文章编号:1671-6833(2024)02-0051-09

基于关键实体和文本摘要多特征融合的话题匹配算法

纪 科[1,2], 张 秀[1,2], 马 坤[1,2], 孙润元[1,2], 陈贞翔[1,2], 邬 俊[3]

(1. 济南大学 信息科学与工程学院,山东 济南 250022;2. 济南大学 山东省网络环境智能计算技术重点实验室,山东 济南 250022;3. 北京交通大学 计算机与信息技术学院,北京 100044)

摘 要:随着网络的快速普及,互联网新闻的数量剧增,在这种情况下,如何有效地找到更加符合特定主题的相关报道成为一个迫切需要解决的问题。针对这一问题,提出了基于关键实体和文本摘要多特征融合的话题匹配算法。首先,使用 W^2NER 模型进行命名实体识别,通过词频、TF-IDF、词的合群性、词词相似度和词句相似度特征,提取关键的实体。其次,使用 Pegasus 模型进行文本摘要,通过 BiLSTM 融合关键实体特征与文本摘要特征,得到新闻文本的深层次语义特征。再次,使用交叉注意力机制对待匹配新闻进行特征交互,增进彼此的联系。最后,融合新闻文本的深层次语义特征和文本交互特征,共同参与文本话题匹配的判断。在来自于搜狐的真实数据上进行了不同算法的对比实验,结果表明:所提算法准确率和精确率均与其他算法效果相近,召回率和 $F1$ 值均有所提升。

关键词:话题匹配; 关键实体; 文本摘要; 文本匹配; 信息检索

中图分类号:TP391.1 **文献标志码**:A **doi**:10.13705/j.issn.1671-6833.2024.02.008

随着现代科技的迅速发展,人们习惯于从网络获取新闻资讯、搜索热点新闻。然而,随着使用网络浏览新闻的频率不断上升,人们对于新闻资讯的要求也越来越高,希望浏览到相关且更加多元化的新闻。

现有的新闻检索往往采用关键词检索算法或文本匹配算法[1]。基于关键词的检索[2]缺少新闻文本语义的概括,检索出的新闻大多不相关,存在检索不准确、相关性差等问题。在基于文本匹配的新闻检索方面,传统的方法[3]更多是在关注文本词形和词汇层面的相似性,但是词汇往往存在一词多义现象,并且用词汇代替整句的语义不够完整。深度学习文本匹配模型[4]侧重于文本语义向量的构建以及交互,但是过分关注文本语义的相似性,忽视新闻文本的话题概括,检索出来的新闻过于相似,极易降低阅读观感,在实际应用中无法满足用户需求。

针对上述问题,本文提出一种基于关键实体和文本摘要多特征融合的话题匹配算法。由于关键实体和摘要可以概括文本的话题,因此该方法基于关键实体提取和文本摘要技术,将融合得到的关键实体特征和摘要特征作为概括文本话题的深层次语义特征,与通过交叉注意力机制(cross-attention)交互的文本语义特征一起,共同参与文本话题匹配的判断。在搜狐公开数据集上进行对比实验,结果表明该算法性能优于目前比较流行的深度学习文本匹配算法。

1 相关工作

1.1 文本匹配

随着自然语言处理(NLP)的发展,文本匹配技术有了很大的进展。文本匹配通过文本中蕴含的语义信息,判别文本之间的矛盾性和相似性,可以应用于很多场景,比如信息检索[5]、对话系统[6]等。

目前,基于深度学习的文本匹配技术可分为表示型文本匹配、交互型文本匹配和预训练语言模型的文本匹配 3 种。表示型文本匹配是先将待匹配的两段文本进行编码得到向量表示,然后计算向量的相似度,更侧重对语义向量的构建,它的优势是结构简单、易于实现,如 SimLSTM[7]模型。交互型文本匹配则是在输入层就进行词语间的匹配,不仅注重整个文本的语义表示,也关注局部文本的表示和交互,注重挖掘语义焦点,ABCNN[8]模型是交互型文

收稿日期:2023-08-10;**修订日期**:2023-10-28

基金项目:国家自然科学基金资助项目(61702216,61772231);山东省重大科技创新工程项目(2021CXGC010103)

作者简介:纪科(1989—),男,辽宁辽阳人,济南大学副教授,博士,主要从事机器学习、推荐系统相关研究,E-mail:ise_jik@ujn.edu.cn。

引用本文:纪科,张秀,马坤,等. 基于关键实体和文本摘要多特征融合的话题匹配算法[J]. 郑州大学学报(工学版),2024,45(2):51-59. (JI K, ZHANG X, MA K, et al. Topic matching algorithm based on multi-feature fusion of key entities and text abstracts [J]. Journal of Zhengzhou University(Engineering Science),2024,45(2):51-59.)

本匹配模型。预训练语言模型的匹配方法则是利用预训练加微调的方式完成文本匹配任务,预训练语言模型利用海量的语料数据可以学习到通用的语义表示,进而实现下游文本匹配任务。

近年来,随着预训练语言模型 BERT[9] 的广泛使用,出现了一系列基于 BERT 的预训练模型,例如 RoBERTa[10]、NEZHA[11],还有将主题信息融入预训练语言模型 BERT 的 tBERT[12] 模型和将主题过滤加入文本匹配任务[13],以及将关键词信息融入预训练语言模型 BERT 的 KeywordBERT[14]、DC-Match[15],这些预训练模型为额外信息融入 BERT 提供了基础。

1.2 命名实体识别

命名实体识别(NER)旨在识别文本中具有特定含义的实体,比如人名、地名、组织名、机构名等。近年来,基于深度学习的 NER 取得了优异的表现。深度学习模型对输入数据进行特征提取,再使用非线性激活函数提高模型的表达能力,完成多层神经网络的训练和预测任务。Huang 等[16] 使用双向长短期记忆网络(BiLSTM)和条件随机场 CRF 的方式解决命名实体识别问题,至今仍在命名实体识别方面被广泛应用。Li 等[17] 提出了 W^{2}NER 模型,通过构建词与词的关系,统一了普通扁平 NER、嵌套 NER 和不连续的 NER 等 3 种 NER 任务模型,在 NER 方面有了很大的进展。

1.3 文本摘要

文本摘要旨在通过算法精练提取文本的主要内容,将文本转化为包含关键信息的简洁摘要。随着文本摘要技术的发展,产生了 2 条技术路线:抽取式文本摘要和生成式文本摘要。抽取式文本摘要通过算法从原文中抽取关键信息组成摘要。Liu[18] 将抽取式摘要分解为序列标注和句子排序任务进行建模。生成式文本摘要是模型根据原文的内容,自动生成文本摘要,它允许摘要中出现新的词语,具有较高的灵活性。Zhang 等[19] 提出了 Pegasus 模型,将输入文本中的重要句子遮蔽,再利用文本其他句子生成被遮蔽的重要句子,加深了模型对文档的理解,实现了生成式摘要任务。

2 问题定义

假设用 $\boldsymbol{S}$ 表示源新闻集,$\boldsymbol{T}$ 表示目标新闻集,s_x 为 $\boldsymbol{S}$ 中的一个源新闻样本,t_x 为 $\boldsymbol{T}$ 中的一个目标新闻样本。给定源新闻-目标新闻对 $\{s_x, t_x\}$,为它设置一个状态标签 $y_x \in \{0,1\}$,其中 1 代表源新闻和目标新闻话题匹配,0 代表源新闻和目标新闻话题不匹配,$\{s_x, t_x, y_x\}$ 为一个训练样本。根据上述定义,n 个训练样本组成了训练数据集,如式(1)所示:

$$\boldsymbol{D}_{\text{train}} = ((s_1,t_1,y_1),(s_2,t_2,y_2),\cdots,(s_n,t_n,y_n))。 \quad (1)$$

本文利用训练数据集 $\boldsymbol{D}_{\text{train}}$ 构建模型,f 为模型损失函数,判断当前源新闻 $s_{x'}$ 和目标新闻 $t_{x'}$ 是否匹配的标签 $y_{x'}$,如式(2)所示:

$$y_{x'} = f(s_{x'}, t_{x'})。 \quad (2)$$

3 话题匹配算法

这一节将介绍本文提出的话题匹配算法,包括基于关键实体和文本摘要的深层次语义特征提取、基于交叉注意力(cross-attention)的文本交互以及融合文本交互特征和深层次语义特征的匹配。话题匹配模型架构如图 1 所示。

3.1 深层次语义特征提取

深层次语义信息是新闻文本的关键实体和文本摘要信息,本文将关键实体特征和文本摘要特征通过双向 LSTM[20] 进行特征融合,得到新闻文本的深层次语义特征。

3.1.1 实体提取器

本文采用 W^{2}NER 模型进行命名实体识别,在人民日报 NER 数据集训练模型,实体生成器如图 2 所示。

首先将新闻文本输入基于预训练语言模型的 BERT,转化成向量形式。给定新闻样本 s,其对应的长度为 l 的字符序列 $s=[x_1,x_2,\cdots,x_l]$,通过 BERT 处理之后获得 s 中的每个字符的表示向量如式(3)所示:

$$\boldsymbol{v} = \text{BERT}(s) = [\boldsymbol{v}_1,\boldsymbol{v}_2,\cdots,\boldsymbol{v}_l]。 \quad (3)$$

为了进一步增强上下文的联系,采用双向 LSTM 来生成包含上下文信息的向量表示,如式(4)所示:

$$\boldsymbol{h} = \text{BiLSTM}(\boldsymbol{v}) = [\boldsymbol{h}_1,\boldsymbol{h}_2,\cdots,\boldsymbol{h}_l]。 \quad (4)$$

接着采用 CLN(conditional layer normalization),构造出词对信息矩阵(word embedding),叠加距离信息矩阵(distance embedding)和区域信息矩阵(region embedding)后,通过不同空洞率的空洞卷积进行特征提取。

然后叠加以上 3 个 embedding,通过不同空洞率的空洞卷积进行特征提取,将得到的特征连接到一起,形成词对网格表征 $\boldsymbol{Q}$。$\boldsymbol{Q}$ 为预测层 MLP 的输入,而预测层中双仿射分类器(biaffine)的输入是来自编码层(BERT+BiLSTM)的输出。词对关系包括 *None*、*NNW*、*THW*-* 3 种关系,*None* 表示 2 个字没

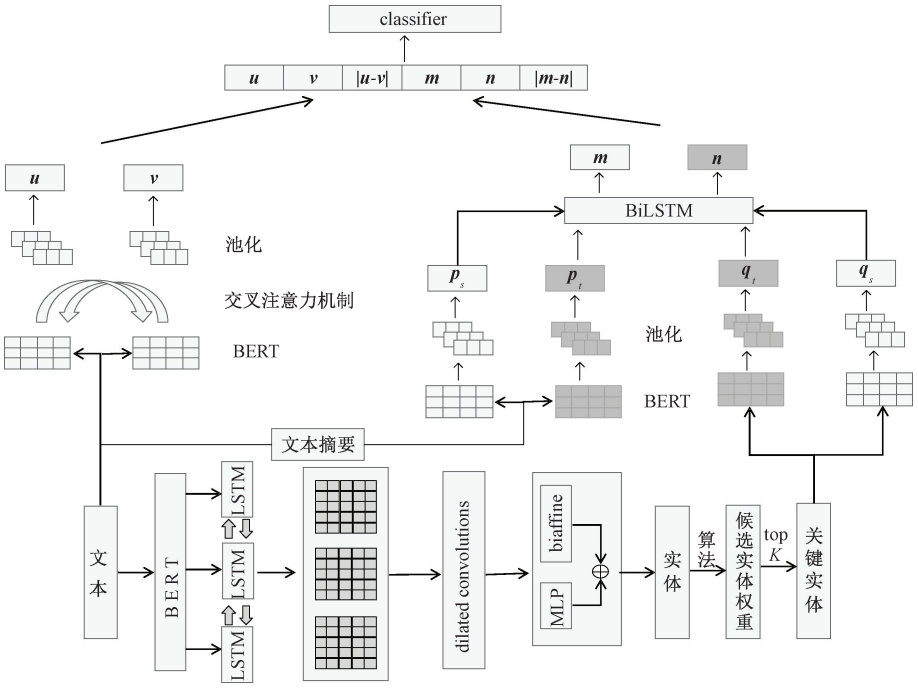

图 1　整体架构图

Figure 1　Overall frame diagram

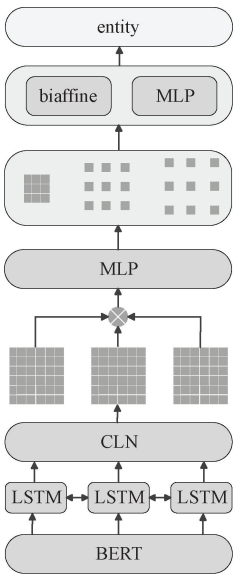

图 2　实体生成器

Figure 2　Entity builder

有关系,不属于同一个实体;*NNW* 表示这 2 个字是在同一个实体中相邻的位置;*THW*-* 表示这 2 个字在同一实体中,且分别是实体的结尾和开始。给定向量表示 $\boldsymbol{h}$,使用 2 个 MLP 来分别计算词对 (x_i,x_j) 的向量表示 $\boldsymbol{s}_i$ 和 $\boldsymbol{o}_j$。然后,使用双仿射分类器来计算词对 (x_i,x_j) 之间的关系分数,如下所示:

$$\boldsymbol{s}_i = \mathrm{MLP}(\boldsymbol{h}_i); \tag{5}$$

$$\boldsymbol{o}_j = \mathrm{MLP}(\boldsymbol{h}_j); \tag{6}$$

$$y'_{ij} = \boldsymbol{s}_i^{\mathrm{T}} U \boldsymbol{o}_j + W[\boldsymbol{s}_i;\boldsymbol{o}_j] + b。 \tag{7}$$

式中:U、W 和 b 为可训练参数;$\boldsymbol{s}_i$ 和 $\boldsymbol{o}_j$ 分别表示第 i 个和第 j 个单词的向量表示。

$R \in \{None, NNW, THW\text{-}*\}$,这里 y'_{ij} 为 R 中预定义的关系的得分。对 $\boldsymbol{Q}_{ij}$ 采用 MLP 计算词对 (x_i, x_j) 的关系得分,如式(8)所示:

$$y''_{ij} = \mathrm{MLP}(\boldsymbol{Q}_{ij})。 \tag{8}$$

通过组合 biaffine 和 MLP 预测器的得分计算词对 (x_i,x_j) 的最终关系概率 y_{ij},如式(9)所示:

$$y_{ij} = \mathrm{softmax}(y'_{ij} + y''_{ij})。 \tag{9}$$

W^2NER 模型预测字和字之间的关系,相当于一个有向的词图。解码的目的是根据确定的路径找到字和字的 NNW 关系,得到预测的实体及其类型,最后得到实体集合 $E = (e_1, e_2, \cdots, e_o)$。

3.1.2　关键实体筛选

将实体词的 TF-IDF 关键性、词出现的频率、词在词的集合中是否合群、词词之间的相似度和词句之间的相似度作为关键实体的筛选特征。

(1)TF-IDF 关键性因子。TF-IDF 是一种统计方法,用以评估一个词对于一个语料库中一份文件的重要程度,常用于关键词检索。TF-IDF 关键性因子权重如式(10)所示:

$$w_i^1 = w_{\mathrm{TF-IDF}}。 \tag{10}$$

(2)词频因子。统计实体集中各个实体词出现的频率,为不同词频的实体赋予不同的权重,如式(11)所示:

$$w_i^2 = 0.1 \cdot n。 \tag{11}$$

式中:n 为实体出现的次数。

(3)词的合群性因子。在数据集上采用 jieba 分词训练 word2vec 词向量模型[21]。通过 word2vec 查找不同类词的方法找出不合群的实体,并为其赋不同的权重。式(12)为不合群词权重,式(13)为合群词权重:

$$w_i^3 = 0.1; \tag{12}$$

$$w_i^3 = 0.3。 \tag{13}$$

(4)词词相似度因子。通过 word2vec 词向量模型编码各实体,以实体为节点,相邻节点的边权重为向量相似度,计算当前词与剩余词的相似性。统计每个实体词的边权重之和,作为此实体的词词相似度,对每个实体的词词相似度进行归一化,如式(14)所示。

$$w_i^4 = \frac{\sum_{j=1}^{n} \cos(\boldsymbol{en}_i, \boldsymbol{en}_j)}{\sum_{k=1}^{n}\sum_{g=1}^{n} \cos(\boldsymbol{en}_k, \boldsymbol{en}_g)}。 \tag{14}$$

式中:$\boldsymbol{en}_i$ 代表第 i 个实体的向量;cos(·)代表余弦相似度的计算;n 为实体个数。

(5)词句相似度因子。通过 word2vec 模型编码各实体和句子,计算其相似度,作为词句相似度因子,其中 $\boldsymbol{sen}$ 代表通过 word2vec 编码的句向量,如式(15)所示:

$$w_i^5 = \cos(\boldsymbol{en}_i, \boldsymbol{sen})。 \tag{15}$$

最后,组合特征权重如式(16)所示:

$$CF_i = w_i^1 + w_i^2 + w_i^3 + w_i^4 + w_i^5。 \tag{16}$$

根据实体集中各个实体的组合特征权重的不同进行排序,取前两个作为关键实体,得到关键实体集合 $K = (k_1, k_2)$。

3.1.3 文本摘要生成器

本文采用 IDEA 研究院 CCNL 提出的基于中文数据集悟道语料库(180 GB 版本)预训练的中文 Pegasus 模型 large 版本[22],对新闻文本进行文本概括,得到新闻准确、简洁的信息。

基于摘要提取的目的,Pegasus 模型首先对于文本中的重要句子进行选取,受词和连续 span mask 的启发,Pegasus 模型选择了遮蔽这些重要句子即间隔句,并且拼接它们形成伪摘要,相应位置遮蔽掉的间隔句用[MASK]来替代。然后通过 decode 恢复这些遮蔽掉的间隔句,加深模型对于文本语义的理解,达到生成文本摘要的目的。

3.1.4 基于 BiLSTM 的深层次语义特征融合

本文设计了一个基于 BiLSTM 的特征提取网络如图 3 所示,对文本摘要特征和关键实体特征进行特征融合,BiLSTM 可以捕获上下文信息,提取更深层的语义特征[23]。

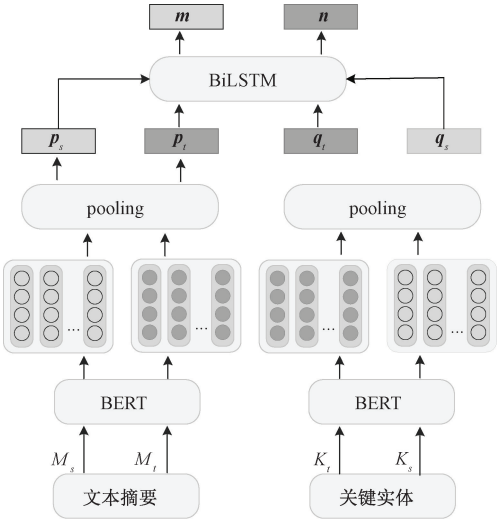

图 3 基于 BiLSTM 的特征提取网络

Figure 3 Feature extraction network based on BiLSTM

该网络是一个并行网络,通过文本摘要提取的新闻文本 s、t 得到摘要 M_s、M_t,通过预训练语言模型 BERT 进行编码,得到摘要中的每个字符的向量表示。然后通过平均池化,分别得到文本摘要特征向量 $\boldsymbol{p}_s$、$\boldsymbol{p}_t$。同样,通过实体提取和关键实体筛选的新闻文本 s、t 得到的关键实体集 K_s、K_t,通过预训练语言模型 BERT 进行编码,得到 K_s、K_t 中的每个字符的向量表示。然后通过平均池化分别得到关键实体特征向量 $\boldsymbol{q}_s$、$\boldsymbol{q}_t$。

接下来,将源新闻文本的文本摘要特征向量 $\boldsymbol{p}_s$ 和源新闻文本的关键实体特征向量 $\boldsymbol{q}_s$ 进行拼接,再通过 BiLSTM 进行特征融合,得到源新闻文本的深层次语义特征向量 $\boldsymbol{m}$。同样,将目标新闻文本的文本摘要特征向量 $\boldsymbol{p}_t$ 和目标新闻文本的关键实体特征向量 $\boldsymbol{q}_t$ 进行拼接,再通过 BiLSTM 进行特征融合,得到目标新闻文本的深层次语义特征向量 $\boldsymbol{n}$。其中,BiLSTM 是双向 LSTM,分别对序列进行正向和反向处理,获取上下文的联系,得到全局特征。

3.2 基于 cross-attention 的文本交互

在这一部分,对源新闻文本和目标新闻文本进行交互,增进彼此的联系,更好地获取待匹配新闻文本之间的差异,如图 4 所示。

首先,对源新闻文本 s 通过预训练语言模型 BERT,得到 s 的特征向量 $\boldsymbol{P}_s$,即 BERT 模型的最后一层输出。同样,将目标新闻文本 t 通过预训练语言模型 BERT,得到 t 的特征向量 $\boldsymbol{P}_t$。虽然 BERT 可以有效地编码语义信息,但是 s 和 t 之间的交互信息没有被探索,缺少彼此的交互和联系。因此,利用交

叉注意模块来增进文本之间的跨序列交互。与自注意力机制不同,交叉注意力机制的输入来自具有相同维度的不同序列,查询来自一个序列,而键和值来自另一个序列。

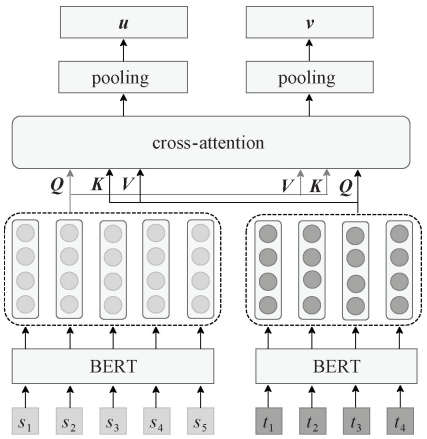

图 4　基于 cross-attention 的文本交互

Figure 4　Text interaction based on cross-attention

具体地,首先将通过 BERT 得到的向量 $\boldsymbol{P}_s$、$\boldsymbol{P}_t$,送入 cross-attention 的输入中,用于计算查询、键和值,将它们分别打包成矩阵 $\boldsymbol{Q}$、$\boldsymbol{K}$、$\boldsymbol{V}$。

$\boldsymbol{Q}$ 和 $\boldsymbol{K}$ 之间的点积的相似性决定了 $\boldsymbol{V}$ 的注意分布。m 个头部的多头注意函数具有 m 个并行的自注意函数。对于第 i 个头部,输入的 $\boldsymbol{Q}$、$\boldsymbol{K}$、$\boldsymbol{V}$ 转换如下:

$$\boldsymbol{Q}_{s_i} = \boldsymbol{P}_s \boldsymbol{W}_i^{\boldsymbol{Q}}; \tag{17}$$

$$\boldsymbol{K}_{s_i} = \boldsymbol{P}_s \boldsymbol{W}_i^{\boldsymbol{K}}; \tag{18}$$

$$\boldsymbol{V}_{s_i} = \boldsymbol{P}_s \boldsymbol{W}_i^{\boldsymbol{V}}; \tag{19}$$

$$\boldsymbol{Q}_{t_i} = \boldsymbol{P}_t \boldsymbol{W}_i^{\boldsymbol{Q}}; \tag{20}$$

$$\boldsymbol{K}_{t_i} = \boldsymbol{P}_t \boldsymbol{W}_i^{\boldsymbol{K}}; \tag{21}$$

$$\boldsymbol{V}_{t_i} = \boldsymbol{P}_t \boldsymbol{W}_i^{\boldsymbol{V}}。 \tag{22}$$

式中:$\boldsymbol{W}_i^{\boldsymbol{Q}}$、$\boldsymbol{W}_i^{\boldsymbol{K}}$、$\boldsymbol{W}_i^{\boldsymbol{V}}$ 表示第 i 个头部对应的 3 个权重矩阵。

多头自注意函数的计算过程表示如下:

$$\boldsymbol{h}_{s_i} = A(\boldsymbol{Q}_{s_i}, \boldsymbol{K}_{t_i}, \boldsymbol{V}_{t_i}) = \mathrm{softmax}\left(\frac{\boldsymbol{Q}_{s_i}\boldsymbol{K}_{t_i}^{\mathrm{T}}}{\sqrt{d_{\mathrm{h}}}}\right)\boldsymbol{V}_{t_i}; \tag{23}$$

$$\boldsymbol{h}_{t_i} = A(\boldsymbol{Q}_{t_i}, \boldsymbol{K}_{s_i}, \boldsymbol{V}_{s_i}) = \mathrm{softmax}\left(\frac{\boldsymbol{Q}_{t_i}\boldsymbol{K}_{s_i}^{\mathrm{T}}}{\sqrt{d_{\mathrm{h}}}}\right)\boldsymbol{V}_{s_i}。 \tag{24}$$

式中:d_{h} 代表每个头部的输出特征的维数。

进一步,将每个头部得到的特征向量拼接,与权重矩阵 $\boldsymbol{W}^o$ 计算后压缩成一个矩阵,m 代表总共的头数,公式如下:

$$\boldsymbol{h}_s = \mathrm{concat}(\boldsymbol{h}_{s_1}, \boldsymbol{h}_{s_2}, \cdots, \boldsymbol{h}_{s_m}); \tag{25}$$

$$\boldsymbol{h}_t = \mathrm{concat}(\boldsymbol{h}_{t_1}, \boldsymbol{h}_{t_2}, \cdots, \boldsymbol{h}_{t_m}); \tag{26}$$

$$\mathrm{MA}(\boldsymbol{Q}_s, \boldsymbol{K}_t, \boldsymbol{V}_t) = \boldsymbol{h}_s \boldsymbol{W}_s^o; \tag{27}$$

$$\mathrm{MA}(\boldsymbol{Q}_t, \boldsymbol{K}_s, \boldsymbol{V}_s) = \boldsymbol{h}_t \boldsymbol{W}_t^o。 \tag{28}$$

再通过由 2 个全连接层和一个 ReLU 激活函数组成的 FFN 层,在各个时序上对特征进行非线性变换,提高网络表达能力。

通过 cross-attention,可以得到新闻文本 s 和 t 各个字的交互向量表达,然后通过平均池化操作,得到新闻文本 s 和 t 的交互特征向量 $\boldsymbol{u}$、$\boldsymbol{v}$。

3.3　融合文本交互特征和深层次语义特征的匹配

基于 cross-attention 的文本交互模块得到了源新闻文本的交互特征 $\boldsymbol{u}$、目标新闻文本的交互特征 $\boldsymbol{v}$,而后用 $|\boldsymbol{u}-\boldsymbol{v}|$ 得到源新闻-目标新闻文本交互特征的差异,$|\boldsymbol{u}-\boldsymbol{v}|$ 为特征向量按位相减并取绝对值的操作。深层次语义特征提取模块得到了源新闻文本的深层次语义特征 $\boldsymbol{m}$、目标新闻文本的深层次语义特征 $\boldsymbol{n}$,而后用特征向量 $|\boldsymbol{m}-\boldsymbol{n}|$ 得到源新闻-目标新闻文本深层次语义特征的差异。将文本交互特征向量及其差异和深层次语义特征向量及其差异拼接,得到融合的特征向量 $(\boldsymbol{u},\boldsymbol{v},|\boldsymbol{u}-\boldsymbol{v}|,\boldsymbol{m},\boldsymbol{n},|\boldsymbol{m}-\boldsymbol{n}|)$。然后,将向量 $(\boldsymbol{u},\boldsymbol{v},|\boldsymbol{u}-\boldsymbol{v}|,\boldsymbol{m},\boldsymbol{n},|\boldsymbol{m}-\boldsymbol{n}|)$ 通过 BN 层进行归一化,通过 ReLU 增加神经网络各层之间的非线性关系,最后通过全连接层进行降维,预测匹配的结果。

4　实验

4.1　数据集

本文的数据集来源于 2021 搜狐校园文本匹配算法大赛话题匹配的真实数据集[24]。该数据集包含源新闻文本、目标新闻文本和标签,若 2 段新闻话题相同或相似,标注为 1,否则标注为 0。

该数据集包括短文本和短文本的匹配、短文本和长文本的匹配、长文本和长文本的匹配 3 部分,数据统计结果如表 1 所示。其中短文本为 100 以内的文本,长文本为 200 字以上的文本。实验将 3 个数据集的训练集和验证集合并,训练一个话题匹配模型,分别在 3 个测试集上测试,实验结果证明模型在不同长度的文本上均有效果。

4.2　基线方法

本节将本文提出的算法和以下 6 个基准的文本匹配算法进行性能比较,其中 ABCNN 和 SimLSTM 模型采用的 word2vec 词向量是使用搜狐数据集进行无监督训练得到的。

ABCNN:交互型文本匹配模型,采用 CNN 抽取上下文信息,在输入层和卷积的输出层添加注意力机制对序列进行交互,进而得到文本匹配结果。

SimLSTM:表示型文本匹配模型,通过2个LSTM网络得到句子表征向量,用全连接层构成的分类层得到匹配结果。

BERT:预训练语言模型,BERT采用基于自注意力机制的Transformer,将源新闻、目标新闻文本通过BERT进行微调,实现文本匹配任务。

SBERT[25]:SBERT模型采用孪生网络结构,分别对源文本和目标文本输入BERT网络,输出2组表征句子语义的向量 $\boldsymbol{u}$、$\boldsymbol{v}$,拼接向量 $\boldsymbol{u}$、$\boldsymbol{v}$、$|\boldsymbol{u}-\boldsymbol{v}|$,预测文本匹配结果。

Erine3.0[26]:基于Ernie、Ernie2.0,百度不断增大语料库,并且融合知识图谱进行知识强化的预训练任务,得到百度的预训练语言模型Erine3.0。通过微调,可进行文本语义匹配任务。

表1 实验中使用的搜狐数据集

Table 1 Sohu dataset used in the experiment

文本类别	数据集	匹配实例数	不匹配实例数	总计实例数
短文本和短文本匹配	训练集	2 773	7 094	9 867
	验证集	485	1 160	1 645
	测试集	2 471	2 463	4 934
短文本和长文本匹配	训练集	3 565	6 371	9 936
	验证集	616	1 040	1 656
	测试集	2 506	2 463	4 969
长文本和长文本匹配	训练集	5 538	4 482	10 020
	验证集	766	904	1 670
	测试集	2 576	2 434	5 010

4.3 评价指标

实验部分综合利用准确率 Acc、精确率 P、召回率 R、$F1$ 指标来评价算法,如下所示:

$$Acc=\frac{TP+TN}{TP+TN+FP+FN};\tag{29}$$

$$P=\frac{TP}{TP+FP};\tag{30}$$

$$R=\frac{TP}{TP+FN};\tag{31}$$

$$F1=\frac{2\times P\times R}{P+R}。\tag{32}$$

本文的实验环境如下:Intel(R) Xeon(R) Platinum 8255C CPU@2.50 GHz+24 GB内存,深度学习框架为Anaconda Python3.0+PyTorch 1.8.1。

4.4 参数设置

在关键实体提取模块,根据实验发现,关键实体个数设置为2时效果最好。在命名实体识别模块,使用BERT获得768维的词向量,学习率设置为 10^{-3},批样本数设置为8,采用Adam优化器,可以获得更好的收敛效果。在话题匹配模块,根据统计,新闻文本平均输入长度为260,文本摘要最大输入长度为100,关键实体最大输入长度为20,以此设置相关参数。同时,学习率设置为 2×10^{-5},批样本数设置为16,迭代次数为2,采用Adam优化器,可以获得较好的收敛效果,但模型容易过拟合,因此采用dropout层解决模型过拟合问题,在训练过程中每200步保存一次模型,保留效果最好的模型。

图5展示了关键实体筛选模块的关键实体个数对实验结果的影响。其中,$k=1$、$k=2$、$k=3$、$k=4$,分别代表了将按照组合特征权重排好序的实体,选取前1、2、3、4个作为关键实体;$k=all$ 代表把文本的所有实体都参与深层次语义特征的提取。综合来看,$k=2$ 时,指标 R、$F1$ 在短短匹配、短长匹配、长长匹配数据集上均为最高,指标 Acc 和 P 也表现较好。因此,选取前2个实体作为关键实体。当 $k=all$ 时,效果较差,同时也证明了关键实体提取模块的有效性。

4.5 对比实验

为了验证本文方法的有效性,选取了表示型、交互型、预训练语言模型3种类型的文本匹配方法进行对比实验。表2展示了所有算法在真实数据集上取得的实验结果。其中,ABCNN模型为交互型文本匹配模型,采用CNN可较好地提取文本局部信息,捕捉文本细节,通过注意力机制增进两句话之间的交互,但是过分注重文本细节的匹配,不利于文本话题匹配。SimLSTM模型为表示型文本匹配模型,仅在匹配层获取向量差异,缺少语义之间的交互,编码句子语义时损失较大。BERT、Ernie3.0模型采用大规模语料库预训练,SBERT模型采用BERT进行语义表示,但是这3个模型忽视文本话题的概括,在话题匹配方面效果不好。

总体上看,本文提出的算法各项指标比较均衡,在3个测试集中,召回率和 $F1$ 都取得最好的效果,在准确率、精确率方面也与其他模型的最优效果相近。因此,证明了本文提出的基于关键实体和文本摘要多特征融合的话题匹配算法的有效性。

4.6 消融实验

为了验证本文方法中各模块的有效性,移除了模型中特定的部分,进行了消融实验。表3展示了消融实验结果,其中,M_1 为移除了深层次语义特征提取模块;M_2 为移除了基于BiLSTM的深层次语义

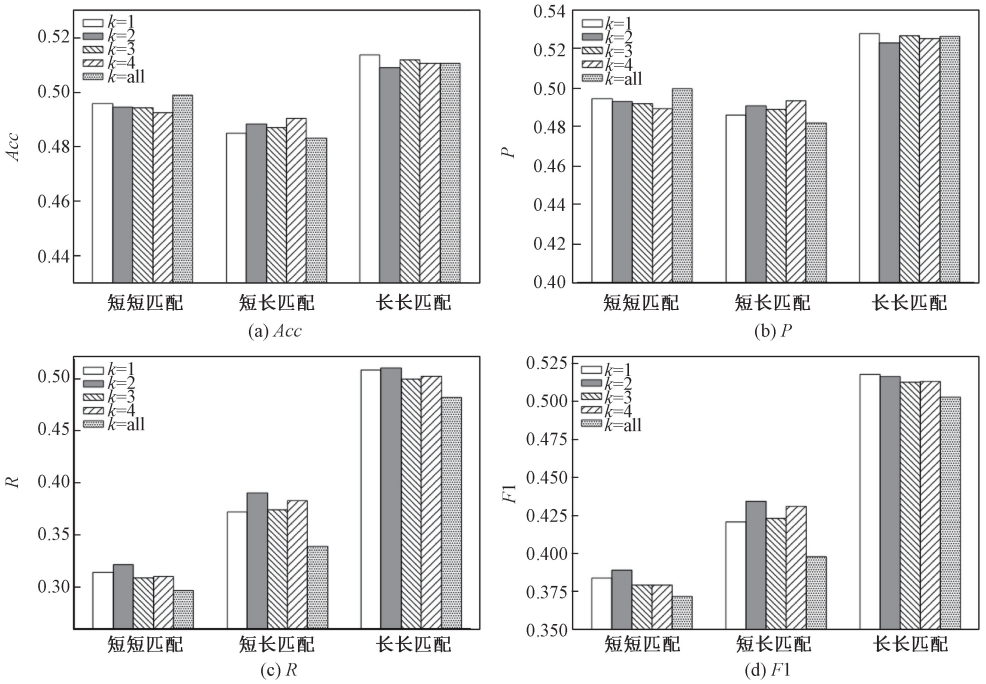

图 5　不同关键实体个数的性能变化

Figure 5　Performance changes for different key entities

特征融合模块,即将文本摘要特征、关键实体特征直接拼接参与话题匹配。M_3 为移除了交叉注意力模块,用 BERT 最后一层的[CLS]向量作为句子语义向量,进行匹配。

从消融实验结果可以看出,各模块对准确率和精确率影响较小,但对召回率和 $F1$ 值有较大的影响。去除了深层次语义特征提取模块后,各指标明显降低。去除 BiLSTM 提取关键实体和文本摘要的全局特征,在不同数据集中,召回率和 $F1$ 也有所降低。去除 cross-attention 文本语义特征交互模块后,短文本和短文本的匹配以及长文本和长文本的匹配中指标均下降,是由于待匹配文本长度相近时,通过 BERT 模型得到的语义特征差距较小,添加交叉注意力机制进行交互,可以更好地关注待匹配文本的语义信息,更好地捕获语义特征的差异,提升话题匹配效果。然而,当待匹配文本长度相差过大时,文本语义特征相差较大,本文模型添加交叉注意力机制后仅在 R 和 $F1$ 指标上有提升。

表 2　对比实验结果

Table 2　Compare experimental results

算法	短文本和短文本的匹配				短文本和长文本的匹配				长文本和长文本的匹配			
	Acc	*P*	*R*	*F*1	*Acc*	*P*	*R*	*F*1	*Acc*	*P*	*R*	*F*1
ABCNN	0.489	0.471	0.172	0.252	**0.498**	**0.504**	0.313	0.386	0.504	0.520	0.472	0.495
SimLSTM	0.499	0.500	0.267	0.348	0.481	0.479	0.328	0.389	0.507	0.522	0.486	0.503
BERT	0.497	0.496	0.285	0.362	0.487	0.487	0.328	0.392	**0.509**	**0.527**	0.453	0.487
SBERT	0.493	0.490	0.301	0.373	0.488	0.488	0.336	0.398	0.505	0.519	0.499	0.509
Ernie3.0	**0.500**	**0.501**	0.282	0.361	0.489	0.490	0.332	0.396	0.507	0.525	0.426	0.470
本文算法	0.495	0.493	**0.321**	**0.389**	0.488	0.491	**0.390**	**0.435**	**0.509**	0.523	**0.510**	**0.516**

表 3　消融实验结果

Table 3　Ablation experimental results

算法	短文本和短文本的匹配				短文本和长文本的匹配				长文本和长文本的匹配			
	Acc	*P*	*R*	*F*1	*Acc*	*P*	*R*	*F*1	*Acc*	*P*	*R*	*F*1
本文算法	**0.495**	**0.493**	**0.321**	**0.389**	0.488	0.491	**0.390**	**0.435**	**0.509**	**0.523**	**0.510**	**0.516**
M_1	0.493	0.492	0.268	0.347	0.482	0.479	0.309	0.376	0.502	0.518	0.449	0.481
M_2	0.494	0.490	0.294	0.368	0.487	0.488	0.339	0.400	0.507	0.523	0.465	0.492
M_3	0.489	0.484	0.308	0.376	**0.493**	**0.496**	0.350	0.410	0.508	**0.523**	0.480	0.501

5 结论

本文提出了基于关键实体和文本摘要多特征融合的话题匹配算法,通过提取文本的关键实体和摘要,获得文本的深层次语义特征,更好地概括新闻文本话题,再通过交叉注意力机制获得文本之间的交互特征,增进文本向量之间的联系,使得文本深层次语义特征和文本交互特征共同作用于文本话题匹配结果。在真实数据集上进行的实验表明,本文的方法要优于目前流行的深度学习文本匹配算法,对文本话题匹配有较好的检测效果。

后续工作可以通过关系抽取或事件抽取的方法进一步提取文本深层次语义特征,提升文本话题匹配结果。目前公开的话题匹配数据集不多,可以制作更大的数据集来进行进一步实验。

参考文献:

[1] MALA V, LOBIYAL D K. Semantic and keyword based web techniques in information retrieval[C]//2016 International Conference on Computing, Communication and Automation (ICCCA). Piscataway: IEEE, 2017: 23-26.

[2] 陈宁. 基于网络的关键词检索技巧[J]. 中国科技信息, 2008(2):115-115, 117.
CHEN N. Key words retrieval skills based on network [J]. China Science and Technology Information, 2008 (2):115-115, 117.

[3] COHEN W W, RAVIKUMAR P, FIENBERG S. A comparison of string distance metrics for name-matching tasks [C]// Proceedings of the 2003 International Conference on Information Integration on the Web. New York:ACM, 2003:73-78.

[4] 庞亮, 兰艳艳, 徐君, 等. 深度文本匹配综述[J]. 计算机学报, 2017, 40(4): 985-1003.
PANG L, LAN Y Y, XU J, et al. A survey on deep text matching[J]. Chinese Journal of Computers, 2017, 40 (4): 985-1003.

[5] LIU J, KONG X, ZHOU X, et al. Data mining and information retrieval in the 21st century: a bibliographic review[J]. Computer Science Review, 2019, 34: 100193.

[6] ARORA S, BATRA K, SINGH S. Dialogue system: a brief review[EB/OL]. (2013-6-18)[2023-06-15]. https://arxiv.org/abs/1306.4134.

[7] MUELLER J, THYAGARAJAN A. Siamese recurrent architectures for learning sentence similarity[C]//Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence. New York: ACM, 2016: 2786-2792.

[8] YIN W P, SCHÜTZE H, XIANG B, et al. ABCNN: attention-based convolutional neural network for modeling sentence pairs[J]. Transactions of the Association for Computational Linguistics, 2016, 4: 259-272.

[9] DEVLIN J, CHANG M W, LEE K, et al. BERT: pre-training of deep bidirectional transformers for language understanding[EB/OL]. (2018-10-11)[2023-06-15]. https://arxiv.org/abs/1810.04805.

[10] LIU Y H, OTT M, GOYAL N, et al. RoBERTa: a robustly optimized BERT pretraining approach[EB/OL]. (2019-7-26)[2023-06-15]. https://arxiv.org/abs/1907.11692.

[11] WEI J Q, REN X Z, LI X G, et al. NEZHA: neural contextualized representation for Chinese language understanding[EB/OL]. (2019-8-31)[2023-06-15]. https://arxiv.org/abs/1909.00204.pdf.

[12] PEINELT N, NGUYEN D, LIAKATA M. TBERT: topic models and BERT joining forces for semantic similarity detection[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: Association for Computational Linguistics, 2020: 7047-7055.

[13] 周澳回, 翁知远, 周思源, 等. 一种基于主题过滤和语义匹配的服务发现方法[J]. 郑州大学学报(工学版), 2022, 43(6):36-41, 56.
ZHOU A H, WENG Z Y, ZHOU S Y, et al. A service discovery method based on topic filtering and semantic matching[J]. Journal of Zhengzhou University (Engineering Science), 2022, 43(6):36-41, 56.

[14] MIAO C Y, CAO Z, TAM Y C. Keyword-attentive deep semantic matching[EB/OL]. (2020-05-11)[2023-06-15]. https://arxiv.org/abs/2003.11516.

[15] ZOU Y C, LIU H W, GUI T, et al. Divide and conquer: text semantic matching with disentangled keywords and intents[EB/OL]. (2022-05-6)[2023-06-15]. https://arxiv.org/abs/2203.02898.

[16] HUANG Z H, XU W, YU K. Bidirectional LSTM-CRF models for sequence tagging[EB/OL]. (2015-08-09)[2023-06-15]. https://arxiv.org/abs/1508.01991.pdf.

[17] LI J Y, FEI H, LIU J, et al. Unified named entity recognition as word-word relation classification[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2022, 36(10): 10965-10973.

[18] LIU Y. Fine-tune BERT for extractive summarization [EB/OL]. (2019-05-25)[2023-06-15]. https://arxiv.org/abs/1903.10318.

[19] ZHANG J Q, ZHAO Y, SALEH M, et al. PEGASUS: pre-training with extracted gap-sentences for abstractive summarization[EB/OL]. (2019-12-18)[2023-06-15]. https://arxiv.org/abs/1912.08777.

[20] YU Y, SI X, HU C, et al. A review of recurrent neural

networks: LSTM cells and network architectures [J]. Neural computation, 2019, 31(7): 1235-1270.

[21] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient estimation of word representations in vector space [EB/OL]. (2013-01-16)[2023-06-15]. https://arxiv.org/abs/1301.3781.

[22] ZHANG J X, GAN R Y, WANG J J, et al. Fengshenbang 1.0: being the foundation of Chinese cognitive intelligence[EB/OL]. (2022-09-07)[2023-06-15]. https://arxiv.org/abs/2209.02970.

[23] 李勇, 金庆雨, 张青川. 融合位置注意力机制和改进 BLSTM 的食品评论情感分析[J]. 郑州大学学报(工学版), 2020, 41(1):58-62.
LI Y, JIN Q Y, ZHANG Q C. Improved BLSTM food review sentiment analysis with positional attention mechanisms[J]. Journal of Zhengzhou University (Engineering Science), 2020, 41(1):58-62.

[24] 搜狐.2021 搜狐校园文本匹配算法大赛[EB/OL]. (2021-03-29)[2023-06-15]. https://www.biendata.xyz/competition/sohu_2021/.
Sohu. 2021 Sohu campus text matching algorithm competition[EB/OL]. (2021-03-29)[2023-06-15]. https://www.biendata.xyz/competition/sohu_2021/.

[25] REIMERS N, GUREVYCH I. Sentence-BERT: sentence embeddings using Siamese BERT-networks [EB/OL]. (2019-8-27)[2023-06-15]. https://arxiv.org/abs/1908.10084.

[26] SUN Y, WANG S H, FENG S K, et al. ERNIE 3.0: large-scale knowledge enhanced pre-training for language understanding and generation[EB/OL]. (2021-07-05)[2023-06-15]. https://arxiv.org/abs/2107.02137.

Topic Matching Algorithm Based on Multi-feature Fusion of Key Entities and Text Abstracts

JI Ke[1,2], ZHANG Xiu[1,2], MA Kun[1,2], SUN Runyuan[1,2], CHEN Zhenxiang [1,2], WU Jun[3]

(1. School of Information Science and Engineering, University of Jinan, Jinan 250022, China; 2. Shandong Provincial Key Laboratory of Network Based Intelligent Computing, University of Jinan, Jinan 250022, China; 3. School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China)

Abstract: With the rapid popularization of the Internet, the amount of Internet news has increased dramatically. In this case, how to effectively find relevant reports that are more in line with a specific topic has become an urgent problem to be solved. To address this issue, a topic matching algorithm based on the fusion of key entities and text abstracts was proposed in this study. Firstly, the W^2NER model was used for named entity recognition to extract key entities using features such as word frequency, TF-IDF, lexical cohesion word-word similarity, and word-sentence similarity. Secondly, the Pegasus model was used for text summarization, and the deep semantic features of news texts were obtained by combining the key entity features with the text summary features using BiLSTM. Next, the cross-attention mechanism was employed to enhance the interaction between the matching news articles by performing feature interaction. Finally, the deep semantic features of the news texts and the text interaction features were fused together to participate in the determination of text topic matching. Comparative experiments were conducted on real data from Sohu, and the results showed that the proposed algorithm achieved similar accuracy and precision compared to other algorithms, while recall and $F1$ score were improved.

Keywords: topic matching; key entity; text summary; text matching; information retrieval