \\ \tilde{c}_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \emptyset \end{bmatrix} (W_g^T [e_t] + b_g) \quad (4)$$

$$c_t = c_{t-1} \odot f_t + \tilde{c}_t \odot i_t \quad (5)$$

$$h_t = o_t \odot \emptyset(c_t) \quad (6)$$

其中 $W_g^T \in R^{(L+F) \times 4F}$, L 为词向量长度, $b_g \in R^{4F}$ 。 σ 为 softmax 激活函数, \emptyset 为 Relu 激活函数。 $t = 1, 2, \dots, n$, n 为句子长度。句子匹配方法是一种句子特征处理方法, 用来寻找句子间的关系, 该方法对从句子级别的特征 S_1, S_2 分别做三种操作:

- 1、特征向量按位相乘
- 2、特征向量按位相减
- 3、特征向量拼接

将经过上述三种操作得到的特征拼接到一起, 得到最终特征。

如图 2 所示, 首先 LSTM1 和 LSTM2 分别对句子 s_i 及 s_j 进行特征提取, 并用最后一个单

词的隐向量作为句子 s_i 和 s_j 的特征。然后对 S_1, S_2 做上述三种操作, 得到三种不同特征(1) $[S_1, S_2]$, (2) $S_1 * S_2$, (3) $S_1 - S_2$ 。对这三种特征进行拼接, 得到句子匹配的特征向量 $S = [[S_1, S_2]; S_1 * S_2; S_1 - S_2]$ 。最后, 将得到的特征 S 输入到前馈神经网络中, 得到前后句关系概率 p_{ij} 。

2.3.2 基于静态注意力机制的特征提取

注意力机制能够对两个句子中词语之间的关系进行建模, 通过计算句1所有词与句2每个词之间的权重, 从而发现不同句子中词与词关系。两个句子中词语的关系有助于捕获句子间的语义逻辑关系, 为此, 本文引入注意力机制提取句子特征, 提升句子排序结果, 增强模型的可视化能力和可解释性。

静态注意力机制是注意力机制中常用的模型之一, 静态注意力机制根据前句所有节点隐藏层的输出及后句最终节点隐藏层的输出计算得到特征。在句子排序任务中, 静态注意力机制能够计算前后两个句子词之间的关系, 在一定程度上能够增强模型捕捉句子语义逻辑关系的能力。

本文使用基于静态注意力机制的序列-序列(sequence to sequence, seq2seq)模型对句子对 s_i 和 s_j 进行特征提取。在神经机器翻译模型中, 序列-序列模型通常用来将某种语言的一个句子翻译成另一种语言的一个句子, 其一般由编码器及解码器组成。本文采用 LSTM 作为中编码器及解码器的构成单元, 首先通过编码器将输入句 s_i 映射成隐向量, 然后结合静态注意力机制, 在解码阶段预测 s_j 的在 s_i 条件下的生成概率。从这个序列到序列映射过程中学习两个句子 s_i 和 s_j 语义逻辑关系特征。图3是基于静态注意力机制的特征提取模型:

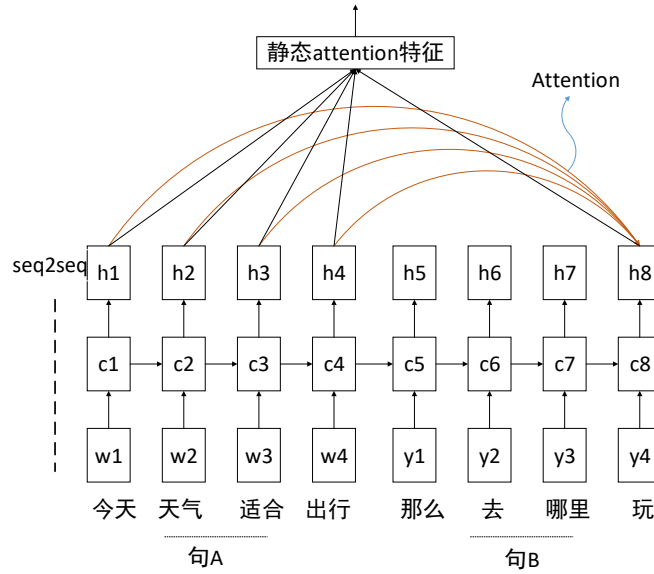


图3 基于静态注意力机制的特征提取

基于静态注意力机制的特征提取计算公式如式(7)(8)(9)(10)所示:

$$M = \tanh(W^y Y + (W^h h_n \times e_L)) \quad (7)$$

$$a = \text{softmax}(w^T M) \quad (8)$$

$$r = Y a_t^T \quad (9)$$

$$h^* = \tanh(W^p r + W^x h_n) \quad (10)$$

其中 $W^y, W^h \in R^{F \times F}, e_L \in R^F, w \in R^F, W^p, W^x \in R^{F \times F}$ 。 $W^y, W^h, W^p, W^x, w, e_L$ 是随机生成的可训练的向量矩阵。 Y 是编码阶段所有的隐藏层输出, h_n 是解码阶段最后一个状态的隐藏层输出。 r 是由编码阶段及 h_n 计算出的权重表示。 h^* 是最终输出的特征, 最终将 h^* 输入到前馈神经网络中。

2.3.3 基于词对齐的注意力机制的特征提取

基于静态注意力机制的特征提取在注意力特征计算阶段，只计算编码阶段隐藏层输出与解码阶段最终输出，没有考虑解码阶段所有节点输出与编码阶段隐藏层输出的关系。为了提升模型捕捉句子语义逻辑关系的能力，本文将词对齐注意力机制引入模型中，词对齐的注意力机制能计算编码阶段与解码阶段词与词的关系，提高模型捕捉语义逻辑关系的能力。

基于词对齐的注意力机制的特征提取模型如图 4 所示：

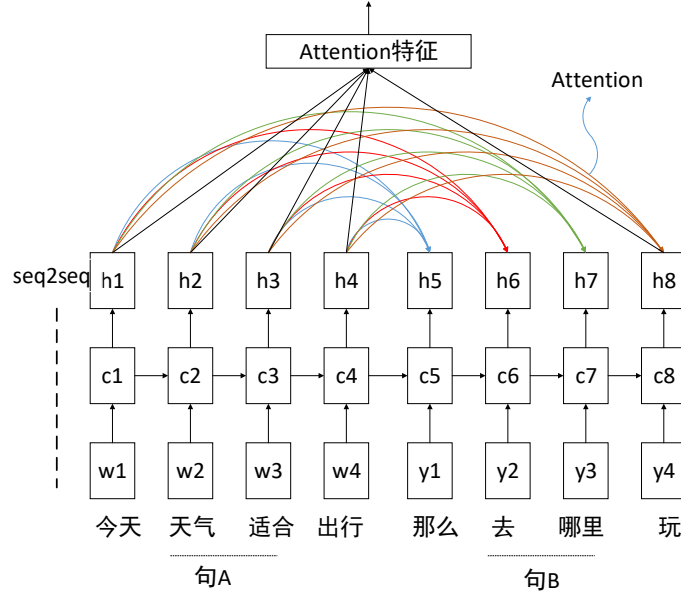


图 4 基于词对齐注意力机制的特征提取

计算公式如式(11)(12)(13)(14)所示：

$$M_t = \tanh(W^y Y + (W^h h_t + W^r r_{t-1}) \times e_L) \quad (11)$$

$$a_t = \text{softmax}(w^T M_t) \quad (12)$$

$$r_t = Y a_t^T + \tanh(W^t r_{t-1}) \quad (13)$$

$$h^* = \tanh(W^p r_N + W^x h_N) \quad (14)$$

其中 $W^y, W^h, W^r \in R^{F \times F}$, $e_L \in R^F$, $w \in R^F$, $W^p, W^x, W^t \in R^{F \times F}$, $W^y, W^h, W^r, W^t, w, e_L$ 是随机生成的可训练的向量矩阵,用来计算注意力权重分布。 W^p, W^x 同样是可训练的向量矩阵,用来计算最后的输出。 h_t 是解码阶段 t 阶段的隐藏层输出。 r_{t-1} 是上一时间节点计算出的权重表示, h^* 是最终输出的特征,然后将 h^* 输入到前馈神经网络中。

2.3.4 基于句内注意力机制的特征提取

Liu^[16]提出了一种基于句内注意力机制的方法，使用双向长短时记忆递归神经网络 (Bidirectional Long Short-Term Memory, BiLSTM)，提取句子级别的特征，将 BiLSTM 所有隐藏层输出 Y 做均值池化 (Mean-Pooling)，得到 R_{mean} ，其相当于词对齐注意力模型的解码阶段的最终输出。 Y 与 R_{mean} 做 attention 计算，得到句子特征表示 S_i 。前后两个句子的句子特征 S_i, S_j 做拼接、按位乘和按位减，最终得到输出特征 S 。

本文尝试将句内注意力机制应用到句子排序中，句内注意力机制提取出的特征相较于 LSTM 特征能更好地保留核心信息，相较于基于句子匹配方法的特征提取，能够更好地捕捉句子间的语义逻辑关系。与静态注意力机制以及词对齐注意力机制的捕捉语义逻辑关系的方式不同，句内注意力机制的核心是静态注意力机制以及句子匹配方法，通过静态注意力机制更好地提取句子级别的特征，同时利用句子匹配方法捕捉句子逻辑关系。

基于句内注意力机制的特征提取如图 5 所示：

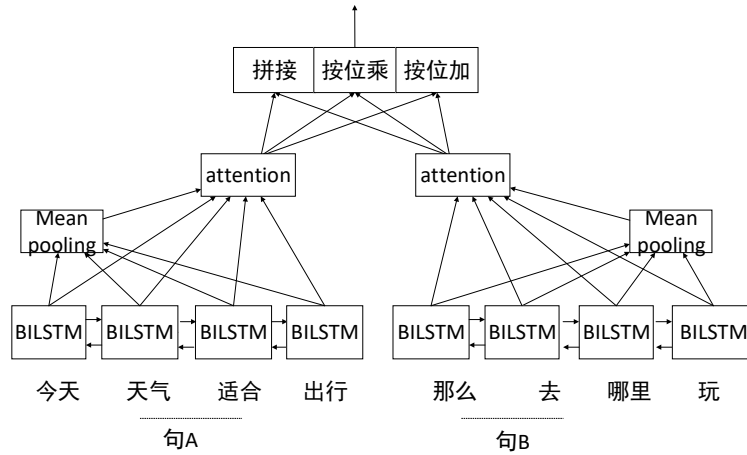


图5 基于句内注意力机制的特征提取

基于句内注意力机制的特征提取的公式如式(15)(16)(17)(18)所示:

$$M = \tanh(W^y Y + W^h R_{mean} \times e_L) \quad (15)$$

$$a = \text{softmax}(w^T M) \quad (16)$$

$$r = Y a_t^T \quad (17)$$

$$h^* = \tanh(W^p r + W^x R_{mean}) \quad (18)$$

其中 $W^y, W^h \in R^{F \times F}, e_L \in R^F, w \in R^F, W^p, W^x \in R^{F \times F}$ 。 $W^y, W^h, W^p, W^x, e_L, w$ 是随机生成的可训练的向量矩阵。 Y 是编码阶段所有隐藏层的输出, R_{mean} 是 Y 做均值池化运算后的输出。 r 是 Y 及 R_{mean} 计算出的权重表示, h^* 是最终注意力特征计算层的输出。

```

1、 Beam Search 算法伪代码
2、 #####S 为待排序句子集合
3、 #Beam 用来记录句子候选序列 o 以及得分 score, Beam 大小设置为 N
4、 Beam = [[o[s], score] for s in S]
5、 For i in range(step):
6、 #get_table 获取 Beam 中所有可能的序列已访问的节点
7、 Hash_table = get_table(Beam)
8、 New_beam = []
9、 #生成候选排序, 后继节点不在 Hash_table 中
10、 For o, score in Beam:
11、     For s in S:
12、         If s not in Hash_table(o):
13、             New_o = o.append(s)
14、             New_score = score + p(o[1], s)
15、             tmp = [New_o, New_score]
16、             New_beam.append(tmp)
17、 #取 New_beam 中最好的 N 个候选生成新的 Beam 表
18、 Beam = get_max(New_beam)

```

算法1 基于集束搜索的句子排序算法

2.4 基于集束搜索的句子排序算法

在求解出所有句子间的语义逻辑关系后，本文使用集束搜索(Beam Search)求解最优排序。基于集束搜索的句子排序算法如算法 1 所示。

2.5 目标函数及优化

在本文中，本文通过捕捉句子间的逻辑关系来完成句子排序任务。因此本文建立的是二分类模型，本文利用句子在文章中的顺序来生成训练数据 (S_i, S_j, p) ，即，如果句 S_i 和句 S_j 在文章中是前后句关系，那么 $p=1$ ，如果不是前后句关系，则 $p=0$ 。训练数据的正负比例为 1:1。

模型训练的目标函数是交叉熵函数，公式如式(19)所示：

$$J(\theta) = -\frac{1}{2m} \sum_{i=1}^{2m} y^i \log p_{x^i} + (1 - y^i) \log(1 - p_{x^i}) \quad (19)$$

其中 p_{x^i} 是模型输出的前后句关系概率， θ 是模型参数。

本文采用 mini-batch 随机梯度下降的方法来训练本文的模型，同时在前馈神经网络每一层中加入 batch normalization^[17]加快训练速度，通过加入 batch normalization 可以使模型可以采用较大的学习率，加快模型训练速度。

3 实验

3.1 实验数据集

本文在百度百科语料和新华社新闻语料进行了句子排序实验来验证本文提出的句子排序模型。本文从两个数据集中抽取句子数量大于 2 的段落作为语料并划分为训练集与测试集。其中新华社语料多是事实的陈述，百度百科语料比较复杂，包含人物介绍、事物介绍、文言文、新闻和书籍介绍，比如莫言的作品介绍。本文将训练集中得到的句子对作为训练数据，其中在原段落中为前后句关系的句子对最为正例，其他情况作为负例，正负比例为 1:1。

各个数据集篇章数量表：

数据集数量篇章	训练集	测试集
新华社语料	9873	295
百度百科语料	4109	204

表 1 数据集篇章数量

各个数据集句子数量表

数据集句子对数量	训练集	测试集
新华社语料	212042	5326
百度百科语料	78068	3202

表 2 数据集句子对数量

3.2 评估准则

本文采用 PMR 准则来评估模型，PMR 计算公式如式(20)：

$$PMR = \frac{1}{M} \sum_{m=1}^M 1\{\hat{o}^m = o^{m*}\} \quad (20)$$

PMR 评价准则表示排序后的句子，如果在原来的位置上，则累加 1，不在，则累加 0，最终除以句子总数。

3.3 实验参数

基线及各个方法的参数设置如下：Dropout 大小设置为 0.5，输入到模型中的句子长度保留 30 词，对于长度不够的句子进行补 0 处理，对于过长的句子进行截断处理。mini-batch 大小设置为 64，学习率设置为 0.3。在前馈神经网络中每一层加入 batch normalization 进行数据归一化。词向量长度为 200，LSTM 及双向 LSTM 隐藏层长度为 200，前馈神经网络隐藏层长度设置为 100。

Dropout 参数影响模型的泛化能力，句子长度通过统计句子的平均长度进行设置。

Mini-batch 影响模型的下降速度，同时在一定程度上影响模型结果。因为本文引入 batch normalization，可以设置较大的学习率，增强模型的收敛速度。LSTM 及双向 LSTM 隐藏层维度与词向量维度保持一致。

3.4 实验结果及分析

PMR	新华社语料	百度百科
基线 LSTM(Chen)	38.01%	35.75%
句子匹配方法	37.04%	39.69%
静态注意力模型	41.49%	35.81%
词对齐注意力模型	42.06%	39.31%
句内注意力模型	41.39%	41.98%

表 3 句子排序的实验结果

如表 3 所示，除了句子匹配方法在新华社语料上的排序结果，本文提出的句子排序模型(包括基于句子匹配方法和基于注意力机制的方法)在句子排序任务中表现优于基线方法，特别是基于词对齐注意力机制和基于句内注意力机制的模型。这些句子排序效果的提升归功于本文提出的句子排序模型有较强的捕捉句子语义逻辑关系的能力。句子匹配模型可以从句子层面描述句子之间的相关性，而基于注意力机制的句子排序模型则通过捕获句子中词语之前的相关关系。这两种相关性信息是句子语义逻辑关系的两个关键因素，因此引入这两种特征能够提升句子排序效果。

基于句子匹配方法的句子排序方法在百度百科语料上表现好于基线，但在新华社语料上比基线差。句子匹配方法捕捉句子语义逻辑关系的核心操作是按位乘、按位减、拼接，通过计算特征向量的差异性来捕捉句子语义逻辑关系。新华社语料多为事实陈述型语料，句子匹配方法不能很好捕捉事实陈述型语料句子间的差异性。因此句子匹配方法在新华社语料上表现较差。

基于静态注意力机制的句子排序方法，在新华社语料上对比基线方法有了显著提高，达到 41.49%，而在百度百科语料上与基线相当。百度百科含有大量噪音，如文言文、诗词歌赋等，静态注意力机制不能很好捕捉文言文、诗词歌赋的语义逻辑关系，仅达到基线水平。

与 Chen 的基线方法相比，词对齐和句内注意力模型在新华社语料和百度百科语料上句子排序结果取得了 4%-6%PMR 提升。词对齐注意力模型可以识别两个句子之间的词与词对齐关系，而这种词与词的对齐关系恰好是句子语义逻辑关系的一种表现形式，因此，词对齐注意力模型捕获句子语义逻辑关系的能力更强。句内注意力模型则是通过句内注意力增强句子特征的语义表示能力，提高语义逻辑判断的准确性，从而实现了更好的句子排序。此外，词对齐和句内注意力模型的上述特点也使得其不同数据集句子排序结果的稳定性(所谓模型稳定性就是在不同数据集上实验结果都获得了提高，并且提高的幅度相当)。

4 结论

本文的主要贡献是将句子匹配方法、静态注意力机制、词对齐注意力机制及句内注意力机制引入句子排序任务中，捕捉句子间的语义逻辑关系，提升句子排序的准确性，增强段落的可读性。实验证明注意力机制能够更有效地提升句子排序效果。

本文主要的目的是探索句子匹配方法和注意力机制在句子排序中发挥的作用，为句子排序提供语义信息更强的特征。而将多种特征融合也是句子排序研究中的一个重要方面，为此在未来的工作中，我们将进一步研究将句子匹配方法和注意力机制等多种特征融合在一起的方法，进而更好地实现句子排序。

参考文献

- [1] 韩永峰, 许旭阳, 李弼程, 等. 基于事件抽取的网络新闻多文档自动摘要[J]. 中文信息学报, 2012, 26(1):58-66.
- [2] 刘平安. 基于 HLDA 模型的中文多文档摘要技术研究[D]. 北京邮电大学, 2013.
- [3] Wang L, Raghavan H, Castelli V, et al. A Sentence Compression Based Framework to Query-Focused Multi-Document Summarization[J]. 2013:1384-1394.
- [4] 刘秉权, 徐振, 刘峰, 等. 面向问答社区的答案摘要方法研究综述[J]. 中文信息学报, 2016, 30(1):1-8.
- [5] Ferreira R, Cabral L D S, Freitas F, et al. A multi-document summarization system based on statistics and linguistic treatment[J]. Expert Systems with Applications, 2014, 41(13):5780-5787.
- [6] 徐永东, 王亚东, 刘杨, 等. 多文档文摘中基于时间信息的句子排序策略研究[J]. 中文信息学报, 2009, 23(4):27-33.
- [7] 姚超, 李生, 张姝, 等. 基于内聚度的多文档文摘句子排序[C]// 中国中文信息学会二十五周年学术会议. 2006
- [8] Xinchu Chen, Xipeng Qiu, and Xuanjing Huang. 2016. Neural sentence ordering. arXiv preprint arXiv:1607.06952.
- [9] Simard P Y, Steinkraus D, Platt J C. Best practices for convolutional neural networks applied to visual document analysis[C]// International Conference on Document Analysis and Recognition, 2003. Proceedings. IEEE, 2003:958.
- [10] Surhone L M, Tennoe M T, Henssonow S F. Long Short Term Memory[J]. Betascript Publishing, 2010.
- [11] Logeswaran L, Lee H, Radev D. Sentence Ordering using Recurrent Neural Networks[J]. arXiv preprint arXiv:1611.02654, 2016.
- [12] Gong J, Chen X, Qiu X, et al. End-to-End Neural Sentence Ordering Using Pointer Network[J]. arXiv preprint arXiv:1611.04953, 2016.
- [13] 康世泽, 马宏, 黄瑞阳. 一种基于神经网络模型句子排序方法[J]. 中文信息学报, 2016, 30(5):195-202.
- [14] Mou L, Rui M, Li G, et al. Recognizing Entailment and Contradiction by Tree-based Convolution[J]. Computer Science, 2015.
- [15] Tim Rocktäschel, Grefenstette E, Hermann K M, et al. Reasoning about Entailment with Neural Attention[J]. 2015.
- [16] Liu Y, Sun C, Lin L, et al. Learning natural language inference using bidirectional LSTM model and inner-attention[J]. arXiv preprint arXiv:1605.09090, 2016.
- [17] Ioffe S, Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift[J]. Computer Science, 2015.