基于 Transformer 模型的问句语义相似度计算

丁 邱[1,2]，迟海洋[3]，严 馨[1,2+]，徐广义[4]，邓忠莹[1]

（1. 昆明理工大学 信息工程与自动化学院，云南 昆明 650500；2. 昆明理工大学 云南省人工智能重点实验室，云南 昆明 650500；3. 昆明学院 信息中心，云南 昆明 650214；4. 云南南天电子信息产业股份有限公司 昆明南天电脑系统有限公司，云南 昆明 650040）

摘 要：针对现有方法准确率不高、不能充分捕捉句子深层次语义特征的问题，提出一种基于 Transformer 编码器网络的问句相似度计算方法。在获取句子语义特征前引入交互注意力机制比较句子间词粒度的相似性，通过注意力矩阵和句子矩阵相互生成彼此注意力加权后的新的句子表示矩阵，将获取的新矩阵同原始矩阵拼接融合，丰富句子特征信息；将拼接后的句子特征矩阵作为 Transformer 编码器网络的输入，由 Transformer 编码器分别对其进行深层次语义编码，获得句子的全局语义特征；通过全连接网络和 Softmax 函数对特征进行权重调整，得到句子相似度。在中文医疗健康问句数据集上模型取得了 90.2%的正确率，较对比模型提升了将近 4.2%，验证了该方法可以有效提高句子的语义表示能力和语义相似度的准确性。

关键词：自然语言处理；Transformer 编码器；交互注意力机制；特征融合；语义相似度；语义编码；句子表示

中图法分类号：TP391 **文献标识号**：A **文章编号**：1000-7024（2023）03-0887-07

doi：10.16208/j.issn1000-7024.2023.03.034

Semantic similarity calculation of questions based on Transformer model

DING Qiu[1,2], CHI Hai-yang[3], YAN Xin[1,2+], XU Guang-yi[4], DENG Zhong-ying[1]

(1. Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China; 2. Yunnan Key Laboratory of Artificial Intelligence, Kunming University of Science and Technology, Kunming 650500, China; 3. Information Center, Kunming University, Kunming 650214, China; 4. Kunming Nantian Computer System Limited Company, Yunnan Nantian Electronic Information Industry Limited Company, Kunming 650040, China)

Abstract: To solve the problem that the existing methods are not accurate and can not fully capture the deep semantic features of sentences, a question similarity calculation method based on transformer encoder network was proposed. Before acquiring sentence semantic features, interactive attention mechanism was introduced to compare the similarity of word granularity between sentences. The attention matrix and sentence matrix were used to generate a new sentence representation matrix weighted by each other's attention. The new matrix was combined with the original matrix to enrich the sentence feature information. The spliced sentence feature matrix was used as the input of the transformer encoder network, and the transformer encoder encoded them in deep level to obtain the global semantic features of sentences. The feature weight was adjusted and the sentence similarity was calculated through the full connection network and Softmax function. On the data set of Chinese medical and health questions, the accuracy of the model is 90.2%, which is nearly 4.2% higher than that of the comparative model. It is verified that this method can effectively improve the semantic representation ability of sentences and the accuracy of semantic similarity.

Key words: natural language processing; Transformer encoder; interactive attention mechanism; feature fusion; semantic similarity; semantic coding; sentence representation

收稿日期：2021-07-07；**修订日期**：2023-03-01

基金项目：国家自然科学基金项目（61562049、61462055）

作者简介：丁邱（1997－），女，云南文山人，硕士研究生，CCF 学生会员，研究方向为自然语言处理；迟海洋（1994－），男，山东日照人，硕士，研究方向为自然语言处理；＋通讯作者：严馨（1969－），女，重庆人，硕士，副教授，CCF 会员，研究方向为自然语言处理；徐广义（1965－），男，云南宣威人，硕士，高级工程师，研究方向为自然语言处理和数据挖掘；邓忠莹（1979－），女，黑龙江双鸭山人，硕士，实验师，研究方向为数据挖掘。E-mail：kg_yanxin@sina.com

0 引 言

句子相似度计算是自然语言处理的核心算法之一，在很多领域中都或多或少会用到句子相似度计算。随着深度学习技术的不断发展，其在自然语言处理（natural language processing，NLP）领域也得到广泛应用并且取得了不错的成绩。在各种各样的问答系统中，用户是问句的提出者。而用户所提问题具有咨询意图复杂、上下文相关性弱、问题多样化、指代缺失、口语化严重等问题，加大了获取问句深层含义的难度，故句子相似度计算还面临重大的挑战。本文提出了一种基于 Transformer 编码器网络的问句相似度计算方法。主要贡献如下：

（1）针对现有的句子相似度计算方法准确率不高、不能充分捕捉句子深层次语义特征的问题，使用基于 Transformer 编码器的网络代替传统神经网络，充分捕捉问句的语义信息。

（2）针对 Siamese 网络在编码阶段只能对单独每个句子提取特征，忽视了句子之间的特征和联系，在获取句子语义特征前引入交互注意力机制，丰富问句的特征。

1 相关工作

常见的相似度计算方法有基于向量空间的方法，将句子表示为向量形式，通过计算没有语义信息的两个句子向量之间的距离来计算相似度。李晓等[1]基于 Word2vec 模型和句子结构信息计算句子相似度；郭胜国等[2]提出了一种基于词向量计算相似度和依存句法相结合的相似度计算方法。基于语义资源的方法，该方法以语义资源为基础，引入知网、WordNet 等语义资源，利用语义关系和语法成分计算句子相似度。朱新华等[3]提出了一种基于知网和同义词林的词语语义相似度计算方法。李蕾等[4]针对现有基于知网的词语语义相似度计算方法，未考虑义原节点的密度对义原距离的影响，或没有考虑义原深度与义原密度的主次关系，提出一种改进的基于知网的词语语义相似度计算方法。基于混合的方法：该方法在已有方法的基础上，综合了多种相似度计算方法。Yang 等[5]将结构特征与平面特征相结合，提高了句子相似度计算的性能。Kadupitiya 等[6]提出了一种混合方法，使用语义相似性度量与词序信息相结合，实现句子相似性计算。吴克介等[7]将词汇的词性作为权重因子融合知网与搜索引擎，改善了词汇语义相似度计算的应用效果。

随着深度学习的发展，构建神经网络模型挖掘文本特征、通过计算文本特征向量的相似度来辨识两个句子的相似度的方法已成为主流。Peng 等[8]提出了一种简单的增强型递归卷积神经网络，不仅模型体系结构简单，而且取得了较好的句子相似性学习性能。Mueller 等[9]将孪生神经网络（Siamese neural network）应用于文本的相似度计算，该模型使用长短期记忆网络（long short-term memory，LSTM）网络并结合 Manhattan 距离来计算句子的相似度。郭浩等[10]提出了一种基于 CNN 和 BiLSTM 相结合的短文本相似度计算方法，利用 CNN 和 BiLSTM 分别提取句子不同粒度特征。纪明宇等[11]设计了一种基于门控循环网络（GRU）的相似度计算方法，在智能客服数据集上取得不错的性能。赵琪等[12]提出了一种基于 Capsule-BiGRU 的文本相似度计算方法，将胶囊网络提取的局部特征矩阵和 BiGRU 网络提取的全局特征矩阵融合，获取多层次相似度向量进行相似度的计算。

随着深度学习在许多领域的成熟应用，研究者们开始优化和改进深度学习。在自然语言处理方面，将注意力机制同神经网络结合起来，提升了模型的准确率、取得了不错效果。孙阳[13]提出了一种基于注意力机制的汉语句子相似度计算方法，利用交互注意力来计算句子对之间的词汇关联信息、通过自注意力提取当前词与其它词的关系，与现有模型对比取得了最佳的效果。冯兴杰等[14]综合考虑两个句子的词语和句子间的语义信息，提出了一种基于多注意力 CNN 的问句相似度计算模型，提高了模型对问句的辨识能力。李霞等[15]提出一种结合门控卷积神经网络和自注意力机制的句子相似度计算模型，利用门控卷积神经网络获取句子局部信息和通过自注意力机制获取句子内的远距离单词之间的语义相关信息，将其拼接融合后作为最后的向量，在不同数据集上取得了较好的性能。胡艳霞等[16]提出了一种基于树模型和 LSTM 的句子语义相似度计算方法。在数据集上的实验结果表明，结合多头注意力机制的句子相似度计算方法优于非注意力方法。谷歌提出的 Transformer 模型[17]大部分采用自注意力机制可以一步到位获取序列的全局特征，且计算效率高、复杂度低、学习能力更强。乔伟涛等[18]使用 Transformer 模型对句子进行深层语义编码，其语义相似度计算性能优于基于 Siamese 的传统网络。

2 模型构建

基于 Transformer 模型的问句相似度计算模型框架如图 1 所示，以处理健康信息问句数据集为例，本文所提出的模型主要分为以下 5 部分：词嵌入层、交互注意力层、Transformer 编码层、特征融合层、输出层。

模型关键步骤和具体流程归纳如下：

首先，将句子编码为词向量组成的矩阵形式，作为模型输入。

其次，利用交互注意力机制比较句子间词粒度的相似性，并通过注意力矩阵和句子矩阵相互生成彼此注意力加权后的新的句子表示矩阵；将获取的新矩阵同原始矩阵拼接融合，丰富句子特征信息。

再将拼接后的句子特征矩阵作为 Transformer 编码器网

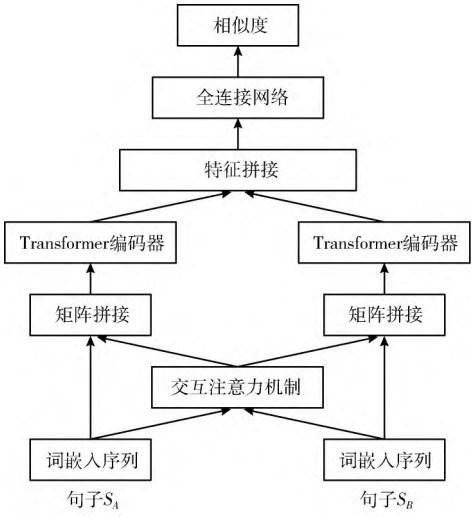

图 1　基于 Transformer 模型的问句相似度计算模型框架

络的输入，由 Transformer 编码器分别对句子 S_A、S_B 进行深层次语义编码，获取句子 S_A、S_B 的全局语义特征。

然后，对句子的语义特征向量进行融合后输入到全连接层。

最后，通过全连接网络对特征进行权重调整和使用 Softmax 函数达到句子相似度计算的目的。

2.1　词嵌入层

本文使用 Word2vec 模型对经过预处理的健康问句语料进行训练，生成 300 维的词向量，得到句子矩阵 $A=(a_1^{\mathrm{T}},a_2^{\mathrm{T}},a_3^{\mathrm{T}},\cdots,a_m^{\mathrm{T}})^{\mathrm{T}}$，$B=(b_1^{\mathrm{T}},b_2^{\mathrm{T}},b_3^{\mathrm{T}},\cdots,b_n^{\mathrm{T}})^{\mathrm{T}}$。其中，$A\in R^{m\times d}$，$B\in R^{n\times d}$，$a_i$、$b_i$ 分别表示句子 S_A、S_B 中第 i 个词的词向量，d 表示词向量维度。

2.2　交互注意力层

基于 Siamese 网络结构只是在最后阶段对两个句子进行交互，编码阶段句子是相互独立的，为了提升网络性能，将交互注意力机制加入网络，在获取句子语义向量前对句子词向量矩阵进行一次交互[19]，生成每个句子的交互注意力矩阵 C、F，C、F 矩阵维度分别与 A、B 相同，最后将矩阵 A 与 C 拼接融合为一个新的矩阵，作为 Transformer 网络的输入。具体计算过程如下：

首先，由 A、B 生成注意力矩阵 E

$$E=AB^{\mathrm{T}} \tag{1}$$

接下来，通过注意力矩阵 E 和句子矩阵 A、B 相互生成彼此注意力加权后的新的句子表示矩阵 C、F，C、F 的行向量生成过程如下

$$c_i=\sum_{j=1}^{n}\frac{\exp(e_{ij})}{\sum_{k=1}^{n}\exp(e_{ik})}\cdot b_j,\ i\in[1,m] \tag{2}$$

$$f_j=\sum_{i=1}^{m}\frac{\exp(e_{ij})}{\sum_{k=1}^{m}\exp(e_{kj})}\cdot a_i,\ j\in[1,n] \tag{3}$$

其中，e_{ij} 为注意力矩阵 E 的第 i 行第 j 列元素，表示 S_A 的第 i 个词与 S_B 的第 j 个词的相似度。$a_i=(a_{i1},a_{i2},a_{i3},\cdots,a_{id})$ 是矩阵 A 的第 i 行的行向量，$b_j=(b_{j1},b_{j2},b_{j3},\cdots,b_{jd})$，是矩阵 B 的第 j 行的行向量。$c_i=(c_{i1},c_{i2},c_{i3},\cdots,c_{id})$，是矩阵 C 的第 i 行的行向量；$f_j=(f_{j1},f_{j2},f_{j3},\cdots,f_{jd})$，是矩阵 F 的第 j 行的行向量，m、n 分别表示句子 S_A、S_B 的长度。

最后，将矩阵 A、B 和矩阵 C、F 对应的行向量 a_i、b_j、c_i、f_j 对应分别融合拼接成新的矩阵 G、P 对应的行向量，G、P 的行向量表示如下

$$g_i=[a_i;c_i] \tag{4}$$

$$p_j=[b_j;f_j] \tag{5}$$

其中，g_i，p_j 分别表示 G、P 的行向量。

2.3　Transformer 编码层

为了获取句子的深层次语义信息，本文利用 Transformer 编码器网络分别对句子矩阵 G、P 进行特征提取，最终得到句子的编码向量。本文只使用 Transformer 的编码器部分，Transformer 编码器由两个子层组成，结构如图 2 所示。

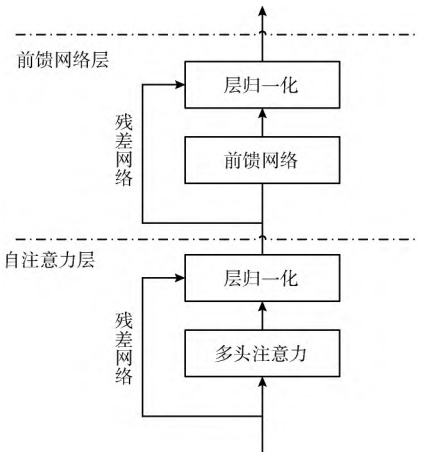

图 2　Transformer 编码器结构

Transformer 获取句子特征过程如下（以句子 S_A 为例）：

（1）将 $G\in R^{m\times d'}$ 作为 Transformer 编码器的输入，并添加一个位置编码向量 Positional Encoding，从而解决 Transformer 模型缺少对输入序列中词语顺序表示的问题，计算方法如下

$$\mathrm{PE}(pos,2i)=\sin(pos/10000^{2i/d'}) \tag{6}$$

$$\mathrm{PE}(pos,2i+1)=\cos(pos/10000^{2i/d'}) \tag{7}$$

其中，$d'=2d$ 表示词向量的维度，pos 是当前词在句子中的位置，i 是向量中每个值的下标。

（2）令 $Q=K=V=G$，则多头自注意力的计算公式如下

$$\mathrm{Attention}(Q,K,V)=\mathrm{softmax}\left(\frac{QK^{\mathrm{T}}}{\sqrt{d_k}}\right)V \tag{8}$$

$$head_i=\mathrm{Attention}(QW_i^Q,KW_i^K,VW_i^V)\ (i=1,\cdots,h) \tag{9}$$

$$Z = \text{MultiHead}(Q,K,V) = \text{Concat}(head_1 \cdots head_h)W^O \tag{10}$$

其中，Q、K、V 分别是多头注意力的查询矩阵、键矩阵和值矩阵，等同于上文得到的融合词向量矩阵，d_k 表示键的维数 $d_k = d_v = d'/h$。$W_i^Q \in R^{d' \times d_k}$、$W_i^K \in R^{d' \times d_k}$、$W_i^V \in R^{d' \times d_v}$ 是变换矩阵，将 Q、K、V 投影到 h 个不同的维度，$W^O \in R^{hd_v \times d'}$ 表示多头注意力的权重矩阵。

(3) 使用残差连接和层归一化调整特征信息的计算公式如下

$$L = \text{LayerNorm}(G + Z) \tag{11}$$

式中：LayerNorm 表示归一化函数。

(4) 对残差连接和层归一化处理的输出 L 使用前馈神经网络做两次线性变换，用 Relu 激活函数进行激活，计算公式如下

$$\text{FFN}(L) = \max(0, LW_1 + b_1)W_2 + b_2 \tag{12}$$

式中：W_1、W_2 表示两次线性变换的权重矩阵，b_1、b_2 表示偏置矩阵。

(5) 最后对线性变换和第一次层归一化的结果再做一次残差连接和层归一化，并将其结果作为 Transformer 编码器的最后输出

$$T = \text{LayerNorm}(L + \text{FFN}(L)) \tag{13}$$

式中：$T \in R^{m \times d'}$。

2.4 特征融合及其输出

Transformer 编码层的输出 T_A、T_B 经过全连接层进行一维化得到 x_A、x_B，x_A、x_B 分别表示句子 S_A、S_B 的语义特征向量，并计算 x_A 和 x_B 的差和乘积。

受 Chen 等增强推理信息方法的启发，在向量拼接前将获取的句子 S_A、S_B 的语义特征 x_A、x_B 进行向量相减和相乘操作，使模型可以关注到两个句子相异和相同的部分[20,21]。特征融合如下

$$u = [x_A; x_B; x_A - x_B; x_A * x_B] \tag{14}$$

将拼接后的特征表示 u 送入两层全连接网络，其中第一层全连接网络的维度为 300 维，使用 Relu 激活函数，将第二层全连接网络的输出经 Softmax 归一化函数，预测句子的相似度，过程如下

$$u' = \text{Relu}(W_u u + b_u) \tag{15}$$

$$\hat{p}(y \mid A,B) = \text{softmax}(W_{u'} u' + b_{u'}) \tag{16}$$

$$y' = \underset{y}{\text{argmax}}(\hat{p}(y \mid A,B)) \tag{17}$$

其中，W_u 和 b_u 为第一层全连接网络的权重和偏置，$W_{u'}$、$b_{u'}$ 为 Softmax 的权重和偏置。

模型损失函数采用交叉熵损失函数，具体计算方法如下

$$J(\theta) = -\frac{1}{k}\sum_{i=1}^{k} r_i \ln(y_i) + \lambda \|\theta\|^2 \tag{18}$$

式中：θ 为参数，k 表示类别个数。r_i 为真实标签，y_i 为预测值，$\lambda\|\theta\|^2$ 为正则项。

3 实验和结果分析

3.1 数据集

本文使用的实验数据集是天池-新冠疫情相似句对判定大赛官方提供的医疗问题数据对，共涉及“肺炎”、“支原体肺炎”、“支气管炎”、“上呼吸道感染”、“肺结核”、“哮喘”、“胸膜炎”、“肺气肿”、“感冒”、“咳血”等多个病种，所含用户问句丰富。为了和本文研究内容相结合以及提高样本质量，首先对大赛提供数据进行筛选，挑选出符合本文的数据。标签表示问句之间的语义是否相同，若相同，标为 1，若不相同，标为 0。最终获得样本总数 10 000 条，其中 8000 条用作训练集，2000 条用作测试集。数据集样例见表 1。

表 1 数据集样例

问句 A	问句 B	标签
肺部发炎是什么原因引起的	肺部炎症有什么症状	0
支原体肺炎症状有哪些	支原体肺炎表现有哪些	1
胸膜炎在复查阶段咳嗽请问是什么病	胸膜炎在复查阶段咳嗽是怎么回事	0
成年人会得支原体肺炎吗	支原体肺炎会有哪些症状	0
感冒了嗓子不舒服怎么办	感冒嗓子不舒服有什么妙招	1
做什么检查确诊新型肺炎	怀疑肺炎，做什么检查	1
小儿支原体肺炎咳嗽吃什么药好	小儿支原体肺炎咳嗽怎么办	1
北京治疗哮喘好的医院是哪家	哪个地区治疗哮喘高效	0
哮喘病在饮食上应该注意什么	哮喘病饮食方面需要注意哪些	1

3.2 实验参数设置

本文实验基于 Pytorch 深度学习框架实现，使用 Adam 优化器，模型具体参数设置见表 2。

相似度计算的结果是一个介于 0 到 1 之间的实数，越接近 1 代表两个句子的相似性越高，反之则越低。为刻画模型的准确率，还需要人为设定一个阈值来界定相似与不

相似。目前，关于衡量句子相似度还未有一个统一和权威的标准，本文根据实验结果统计发现，阈值设为 0.7 时句子相似度计算的结果准确度较高，因此，本文将句子相似度的阈值设为 0.7。

表 2　实验参数设置

参数	数值
词嵌入维度	300
多头自注意力机制的“头数”	6
编码器数量	2
学习率	0.001
序列最大长度	32
dropout	0.5
batch size	64
epoch	10

3.3　评价指标

本文采用正确率（Accuracy）和 F_1 指标对相似度计算效果进行评价。其中，正确率计算方法如式（19）所示，P、R 及 F_1 值计算方法如式（20）～式（22）所示

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \tag{19}$$

$$P = TP/(TP + FP) \tag{20}$$

$$R = TP/(TP + FN) \tag{21}$$

$$F_1 = 2PR/(P + R) \tag{22}$$

其中，$Accuracy$ 表示预测成功的问句对数量占问句对总数的比例；F_1 是精确率 P 和召回率 R 的调和均值。TP 表示将相似问句成功预测为相似的样本数量；TN 表示将不相似问句成功预测为不相似的样本数量；FP 表示将不相似问句错误预测为相似的样本数量；FN 表示将相似问句错误预测为不相似的样本数量。

3.4　实验结果及分析

为验证模型的有效性，将本文提出的方法在同一个数据集上与其它多种模型做对比实验：

Siamese-CNN：该模型使用卷积神经网络分别对两个句子建模和提取句子特征，并将其作为 MLP（多层感知机）的输入，最终获得两个句子的相似度。

Siamese-LSTM：该模型使用 LSTM 分别对问句进行语义编码和特征提取，通过比较两个句子向量的距离来确定两个句子的语义是否相同。

Siamese-BiLSTM：该模型使用双向 LSTM 分别对问句进行语义编码和特征抽取，由句子向量间的距离来确定两个句子的语义是相同。

Siamese-BiLSTM-Attention：该模型在 Siamese-BiLSTM 基础上使用注意力机制产生权重向量，将单词级别的特征拼接成句子级别语义向量，获取更多语义特征。

本文通过分析和研究 Transformer 网络模型，针对问句语义相似度计算任务需同时考虑句子表示特征及句子间的交互信息，提出了一种基于 Transformer 的问句相似度计算模型（question similarity transformer，QSTransformer）来解决相似问句匹配问题。

各模型的损失收敛曲线和正确率收敛曲线如图 3 和图 4 所示。

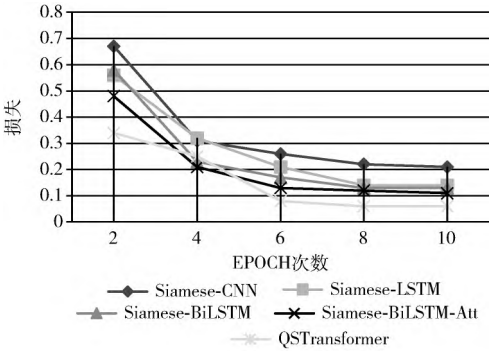

图 3　损失收敛曲线

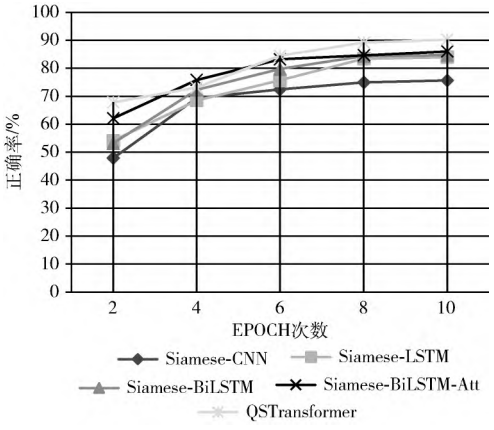

图 4　正确率收敛曲线

从模型的损失收敛曲线和正确率收敛曲线，可见当 epoch 为 8 时，模型的损失和正确率都开始趋于平稳，epoch 为 10 时，达到收敛状态。因此为了减少计算开销，我们将 epoch 设置为 10。从总体来看，本文提出的模型损失要小于其它的对比模型，且正确率也优于其它模型。

不同模型在测试语料上整体性能对比实验结果见表 3，本文模型基于表 1 数据集样例的混淆矩阵见表 4。

由表 3 的实验结果可发现，在使用同样输入的前提下：LSTM 和 BiLSTM 的效果要比使用 CNN 网络效果好，这是因为卷积神经网络只能提取句子的局部特征，LSTM 和 BiLSTM 可以获取整个句子序列信息，比起卷积神经网络循环神经网络语义编码能力更强。BiLSTM 是双向长短时记忆网络，可以缓解长距离特征获取问题，提取到长句子中

表 3 不同模型对比实验结果

实验模型	P/%	R/%	F_1/%	Accuracy/%
Siamese-CNN	72.9	73.6	73.2	75.6
Siamese-LSTM	86.0	81.0	83.4	83.9
Siamese-BiLSTM	86.8	81.9	84.3	84.7
Siamese-BiLSTM-Att	88.8	83.5	86.1	86.0
QSTransformer	**90.8**	**85.2**	**87.9**	**90.2**

表 4 本文模型基于表 1 数据集样例的混淆矩阵

		QSTransformer	
		相似	不相似
实际	相似	4	1
	不相似	0	4

距离较远的有效特征。而加入注意力机制的 Siamese-BiLSTM-Attention 网络模型较 Siamese-BiLSTM 模型正确率有所提升，但提升效果不明显，可以看出单纯引入局部注意力机制并没有很好提升模型表现。本文引入交互注意力机制获取了句子之间的关联信息并使用 Transformer 编码器网络获取句子特征，较 Siamese-BiLSTM-Attention 网络模型有显著提升，正确率提升了 4.2%，提高了模型的语义理解、匹配能力。由表 4 的混淆矩阵也可以看出本文模型性能较好，取得了较高的准确率。

为进一步验证引入交互注意力的有效性，我们又在数据集上进行对比实验，验证加入句子间交互注意力对模型性能的影响，实验结果见表 5。

表 5 引入交互注意力对比实验结果

实验模型	F_1/%	Accuracy/%
QSTransformer	**87.9**	**90.2**
QSTransformer-1	87.1	87.4

QSTransformer-1 表示在本文模型的基础上去除交互注意力的模型。由表 5 实验结果可以分析出，引入交互注意力机制可以有效提升模型的效果，验证了交互注意力可以使模型关注到句子间的相似特征。

使用 Transformer 网络模型来获取句子语义特征效果最佳，这是因为基于 Transformer 的模型能够从句子中获取更为丰富的特征信息。比起传统的卷积神经网络和循环神经网络 Transformer 的语义编码能力更强。另外在使用 Transformer 编码器提取句子特征之前引入交互注意力机制，这样既可以充分提取句子内的语义信息，也考虑了句子间的联系。在丰富问句特征的同时，实现了对句子的深层次语义编码，从而提升了模型性能。本文使用的天池大赛官方提供的医疗问题数据对整体长度较短，训练样本规模整体较小，基于 Transformer 的网络模型在大规模数据集上训练效果提升会更为明显。

4 结束语

本文提出了一种基于 Transformer 编码器的问句相似度计算模型（QSTransformer），同时利用交互注意力机制将句子间的相似特征加入到模型的相似度计算中，综合考虑了句子的语义信息、句子间交互信息，在数据集上的实验结果表明，本文提出的方法较已有的经典网络模型有不错的提升效果，验证了该模型的有效性。

在下一步研究工作中，可以尝试进一步改善 Transformer 网络结构，增强问句的语义表示；同时尝试结合知识图谱、医学词典来丰富问句信息，提升相似度计算性能。

参考文献：

[1] LI Xiao, XIE Hui, LI Lijie. Research on sentence semantic similarity calculation based on word2vec [J]. Computer Science, 2017, 44 (9): 256-260 (in Chinese). [李晓，解辉，李立杰. 基于 Word2vec 的句子语义相似度计算研究 [J]. 计算机科学，2017，44 (9)：256-260.]

[2] GUO Shengguo, XING Dandan. Sentence similarity calculation based on word vector and its application [J]. Modern Electronic Technology, 2016, 39 (13): 99-102 (in Chinese). [郭胜国，邢丹丹. 基于词向量的句子相似度计算及其应用研究 [J]. 现代电子技术，2016，39 (13)：99-102.]

[3] ZHU Xinhua, MA Runcong, SUN Liu, et al. Semantic similarity calculation of words based on CNKI CI forest [J]. Chinese Journal of Information, 2016, 30 (4): 29-36 (in Chinese). [朱新华，马润聪，孙柳，等. 基于知网与词林的词语语义相似度计算 [J]. 中文信息学报，2016，30 (4)：29-36.]

[4] LI Lei, YANG Lihua. Improved algorithm of word semantic similarity based on HowNet [J]. Computer Technology and Development, 2019, 29 (4): 42-46 (in Chinese). [李蕾，杨丽花. 基于知网的词语语义相似度改进算法 [J]. 计算机技术与发展，2019，29 (4)：42-46.]

[5] Yang M, Li P, Zhu Q. Sentence similarity on structural representations [M] //Natural Language Understanding and Intelligent Applications. Springer, Cham, 2016: 481-488.

[6] Kadupitiya JCS, Ranathunga S, Dias G. Sinhala short sentence similarity calculation using corpus-based and knowledge-based similarity measures [C] //Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing, 2016: 44-53.

[7] WU Kejie, WANG Jiawei. Calculation of lexical semantic similarity based on HowNet and search engine [J]. Computer and Modernization, 2018 (4): 90-94 (in Chinese). [吴克介，王家伟. 基于知网与搜索引擎的词汇语义相似度计算 [J]. 计算机与现代化，2018 (4)：90-94.]

[8] Peng S，Cui H，Xie N，et al. Enhanced-RCNN：An efficient method for learning sentence similarity [C] //Proceedings of the Web Conference，2020：2500-2506.

[9] Mueller J，Thyagarajan A. Siamese recurrent architectures for learning sentence similarity [C] //Proceedings of the AAAI Conference on Artificial Intelligence，2016：2786-2792.

[10] GUO Hao，XU Wei，LU Kai，et al. Short text similarity calculation method based on CNN and BiLSTM [J]. Information Technology and Network Security，2019，38 (6)：61-64 (in Chinese). [郭浩，许伟，卢凯，等. 基于 CNN 和 BiLSTM 的短文本相似度计算方法 [J]. 信息技术与网络安全，2019，38 (6)：61-64.]

[11] JI Mingyu，WANG Chenlong，AN Xiang，et al. Sentence similarity calculation method for intelligent customer service [J]. Computer Engineering and Application，2019，55 (13)：123-128 (in Chinese). [纪明宇，王晨龙，安翔，等. 面向智能客服的句子相似度计算方法 [J]. 计算机工程与应用，2019，55 (13)：123-128.]

[12] ZHAO Qi，DU Yanhui，LU Tianliang，et al. Text similarity analysis algorithm based on capsule-BiGRU [J/OL]. Computer Engineering and Application：1-9. [2021-06-03]. http://kns.cnki.net/kcms/detail/11.2127.TP.20200826.1635.010.html (in Chinese). [赵琪，杜彦辉，芦天亮，等. 基于 capsule-BiGRU 的文本相似度分析算法 [J/OL]. 计算机工程与应用：1-9. [2021-06-03]. http://kns.cnki.net/kcms/detail/11.2127.TP.20200826.1635.010.html.]

[13] SUN Yang. Chinese sentence similarity calculation based on convolutional neural network [D]. Hefei：University of Science and Technology of China，2019：37-46 (in Chinese). [孙阳. 基于卷积神经网络的中文句子相似度计算 [D]. 合肥：中国科学技术大学，2019：37-46.]

[14] FENG Xingjie，ZHANG Le，ZENG Yunze. Problem similarity calculation model based on multi attention CNN [J]. Computer Engineering，2019，45 (9)：284-290 (in Chinese). [冯兴杰，张乐，曾云泽. 基于多注意力 CNN 的问题相似度计算模型 [J]. 计算机工程，2019，45 (9)：284-290.]

[15] LI Xia，LIU Chengbiao，ZHANG Youhao，et al. Cross language sentence semantic similarity calculation model based on local and global semantic fusion [J]. Chinese Journal of Information，2019，33 (6)：18-26 (in Chinese). [李霞，刘承标，章友豪，等. 基于局部和全局语义融合的跨语言句子语义相似度计算模型 [J]. 中文信息学报，2019，33 (6)：18-26.]

[16] HU Yanxia，WANG Cheng，LI Bicheng，et al. Sentence semantic similarity calculation based on multi attention mechanism Tree-LSTM [J]. Chinese Journal of Information，2020，34 (3)：23-33 (in Chinese). [胡艳霞，王成，李弼程，等. 基于多头注意力机制 Tree-LSTM 的句子语义相似度计算 [J]. 中文信息学报，2020，34 (3)：23-33.]

[17] Vaswani A，Shazeer N，Parmar N，et al. Attention is all you need [J]. arXiv preprint arXiv：1706.03762，2017.

[18] QIAO Weitao，HUANG Haiyan，WANG Shan. Research on semantic similarity algorithm based on Transformer encoder [J/OL]. Computer Engineering and Application：1-10. [2021-06-03]. http://kns.cnki.net/kcms/detail/11.2127.TP.20200701.1721.026.html (in Chinese). [乔伟涛，黄海燕，王珊. 基于 Transformer 编码器的语义相似度算法研究 [J/OL]. 计算机工程与应用：1-10. [2021-06-03]. http://kns.cnki.net/kcms/detail/11.2127.TP.20200701.1721.026.html.]

[19] TAN Jie. Research on text similarity calculation method based on Attention-Based BertCNN [D]. Wuhan：Wuhan Research Institute of Posts and Telecommunications，2020：17-33 (in Chinese). [谭杰. 基于 Attention-Based BertCNN 的文本相似度计算方法研究 [D]. 武汉：武汉邮电科学研究院，2020：17-33.]

[20] Chen Q，Zhu X，Ling Z，et al. Enhanced LSTM for natural language inference [J]. arXiv preprint arXiv：1609.06038，2016.

[21] Parikh AP，Täckström O，Das D，et al. A decomposable attention model for natural language inference [J]. arXiv preprint arXiv：1606.01933，2016.