

# 利用上下文相似度增强词对齐效果的自然语言推理方法

杜倩龙<sup>1,2</sup>, 宗成庆<sup>1,2</sup>, 苏克毅<sup>3</sup>

(1. 中国科学院自动化研究所模式识别国家重点实验室, 北京 100190; 2. 中国科学院大学, 北京 100049;  
3. 台湾“中央研究院”资讯科学研究所, 台湾 台北 11529)

**摘要:** 当前的自然语言推理方法在进行词对齐时往往只关注如何利用上下文信息改善词汇的表示, 而忽略了上下文相似度对词汇对齐的帮助。推理方法在融合局部判定信息进行结果预测时, 通常赋予局部判定信息相同的权重, 而实际上不同词汇对应的局部判定信息对最终结果具有不同的影响。因此, 提出一种通过增强词对齐性能改善自然语言推理性能的方法, 该方法在词对齐时同时考虑词汇的词义相似度和上下文相似度, 并且利用选择门机制对不同的局部判定信息进行加权。实验表明, 该方法能够获得与当前最优的模型相当的准确率, 而且该方法更加符合人脑推理判定行为过程, 且具有很好的可解释性。

**关键词:** 文本蕴含; 自然语言推理; 词对齐

**中图分类号:** TP391.1

**文献标识码:** A

**doi:** 10.11959/j.issn.2096-6652.202003

## Enhancing alignment with context similarity for natural language inference

DU Qianlong<sup>1,2</sup>, ZONG Chengqing<sup>1,2</sup>, SU Keh-Yih<sup>3</sup>

1. National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China  
2. University of Chinese Academy of Sciences, Beijing 100049, China  
3. Institute of Information Science, Academia Sinica, Taipei 11529, China

**Abstract:** Previous approaches generally use context information to improve the word representation but ignore the importance of context similarity in aligning tokens. Furthermore, most of them uniformly weight various local decisions during aggregation for the global judgment. However, local decisions related to various tokens can influence the final decision differently. In order to process these problems, an enhanced alignment mechanism was proposed, which jointly considers both token content and context similarity in computing the alignment weight for each token pair. Besides, a selection gate mechanism to weight local decisions differently was also proposed. Experimental results show that our performance is comparable to state-of-the-art approaches but better mimics human behavior, making it more interpretable.

**Key words:** textual entailment, natural language inference, word alignment

## 1 引言

自然语言推理 (natural language inference) 任务通常用文本蕴含识别<sup>[1]</sup>测试系统的推理性能。在该任务中, 通常给定一个前提文本  $P$ , 然后判定另一个假设文本  $H$  是否可以从中推理出来, 即判定这

2 个文本片段之间的关系是蕴含 (entailment)、中立 (neutral) 还是矛盾 (contradiction)。作为自然语言理解中的一个基础性任务<sup>[2]</sup>, 自然语言推理在各种任务中得到了广泛应用, 如自动问答<sup>[3]</sup>、信息抽取<sup>[4]</sup>、机器翻译<sup>[5]</sup>和自动摘要<sup>[6]</sup>等。在过去的几十年间, 人们针对语言理解问题提出了若干方法, 包

收稿日期: 2020-01-21; 修回日期: 2020-02-27

通信作者: 宗成庆, cqzong@nlpr.ia.ac.cn

基金项目: 国家重点研发计划基金资助项目 (No.2017YFB1002103)

**Foundation Item:** The National Key Research and Development Program of China (No.2017YFB1002103)

括早期的统计框架和规则方法<sup>[7-8]</sup>。

随着神经网络方法的快速发展,在 Bowman 等人<sup>[9]</sup>于 2015 年公开了一个大规模的自然语言推理数据集 (stanford natural language inference, SNLI) 之后,出现了大量基于神经网络的方法<sup>[10-12]</sup>。在这些方法中, Parikh 等人<sup>[11]</sup>在 2016 年提出的基于注意力机制的方法成为该领域的代表。该方法将自然语言推理过程分解成对齐、比较和融合 3 个子任务进行处理,并且在很多测试集上取得了很好的性能。当前大多数性能较好的模型均受到了他们工作的启发。

然而,对这些基于注意力机制的模型<sup>[11-13]</sup>进行分析后发现,这些模型在文本片段  $P$  和  $H$  之间进行对齐比较时并不能得到精确的词对齐结果。这是因为这些模型通常只关注采用改进词汇表示的方法改善对齐效果,而没有考虑词汇之间上下文相似度 (context similarity) 的影响。

以下面的文本片段对为例 (该文本片段对之间的关系为中立)。

$P$ : An older man sits with his orange juice at a small table in a coffee shop.

$H$ : An elderly man sits in a small shop.

针对该文本片段对  $P$  和  $H$ , 基于注意力机制的方法会利用软注意力机制计算词向量之间的相似度,并将其作为词汇之间的对齐权重。这种方法在进行词汇对齐时仅仅考虑了每个词汇在该文本片段中的语义信息。因此,  $H$  中的词汇 small 将与  $P$  中的词 small 对齐 (相比于其他词,这 2 个词之间的词向量更相似)。在这种对齐情况下,  $H$  中所有的词汇都能在  $P$  中对应到一个近似的词,导致该文本对之间的关系被错误地预测为蕴含关系。然而,由于 small 在这 2 个文本片段中分别修饰不同的词汇 (small 在  $P$  中修饰词汇 table, 在  $H$  中修饰词汇 shop), 这 2 个 small 不应该对齐。因此,仅仅依靠词汇间的词义信息进行对齐并不能达到预期的对齐结果。

通过对随机挑选的 50 个样例进行分析之后发现,在对 1 个文本对之间的关系进行人工判定时,词汇对齐往往会考虑 2 个词汇之间的上下文相似度。例如,对于上述例子,考虑词汇之间上下文的相似度时,会发现  $H$  中的 small 应该与  $P$  中的 coffee 对齐,因为这 2 个词汇具有相同的上下文词汇信息 (例如,相同的上下文词汇 in、a 和 shop)。在该对

齐下,基于  $H$  中的词汇 small 的局部判定结果为中立 (该结果与整个句对间的预测结果一致)。因此,上下文间的相似度能够很好地帮助现有模型改善词汇对齐的效果。

综上所述,文本片段  $P$  和  $H$  之间的词对齐不仅依赖于词汇间词义的相似度,同时还与词汇的上下文间的相似度有密切关系。基于该思想,本文提出了一种新的增强型的词汇对齐机制,该机制会同时考虑词汇间的词义相似度和上下文相似度。

另外,在预测文本对之间的关系时,当前的方法<sup>[11-13]</sup>直接将所有基于词汇级别的判定信息进行融合。实际上,不同词汇对应的判定信息对最终结果的影响是不同的。因此,不同词汇对应的判定信息在进行融合时应该被赋予不同的权重,该权重依赖于对应的词汇以及该词汇在另一个文本片段中对应的信息。

例如,对于上述样例  $H$  中的词汇 An, 由于其完全匹配于  $P$  中的词汇 An, 因此它对应的词汇级别的判定信息倾向于蕴含关系。然而,该结果与文本  $P$  和  $H$  之间的关系标签并不一致,因此该局部判定信息在进行融合时应该被弱化。对于  $H$  中的词汇 small, 由于其与  $P$  中的对齐词汇 coffee 为中立关系, 因此它对应的局部判定信息应该被增强。针对该问题, 笔者为神经网络模型提出了一种选择门 (selection-gate) 机制, 该机制利用词汇本身和它在另一个文本片段中的相关信息来显式地对该词汇对应的判定信息进行加权。通过该机制,可以在弱化不重要的词汇对应的局部判定信息的同时,对来自重要词汇的局部判定信息进行增强。

众所周知,深度神经网络结构能够有效地抽取表象特征<sup>[14]</sup>,甚至可以抽取到一些对于人来说也很难注意到的特征。然而,为了处理现有模型难以解决的推理问题,需要进一步融合人工判定时使用的深层次的信息,这些信息更有意义,也具有更强的可解释性。基于该思想,通过对当前模型出现推理错误的样例进行分析,本文提出了利用上下文相似度增强词对齐效果的自然语言推理方法。相比于以前的方法,该方法能够更好地拟合人的行为,因此也具有很好的可解释性。

本文的贡献主要可以归纳为以下 2 点。

- 提出了一种增强型的词汇对齐方法,该方法同时考虑词汇的词义相似度和上下文相似度,具有更强的可解释性。

• 提出了一种选择门结构, 弱化来自不重要词汇的局部判定信息对最终结果的影响。

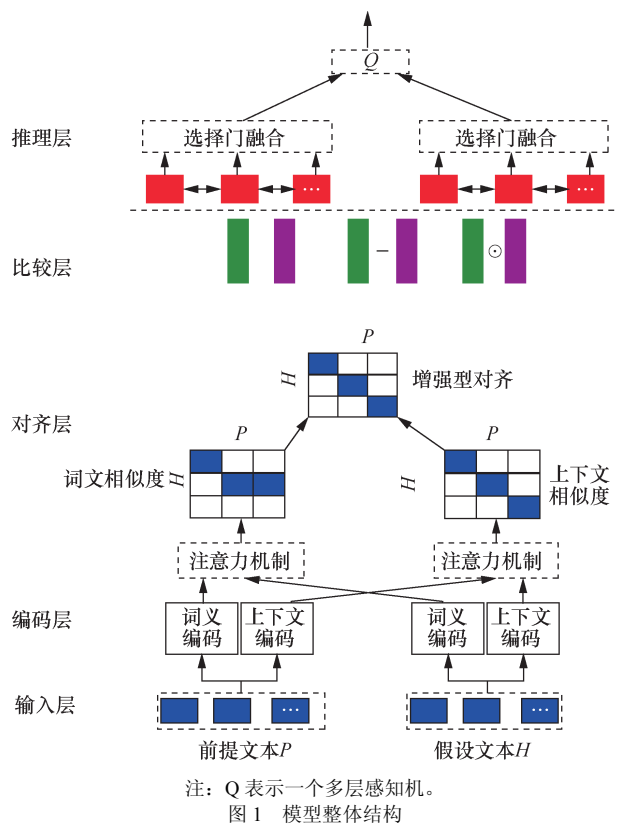
## 2 相关工作

早期的自然语言推理方法主要注重采用传统的统计学方法或者规则方法进行推理<sup>[7-8]</sup>。利用一些人人工定义的特征(如词匹配、否定词检测等)在一些小规模的数据集上进行实验。在 Bowman 等人<sup>[9]</sup>于 2015 年公开了 SNLI 后, 涌现出了大量的基于神经网络结构的方法<sup>[10-13]</sup>。在这些方法中, Parikh 等人<sup>[11]</sup>提出了一个基于注意力机制的模型。不同于以前的方法直接对 2 个文本的向量表示进行比较, 该模型更关注文本中重要的局部信息的比较。受到 Parikh 等人<sup>[11]</sup>的模型的启发, Chen 等人<sup>[12]</sup>在 2017 年提出了一个增强型序列推理模型(enhanced sequential inference model, ESIM), 该模型首先利用双向长短期记忆(bidirectional long short-term memory, BiLSTM)网络对文本中的每个词进行编码, 然后利用类似于 Parikh 的模型中的方法对编码后的表示进行比较。Peters 等人<sup>[15]</sup>在 2018 年利用语言模型在一个大规模的数据集上对所有词汇的词向量进行预训练。将预训练的词向量作为输入来改善词汇的表示, 从而进一步提升 ESIM 的性能。此外, Wang 等人<sup>[16]</sup>在 2017 年提出了一个 BiMPM 模型, 该模型采用一种多角度余弦匹配的方法对每一个文本对进行比较。在 Vaswani 等人<sup>[17]</sup>于 2017 年基于注意力机制提出 Transformer 结构后, Devlin<sup>[18]</sup>等人提出了使用 Transformer 进行自然语言推理的方法。他们首先使用大规模的数据对词向量和模型参数进行预训练, 然后利用预训练后的参数在特定任务上进行参数的微调。通过对当前这些基于注意力机制的模型进行分析后发现, 这些模型都注重采用更复杂的网络结构、更大规模的数据、更多样的比较方法来改善推理性能。

此外, Du 等人<sup>[19]</sup>在 2018 年提出了一种利用词汇依存关系改善对齐权重的方法。在该方法中, 他们利用依存词之间的相似度改进词汇依存三元组之间的对齐和比较效果。不同于以上这些主要利用词向量信息进行词对齐的模型, 本文首次提出利用上下文相似度改进词汇的对齐效果, 同时提出利用选择门机制对词汇级别的判定信息进行处理。

## 3 模型方法

图 1 展示了模型的整体结构。它主要包含 4 层网络结构。一是编码层, 该层利用每个词汇对应的原始输入向量生成对应的词义向量和上下文向量。二是对齐层, 对于  $P$  和  $H$  之间的每个词对, 利用它们的词义向量计算词义相似度, 同时利用它们的上下文向量计算上下文相似度。通过将词汇的词义相似度和上下文相似度进行融合, 便可以为每个词对计算一个增强型的对齐权重。然后, 利用该权重生成每个词汇对应的比较向量(比较向量表示另一个文本片段中与该词汇相关的信息)。三是比较层, 该层将每个词的词义表示向量与其对应的比较向量进行比较。四是推理层, 该层首先利用上一步的比较结果生成对应的词级别的判定信息, 然后利用选择门结构对所有词汇级别的判定信息进行处理, 最后通过将所有处理后的信息进行融合来预测最终的关系标签。



如图 1 所示, 该模型是一个对称的结构, 即同时从 2 个方向对文本对进行比较(将  $P$  中的词汇与文本片段  $H$  进行比较, 以及将  $H$  中的词汇与文本片段  $P$  进行比较)。

### 3.1 编码层

给定 2 个词汇序列  $P = (p_1, p_2, \dots, p_m)$  和  $H = (h_1, h_2, \dots, h_n)$  (其中  $p_i$  和  $h_j$  分别对应  $P$  和  $H$  中第  $i$  个和第  $j$  个词), 定义  $\hat{P} = (\hat{p}_1, \hat{p}_2, \dots, \hat{p}_m)$  和  $\hat{H} = (\hat{h}_1, \hat{h}_2, \dots, \hat{h}_n)$  分别表示  $P$  和  $H$  对应的原始输入向量序列。 $\hat{p}_i$  和  $\hat{h}_j$  分别是  $p_i$  和  $h_j$  的输入向量。 $m$  和  $n$  分别表示  $P$  和  $H$  中的词汇数目。

词汇对应的原始输入向量  $\hat{p}_i$  和  $\hat{h}_j$  包含以下几个部分。

- 词向量：笔者使用 300 维的 Glove<sup>[20]</sup>词向量对所有词汇进行向量初始化。这些词向量是利用从网络上爬取到的 840 B 的数据训练得到的。

- 词汇特征：Chen 等人<sup>[21]</sup>发现词汇的词性等语言学特征可以改善自动问答系统的性能，因此本文在进行词向量初始化时也添加了一些人工选择的特征，这些特征对应词汇的某些特性，包含词性 (POS)、命名实体 (NE) 和归一化的词频 (TF)。将每一个特征都表示为一个 one-hot 向量。

- 词汇匹配特征：本文使用 3 种简单的二值化特征，这些二值化特征表示该词汇是否可以在另一个文本片段中找到对应的原始 (original) 形式、小写 (lowercase) 形式和词干 (lemma form) 形式。这 3 类特征都表示为一位的 0 或者 1 值。

利用上述特征拼接得到原始的输入向量  $\hat{p}_i$  和  $\hat{h}_j$  后，将在下面 2 个子层编码词汇所对应的词义向量和上下文向量。

#### 3.1.1 编码词义向量

对于 2 个原始的输入向量序列  $\hat{P}$  和  $\hat{H}$ ，将其输入 BiLSTM 网络中并计算其对应的词义向量序列。

$$\{\tilde{p}_1, \dots, \tilde{p}_m = \text{BiLSTM}(\{\hat{p}_1, \dots, \hat{p}_m\})\} \quad (1)$$

$$\{\tilde{h}_1, \dots, \tilde{h}_n = \text{BiLSTM}(\{\hat{h}_1, \dots, \hat{h}_n\})\} \quad (2)$$

其中， $\tilde{p}_i \in R^{2d}$  和  $\tilde{h}_j \in R^{2d}$  分别是  $p_i$  和  $h_j$  在 BiLSTM 中的隐层表示， $d$  表示 BiLSTM 网络结构中隐层的维度。 $\tilde{p}_i$  和  $\tilde{h}_j$  分别表示词汇  $p_i$  和  $h_j$  的词义向量，它们将被用来在对齐层求取词对间的词义相似度，以及在比较层与另一个文本片段中的相关信息进行比较。

#### 3.1.2 编码上下文向量

使用自注意力机制计算每个词对应的上下文向量。首先，同第 3.1.1 节类似，将原始输入向量

序列  $\hat{P}$  和  $\hat{H}$  输入另一个 BiLSTM 中，从而为每个词生成一个新的表示  $q$ ，具体如下：

$$\{q_1^p, \dots, q_m^p = \text{BiLSTM}(\{\hat{p}_1, \dots, \hat{p}_m\})\} \quad (3)$$

$$\{q_1^h, \dots, q_n^h = \text{BiLSTM}(\{\hat{h}_1, \dots, \hat{h}_n\})\} \quad (4)$$

其中， $q_i^p \in R^{2d}$ 、 $q_j^h \in R^{2d}$  分别是  $p_i$ 、 $h_j$  对应的隐层表示。

对于词汇  $p_i$ ，为了抽取其上下文向量，首先利用自注意力机制计算  $p_i$  与它的所有上下文词汇之间的关联程度。例如，对于  $P$  中的任意一个词汇  $p_i$ ， $p_i$  与  $p_l$  间的关联权重可以通过两者隐层表示的内积来计算，具体如下：

$$r_{i,l} = (q_i^p)^T q_l^p \quad (5)$$

其中， $r_{i,l}$  表示  $p_i$  和它的上下文词汇  $p_l$  间的关联权重。当计算得到  $p_i$  和它的所有上下文词汇间的关联权重后，通过对局部上下文窗口内的所有词汇对应的隐层表示进行加权求和，得到  $p_i$  的上下文向量，具体计算过程见式 (6) (式 (7) 对应  $h_j$  的上下文向量)。

$$a_i = \sum_{l=i-L}^{i+L} \frac{\exp(r_{i,l})}{\sum_{k=i-L}^{i+L} \exp(r_{i,k})} q_l^p \quad (6)$$

$$b_j = \sum_{l=j-L}^{j+L} \frac{\exp(r_{j,l})}{\sum_{k=j-L}^{j+L} \exp(r_{j,k})} q_l^h \quad (7)$$

其中， $(2L+1)$  表示局部上下文窗口的大小。向量  $a_i$  和  $b_j$  分别表示词汇  $p_i$  和  $h_j$  的上下文向量，它们包含这 2 个词汇对应的上下文信息，这些信息将用于计算下一层词对之间的上下文相似度。

### 3.2 对齐层

该层使用每个词汇对应的词义向量和上下文向量计算词对之间的对齐权重，然后利用对齐权重生成每个词对应的比较向量。

首先，利用向量的内积操作分别计算 2 个词的词义相似度和上下文相似度。

$$e_{i,j}^t = \tilde{p}_i^T \tilde{h}_j \quad (8)$$

$$e_{i,j}^c = a_i^T b_j \quad (9)$$

其中， $e_{i,j}^t$  表示词汇  $p_i$  和  $h_j$  之间的词义相似度，而  $e_{i,j}^c$  表示它们之间的上下文相似度。在获取  $p_i$  和  $h_j$  之间的这 2 种相似度后，通过将它们进行融合，

可以计算得到一个增强型的对齐权重, 如式 (10) 所示。

$$e_{i,j} = \lambda_1 \times e_{i,j}^t + \lambda_2 \times e_{i,j}^c + \beta \quad (10)$$

其中,  $e_{i,j}$  表示  $p_i$  和  $h_j$  之间的增强型对齐权重。 $\lambda_1 \in R$ 、 $\lambda_2 \in R$  和  $\beta \in R$  是要学习的参数。在这里,  $\beta$  用来对对齐权重进行调节, 可以有效地改进模型效果。

对于词汇  $p_i$ , 可以计算得到它与  $H$  中每个词之间的增强型对齐权重。利用该对齐权重对  $H$  中所有词的词义向量进行加权求和, 从而生成  $p_i$  相对于  $H$  的比较向量  $u_i$ , 具体如式 (11) 所示。相应地, 也可以计算得到  $h_j$  相对于  $P$  的比较向量  $v_j$ , 见式 (12)。

$$u_i = \sum_{j=1}^n \frac{\exp(e_{i,j})}{\sum_{k=1}^n \exp(e_{i,k})} \tilde{h}_j \quad (11)$$

$$v_j = \sum_{i=1}^m \frac{\exp(e_{i,j})}{\sum_{k=1}^m \exp(e_{j,k})} \tilde{p}_i \quad (12)$$

其中,  $\tilde{p}_i$  和  $\tilde{h}_j$  表示词汇对应的词义向量 (具体见式 (1) 和式 (2))。比较向量  $u_i$  表示  $H$  中与词汇  $p_i$  相关的语义信息; 同理,  $v_j$  表示  $P$  中与词汇  $h_j$  相关的语义信息。

### 3.3 比较层

对于词汇  $p_i$ , 将它的词义向量  $\tilde{p}_i$  与比较向量  $u_i$  进行比较, 具体见式 (13)。相应地, 也对  $h_j$  的词义向量和比较向量进行比较, 见式 (14)。

$$p_i^c = G([\tilde{p}_i; u_i; \tilde{p}_i - u_i; \tilde{p}_i \odot u_i]) \quad (13)$$

$$h_j^c = G([\tilde{h}_j; v_j; \tilde{h}_j - v_j; \tilde{h}_j \odot v_j]) \quad (14)$$

其中,  $G$  表示一个单层的前向网络结构, 其激活函数为 ReLu。括号中的 “;” 表示拼接操作, “-” 和 “ $\odot$ ” 分别表示向量的减法和点乘操作。 $p_i^c$  表示  $p_i$  和  $H$  之间的比较结果; 同理,  $h_j^c$  表示  $h_j$  和  $P$  之间的比较结果。

### 3.4 推理层

利用比较层可以得到每个词对应的比较结果, 因此在这一层将利用比较结果生成对应的词汇级别的判定信息, 然后通过将所有词汇级别的判定信息进行融合来预测最终的关系标签。

#### 3.4.1 推理词汇级别的判定信息

为了推理得到每个词对应的词汇级别的判定信息, 将  $P$  中每个词对应的比较结果输入一个 BiLSTM 网络结构中进行处理, 具体见式 (15)。相应地, 也将  $H$  中所有词对应的比较结果输入 BiLSTM 结构中进行处理, 见式 (16)。

$$\{p_1^l, \dots, p_m^l = \text{BiLSTM}(\{p_1^c, \dots, p_m^c\})\} \quad (15)$$

$$\{h_1^l, \dots, h_n^l = \text{BiLSTM}(\{h_1^c, \dots, h_n^c\})\} \quad (16)$$

其中,  $p_i^l$  表示  $p_i$  对应的词汇级别的判定信息,  $h_j^l$  表示  $h_j$  对应的词汇级别的判定信息。得到这些词汇级别的判定信息后, 只需要将其融合便可以预测最终的结果。

#### 3.4.2 基于选择门的融合机制

为了增强来自重要词汇的词汇级别判定信息对最终预测的影响, 同时弱化来自不重要词汇的词汇级别判定信息对最终结果的影响, 本文提出了一种选择门的机制来对不同的词汇级别判定信息进行加权。

对于词汇级别的判定信息, 该判定信息是否重要是由其对应的词以及该词在另一个文本片段中的相关信息决定的。由于比较层的输入信息很好地描述了这种重要性, 因此采用与其相同的输入信息生成一个选择门向量。该向量描述对应词汇级别判定信息的重要程度, 具体见式 (17)。

$$\begin{aligned} g_i &= R([\tilde{p}_i; u_i; \tilde{p}_i - u_i; \tilde{p}_i \odot u_i]) \\ g_j &= R([\tilde{h}_j; v_j; \tilde{h}_j - v_j; \tilde{h}_j \odot v_j]) \end{aligned} \quad (17)$$

其中,  $R$  是一个单层的前向网络, 其激活函数为 Sigmoid。在生成对应的选择门向量后, 利用该选择门向量可以对所有的词汇级别判定信息进行加权。最终, 利用均值和最大值池化操作对所有加权后的结果进行融合。

$$\begin{aligned} O_{P, \text{avg}} &= \sum_{i=1}^m g_i \odot p_i^l / m, O_{P, \text{max}} = \max_{i=1}^m g_i \odot p_i^l \\ O_P &= [O_{P, \text{avg}}; O_{P, \text{max}}] \\ O_{H, \text{avg}} &= \sum_{j=1}^n g_j \odot h_j^l / n, O_{H, \text{max}} = \max_{j=1}^n g_j \odot h_j^l \\ O_H &= [O_{H, \text{avg}}; O_{H, \text{max}}] \end{aligned} \quad (18)$$

基于上述操作得到了 2 个全局推理向量  $O_P$  和

$O_H$ , 这2个向量代表不同的比较方向( $O_P$ 表示将 $P$ 中的词汇与 $H$ 进行比较,  $O_H$ 表示将 $H$ 中的词汇与 $P$ 进行比较)。将这2个不同方向的推理向量进行拼接操作, 然后将结果输入一个多层感知机 $Q$ 中, 以预测最终的关系标签。多层感知机 $Q$ 是一个2层的前向网络, 每一层分别对应 ReLu 和 Softmax 这2种激活函数。

## 4 实验

### 4.1 实验数据

为了对本文提出的模型进行评估, 笔者采用了3个自然语言推理数据集进行测试, 分别为: SNLI<sup>[9]</sup>、MultiNLI<sup>[22]</sup>和 New-test-set<sup>[23]</sup>。下面对这3个数据集进行简单介绍。

- SNLI: 该数据集是第一个大规模的英文自然语言推理数据集, 其包含57万个文本片段对。每个文本片段对之间的关系被标注为蕴含、矛盾或者中立。在原始的语料中, 由于人工标注者在结果上无法达成一致, 有部分数据被标注为“—”。依照 Bowman 等人<sup>[9]</sup>在2015年的处理方式, 在本实验中也删除这部分数据。

- MultiNLI: 作为针对自然语言推理任务的一个大规模数据集, MultiNLI 数据集包含43.3万个文本对。为了更好地评估自然语言推理系统在复杂语言结构上的性能, 该数据集包含了多个类型的英文数据, 包括非小说的书面语类(written non-fiction genres)、口语类(spoken genres)、非正规书写类(less formal written genres)以及关于“911”的新闻报道等。另外, 依据是否与训练集中的数据来自相同的类别, 该数据的开发集和测试集包含了领域内数据(matched)和领域外数据(mismatched)2种。这样能够更好地测试模型的推理性能。

- New-test-set: 上述2个数据集在构造时会给标注者提供一个前提文本 $P$ , 然后要求他们构造3个假设文本 $H$ 。然而, Gururangan 等人<sup>[29]</sup>在2018年发现这种方式会产生一些问题: 一些特别的语言学现象(如否定词)会与某类类别标签高度相关。在这种情况下, 仅利用一个简单的文本分类模型单独对假设文本 $H$ 进行分类就可以在 SNLI 上达到67%的准确率, 同时在 MultiNLI 上达到53%的准确率。因此, Glockner 等人<sup>[23]</sup>在2018年公开了一个新的测试集 New-test-set, 可以更好地评估自然语言推理模型的性能。该数据集中的前提文本 $P$ 都来自 SNLI

的训练语料。针对这些前提文本 $P$ , 通过将 $P$ 中的一个词进行替换, 构建了一批假设文本 $H$ 。这些文本分别对应蕴含、矛盾和中立这3类关系标签。该数据集一共包含8193个文本对。

### 4.2 实验设置

在模型的实现过程中, 采用适应性矩估计算法(Adam)进行优化。在训练的过程中, 使用带 dropout<sup>[30]</sup>的多类交叉熵损失函数进行训练, 采用概率为0.2的 dropout 策略, 同时将学习率设置为0.0003, 在 SNLI 以及 MultiNLI 的领域内数据和领域外数据上的批大小分别为32、25、20。同时, 在上述3个数据集上, 网络结构的隐层大小分别为400、373和450。在进行词汇输入向量的初始化时, 利用 Spacy 工具生成词汇的词性标签和命名实体标签, 并将它们分别初始化为一个 one-hot 向量。在选取局部上下文时, 将上下文窗口的尺寸 $L$ 设置为2。以上所有超参数均为基于相应的开发集得到的。

### 4.3 实验结果分析

表1展示了一些在 SNLI 数据集上的代表性相关模型。模型(1)~模型(10)主要为基于句子间注意力机制的方法。在这些方法中, Chen 等人<sup>[12]</sup>在2017年提出的 ESIM(模型(6))利用 BiLSTM 结构对文本中的每个词进行编码, 生成词的表示, 然后利用这些词的表示进行比较, 从而预测最终的结果。该模型也成为自然语言推理领域的一个代表制作。基于 ESIM, 本文提出增加额外的语言学信息来改善词的表示, 同时利用增强型词对齐机制来改进对齐效果, 并利用选择门机制对词汇级别的判定信息进行过滤。

从表1可以看出, 本文提出的方法(proposed approach)在 SNLI 数据集上的准确率达到88.8%, 优于表1中的所有其他模型。通过与基线模型 ESIM 比较, 可以发现模型的性能明显优于 ESIM。另外, 模型(10)基于 ESIM 提出了利用上下文信息改进词的表示, 从而提升系统的性能。相比于模型(10)所提方法, 本文提出的方法仍然取得了更好的效果。

表2展示了在 MultiNLI 数据集上性能较好的代表性模型。从表2中可以看出, 本文提出的方法在 matched 数据集上优于所有的相关模型, 同时在 mismatched 数据集上优于模型(1)~模型(8)。特别地, 相比于模型(7)仅采用上下文信息改善

表 1 模型在数据集 SNLI 上的性能

| 模型                                                     | 训练集准确率 | 测试集准确率       |
|--------------------------------------------------------|--------|--------------|
| (1) mLSTM word-by-word attention model <sup>[24]</sup> | 92.0%  | 86.1%        |
| (2) decomposable attention model <sup>[11]</sup>       | 90.5%  | 86.8%        |
| (3) RHD with Individual Comparison <sup>[19]</sup>     | 90.3%  | 87.4%        |
| (4) BiMPM <sup>[16]</sup>                              | 90.9%  | 87.5%        |
| (5) re-read LSTM <sup>[25]</sup>                       | 90.7%  | 87.5%        |
| (6) ESIM <sup>[12]</sup>                               | 92.6%  | 88.0%        |
| (7) DIIN <sup>[26]</sup>                               | 91.2%  | 88.0%        |
| (8) DR-BiLSTM <sup>[27]</sup>                          | 94.1%  | 88.6%        |
| (9) KIM <sup>[28]</sup>                                | 94.1%  | 88.6%        |
| (10) ESIM + ELMo <sup>[15]</sup>                       | —      | 88.7%        |
| (11) Human Performance <sup>[26]</sup>                 | —      | <b>87.7%</b> |
| (12) proposed approach                                 | 95.4%  | <b>88.8%</b> |

词表示的方法, 本文提出的方法在 MultiNLI 数据集上取得了更好的效果。

表 2 模型在数据集 MultiNLI 上的性能

| 模型                                                 | 测试集准确率       |              |
|----------------------------------------------------|--------------|--------------|
|                                                    | Matched      | Mismatched   |
| (1) BiLSTM <sup>[22]</sup>                         | 67.0%        | 67.6%        |
| (2) Inner Att <sup>[31]</sup>                      | 72.1%        | 72.1%        |
| (3) ESIM <sup>[22]</sup>                           | 72.3%        | 72.1%        |
| (4) Gated-Att BiLSTM <sup>[32]</sup>               | 73.2%        | 73.6%        |
| (5) Shortcut-Stacked encoder <sup>[33]</sup>       | 74.6%        | 73.6%        |
| (6) RHD with Individual Comparison <sup>[19]</sup> | 75.1%        | 74.7%        |
| (7) BiLSTM + ELMo + Attn <sup>[18]</sup>           | 76.4%        | 76.1%        |
| (8) KIM <sup>[28]</sup>                            | 77.2%        | 76.4%        |
| (9) CAFÉ <sup>[13]</sup>                           | 78.7%        | <b>77.9%</b> |
| (10) DIIN <sup>[26]</sup>                          | 78.8%        | 77.8%        |
| (11) Human Performance <sup>[26]</sup>             | 88.5%        | <b>89.2%</b> |
| (12) proposed approach                             | <b>79.0%</b> | <b>77.5%</b> |

表 3 展示了一些当前性能较好的自然语言推理方法在 New-test-set 数据集上的效果。由于对数据构建方法进行了改进, 因此该数据集能更好地评估自然语言推理系统的性能。与表 1 相比, 可以看出在该数据集上, 模型 (1)、模型 (2) 和模型 (5) 的性能都有显著的下降。然而, 本文提出的方法在该数据集上仍然取得了很好的效果。尽管本文提出的方法是基于 ESIM 的, 但相比于 ESIM, 本文提出的方法在该数据集上的性能有非常显著的提升,

这也说明其具有非常强的鲁棒性<sup>[29]</sup>。此外, 表 3 中的 BERT 模型<sup>[18]</sup>也取得了很好的效果。然而, BERT 模型需要利用大规模的额外数据对词向量和模型参数进行预训练。相比于该模型, 尽管本文提出的方法使用了较小规模的数据, 但是在该数据集上仍然取得了更高的准确率。

表 3 模型在 New-test-set 数据集上的性能

| 模型                                               | 训练数据集           | New-test-set |
|--------------------------------------------------|-----------------|--------------|
| (1) decomposable attention model <sup>[11]</sup> | SNLI            | 51.9%        |
|                                                  | MultiNLI + SNLI | 65.8%        |
| (2) ESIM <sup>[12]</sup>                         | SNLI            | 65.6%        |
|                                                  | MultiNLI + SNLI | 74.9%        |
| (3) Residual-Stacked-Encoder <sup>[33]</sup>     | SNLI            | 62.2%        |
|                                                  | MultiNLI + SNLI | 68.2%        |
| (4) WordNet Baseline <sup>[23]</sup>             | —               | 85.8%        |
| (5) KIM <sup>[28]</sup>                          | SNLI            | 83.5%        |
| (6) BERT <sup>[18]</sup>                         | SNLI            | 93.0%        |
| (7) proposed approach                            | SNLI            | <b>93.4%</b> |

#### 4.4 消融分析

表 4 展示了模型中不同策略对最终结果的影响, 表 4 中的模型 (1) 是最终的模型。在模型 (2) 中, 计算对齐权重时不考虑上下文相似度, 即仅利用词向量的相似度计算词汇之间的对齐权重。结果表明, 当移除上下文相似度时, 模型的性能会有显著的下降, 这也说明增强型词对齐机制在推理过程中发挥了作用。

表 4 消融分析

| 模型                                                               | SNLI  | Matched |
|------------------------------------------------------------------|-------|---------|
| (1) proposed approach                                            | 88.8% | 79.0%   |
| (2) w/o context similarity                                       | 88.4% | 77.9%   |
| (3) w/o selection gate                                           | 88.6% | 77.8%   |
| (4) w/o context similarity & selection gate                      | 88.2% | 76.9%   |
| (5) w/o context similarity, selection gate & additional features | 88.0% | 76.3%   |

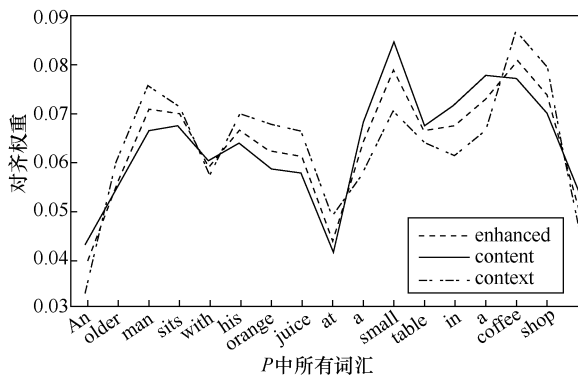
在模型（3）中，不再利用选择门机制对所有词级别的判定信息进行加权处理，而是直接将所有词汇级别的判定信息进行融合。此时，该模型在 SNLI 和 matched 数据集上的准确率分别下降到 88.6% 和 77.8%，该结果说明选择门机制能够弱化噪声信息对最终结果的影响。

在模型（4）中，通过同时移除上下文相似度和选择门机制来查看这 2 个结构的组合对最终结果的影响。将这 2 个结构同时删除后，该模型在 SNLI 和 matched 数据集上的准确率分别下降到 88.2% 和 76.9%。

最后，模型（5）移除了所有新添加的结构。此时，模型在 SNLI 和 matched 数据集上的准确率下降到 88.0% 和 76.3%。通过与模型（4）比较可以发现，该模型在这 2 个数据集上的准确率分别下降了 0.2% 和 0.6%，这说明笔者在对输入向量进行初始化时添加的语言学特征信息对该任务也有作用。

#### 4.5 对齐权重的可视化

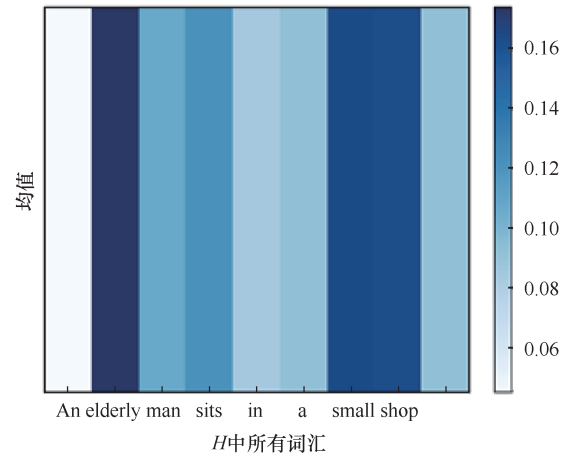
针对第 1 节中的样例  $P$  和  $H$ ，图 2 展示了  $H$  中的词 small 与  $P$  中所有词汇之间的对齐权重（曲线 enhanced、content 和 context 分别表示增强型对齐权重、词义相似度和上下文相似度）。

图 2 文本  $H$  中的词汇 small 与  $P$  中所有词汇之间的对齐权重

从图 2 可以看出，上下文相似度可以有效地改善对齐性能。词义相似度曲线展示了只利用词义相似度进行词对齐时， $P$  中的词汇 small 会得到最大的对齐权重，从而导致一个错误的预测结果。将上下文相似度也考虑进去后， $P$  中的词汇 coffee 将会获得最大的对齐权重（图 2 中的 enhanced 曲线），这也非常符合人为的判定结果。

#### 4.6 选择门的可视化

针对第 1 节中的样例  $P$  和  $H$ ，图 3 展示了  $H$  中所有词汇的选择门向量在所有维度上的均值。从图 3 可以看出，词汇 small 对应的选择门向量的值要明显大于词汇 An。这非常符合人脑的直觉：在进行最终的结果预测时，small 对应的局部判定信息比 An 对应的局部判定信息更重要。

图 3 文本  $H$  中所有词汇的选择门向量在所有维度上的均值

### 5 结束语

本文提出了一种针对自然语言推理的增强型词对齐方法。当词汇在 2 个不同文本中拥有相似的上下文信息时，这 2 个词汇有很大概率对齐。因此，本文首次提出同时利用词义相似度和上下文相似度这 2 种信息计算一个增强型的对齐权重。利用该对齐权重可以更好地进行词汇对齐，从而改善最终的预测结果。此外，在进行最终结果的预测时，本文设计了一种选择门机制对来自非关键词的信息进行弱化，从而改善预测效果。相比于以前的方法，本文提出的方法更符合人工判定时的行为，也具有很好的可解释性。实验表明，该方法与当前最优的方法相比，在准确率方面的性能是相当的，但前者具有更好的鲁棒性。

## 参考文献:

- [1] KOUYLEKOV M, MAGNINI B. Recognizing textual entailment with tree edit distance algorithms[J]. *Pascal Challenges on RTE*, 2005: 17-20.
- [2] BOS J, MARKERT K. Recognizing textual entailment with robust logical inference[C]// *The Conference Human Language Technology and Empirical Methods in Natural Language Processing*. Stroudsburg: Association for Computational Linguistics, 2005: 404-426.
- [3] BHASKAR P, BANERJEE S, PAKRAY P, et al. A hybrid question answering system for multiple choice question(MCQ)[C]// *QA4MRE at Conference and Labs of the Evaluation Forum*. [S.l.:s.n.], 2013.
- [4] ROMANO L, KOUYLEKOV M, SZPEKTOR I, et al. Investigating a generic paraphrase-based approach for relation extraction[C]// *The 11th Conference of the European Chapter of the Association for Computational Linguistics*. Stroudsburg: Association for Computational Linguistics, 2006: 409-416.
- [5] PADÓ S, CER D, GALLEY M, et al. Measuring machine translation quality as semantic equivalence: a metric based on entailment features[J]. *Machine Translation*, 2009, 23(2-3): 181-193.
- [6] HARABAGIU S, HICKL A. Methods for using textual entailment in open-domain question answering[C]// *The 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*. Stroudsburg: Association for Computational Linguistics, 2006: 905-912.
- [7] HARABAGIU S, HICKL A, LACATUSU F. Satisfying information needs with multi-document summaries[J]. *Information Processing and Management*, 2007, 43(6): 1619-1642.
- [8] HEILMAN M, SMITH N A. Tree edit models for recognizing textual entailments, paraphrases, and answers to questions[C]// *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics*. Stroudsburg: Association for Computational Linguistics, 2010: 1011-1019.
- [9] BOWMAN S R, ANGELI G, POTTS C, et al. A large annotated corpus for learning natural language inference[C]// *The 2015 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg: Association for Computational Linguistics, 2015: 632-642.
- [10] MOU L L, MEN R, LI G, et al. Natural language inference by tree-based convolution and heuristic matching[C]// *The 54th Annual Meeting of the Association for Computational Linguistics*. Stroudsburg: Association for Computational Linguistics, 2016: 130-136.
- [11] PARIKH A P, TACKSTROM O, DAS D, et al. A decomposable attention model for natural language inference[C]// *The 2016 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg: Association for Computational Linguistics, 2016: 2249-2255.
- [12] CHEN Q, ZHU X, LING Z H, et al. Enhanced LSTM for natural language inference[C]// *The 55th Annual Meeting of the Association for Computational Linguistics*. Stroudsburg: Association for Computational Linguistics, 2017: 1657-1668.
- [13] TAY Y, LUU A T, HUI S C, et al. Compare, compress and propagate: enhancing neural architectures with alignment factorization for natural language inference[C]// *The 2018 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg: Association for Computational Linguistics, 2018: 1565-1575.
- [14] POLIAK A, NARADOWSKY J, HALDAR A, et al. Hypothesis only baselines in natural language inference[C]// *The 7th Joint Conference on Lexical and Computational Semantics*. [S.l.:s.n.], 2018.
- [15] PETERS M E, NEUMANN M, IYYER M, et al. Deep contextualized word representations[C]// *The 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Stroudsburg: Association for Computational Linguistics, 2018: 2227-2237.
- [16] WANG Z, HAMZA W, FLORIAN R, et al. Bilateral multi-perspective matching for natural language sentences[C]// *The 26th International Joint Conference on Artificial Intelligence*. Los Altos: William Kaufman, 2017: 4144-4150.
- [17] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]// *The 31st Conference on Neural Information Processing System (NIPS 2017)*. [S.l.:s.n.], 2017.
- [18] DEVLIN J, CHANG M, LEE K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding[C]// *The 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Stroudsburg: Association for Computational Linguistics, 2019: 4171-4186.
- [19] DU Q L, SU K Y, ZONG C Q. Adopting the word-pair-dependency-triplets with individual comparison for natural language inference[C]// *The 27th International Conference on Computational Linguistics (COLING)*. Stroudsburg: Association for Computational Linguistics, 2018: 414-425.
- [20] PENNINGTON J, SOCHER R, MANNING C D. GloVe: global vectors for word representation[C]// *The 2014 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg: Association for Computational Linguistics, 2014: 1532-1543.
- [21] CHEN D, FISCH A, WESTON J, et al. Reading Wikipedia to answer open-domain questions[C]// *The 55th Annual Meeting of the Association for Computational Linguistics*. Stroudsburg: Association for Computational Linguistics, 2017: 1870-1879.
- [22] WILLIAMS A, NANGIA N, BOWMAN S R, et al. A broad-coverage challenge corpus for sentence understanding through inference[C]// *The 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Stroudsburg: Association for Computational Linguistics, 2018: 1112-1122.
- [23] GLOCKNER M, SHWARTZ V, GOLDBERG Y. Breaking NLI systems with sentences that require simple lexical inferences[C]// *The 56th Annual Meeting of the Association for Computational Linguistics*. Stroudsburg: Association for Computational Linguistics, 2018: 650-655.
- [24] WANG S, JIANG J. Learning natural language inference with LSTM[C]// *The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Stroudsburg: Association for Computational Linguistics, 2016: 1442-1451.
- [25] SHA L, CHANG B B, SUI Z F, et al. Reading and thinking: reread LSTM unit for textual entailment recognition[C]// *The 26th International Conference on Computational Linguistics (COLING)*. Stroudsburg: Association for Computational Linguistics, 2016: 2870-2879.
- [26] GONG Y, LUO H, ZHANG J, et al. Natural language inference over interaction space[C]// *The 6th International Conference On Learning Representations*. [S.l.:s.n.], 2018.

- [27] GHAEINI R, HASAN S A, DATLA V, et al. DR-BiLSTM: dependent reading bidirectional LSTM for natural language inference[C]// The 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg: Association for Computational Linguistics, 2018: 1460-1469.
- [28] CHEN Q, ZHU X D, LING Z H, et al. Neural natural language inference models enhanced with external knowledge[C]// The 56th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: Association for Computational Linguistics, 2018: 2406-2417.
- [29] GURURANGAN S, SWAYAMDIPTA S, LEVY O, et al. Annotation artifacts in natural language inference data[C]// The 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg: Association for Computational Linguistics, 2018: 107-112.
- [30] SRIVASTAVA N, HINTON G, KRIZHEVSKY A, et al. Dropout: a simple way to prevent neural network from overfitting[J]. The Journal of Machine Learning Research, 2014, 15(1): 1929-1958.
- [31] BALAZS J A, MARRESE-TAYLOR E, LOYOLA P, et al. Refining raw sentence representations for textual entailment recognition via attention[C]// The 2nd Workshop on Evaluating Vector Space Representations for NLP. Stroudsburg: Association for Computational Linguistics, 2017: 51-55.
- [32] CHEN Q, ZHU X D, LING Z H, et al. Recurrent neural network-based sentence encoder with gated attention for natural language inference[C]// The 2nd Workshop on Evaluating Vector Space Representation for NLP. Stroudsburg: Association for Computational Linguistics, 2017: 36-40.
- [33] NIE Y, BANSAL M. Shortcut-stacked sentence encoders for multi-domain inference[C]// The 2nd workshop on Evaluating Vector

Space Representations for NLP. Stroudsburg: Association for Computational Linguistics, 2017: 41-45.

#### [作者简介]



杜倩龙(1989-), 男, 中国科学院自动化研究所博士生, 主要研究方向为词义消歧、自然语言推理。



宗成庆(1963-), 男, 博士, 中国科学院自动化研究所研究员, 中国科学院大学岗位教授、博士生导师, 主要研究方向为自然语言处理理论与方法、机器翻译、文本数据挖掘、人机对话系统。



苏克毅(1955-), 男, 博士, 台湾“中央研究院”资讯科学研究所研究员, 主要研究方向为机器翻译、机器阅读和自然语言理解。