计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2023,32(5):291−299 [doi: 10.15888/j.cnki.csa.009066]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于 Transformer 与 HowNet 义原知识融合的双驱动语义蕴含识别①

陈　帆[1], 黄　炎[2], 张新访[1]

[1](华中科技大学 机械科学与工程学院, 武汉 430074)
[2](华中科技大学 人工智能与自动化学院, 武汉 430074)
通信作者: 黄　炎, E-mail: platanus@hust.edu.cn

摘　要: 语义蕴含识别旨在检测和判断两个语句的语义是否一致, 以及是否存在蕴含关系. 然而现有方法通常面临中文同义词、一词多义现象困扰和长文本难理解的挑战. 针对上述问题, 本文提出了一种基于 Transformer 和 HowNet 义原知识融合的双驱动中文语义蕴含识别方法, 首先通过 Transformer 对中文语句内部结构语义信息进行多层次编码和数据驱动, 并引入外部知识库 HowNet 进行知识驱动建模词汇之间的义原知识关联, 然后利用 soft-attention 进行交互注意力计算并与义原矩阵实现知识融合, 最后用 BiLSTM 进一步编码文本概念层语义信息并推理判别语义一致性和蕴含关系. 本文所提出的方法通过引入 HowNet 义原知识手段解决多义词及同义词困扰, 通过 Transformer 策略解决长文本挑战问题. 在 BQ、AFQMC、PAWSX 等金融和多语义释义对数据集上的实验结果表明, 与 DSSM、MwAN、DRCN 等轻量化模型以及 ERNIE 等预训练模型相比, 该模型不仅可以有效提升中文语义蕴含识别的准确率 (相比 DSSM 模型提升 2.19%), 控制模型的参数量 (16 M), 还能适应 50 字及以上的长文本蕴含识别场景.

关键词: 义原知识融合; Transformer; HowNet; 蕴含识别

引用格式: 陈帆,黄炎,张新访.基于 Transformer 与 HowNet 义原知识融合的双驱动语义蕴含识别.计算机系统应用,2023,32(5):291–299. http://www.c-s-a.org.cn/1003-3254/9066.html

Co-driven Recognition of Semantic Entailment Based on Fusion of Transformer and HowNet Sememe Knowledge

CHEN Fan[1], HUANG Yan[2], ZHANG Xin-Fang[1]

[1](School of Mechanical Science & Engineering, Huazhong University of Science and Technology, Wuhan 430074, China)
[2](School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract: Semantic entailment recognition aims to detect and judge whether the semantics of two Chinese sentences are consistent and whether there is an entailment relationship. The existing methods, however, usually face the challenges of Chinese synonyms, polysemy, and difficulty in understanding long texts. To solve the above problems, this study proposes a co-driven Chinese semantic entailment recognition method based on the fusion of Transformer and sememe knowledge of HowNet. First, the internal structural semantic information of Chinese sentences is encoded at multiple levels and undergoes data-driven processing by Transformer. The external knowledge base HowNet is introduced for knowledge-driven modeling of the sememe knowledge correlations between words. Then, the interaction attention is calculated by Soft-Attention and achieves knowledge fusion with the sememe matrix. Finally, BiLSTM is used to encode the semantic information of the conceptual layer of texts and infer and judge the semantic consistency and entailment

① 基金项目: 国家重点研发计划 (2021YFB2012202); 湖北省科技重大专项 (2020AEA011); 湖北省重点研发计划 (2020BAB100, 2021BAA171, 2021BAA038)
收稿时间: 2022-10-02; 修改时间: 2022-11-04, 2022-11-15; 采用时间: 2022-11-16; csa 在线出版时间: 2023-03-01
CNKI 网络首发时间: 2023-03-02

relationship. The proposed method employs the sememe knowledge of HowNet to solve the problems of polysemy and synonyms and uses the Transformer strategy to resolve the challenge of long texts. The experimental results on financial and multi-semantic interpretation pair data sets such as BQ, AFQMC, and PAWSX show that compared with lightweight models such as DSSM, MwAN, and DRCN and pre-trained models such as ERNIE, this model can effectively improve the recognition accuracy of Chinese semantic entailment (an increase of 2.19% compared with that of the DSSM model) and control the number of model parameters (16 M). In addition, it can also adapt to entailment recognition scenarios of long texts with no less than 50 words.

Key words: sememe knowledge fusion; Transformer; HowNet; entailment recognition

1 引言

文本蕴含识别任务, 是自然语言处理重要的子任务之一, 并且可以应用于大量的自然语言处理中. 其看似是一个简单的任务, 但在不同的下游任务中其变种十分广泛, 如信息检索、问答系统、对话系统、机器翻译等中. 该任务的输入主要为句子对, 最终结果是判断此句子对之间的逻辑关系 (中立, 蕴含, 矛盾) 或者句子对是否相似.

中文文本蕴含识别任务是比较困难的任务之一, 原因在于中文里一词多义、同义词等特点相对于其他语言更为突出, 从而导致在日常交流中也常有“词不达意”的情况出现. 然而传统的算法并不能获取中文里的语义信息, 比如 Bow、VSM、TF-IDF、BM25、Jaccord、SimHash 等经典算法, 其主要解决的是词汇层面的匹配问题, 或者说词汇层面的相似度问题. 而基于词汇重合度的匹配算法有很大的局限性, 原因包括:

(1) 词义局限. 同一词语在不同的语境下所表达的意思会不同.

(2) 结构局限. 对于一个词语, 调转先后顺序表达的意思会完全不同, 如“奶牛”“牛奶”.

(3) 知识局限. 一个句子结合常识看是错误的, 但从词法和句法上看都是正确的.

为了解决上述的问题, 本文提出了一种基于 Transformer[1] 和 HowNet 的文本蕴含识别方法, 拓展了对于句子语义信息获取方面的研究. 我们首先通过 Transformer 对中文语句内部结构语义信息进行多层次编码和数据驱动, 并引入外部知识库 HowNet 进行知识驱动建模词汇之间的义原知识关联, 然后利用 soft-attention 进行交互注意力计算并与义原矩阵实现知识融合, 最后用 BiLSTM 进一步编码文本概念层语义信息并推理判别语义一致性和蕴含关系. 通过多项实验表明, 与轻量化模型和预训练模型相比, 合理使用 HowNet 的义原知识以及 Transformer 对于长文本的优势, 能够使得模型准确率得到了一定的提升.

本文提出的模型有以下创新点.

● 针对中文语义蕴含识别任务存在的中文同义词、一词多义现象困扰和长文本难理解的挑战, 提出了一种基于 Transformer 和 HowNet 义原知识融合的双驱动中文语义蕴含识别方法, 在利用数据样本驱动进行 Transformer 编码和模型预训练之外, 利用 HowNet 义原知识进行知识驱动以增强对同义词、多义词等的深层语义理解.

● 提出一种义原知识融合的技术手段, 通过 Transformer 对中文语句内部结构语义信息进行多层次编码, 并引入外部知识库 HowNet 建模词汇之间的义原知识库, 最后利用 soft-attention 进行交互注意力计算并于义原矩阵实现知识融合.

● 将所提出的模型应用到了蚂蚁金融、银行金融行业和多语义应用场景, 与 DSSM、MwAN、DRCN 等轻量化模型以及 ERNIE 等预训练模型进行了实验对比分析, 不仅可以有效提升中文语义蕴含识别的准确率, 控制模型的参数量, 还能适应 50 字以内的长文本蕴含识别场景.

2 相关工作

近年来, 随着深度学习的快速发展, 用机器学习解决文本蕴含识别问题的方法被大量提出. 在语义信息获取方面, 经典的短文本匹配模型 DSSM[2], 解决了 LSA、LDA 等方法存在的字典爆炸的问题, 但也因为采用了词袋模型, 损失了上下文信息. 2016 年提出的 ESIM 模型[3], 综合应用了 BiLSTM 和注意力机制, 首次在局部推理中让两个句子之间产生了交互. 2018 年提出的 DIIN 模型[4], 采用 CNN 和 LSTM 来做特征提取, 但作者在其输入层同时采用了词向量和局向量, 额

外输入了一些句法特征, 并采用 DenseNet[5] 来进行特征提取. 2018 年提出的 DRCN 模型[6], 借鉴了图像识别中 DenseNet 的密集连接操作, 通过密集连接 RNN, 保留了文本的最原始信息, 通过多次循环, 不断向矩阵向量添加交互信息, 最后全连接层输出. 2018 年提出的 KIM 模型[7], 利用了外部知识库 WordNet[8] 来判断两个句子之间的逻辑关系, 将外部先验知识嵌入相似度矩阵中. 2018 年提出的 MwAN 模型[9], 作者在模型中利用了多种注意力机制 (拼接、双线性、点乘、相减), 充分捕捉了句子对之间的关系, 最后对多个结果进行权重化组合, 通过 GRU 和全连接输出最终的概率.

而在句子结构方面, 2015 年提出的 CT-LSTM[10], 为了解决 LSTM 无法对句子的结构信息进行提取的问题, 提出了一种树形的 LSTM, 针对序列问题进行了探讨, 与常用的 RNN 顺序建模不同, 其利用句子的依存关系作为 LSTM 的输入, 对未来研究也有一定的启发.

随着 2018 年 BERT 模型[11] 的提出, 一股预训练模型的潮流席卷了整个 NLP 界, 在各大 NLP 榜单中名列前茅. BERT 拥有一个完整的 encoder-decoder 框架, 其基本组成为 Transformer, 主要由多头注意力 (multi-head attention, MA) 构成, 它是一个用纯注意力搭建的模型, 可以解决 RNN 及其变体存在的长距离依赖问题, 也就是注意力机制有更好的记忆力, 能够记住更长距离的信息. BERT 模型的优点便是更能学习到句子中的一些语法和语义信息, 使输出的词语向量更具有代表性, 更大的参数量也使得其在各种下游任务中表现优秀. 为了解决预训练模型参数量大, 难以在消费级 GPU 上应用的问题, Dettmers 等人[12] 为 Transformer 提出了 LLM.Int8(), 使得普通消费级 GPU 上也能使用非常大的模型, 而不会降低性能. 而除了 Transformer 外, Liu 等人[13] 提出了另一种简单的、与注意力无关的架构 gMLP, 在预训练某些指标上达到了与 Transformer 同等的水平, 甚至在某些下游任务上更胜一筹.

由于汉语词汇特殊的多义性, 中文文本匹配存在着一定的困难, 不同的词语往往包含有相同的意思, 如“中国”与“华夏”, 从语义上来讲是一致的, 而从字形上讲却没有关联. 为了解决这类同义词的问题, 许多研究人员选择利用中文的词性、依存句法等信息来计算相似度. 如严娇等人[14] 尝试将文本进行词性标注后, 仅保留名词、动词和形容词, 结合依存句法分析获得词语对, 以 PageRank[15]、度中心性等作为指标, 对大量文本建立语法网络, 提出了结合句法关系和词汇语义的文本相似度计算方法. 黄炎等人[16] 提出了一种基于主题约束的篇章级文本自动生成模型, 利用关键词集的同义性生成了多个文章主题规划, 李琳等人[17] 提出了概念向量空间的概念, 将文档表示为概念词的集合并建立向量空间, 再通过余弦相似度计算语义相似度, 效果优于词袋模型+Word2Vec[18].

同样使用 HowNet 进行中文蕴含识别的还有 Lyu 等人[19] 于 2021 年提出的 LET, 他们对每一个中文词汇下的所有义原的初始向量进行了图注意力转化, 随后通过注意力池化获得每一个词的义原向量, 通过 GRU 与 BERT 词向量融合获得最终的词向量. 虽然中文词语常有多个语义, 但我们辨别语义所需要使用的正确义原往往只有少数, 因此会导致其获得的义原向量不能与实际句子很好地相匹配, 会含有多余的语义信息. 而本文是将义原信息进行初步筛选, 再将其融入交互矩阵中, 避免了加入多余信息.

3 研究方法

本节主要介绍了基于 Transformer 和 HowNet 义原知识融合的双驱动中文蕴含识别模型, 分析了其主要结构以及其主要作用. 其主要结构如图 1 所示, 一共分为 6 层, 分别为 Transformer 层、Attention 层、BiLSTM 层、平均池化和最大池化层、全连接层.

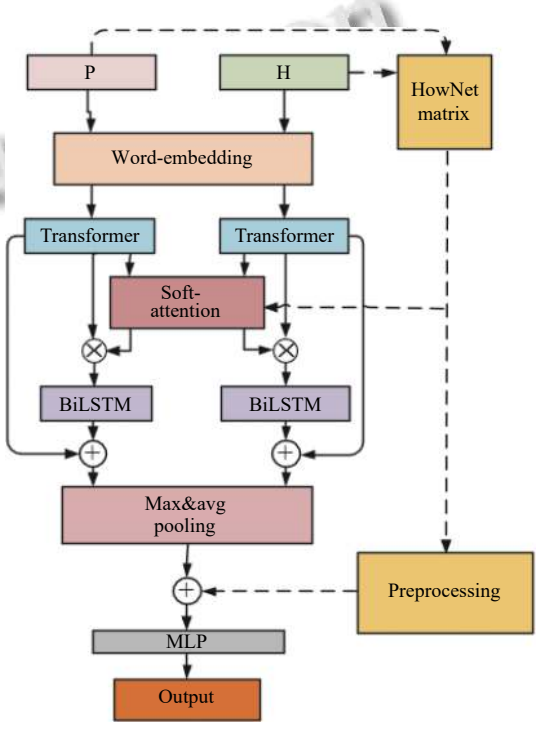

图 1 模型结构图

3.1 Transformer 层

Transformer 层的作用是使向量化表示的文本通过神经网络, 获取深层次的语义信息. 常用的神经网络有卷积神经网络、长短时记忆网络等. 本文模型使用了 Transformer 架构作为文本的编码层, 对句子向量进行处理, 其主要由多头注意力机制以及前馈神经网络组成, 可以缓解梯度消失的问题, 其单头注意力机制计算方式如下.

$$Query = W^Q X \tag{1}$$

$$Key = W^K X \tag{2}$$

$$Value = W^V X \tag{3}$$

$$Attention(Q,K,V) = Softmax\left(\frac{QK^{\mathrm{T}}}{\sqrt{d_k}}\right)V \tag{4}$$

其中, d_k为输入向量的维度. $W^Q, W^K, W^V \in R^{d_{\text{model}} \times d_k}$.

但由于其并没有获取文本序列的位置信息, 因此需要在词向量中加入绝对位置编码, 其计算方式如式 (5), 式 (6) 所示:

$$PE_{(p,2m)} = \sin\left(\frac{p}{10000^{2m/d}}\right) \tag{5}$$

$$PE_{(p,2m+1)} = \cos\left(\frac{p}{10000^{2m/d}}\right) \tag{6}$$

其中, d表示当前词向量的维度, p表示当前词在句子中的位置.

3.2 Attention 层

Attention 层是文本蕴含识别模型中的重要组成, 具有速度快、效果好、参数少的优点. 目前有很多不同的类型, 如 soft-attention、hard attention、self-attention 等, 也有如 Sun 等人[20] 提出的专注与局部的注意力机制. 本文采用了常用的软注意力机制 (soft-attention), 但向其中添加了基于 HowNet 生成的语义矩阵信息, 并且加入了可训练权重γ.

HowNet 是一个基于汉语和英语的常识知识库, 解释了概念与概念之间以及概念所拥有的属性之间的关系. 本文主要利用该外部知识库, 获得句子对中对应两个词语的所有中文义原. 如果两个词语存在相同的义原, 则其矩阵中的对应位置的值会被设定为 1, 否则设定为 0. 图 2 以两个例句为例, 进行义原分析.

图 2 中最上框代表句子分词后的结果, 中间框代表当前词语对应的多种义原信息, 下侧框代表两个词语义原信息的交集. 由图 2 框图可知, “中国”有中国、与特定国家相关、地方、亚洲等义原, “华夏”有借入、金融、留存、中国、国家等义原, 两个词语义原的交集为中国、国家、地方, 所以此时$HowNet$(中国, 华夏) $\neq 0$.

$$M_{i,j} = \begin{cases} 1, & HowNet\left(P_i, H_j\right) \neq 0 \\ 0, & HowNet\left(P_i, H_j\right) = 0 \end{cases} \tag{7}$$

$$M = \begin{bmatrix} 0 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 0 \end{bmatrix} \tag{8}$$

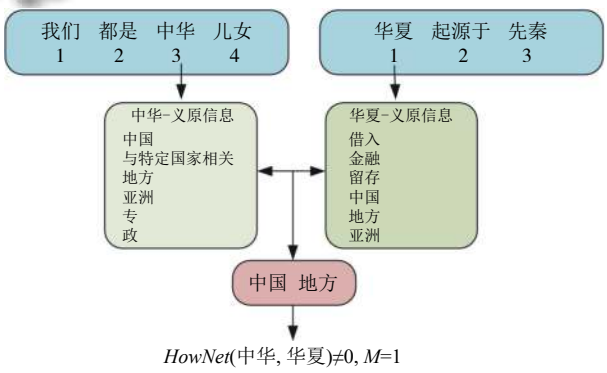

图 2 HowNet 义原分析

Attention 矩阵e生成公式如式 (9):

$$e = PH^{\mathrm{T}} + \gamma \cdot M \tag{9}$$

其中, γ为一个可训练参数. 此时的 Attention 矩阵e不仅融合了句子间的文本信息, 还获得了句子间词语对的语义信息, 矩阵变化热力图如图 3 所示, 加入义原信息之后, 某些位置的权重被提高, 这表明该位置获取了义原矩阵的信息. 获得了改良的注意力矩阵e后, 软注意力计算方式如下:

$$\hat{P} = \sum_{j=1}^{l_h} \frac{\exp\left(e_{ij}\right)}{\sum_{k=1}^{l_h} \exp\left(e_{ik}\right)} P_{tf}, \ \forall i \in \left[1,2,\cdots,l_p\right] \tag{10}$$

$$\hat{H} = \sum_{j=1}^{l_p} \frac{\exp\left(e_{ij}\right)}{\sum_{k=1}^{l_p} \exp\left(e_{ik}\right)} H_{tf}, \ \forall i \in [1,2,\cdots,l_h] \tag{11}$$

在式 (10), 式 (11) 中, P_{tf}、H_{tf}为句子对经过 Transformer 后的矩阵向量. l_p, l_h表示句子长度, $\hat{P}$, $\hat{H}$表示经过软注意力机制后的输出.

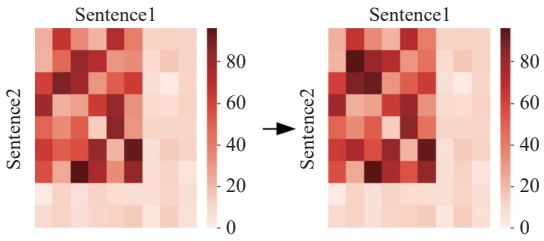

图 3 Attention 矩阵变化

3.3 BiLSTM 层

这一层用于将经过软注意力机制的 Transformer 层输出$\hat{P}$和$\hat{H}$进行处理, 经过双向长短时记忆网络, 将前向编码和后向编码进行拼接, 能够进一步获取上下文信息.

长短时记忆网络的输入如下:

$$P_{\text{BiLSTM}} = BiLSTM\left(\hat{P}\right) \tag{12}$$

其中, $\hat{P}$指的是句子P经过软注意机制后的输出向量, P_{BiLSTM}为经过长短时神经网络后的向量.

3.4 平均池化和最大池化层

为了使经过 Transformer 和 BiLSTM 后的文本信息相融合, 本模型通过最大池化以及平均池化将多个输入拼接, 该层的目的在于使两个句子的向量维度由$R^{l\times d}$变为R^{l}, 便于后续输入全连接层:

$$P_o = Concat\left(\left[P_{tf}; P_{\text{BiLSTM}}\right]\right) \tag{13}$$

$$P_{\text{rep}} = [\max(P_o); \text{mean}(P_o)] \tag{14}$$

3.5 全连接层

在获得句子对的完整句向量表达P_{rep}和H_{rep}后, 常用的向量拼接方法为直接拼接并且将其输入多层前馈神经网络并获得结果. 而提出的模型在拼接时, 考虑了 HowNet 矩阵中的信息, 分别获取了 HowNet 矩阵中两个维度之和HN_{row}和HN_{col}. 并将其与P_{rep}和H_{rep}进行拼接. 获得前馈神经网络的最终输入 H.

$$HN_{\text{row}} = \text{sum}(M, axis = 0)$$

$$HN_{\text{col}} = \text{sum}(M, axis = 1)$$

$$H = concat\left(P_{\text{rep}}; H_{\text{rep}}; P_{\text{rep}} - H_{\text{rep}}; HN_{\text{col}}; HN_{\text{row}}\right) \tag{15}$$

其中, sum(·)表示按照 $axis$ 维度求和, 以HN_{row}为例, HN_{row}代表 HowNet 矩阵沿着第 1 个维度进行求和后的结果, 即为句子 1 对应的 HowNet 信息, 通过向量拼接, H获得了两个句子对应的义原信息, 其中的$P_{\text{rep}}-H_{\text{rep}}$也表示了两个句向量的差异性.

3.6 预测层

获得句子对的最终句向量表达H后. 模型采用了一个两层全连接神经网络, 获得句其配结果. 采用的损失函数为交叉熵损失函数, 其计算方式如式 (16).

$$\begin{cases} output = FFN(H) \\ Loss = \dfrac{1}{N}\sum\limits_{i} -[y_i \times \log(p_i) + (1-y_i) \times \log(1-p_i)] \end{cases} \tag{16}$$

其中, y_i表示样本i的标签, 正样本为 1, 负样本为 0. p_i表示样本i预测为正样本的概率.

除了常用的交叉熵损失函数外, 我们也尝试了苏剑林提出的 CoSent 损失函数, 其让正样本对的相似度大于负样本的相似度, 使正负样本在向量空间的距离尽量远, 实验表明, 使用 CoSent 损失函数[21] 对预训练方法如 BERT、SentenceBERT[22] 有一定效果, 使得预训练模型收敛更快, 但对于本文提出的模型, 在非预训练情况下其效果不如交叉熵损失函数.

在训练阶段, 我们使用了 MultiStepLR 来动态调整学习率, 在实验的第 20、50、80、100、150 次迭代, 以 0.5 的衰减率来进行学习率的更新. 通过随着迭代次数增加动态调整学习率, 模型的收敛速度增加, 其变化趋势如图 4 所示.

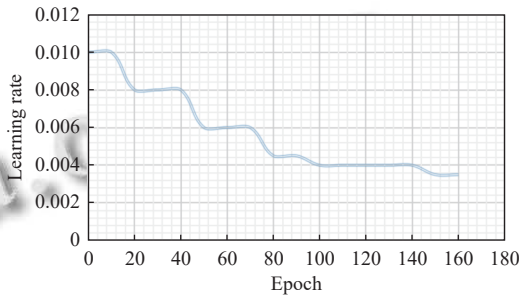

图 4 学习率衰减

4 实验与分析

4.1 数据集

为了验证本文提出的基于 Transformer 和 HowNet 双驱动的文本蕴含识别模型的有效性, 本文分别在 3 个公开的数据集上进行实验. 数据集分别为 PAWSX 数据集, AFQMC 数据集, BQ Corpus 数据集. 其数据集大小及样例如表 1 和表 2 所示.

PAWSX-zh 数据集是谷歌发布的多语言释义对的数据集, 特点是具有高度重叠词汇. AFQMC 数据集是

蚂蚁金融相似度数据集, 其中包含 34 334 条训练数据, 4 316 条验证数据和 3 861 条测试数据. BQ Corpus 是银行金融领域的问题匹配数据集, 包括了从一年的线上银行系统日志中抽取的问题对, 是目前最大的银行领域的问题匹配数据, 包含 10 000 条训练数据, 10 000 条验证数据, 10 000 条测试数据.

表 1 数据集大小

数据集名称	训练集大小	验证集大小	测试集大小
PAWSX	49 401	2 000	2 000
AFQMC	34 334	4 316	3 861
BQ Corpus	100 000	10 000	10 000

表 2 BQ 数据集样例

句子1	句子2	标签
微信消费算吗	还有多少钱没还	0 (矛盾)
下周有什么好产品?	元月份有哪些理财产品	1 (蕴含)
能查账单吗	可以查询账单	0
1无法借款	QQ有微粒代	0

4.2 评价指标

本文采用了准确率Acc和$F1$作为评价指标, 其计算公式如下所示:

$$Acc = \frac{TP+TN}{TP+FN+FP+TN} \tag{17}$$

$$P = \frac{TP}{TP+FP} \tag{18}$$

$$R = \frac{TP}{TP+FN} \tag{19}$$

$$F1 = 2\times\frac{P\times R}{P+R} \tag{20}$$

其中, TP为真正例, TN为真负例, FP为假正例, FN为假负例.

4.3 实验设置

本文的实验采用了型号为 RTX2080ti 的 4 卡 GPU 服务器. 模型训练参数如表 3 所示. 软件版本如下: Python 3.6.13, PyTorch 1.10.2, OpenHowNet 2.0, Transformers 4.18.0.

表 3 模型初始训练参数

参数	数值
词嵌入层维度	300
隐藏层数	128
最大序列长度	100
批次大小batch_size	64
Transformer编码器层数	10
模型优化器	Adam
初始学习率	0.01

4.4 对比实验结果

为了验证本文提出的模型的实际效果, 在非预训练模型方面, 选择了 3 个经典的文本匹配模型 DSSM、MwAN 和 DRCN. 而预训练模型方面, 我们选择了 BERT-wwm-ext, BERT 以及百度 ERNIE.

选择的数据集为 PAWSX、AFQMC 以及 BQ Corpus. 为了保证实验的统一性, 对同一数据集, 所有模型均采用了相同的 jieba 词表, 对比的指标为准确率 Acc和$F1$值. 表 4–表 6 为实验结果, 其中, Improvement 表示相较于所选择效果最优的基线模型的提升百分比.

表 4 BQ 数据集实验结果 (%)

模型	是否预训练	Acc	$F1$
DSSM	×	77.12	76.47
MwAN	×	73.99	73.29
DRCN	×	74.65	76.02
Ours	×	**78.81**	**76.62**
Improvement	×	**+2.19**	**+1.96**
BERT-wwm-ext	√	84.71	83.94
BERT	√	84.50	84.00
ERNIE	√	84.67	84.20
Ours-BERT	√	**84.82**	**84.33**
Improvement	√	**+0.177**	**+0.464**

表 5 AFQMC 数据集实验结果 (%)

模型	是否预训练	Acc	$F1$
DSSM	×	57.02	30.75
MwAN	×	65.43	28.63
DRCN	×	66.05	40.60
Ours	×	**66.62**	**42.93**
Improvement	×	**+0.86**	**+5.7**
BERT-wwm-ext	√	81.76	80.62
BERT	√	81.43	79.77
ERNIE	√	81.54	80.81
Ours-BERT	√	**81.84**	**81.93**
Improvement	√	**+0.097**	**+1.38**

由表 4 可以获得, 本文提出的模型, 在 BQ 数据集上的准确率Acc和$F1$均高于其他模型. 如表 5 所示, 从数据集上看, 3 个模型在 AFQMC 上的结果均不太好, 初步分析原因, 其样本数据的语言规范性较差, 比如会出现不完整句子的情况, 如“可以帮我冻结花呗吗”和“里冻结花呗额度”, 其标签为 1, 该种数据导致了在训练集和测试集上的效果均不理想. 但本文提出的模型

融合了 Transformer 和 HowNet 外部知识库, 效果更优, 结果表明, 对于不规范的文本, 通过获取某些词语的义原信息并进行匹配, 能够提升一定性能. 如表 6 所示, 而对于数据样本都为难样例的 PAWSX 数据集, 传统的 DSSM 模型无法获取交互信息以及上下文信息, 所以效果较差, 对于难负样例, 其句子对的高度相似导致了义原信息过于类似, 每一对句子所生成的 HowNet 矩阵都几乎一致, 所以获取义原知识方法对判断难样本的正负样例效果不好.

表 6 PAWSX 数据集实验结果 (%)

模型	是否预训练	*Acc*	*F*1
DSSM	×	42.64	59.43
MwAN	×	52.70	52.65
DRCN	×	61.24	56.52
Ours	×	**62.55**	**59.72**
Improvement	×	**+2.13**	**+0.48**
BERT-wwm-ext	√	77.23	76.52
BERT	√	77.06	77.16
ERNIE	√	78.02	77.59
Ours-BERT	√	**78.33**	**77.96**
Improvement	√	**+0.397**	**+0.476**

从误差分析上看, 由于都直接使用了 jieba 分词来进行文本的预处理, 其产生的分词误差, 对实验结果有不同程度的影响. 虽然有分词误差, 但对于同一数据集, 所有的模型都采用了同一个词表, 相比较之下, 本文提出的模型效果更优.

4.5 消融研究

在本节中, 为了理解其各个部分的相对重要性以及有效性, 对本文提出模型的不同结构进行了消融研究. 共分为两个实验, 都使用了 BQ 数据集, 其中, 实验 1 评估了使用不同的分词工具以及是否使用外部知识库 HowNet 结果对于实验结果的影响; 实验 2 评估了 BQ 数据集中不同文本长度对 HowNet 的影响; 实验 3 改变了 Transformer 层的层数, 评估 Transformer 编码层层数对于实验结果的影响.

从实验结果表 7 可以获得, 使用 HowNet 一定程度上可以提升模型的性能, 与不使用 HowNet 相比, 在各种分词工具下的准确率都提高了. 如果数据样本中存在一些义原复杂的词汇, 通过引入外部知识库能够明显提升模型对于多义词、近义词的敏感度, 能够显著提高模型的性能.

由表 8 结果可获得, 无论是对于文本长度小于 15 的文本还是大于 15 小于 50 的文本, HowNet 都能有效提升性能, 并且在较长的文本中能够获得更多的有效义原信息, 获得更优的效果. 经过该数据集中最长文段实验, 本模型在保证实验效果的基础上能够处理的最长文本长度为 50.

表 7 消融实验 1 实验结果

所用分词工具	是否使用HowNet	*Acc*	*F*1
jieba	√	**0.7881**	**0.7662**
	×	0.7783	0.7624
PKUseg	√	**0.7869**	**0.7653**
	×	0.7792	0.7611
HanLP	√	**0.7853**	**0.7599**
	×	0.7735	0.7512

表 8 消融实验 2 实验结果

文本长度	HowNet	*Acc*	*F*1
1–15	√	**0.7869**	**0.7652**
	×	0.7763	0.7521
15–50	√	**0.7884**	**0.7684**
	×	0.7792	0.7545

从表 9 结果数据可以获得, Transformer 层数越高, 模型的效果越好. 通过堆叠 Transformer 编码层, 一定程度上可以提高模型的性能, 但同时模型的参数量、模型的训练时长都会显著增加, 其收敛速度也会明显变慢. 我们取编码层数为 6 的模型为最优模型, 其参数量为 16M, 而非预训练模型中效果最优的模型 DRCN 则达到了 19M. 在实际应用中样例如表 10.

表 9 消融实验 3 实验结果

Transformer编码层层数	*Acc*	*F*1
2	0.7433	0.7353
4	0.7648	0.7572
6	0.7752	**0.7731**
8	0.7853	0.7658
10	**0.7881**	0.7662

表 10 实际应用结果

句子对	是否加入HowNet	预测结果	正确结果
A: 解除花呗的绑定 B: 取消开通花呗	是 否	1 0	1 (蕴含)

在训练过程中, 可训练参数γ的变化如图 5.

由图 5 可知, 在实验过程中, 通过观察注意力矩阵的可训练参数γ的变化, 可获得随着迭代次数的增加, HowNet 所获得的义原信息矩阵在注意力矩阵中所占的权重是逐步提高的, 这表明原始文本生成的 HowNet 义原信息对于模型效果提升是有积极作用的.

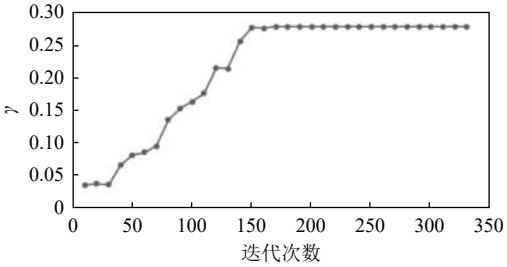

图 5 可训练参数γ变化图

5 结束语

本文提出的基于 Transformer 以及 HowNet 双驱动文本蕴含识别模型, 使用了 HowNet 外部知识库, 针对句子对中的词语对进行义原匹配, 同时使用了基于多头注意力的 Transformer 获取文本信息, 以及能够更好获取序列信息的 BiLSTM, 在模型中融合了多种信息, 使得模型对于概念信息较为敏感. 实验表明, 与 DSSM、MwAN、DRCN 模型和没有加入义原信息的预训练模型相比, 该模型在 BQ, AFQMC 和 PAWSX 数据集上有一定的提升, 通过堆叠 Transformer 层数, 能够有效提升实验结果, 但同时参数量也随之增加, 相比之下本文提出的模型参数更少, 效果更优. 与同样采用 HowNet 作为外部信息库的 LET 相比, 本文对 HowNet 义原信息的使用上只筛选了相关的义原信息, 避免了其他义原对于结果的影响, 更加准确且更加直观. 本文通过理论创新、技术创新、应用创新, 提出了一种 Transformer 与 HowNet 义原知识双驱动的中文语义蕴含研究方法, 将模型应用到了金融等具体应用领域, 实验表明, 本文提出的模型能够有效利用义原知识, 有效提升了中文蕴含识别的准确率, 还能适应 50 字以内的长文本蕴含识别场景, 不论是在轻量化模型还是在预训练模型中都有一定的提升效果.

在未来的研究工作中, 笔者还将继续完善该模型, 将其义原知识添加到 Transformer 内部结构中, 进一步提升针对中文文本的效果, 除此之外, 对于中文的外部知识库, 还需要有更多的工作去补充.

参考文献

1 Vaswani A, Shazeer N, Parmar N, *et al*. Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6000–6010.

2 Huang PS, He XD, Gao JF, *et al*. Learning deep structured semantic models for Web search using clickthrough data. Proceedings of the 22nd ACM International Conference on Information & Knowledge Management. San Francisco: ACM, 2013. 2333–2338.

3 Chen Q, Zhu XD, Ling ZH, *et al*. Enhanced LSTM for natural language inference. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver: Association for Computational Linguistics, 2017. 1657–1668.

4 Gong YC, Luo H, Zhang J. Natural language inference over interaction space. Proceedings of the 6th International Conference on Learning Representations. Vancouver: OpenReview.net, 2018. 1–15.

5 Huang G, Liu Z, van der Maaten L, *et al*. Densely connected convolutional networks. Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 2261–2269.

6 Kim S, Kang I, Kwak N. Semantic sentence matching with densely-connected recurrent and co-attentive information. Proceedings of the 2019 AAAI Conference on Artificial Intelligence, 2019, 33(1): 6586–6593. [doi: 10.1609/aaai.v33i01.33016586]

7 Chen Q, Zhu XD, Ling ZH, *et al*. Neural natural language inference models enhanced with external knowledge. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Melbourne: Association for Computational Linguistics, 2018. 2406–2417.

8 Lin DK. Review of "WordNet: An electronic lexical database" by Christiane Fellbaum. Computational Linguistics, 1999, 25(2): 292–296.

9 Tan CQ, Wei FR, Wang WH, *et al*. Multiway attention networks for modeling sentence pairs. Proceedings of the 27th International Joint Conference on Artificial Intelligence. Stockholm: AAAI Press, 2018. 4411–4417.

10 Tai KS, Socher R, Manning CD. Improved semantic representations from tree-structured long short-term memory networks. Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. Beijing: Association for Computational Linguistics, 2015. 1556–1566.

11 Raffel C, Shazeer N, Roberts A, *et al*. Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research, 2020, 21(1): 140.

12 Dettmers T, Lewis M, Belkada Y, *et al*. LLM.int8(): 8-bit

matrix multiplication for transformers at scale. arXiv:2208.07339, 2022.
13 Liu HX, Dai ZH, So DR, *et al*. Pay attention to MLPs. Proceedings of the 35th Conference on Neural Information Processing Systems. NeurIPS, 2021. 1–12.
14 严娇, 马静, 房康. 基于融合共现距离的句法网络下文本语义相似度计算. 现代图书情报技术, 2019, 3(12): 93–100.
15 Page L, Brin S, Motwani R, *et al*. The PageRank Citation Ranking: Bringing Order to the Web. San Francisco: Stanford InfoLab, 1998.
16 黄炎, 孙海丽, 徐科, 等. 基于主题约束的篇章级文本生成方法. 北京大学学报 (自然科学版), 2020, 56(1): 9–15. [doi: 10.13209/j.0479-8023.2019.103]
17 李琳, 李辉. 一种基于概念向量空间的文本相似度计算方法. 数据分析与知识发现, 2018, 2(5): 48–58.
18 Mikolov T, Chen K, Corrado G, *et al*. Efficient estimation of word representations in vector space. Proceedings of the 1st International Conference on Learning Representations. Scottsdale: ICLR, 2013. 1–12.
19 Lyu B, Chen L, Zhu S, *et al*. LET: Linguistic knowledge enhanced graph transformer for Chinese short text matching. Proceedings of the 2021 AAAI Conference on Artificial Intelligence, 2021, 35(15): 13498–13506. [doi: 10.1609/aaai.v35i15.17592]
20 Sun HL, Huang Y, Lu SF. Improving fine-grained text sentiment transfer for diverse review generation. Proceedings of 2020 IEEE International Conference on Artificial Intelligence and Computer Applications. Dalian: IEEE, 2020. 261–266.
21 苏剑林. CoSENT(一): 比 Sentence-BERT 更有效的句向量方案. https://spaces.ac.cn/archives/8847. (2022-01-06).
22 Reimers N, Gurevych I. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong: Association for Computational Linguistics, 2019. 3982–3992.

(校对责编: 牛欣悦)