

文章编号: 1003-0077(2020)06-0081-08

面向多类型问题的阅读理解方法研究

谭红叶^{1,2}, 屈保兴¹

(1. 山西大学 计算机与信息技术学院, 山西 太原 030006;
2. 山西大学 计算智能与中文信息处理教育部重点实验室, 山西 太原 030006)

摘要: 机器阅读理解是基于给定文本, 自动回答与文本内容相关的问题。针对此任务, 学术界与工业界提出多个数据集与模型, 促使阅读理解取得了一定的进步, 但提出的模型大多只是针对某一类问题, 不能满足现实世界问题多样性的需求。因此, 该文针对阅读理解中问题类型多样性的解答展开研究, 提出一种基于 Bert 的多任务阅读理解模型, 利用注意力机制获得丰富的问题与篇章的表示, 并对问题进行分类, 然后将分类结果用于任务解答, 实现问题的多样性解答。该文在中文公共阅读理解数据集 CAIL2019-CJRC 上对所提模型进行了实验, 结果表明, 系统取得了比所有基线模型都要好的效果。

关键词: 阅读理解; 分类; 注意力机制; 多类型问题
中图分类号: TP391 **文献标识码:** A

An Approach to Multi-Type Question Machine Reading Comprehension

TAN Hongye^{1,2}, QU Baoxing¹

(1. School of Computer and Information Technology, Shanxi University, Taiyuan, Shanxi 030006, China;
2. Key Laboratory of Intelligence and Chinese Information, Ministry of Education
Processing, Shanxi University, Taiyuan, Shanxi 030006, China)

Abstract: Machine reading comprehension (MRC) enables the machine read a given passage and then answer some relevant questions. A number of data sets and models have been proposed for a specific type of problems, without dealing with the diversity of problems in real-world. In this paper, we propose a multi-task reading comprehension model based on Bert. It uses the attention mechanism to obtain multi representations of questions and passages and then classify the questions. Then the model utilizes the classification results to answer the various questions. Experiments on Chinese public machine reading comprehension dataset CAIL2019-CJRC show that our system achieves better results than all the baseline models.

Keywords: reading comprehension; classification; attention mechanism; multi-type questions

0 引言

机器阅读理解(machine reading comprehension)作为问答社区中一个新的任务, 是人工智能核心问题之一。其目的是让机器阅读给定文本, 然后回答相关问题。该任务需要对文本进行全面理解, 并具备推理和分析能力, 才能准确回答相关问题。随着越来越多的阅读理解与问答数据集的发

布, 各种深度学习方法相继被提出, 如 BiDAF^[1] 模型、Match-LSTM^[2] 模型, 以及近期火热的预训练模型 Bert^[3] 等, 这极大地促进了阅读理解的进步。但是到目前为止, 大多数模型只针对抽取型问题进行解答, 不能满足现实世界中问题多样性的需求。分析相关阅读理解数据集发现, 问题一般有三种类型, 分别为 Unknown 型、Yes/No 型、Span 型。其中, Unknown 型问题是指不能基于给定文本回答的问题, 即问题答案为空; Yes/No 型问题也称作是

否型或验证型问题,这种问题的答案为“是”或者“否”;Span 型问题是指需要从阅读文本中抽取答案的问题,答案可以是一个实体、一个片段或者是多个片段的组合。表 1 给出了三类问题的具体示例。

表 1 各类问题示例表

问题类型	问题	答案
Unknown	沈 x4 拖欠多久未还款?	无
Yes/No	投保人是否将向第三者的追偿权完全转让原告?	YES
Span	未付保费如何计算?	已欠保费+当期应缴保费×当期实际承保天数/30

针对问题多样性,本文提出了一种基于 Bert^[3]的多任务阅读理解模型,该模型的主要思想是:首先对问题进行分类预测,然后将预测结果与注意力层捕获的问题与篇章之间的关系结合起来,进而对多样性问题进行解答。相关数据集上的实验表明,本文所提模型的性能得到了明显提升。

本文主要贡献有以下几点:

- (1) 对阅读理解中的问题进行分类(Unknown 型、Yes/No 型、Span 型);
- (2) 基于注意力机制获得问题与篇章交互表示,在此基础上进一步获得包含字重要程度的篇章表示;
- (3) 提出一种基于 Bert^[3]的多任务阅读理解模型,模型利用分类结果指导问题解答,实现了多类型问题的解答。

1 相关工作

1.1 主流数据集

近年来,工业界和学术界发布了多个机器阅读理解数据集,促进了机器阅读理解技术的进步。常见的任务形式主要有三种:①填空题,这种题型就是在原文中去除一个词,根据阅读上下文完成填词任务,主要代表数据集有 PD&.CFT^[4]、CNN/Daily Mail^[5];②选择题,给定一段文本,对应的每个问题会提供 4 个选项,Race2^[6]、MCTest^[7]就是这类数据的代表;③简答题,根据上下文,回答所提出的问题,这种题型最为复杂,也是最接近实际的,主要代

表数据集有 SQuAD^[8]、Cmrc2018^[9]、DuReader^[10]、CAIL2019-CJRC^① 机器阅读理解数据集。具体信息如表 2 所示。

表 2 主流阅读理解数据集

任务形式	代表数据集
填空题	People Daily & Children's Fairy Tale ^[4] ,CNN/Daily Mail ^[5]
选择题	Race2 ^[6] ,MCTest ^[7]
简答题	SQuAD ^[8] , Cmrc2018 ^[9] , DuReader ^[10] , CAIL2019-CJRC

1.2 主流方法

在上述数据集的支持下,很多基于深度学习的阅读理解方法被提出,这些方法主要分为以下两大类。

(1) 基于注意力机制的方法。由于注意力机制在多个不同领域的任务中取得非常好的效果,学者们也在阅读理解任务中使用多种注意力机制。例如,2015 年,Hermann 等人^[11]将注意力机制应用到阅读理解任务中;2016 年 Chen 等人^[12]设计了双线性函数注意力,在 CNN/Daily Mail^[5]数据集上取得了不错的效果;Cui 等人提出 AoA^[13]注意力机制,更好地将问题和篇章的表述结合起来;Seo 等人^[1]提出一种双向注意流模型来捕获问题篇章不同粒度的相关性;2017 年,Wang 等人^[14]提出 R-NET 模型,引入自注意力机制,通过对文章本身的匹配来细化表示,更好地捕捉全局信息;Chen 等人在 2018 年提出 CSA^[15]模型,通过卷积空间注意力机制来提取多种问题篇章表示之间的信息。

(2) 预训练模型。2018 年,谷歌提出 Bert^[3]模型,该模型是基于多层双向 Transformer^[16]编码器框架,利用 Masked LM 和 Next Sentence Prediction 两项预训练任务,在大数据集上进行训练得到的语言模型。使用时,一般需要在下游任务上进行微调。2019 年,百度提出 ERNIE^[17]模型,对预训练 Masked LM 任务进行改进,除了对字掩盖,还对词、实体等语义单元进行掩盖,使模型可以学习完整概念的语义表示。该模型在中文任务上全面超越 Bert。

① <https://github.com/china-ai-law-challenge/CAIL2019/tree/master/阅读理解>

2 方法

2.1 任务定义

典型的机器阅读理解系统以篇章和问题作为输入,然后在所给篇章范围内进行答案预测。该任务形式化定义为:给定篇章 $d = \{w_1^d, w_2^d, \dots, w_n^d\}$, 问题 $q = \{w_1^q, w_2^q, \dots, w_m^q\}$, 依据式(1)输出答案 $a \in A$ 。

$$f(q, d) = \arg \max_{a \in A} P(a | q, d) \quad (1)$$

其中, n 为篇章长度, m 为问题长度。对于答案 A , 根据问题类型不同,可以有不同表示。Unknown 型问题答案 $A = \text{None}$, Yes/No 型问题答案 $A = \{\text{Yes/No}\}$, Span 型问题答案 $A = \{w_i^d, w_{i+1}^d, \dots, w_j^d\}$,

从 d 中抽取。

2.2 模型框架

本文模型框架如图 1 所示,由编码层、问题分类层、注意力层和输出层 4 个部分组成,各层主要功能如下所示:

- (1) 编码层: 对输入的问题和篇章进行语义化表示;
- (2) 问题分类层: 对问题进行分类预测;
- (3) 注意力层: 进一步捕获问题与篇章之间的关系;
- (4) 解答层: 根据问题类型以及注意力层所捕获的问题篇章之间的关系,对答案进行多样性解答。

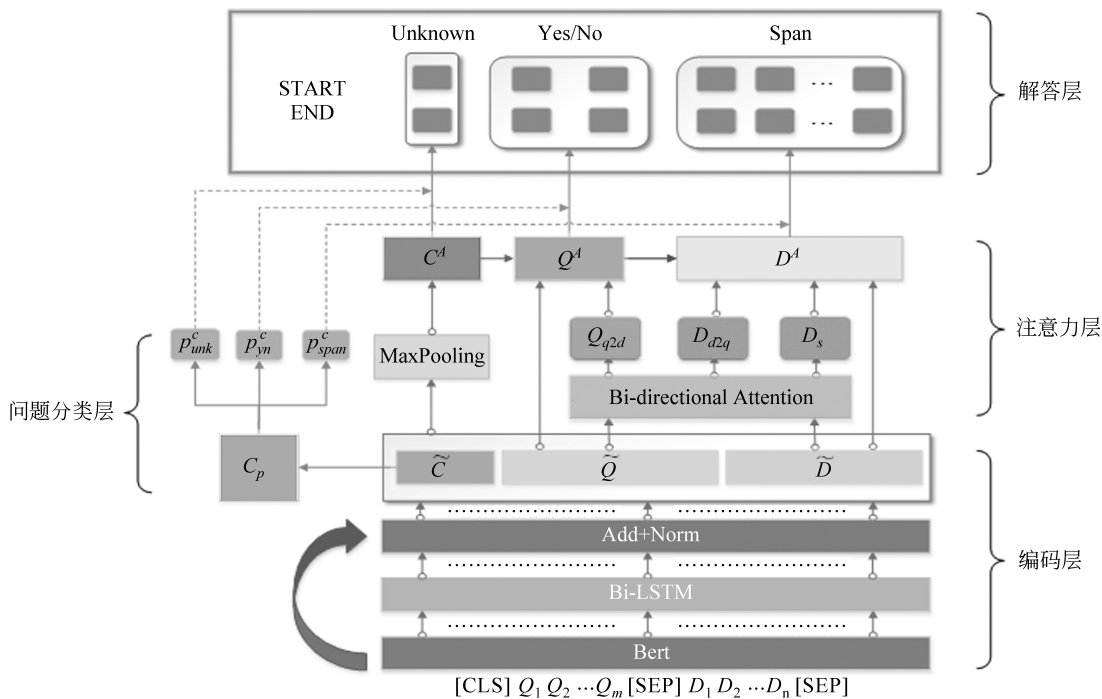


图 1 本文模型示意图

2.3 编码层

编码层将问题和篇章的离散符号转换为连续向量表示序列。这里我们利用 Bert、Bi-LSTM 以及 LayerNorm 这三层来实现编码层。

首先采用预训练模型 Bert 得到问题与篇章的语义化表示,具体如式(2)、式(3)所示。输入采用 $\text{input} = [\text{cls}] + Q + [\text{sep}] + D + [\text{sep}]$ 的方式,其中 $[\text{sep}]$ 为分隔符, $[\text{cls}]$ 为输入的聚合表示,用于分类任务。

$$U = \text{Bert}(\text{input}) \quad (2)$$

$$\{C, Q', D'\} = U \quad (3)$$

式(2)中, $U \in R^{l \times h}$, $l = m + n + 3$, 为输入长度, h 为向量维度。基于 U , 可以得到 $[\text{cls}]$ 、 Q 、 D 的向量化表示: $C = \{e_0^u\}$ 、 $Q' = \{e_1^u, e_2^u, \dots, e_m^u\}$ 、 $D' = \{e_{m+2}^u, e_{m+3}^u, \dots, e_{m+n+2}^u\}$; 其中, $e_i^u \in R^h$, $i \in [0, l-1]$ 。

为了进一步细化字词的表示,我们在 Bert 层之后加了一层共享的 Bi-LSTM 层,利用周围词汇的语境线索来学习字词间的相互关系,得到 U 的进一步表示 U' 。

接下来,将 U' 与 U 进行残差连接,并进行层归一化(LayerNorm^[18]),具体计算如式(4)~式(6)所示。

$$\tilde{U} = \text{LayerNorm}(U'') \quad (4)$$

$$U'' = U + U' \quad (5)$$

$$U' = \text{Bi-LSTM}(U) \quad (6)$$

其中, $\{\tilde{C}, \tilde{Q}, \tilde{P}\} = \tilde{U}$, 得到 $[\text{cls}]$ 、 \tilde{Q} 、 \tilde{D} 的在编码层的最终表示 $\tilde{C} = \{e_0^{\tilde{u}}, \tilde{Q} = \{e_1^{\tilde{u}}, e_2^{\tilde{u}}, \dots, e_m^{\tilde{u}}\}$ 、 $\tilde{D} = \{e_{m+2}^{\tilde{u}}, e_{m+3}^{\tilde{u}}, \dots, e_{m+n+2}^{\tilde{u}}\}$ 。

2.4 问题分类层

问题分类层就是根据所给问题和篇章, 对问题进行分类预测。本文将问题分为 Unknown 型、Yes/No 型和 Span 型三类。将 $[\text{cls}]$ 在编码层中得到的最终表示 \tilde{C} 放入全连接层 Dense, 得到表示 C_p , 然后进行线性分类。具体如式 (7) ~

式 (9) 所示。

$$C_p = \text{Dense}(\tilde{C}) \quad (7)$$

$$P^c = W_c C_p + b_c \quad (8)$$

$$\{p_{\text{unk}}^c, p_{\text{yn}}^c, p_{\text{span}}^c\} = P^c \quad (9)$$

其中, $C_p \in R^h$, W_c 为权重矩阵, b_c 为偏置, 且 $W_c \in R^{h \times 3}$, $b_c \in R^3$ 。 P^c 为预测的类别分布概率向量, p_{unk}^c 、 p_{yn}^c 、 p_{span}^c 为相应问题类别的概率。

2.5 注意力层

注意力层进一步捕获问题与篇章的关系, 支持问题回答。该层采用 bi-directional 注意力机制, 获得了问题与篇章的交互表示, 以及包含字重要程度的篇章表示。具体结构如图 2 所示。

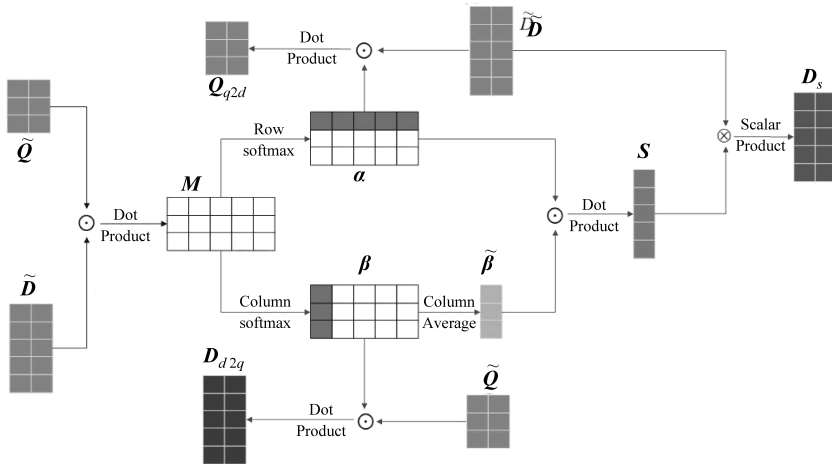


图 2 模型使用的注意力层示意图

2.5.1 问题与篇章的交互表示

首先, 以点积来计算问题与篇章中字词之间的匹配程度, 构成匹配矩阵 M 。具体计算如式 (10) 所示。

$$M_{i,j} = \tilde{Q}_{:,i}^T \cdot \tilde{D}_{:,j} \quad (10)$$

其中, $\tilde{Q}_{:,i}$ 表示 \tilde{Q} 的第 i 列向量, $\tilde{D}_{:,j}$ 表示 \tilde{D} 的第 j 列向量, $M_{i,j}$ 表示问题中的第 i 个字和篇章中的第 j 个字的匹配得分, $M_{i,j} \in R$; 匹配矩阵 $M \in R^{m \times n}$ 中, 行代表问题中的某个字与篇章中所有字的匹配得分, 列代表篇章中的某个字与问题中的所有字的匹配得分。

然后, 对匹配矩阵的每一行做 softmax, 得到问题到篇章的注意力 $\alpha = \{\alpha_i\}_{i=1}^m$, 并与篇章 \tilde{D} 做点积运算, 得到与篇章相关的问题表示 Q_{q2d} ; 同理, 对匹配矩阵的每一列做 softmax, 得到篇章到问题的注意力 $\beta = \{\beta_j\}_{j=1}^n$, 然后与问题 \tilde{Q} 做点积运算, 得到与

问题相关的篇章表示 D_{d2q} 。如式 (11)、式 (12) 所示。

$$\alpha_i = \text{softmax}(M_{i,:}), \quad \beta_j = \text{softmax}(M_{:,j}) \quad (11)$$

$$Q_{q2d} = \alpha \tilde{D}, \quad D_{d2q} = \beta^T \tilde{Q} \quad (12)$$

2.5.2 包含字重要程度的篇章表示

根据篇章回答问题时, 需要得到能够包含问题与篇章中每个字重要程度的篇章表示。具体通过对 β 求和求平均, 获得问题中每个字的重要程度 $\tilde{\beta} \in R^{m \times 1}$; 然后与问题到篇章的注意力 α 点积, 得到篇章中每个字的重要程度 $s \in R^{n \times 1}$, 最后与 \tilde{D} 标量相乘得到新的篇章表示 D_s 。具体如式 (13) ~ 式 (15) 所示。

$$\tilde{\beta} = \frac{1}{n} \sum_{j=1}^n \beta_j \quad (13)$$

$$s = \alpha^T \tilde{\beta} \quad (14)$$

$$\mathbf{D}_s = \tilde{\mathbf{D}} \cdot \mathbf{s} \quad (15)$$

2.6 解答层

根据问题分类层得到的问题类型概率以及编码层、注意力层,得到多种问题与篇章的表示,来对问题进行多样性解答。

2.6.1 Unknown 型问题的解答

Unknown 型问题一般比较简单,只需根据篇章判断该问题是否有答案。具体是对编码层得到的 $\tilde{\mathbf{U}}$ 做 MaxPooling,获得问题与篇章的聚合表示,再根据分类层预测得到的 $\mathbf{P}_{\text{unk}}^c$,求得该问题答案为 None 的概率 $\mathbf{P}_{\text{none}}^A$,如式(16)、式(17)所示。

$$\mathbf{C}^A = \text{MaxPooling}(\tilde{\mathbf{U}}) \quad (16)$$

$$\mathbf{p}_{\text{none}}^A = (\mathbf{W}_{\text{unk}}\mathbf{C}^A + \mathbf{b}_{\text{unk}})\mathbf{P}_{\text{unk}}^c \quad (17)$$

2.6.2 Yes/No 型问题的回答

该类问题的回答需要准确理解问题,因为不同的提问方式可能有不同的回答。具体做法是将问题与篇章的聚合表示 \mathbf{C}^A 和问题的多种表示进行拼接融合,加强对问题的理解,在此基础上进行答案预测,得到答案为 Yes 与 No 的概率 \mathbf{P}_y^A 、 \mathbf{P}_n^A 。计算如式(18)~式(20)所示。

$$\mathbf{Q}^A = \mathbf{W}_q(\mathbf{C}^A; \mathbf{W}_q[\tilde{\mathbf{Q}}; \mathbf{Q}_{q2d}]) \quad (18)$$

$$\mathbf{P}_{\text{yn}}^A = (\mathbf{W}_{\text{yn}}\mathbf{Q}^A + \mathbf{b}_{\text{yn}})\mathbf{P}_{\text{yn}}^c \quad (19)$$

$$\{\mathbf{P}_y^A, \mathbf{P}_n^A\} = \mathbf{P}_{\text{yn}}^A \quad (20)$$

2.6.3 Span 型问题的回答

Span 型问题最为复杂,需要充分地对问题与篇章进行理解,才能准确地从篇章中抽取答案。结合上述两种类型问题得到的表示 \mathbf{C}^A 、 \mathbf{Q}^A 与,与篇章的多种表示进行拼接、融合,实现问题与篇章的充分理解,然后再在篇章中抽取答案片段。如式(21)~式(23)所示。

$$\mathbf{D}^A = \mathbf{W}_d([\mathbf{C}^A, \mathbf{Q}^A]; \mathbf{W}_d[\tilde{\mathbf{D}}; \mathbf{D}_{q2d}; \mathbf{D}_s]) \quad (21)$$

$$\mathbf{P}_{\text{span}}^A = (\mathbf{W}_{\text{span}}\mathbf{D}^A + \mathbf{b}_{\text{span}})\mathbf{P}_{\text{span}}^c \quad (22)$$

$$\{\mathbf{P}_s^A, \mathbf{P}_e^A\} = \mathbf{P}_{\text{span}}^A \quad (23)$$

其中, $\mathbf{P}_s^A = [\mathbf{P}_{s_0}^A, \mathbf{P}_{s_1}^A, \dots, \mathbf{P}_{s_{n-1}}^A]$, $\mathbf{P}_e^A = [\mathbf{P}_{e_0}^A, \mathbf{P}_{e_1}^A, \dots, \mathbf{P}_{e_{n-1}}^A]$ 分别为篇章中每个位置作为答案开始与结束的概率;开始和结束的位置通过式 $\text{pos}_{\text{start}}^A = \text{argmax}([\mathbf{P}_s^A])$ 与 $\text{pos}_{\text{end}}^A = \text{argmax}([\mathbf{P}_e^A])$ 得到,且 $\text{pos}_{\text{start}}^A \leq \text{pos}_{\text{end}}^A$;同时获得起始位置的概率分别为 $\mathbf{P}_{\text{start}}^A = \mathbf{P}_s^A[\text{pos}_{\text{start}}^A]$ 与 $\mathbf{P}_{\text{end}}^A = \mathbf{P}_e^A[\text{pos}_{\text{end}}^A]$ 。

最后从多种答案中选取概率最大的作为最终答案。如式(24)~式(26)所示。

$$\mathbf{P}^A = \text{argmax}([\mathbf{P}_{\text{none}}^A, \mathbf{P}_y^A, \mathbf{P}_n^A, \frac{1}{2}[\mathbf{P}_{\text{start}}^A + \mathbf{P}_{\text{end}}^A]]) \quad (24)$$

$$\text{ans} = [\text{None}, \text{Yes}, \text{No}, D[\text{pos}_{\text{start}}^A; \text{pos}_{\text{end}}^A]] \quad (25)$$

$$\mathbf{A} = \text{ans}[\mathbf{P}^A] \quad (26)$$

2.6 模型损失函数

模型以交叉熵作为损失函数,主要由问题分类函数 L_C 和答案预测损失函数 L_A 两部分组成。具体如式(27)~式(31)所示。

$$L = L_A + L_C \quad (27)$$

$$L_A = -\log P_s^L(y^s) - \log P_e^L(y^e) \quad (28)$$

$$L_C = -\log P^c(y^c) \quad (29)$$

$$\mathbf{P}_s^L = [\mathbf{P}_{s_0}^A, \mathbf{P}_{s_1}^A, \dots, \mathbf{P}_{s_{n-1}}^A, \mathbf{P}_{\text{none}}^A, \mathbf{P}_y^A, \mathbf{P}_n^A] \quad (30)$$

$$\mathbf{P}_e^L = [\mathbf{P}_{e_0}^A, \mathbf{P}_{e_1}^A, \dots, \mathbf{P}_{e_{n-1}}^A, \mathbf{P}_{\text{none}}^A, \mathbf{P}_y^A, \mathbf{P}_n^A] \quad (31)$$

式(28)中, y^s , y^e 为真实答案的起始位置与结束位置。式(29)中, y^c 为真实的问题类别,式(30)、式(31)为将所有类型答案的预测概率组合起来,统一计算损失。

3 实验

3.1 数据集

本文针对问题的多样性展开研究,因此选择具有多样性问题的数据集来验证所提模型的有效性。具体选择“中国法研杯”挑战赛中文阅读理解任务数据集 CAIL2019-CJRC 进行实验。

该数据集具有多种问题类型,来源于“中国裁判文书网”公开的法律文书,主要为民事和刑事案件的一审判决书。数据集分别有 4 000 条民事案件样例与 4 000 条刑事案件样例,每条数据样例中有 4~5 个问题,共有 39 000 多个问题。具体示例如表 3 所示。其中,问题(1)~(3)为 Span 型问题,需要从文本中抽取片段来回答;问题(4)为 Yes/No 型问题,可以根据文本来判断其答案;问题(5)为 Unknown 型问题,该类问题不能根据所给文本做出回答。

由于评测主办方只公开了训练集,没有公开发集和测试集,因此本文只能在训练集上对模型进行测试。按照 4 : 1 的比例对训练集数据划分,形成训练集和测试集。

表 3 CAIL2019-CJRC 数据集示例

文本	
经审理查明,原告许 0、被告许 0 甲、许 0 乙与许 0 某系同胞兄妹关系,许 0 某 终身未婚 ,无子女,父母早年病故。2012 年 4 月 4 日,许 0 某因病去世,遗产有工资 4 800 元、丧葬费补助款 3 735 元,合计 8 535 元,该款存于银行卡,现银行卡在被告许 0 乙手里保存,银行卡密码由被告许 x14 设定掌控以上事实,有原告提供的协议书一份、武官村村民商森林证明材料一份、土地承包人王 15 明证言一份、银行卡账户历史明细清单一份,被告许 0 甲、许 0 乙提供的 2012 年 3 月 10 日、12 日 16 日签订的协议三份、录音光盘一张、龙凤居公墓的证明和政府文件,以及原、被告当庭陈述予以证实以上证据经庭审质证、认证,可以作为认定本案事实的依据。	
问题	答案
(1) 许 0 某婚姻状况如何?	• 终身未婚
(2) 许 0 共有多少遗产?	• 遗产有工资 4 800 元、丧葬费补助款 3 735 元,合计 8 535 元
(3) 银行卡密码谁知道?	• 许 x14
(4) 原被告和死者是否为同胞兄妹关系?	• YES
(5) 存有遗产款项的银行卡属于哪个银行?	• None(无答案)

3.2 训练细节

训练过程中,模型采用 Bert 中所提出的带有权重衰减的自适应动量估计算法(Adam weight decay optimizer)作为优化算法,初始学习率设为 3e-5,批处理大小为 32,共训练 5 轮。

3.3 主要实验结果

本文与在阅读理解中近期主流的 5 个模型进行对比。这 5 个模型分别为:

- (1) Bert^[3] 模型,谷歌公开的预训练模型(BERT-Base, Chinese);
- (2) ERNIE^[17] 模型,百度公开的预训练模型(ERNIE 中文 base 模型);
- (3) Chinese BERT-wwm^[19] 模型,为 Cui 等人以全词掩盖进行预训练的模型;
- (4) BertQA^①模型,为在 Bert 上进行微调的模型,在 SQuAD^[8]数据集上取得不错的结果;
- (5) Cail2019-top3^② 模型,为一种端到端的多任务学习模型,在 Cail2019 阅读理解评测中最终排名第 3。

本文采用 Chinese BERT-wwm^[19] 作为预训练模型。实验的评价指标为与 CoQA^[20]数据集一致的宏平均(macro-average F1)。具体结果如表 4 所示。

从表 4 可以看出,本文所提模型取得最好结果,在民事案件数据上相比于基线 Bert 模型提升了 3.6 个百分点,在刑事案件数据上相比于基线 Bert 模型提升了 2.7 个百分点,充分说明本文所提模型通过

注意力机制获得丰富的问题与篇章的表示,并对问题进行分类,然后将分类结果用于解答任务,最终实现问题的多样性解答的有效性。

表 4 模型实验结果

模型	民事 macro-F1	刑事 macro-F1
Bert ^[3]	78.9	75.9
ERNIE ^[17]	78.8	76.1
Chinese BERT-wwm ^[19]	79.3	76.9
BertQA	80.7	77.0
cail2019-top3	81.9	78.1
Our-model	82.5	78.6

此外,从表 4 中还可看出,所有基线模型中 Cail2019-top3 取得最好结果,预训练模型 Chinese BERT-wwm 相比于 Bert、ERNIE 模型结果更好。

3.4 消融实验结果

为了评估模型每部分的贡献,我们进行了消融实验,分别去掉 Bi-LSTM、分类任务与解答任务的联系(task_association)、分类任务(classification)、对问题的多样性解答(multiple_answer)等模块,来考察系统性能的变化,具体结果如表 5 所示。从表 5 中可看出,模型每一部分对系统性能都有效。其中,对问题进行多样性解答对系统性能影响最大,可

① <https://github.com/ankit-ai/BertQA-Attention-on-Steroids>

② <https://github.com/NoneWait/cail2019>

能的原因：不同类型问题的难易程度不同，回答的方式不同，针对每类问题的特性去分别解答，可以获得更好的结果。

表 5 消融实验结果

模型	macro-F1 民事	macro-F1 刑事
Our-model	82.5	78.6
—Bi_LSTM	82.1	78.0
—Task_association	81.9	77.9
—Classification	80.1	77.2
—Multiple_answer	79.4	76.9

4 结论

本文针对阅读理解中问题多样性的任务展开研究，提出一种基于 Bert 的多任务阅读理解模型，将问题分为 Unknown 型、Yes/No 型、Span 型三类，并针对每类问题分别进行解答。模型训练过程中，对每个问题分类预测，并将预测的结果作用于解答任务，通过注意力机制获得多种问题篇章的表示，将其融合后进行多样性解答，最终在 CAIL2019-CJRC 数据集上取得不错的效果。

但是，问题篇章之间的关系仅仅根据注意力机制来捕获并不充分，在未来的工作中，计划尝试用图网络的方法，对问题篇章进行建模，进一步加强对问题篇章的理解，更好地捕获关系，从而提升模型的性能。

参考文献

[1] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, et al. Bidirectional attention flow for machine comprehension[J]. arXiv preprint arXiv:1611.01603, 2016.

[2] Shuohang Wang, Jing Jiang. Machine comprehension using match-LSTM and answer pointer[J]. arXiv preprint arXiv:1608.07905, 2016.

[3] Jacob Devlin, Mingwei Chang, Kenton Lee, et al. BERT: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv: 1810.04805, 2018.

[4] Yiming Cui, Ting Liu, Zhipeng Chen, et al. Consensus attention-based neural networks for chinese reading comprehension[C]//Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics, 2016: 1777-1786.

[5] Karl Moritz Hermann, Tomas Kocisky, Edward

Grefenstette, et al. Teaching machines to read and comprehend[C]//Proceedings of Advances in Neural Information Processing Systems, 2015:1693-1701.

[6] Guokun Lai, Qizhe Xie, Hanxiao Liu, et al. RACE: Large-scale reAding comprehension dataset from examinations[C]//Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017:785-794.

[7] Matthew Richardson, J C Christopher Burges, Erin Renshaw. MCTest: A challenge dataset for the open-domain machine comprehension of text[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2013:193-203.

[8] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, et al. SQuAD: 100,000+ questions for machine comprehension of text[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2016: 2383-2392.

[9] Yiming Cui, Ting Liu, Wanxiang Che, et al. A span-extraction dataset for chinese machine reading comprehension[J]. arXiv preprint arXiv:1810.07366, 2018.

[10] Wei He, Kai Liu, Jing Liu, et al. DuReader: A Chinese machine reading comprehension dataset from real-world applications[J]. arXiv preprint arXiv: 1711.05073, 2017.

[11] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, et al. Teaching machines to read and comprehend[C]//Proceedings of Advances in Neural Information Processing Systems, 2015:1693-1701.

[12] Danqi Chen, Jason Bolton, Christopher D Manning. A thorough examination of the CNN/Daily Mail reading comprehension task[J]. arXiv preprint arXiv: 1606.02858, 2016.

[13] Yiming Cui, Zhipeng Chen, Si Wei, et al. Attention-over attention neural networks for reading comprehension[J]. arXiv preprint arXiv:1607.04423, 2016.

[14] Wenhui Wang, Nan Yang, Furu Wei, et al. Gated self-matching networks for reading comprehension and question answering[C]//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017:189-198.

[15] Zhipeng Chen, Yiming Cui, Wentao Ma, et al. Convolutional spatial attention model for reading comprehension with multiple-choice questions[J]. arXiv preprint arXiv: 1811.08610, 2018.

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. Attention is all you need[J]. arXiv preprint arXiv: 1706.3762, 2017.

[17]

Yu Sun, Shuohuan Wang, Yukun Li, et al. ERNIE: Enhanced representation through knowledge integration[J]. arXiv preprint arXiv:1904.09223, 2019.

[18]

Jimmy Lei Ba, Jamie Ryan Kiros, Geoffrey E Hinton. Layer normalization[J]. arXiv preprint arXiv:1607.06450, 2016.

[19]

Yiming Cui, Wanxiang Che, Ting Liu, et al. Pre-training with whole word masking for chinese BERT[J]. arXiv preprint arXiv:1906.08101, 2019.

[20]

Siva Reddy, Danqi Chen, Christopher D Manning. CoQA: A conversational question answering challenge[J]. arXiv preprint arXiv:1808.07042, 2018.



谭红叶(1971—),通信作者,博士,副教授,主要研究领域为中文信息处理、信息检索。
E-mail: hytan_2006@126.com



屈保兴(1995—),硕士研究生,主要研究领域为中文信息处理。
E-mail: 1025505854@qq.com



(上接第 80 页)

[20]

Laine S, Aila T. Temporal ensembling for semi-supervised learning[C]//Proceedings of the International Conference on Learning Representations, 2017.

[21]

Zhang X, Lecun Y. Text understanding from scratch [J/OL]. arXiv preprint arXiv: 1502.01710, 2015.

[22]

Yogatama D, Dyer C, Ling W, et al. Generative and discriminative text classification with recurrent neural networks [J/OL]. arXiv preprint arXiv: 1703.01898, 2017.

[23]

Paszke A, Gross S, Chintala S, et al. Automatic differentiation in PyTorch[C]//Proceedings of the NIPS Autodiff Workshop, 2017.

[24]

Pennington J, Socher R, Manning C D, et al. Glove: Global vectors for word representation[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2014: 1532-1543.

[25]

Kingma D P, Ba J. Adam: A method for stochastic optimization [C]//Proceedings of the International Conference on Learning Representations, 2015.



线岩团(1981—),博士研究生,副教授,主要研究领域为自然语言处理、信息抽取、机器翻译。
E-mail: xianyantuan@qq.com



相艳(1979—),博士研究生,讲师,主要研究领域为自然语言处理、信息抽取、机器翻译。
E-mail: 50691012@qq.com



余正涛(1970—),通信作者,博士,教授,主要研究领域为自然语言处理、信息检索、机器翻译、机器学习。
E-mail: ztyu@hotmail.com