

基于BERT的端到端中文篇章事件抽取

张洪宽 宋晖[✉] 王舒怡 徐波

(东华大学 计算机科学与技术学院, 中国 上海 201620)

{2181729,2181754}@mail.dhu.edu.cn, {songhui,xubo}@dhu.edu.cn

摘要

篇章级事件抽取研究从整篇文档中检测事件, 识别出事件包含的元素并赋予每个元素特定的角色。本文针对限定领域的中文文档提出了基于BERT的端到端模型, 在模型的元素和角色识别中依次引入前序层输出的事件类型以及实体嵌入表示, 增强文本的事件、元素和角色关联表示, 提高篇章中各事件所属元素的识别精度。在此基础上利用标题信息和事件五元组的嵌入式表示, 实现主从事件的划分及元素融合。实验证明本文的方法与现有工作相比具有明显的提升。

关键词: 篇章级别事件抽取; 端到端; 增强嵌入表示; 主从事件

A BERT-based End-to-End Model for Chinese Document-level Event Extraction

Hongkuan Zhang, Hui Song[✉], Shuyi Wang, Bo Xu

(School of Computer Science and Technology, Donghua University, Shanghai, 201620, China)

{2181729,2181754}@mail.dhu.edu.cn, {songhui,xubo}@dhu.edu.cn

Abstract

Document-level event extraction aims at discovering event mentions and extracting events which contain event arguments and their roles from texts. This paper proposes an end-to-end model for closed-domain based on BERT. We introduce the embedding of event type and entity nodes to the subsequent layer for event argument and role identification, which represents the relation between event, arguments and roles and improves the accuracy of classifying multi-event arguments. With the title, the quintuple of event, we calculate the master slave structure between multiple events with the embedding presentation. Experimental results show that our model outperforms the state of the art.

Keywords: Document-level Event Extraction, End-to-end, Enhancing embedding, Master slave event

1 引言

©2020 中国计算语言学大会

根据《Creative Commons Attribution 4.0 International License》许可出版

基金项目: 国家自然科学基金青年项目 (61906035), 上海市青年科技英才扬帆计划项目 (19YF1402300)

近年来互联网快速发展，网络媒体每天产生大量的新闻、公告等非结构化信息。信息抽取技术研究如何从海量的信息中快速有效地捕获有价值的信息，以帮助人们针对特定信息做分析、决策。事件抽取是信息抽取的分支，旨在从非结构化的自然语言文本中抽取出用户感兴趣的事件信息并以结构化的形式展示 (Ahn, 2006)。事件抽取在很多领域有着广泛的应用，如构建事件知识图谱、信息检索、自动问答以及辅助其他自然语言处理任务等。

事件抽取分为开放域和限定域事件抽取 (Wei and Wang, 2019)。开放域事件抽取研究通常没有领域范围限制，事件类型及事件的框架结构未知，主要利用无监督方法从文本中发现事件 (Piskorski et al., 2011; Ribeiro et al., 2017; Yu and Wu, 2018)。限定域事件抽取往往针对特定领域（如医疗、金融、司法等）的数据进行建模，识别用户感兴趣的信息。与开放域相比，限定域事件抽取有清晰的事件类型定义及对应的事件框架，能够获得具有实用价值的信息，近年来成为研究和应用的热点。

事件抽取从文本粒度上也可分为句子级别和篇章级别。句子级别事件抽取研究从句子中识别所关注的内容，首先采用深度神经网络，如动态多池化卷积神经网络 (Chen et al., 2015)、结合LSTM和CNN的卷积双向LSTM神经网络 (Zeng et al., 2016)等方法提取特征，然后识别事件元素和角色。事件及其元素识别通常采用管道式方法，分为两个子任务实现，忽略了他们之间的联系，容易导致错误传播问题。针对管道式模型存在的问题，Li (2013)和Nguyen (2016)采用联合模型捕获实体与事件之间的语义关系，同时识别事件和实体，提高了事件抽取的准确率。

篇章级的事件抽取旨在发现从整篇文档识别事件并提取出相应的事件元素，抽取困难之处在于文档中可能存在多个事件（如图1所示），事件元素分布在不同句子中，多个事件之间存在元素重叠（如图1所示）。

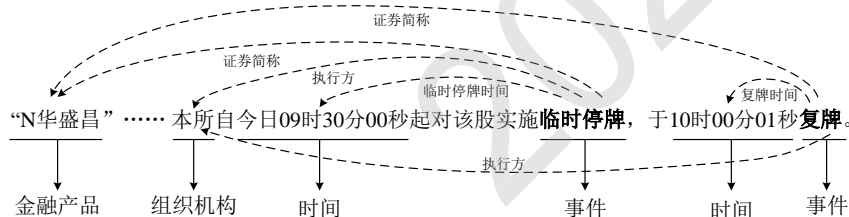


图1.文档中存在“临时停牌”与“复牌”两个事件，且共享事件元素“N华盛昌（角色为证券简称）”和“本所（角色为执行方）”。

目前篇章级的事件抽取大多采用管道式方法，即分别对事件和事件元素进行识别，然后判断元素的角色。如Yang (2018)基于句子抽取结果及文本特征发现核心事件，采用元素补齐策略得到篇章级别的事件信息。ZHONG (2019)在句子级抽取结果的基础上利用整数线性规划进行全局推理，融合共指事件的元素，实现篇章级事件抽取。由于管道模型本身的局限性，导致这些方法整体效果不理想。Yang (2016)提出一个端到端学习模型，通过采用联合因子图模型来联合学习每个事件内部的结构化信息、篇章内不同事件间的关系和实体信息，但是该工作使用了大量人工特征，不利于领域泛化。

Zheng (2019)采用端到端模型首先识别句子级别的事件元素，再利用二分类策略判定事件类型，然后将元素角色识别转化为有向无环图的生成过程。该文能够较好地处理单事件单实例和单事件多实例的样本，但在元素分布较分散且存在元素重合的多事件多实例样本上性能略差。针对以上问题，本文在已有篇章级的抽取研究工作基础上，提出了基于BERT (Devlin et al., 2019)的端到端事件抽取模型DLEMC (Document-Level End-to-end Model in Chinese)，该模型无需将文章按句子分割分别处理，尽可能保留完整的文本，减少信息损失；在事件元素识别中引入了事件类型嵌入增强文本特征，并在元素角色分类中利用注意力机制引入事件类型和实体嵌入表示，以便准确识别事件元素及其在不同事件中的角色。

DLEMC模型分为4层：输入编码层、事件检测层、事件元素识别层以及元素角色识别层。输入编码层接收输入的文本信息并输出对应的嵌入表示；事件检测层采用多个分类器对同一个文本特征向量进行多标签事件分类；事件元素识别层通过引入事件特征学习不同事件中元素的语义信息，进行事件元素的识别；元素角色识别层利用注意力机制来提高模型对确定事件中每

个元素特征的关注度，判断其在对应事件中扮演的角色。最后基于事件的嵌入表示计算余弦相似度，进行主从事件划分及融合共指元素，得到篇章级结构化事件信息。实验证明DLEMC的性能与基准模型相比有明显的提升。

本文的贡献总结如下：

(1) 本文依据金融领域上市公司公告组织了一个篇章级别事件抽取语料集，并针对该语料定义了事件及事件表示框架。

(2) 本文构建了篇章级别的事件抽取模型，该模型采用端到端方式进行联合学习，同时对事件检测、事件元素识别及元素角色分类进行训练，通过实验证明了该模型的有效性。

(3) 本文在事件元素识别中引入事件类型特征，以提高不同类型事件下的元素识别能力。为了更准确识别多事件文档中元素角色，我们将事件检测层输出的事件类型及事件元素的嵌入表示作为注意力引入模型的角色分类层，以提高不同类型下元素角色识别的准确率。

2 事件定义及表示

随着金融科技的发展，在金融领域每天都有海量的数据产生，金融事件抽取研究能够帮助人们进行金融风险监控、辅助投资决策、大数据分析等。本文研究的事件抽取需要提取事件类别及参与者，目前该领域的中文事件抽取研究缺乏数据支持，前人的相关研究大多没有公开数据集，研究的事件类型比较集中且类别较少(Yang (2018)4类, Zheng (2019)5类, 去除重复后共5类。), 为扩充金融领域事件研究的数据，本文组织构建了一定规模的金融领域中文篇章事件抽取数据集，并依据自动内容抽取(Automatic Context Extraction, ACE)⁰定义的事件抽取任务，说明如下：

事件(event): 在某个时间点或时间段，一个或多个机构的金融产品的状态主动地或被动地发生了变化。

实体(entity): 语义类别中的一类或一组对象，本文讨论的实体包括命名实体、金融产品、时间和数值。

事件元素(event argument): 在事件中具有特定作用的实体。

元素角色(argument role): 事件元素在事件中承担的角色。

针对本文研究的金融公告信息，事件定义为 $event = def(T, O, F, D, N)$ 其中 T 为事件类型， O 、 F 、 D 、 N 为事件中的4类角色，分别表示组织机构、金融产品、时间、数值，每一类下有若干小类，共计22类事件角色。

(1) 事件类型，事件所属的类别，如“临时停牌”、“复牌”、“上市交易”等。

(2) 组织机构，参与事件的一类实体，如“东洋科技集团股份有限公司”。

(3) 金融产品，金融领域中的相关产品，如“证券名称”，“证券简称”。

(4) 时间，指事件发生的具体时间点或者事件持续发生且产生作用的时间间隔，如“10时00分01秒”。

(5) 数值，衡量事件中某一属性具体量的多少，如“票面利率4%”、“标准交易单位10张”等。

其中事件类型、组织机构、金融产品和时间的实例在事件文本中一定会出现，数值实例不一定会出现。

3 事件抽取模型DLEMC

篇章级事件抽取研究识别文档中存在的事件和相关元素，并判断元素在事件中扮演的角色。给定文档集 $doc = \{s_0, s_1, \dots, s_{N_s}\}$ ，每篇文档包含标题句 s_0 和内容句 $\{s_1, \dots, s_{N_s}\}$ ， N_s 为句子数量。模型DLEMC首先对文档标题和内容分别进行事件检测，得到文档包含的事件类型 $\{t_0, t_1, t_2, \dots\}$ ，其中 t_0 为标题中的事件，其他为内容中的事件，然后识别出文档内容中每类事件的相关元素 $\{e_1, e_2, \dots\}$ 及其对应的角色 $\{role_1, role_2, \dots\}$ 。

DLEMC模型由4部分组成（如图2所示），包括输入编码层、事件检测层、事件元素识别层和元素角色识别层。

输入编码层，基于BERT对输入的句子进行编码，得到句子对应的向量以及句子中每个token的向量。

⁰<http://projects.ldc.upenn.edu/ace>

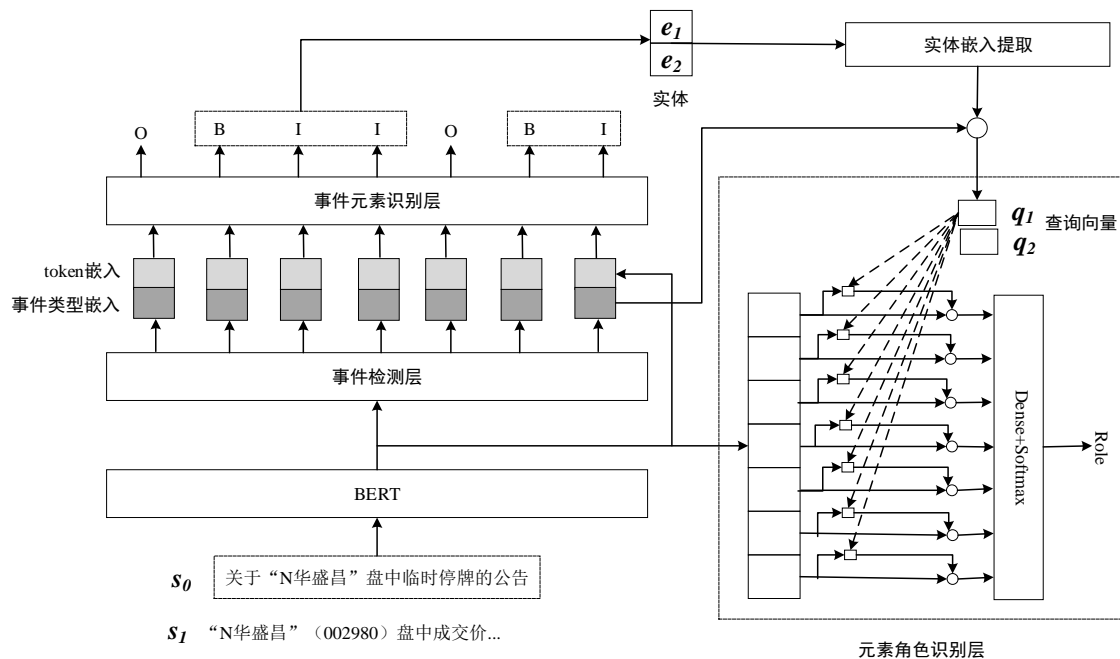


图2.篇章级事件抽取模型DLEMC

事件检测层，将编码层输出的句向量作为输入，预测该句中包含的事件，一个句子中可能存在多个事件。

事件元素识别层，识别句子中参与事件的实体。将句子的 $token$ 向量与事件类型对应的向量进行拼接作为输入，预测每个 $token$ 对应的 BIO 标签，从而识别出事件元素对应的实体。

元素角色识别层，对上一步识别出的确定事件类型下的实体进行角色分类。将事件类型 t 和实体 e 对应的嵌入表示求平均之后作为注意力的查询向量，重新计算 $token$ 的向量表示，再对每个事件元素的角色进行识别。

模型训练时分别计算事件检测层、事件元素识别层以及元素角色识别层的损失，并将三者求和作为模型最终的优化目标。

3.1 输入编码层

本文采用预训练语言模型BERT对文档进行编码，考虑BERT模型的有效位置编码序列长度以及实际训练的模型规模，我们设置最大序列长度为 max_length 。对于给定的文档，将标题作为独立的句子；对于文档内容，若文本序列长度大于 max_length ，则依据中文标点符号将其切分成多个句子，反之则将整篇文档作为一个句子。

经过以上处理，本文将一篇文档表示为一系列句子集合 $doc = \{s_0, s_1, \dots, s_{N_s}\}$ ， N_s 为句子总数， s_j 为文档中第 j 个句子， s_0 为文档标题。每个句子由一系列 $token$ 组成 $\{tok_{1,j}, \dots, tok_{N_w,j}\}$ ，其中 $tok_{i,j}$ 为第 j 个句子中第 i 个 $token$ ， N_w 为第 j 个句子的序列长度。每个句子经BERT编码后得到的 $token$ 向量序列为 $H_{tok} = \{h_{1,j}, h_{2,j}, \dots, h_{N_w,j}\}$ ，其中 $h_{i,j}$ 为第 j 个句子中第 i 个 $token$ 对应的向量，维度为 d ，句子向量序列为 $\{h_0, h_1, \dots, h_{N_s}\}$ ， h_0 为标题句向量表示， h_j 为文档中第 j 个句子的向量表示，维度为 d 。

3.2 事件检测层

事件检测的目的是检测句子中包含的事件，本文的数据集样本中可能存在多个事件，受 (Liu al et., 2019)启发，我们将事件检测建模为多标签分类任务。事件检测样本的标注形式如表1所示。

其中 s 为句子， t_1 、 t_2 为不同的事件类型，标签为1表示句子 s 中包含事件 t_1 ，标签为0则表示句子 s 不包含对应的事件。

句子	事件类型	标签
s	t_1	1
s	t_2	0
s	t_3	1

表1.事件检测数据标注实例

对于给定文档的句子向量表示 $\{h_0, h_1, \dots, h_{N_s}\}$ ，我们依次将文档中的句子向量作为全连接层的输入，如式(1)所示。

$$H_{ed} = W_{ed}h_j + b_{ed} \quad (1)$$

其中 W_{ed} 为参数矩阵， b_{ed} 为偏置， h_j 为第 j 个句子 s_j 的隐层向量表示。对所有事件类型使用Sigmoid分类器进行分类，式(2)给出了对某类事件预测的计算方法。

$$y = \frac{1}{1 + e^{-H_{ed}}} \quad (2)$$

此层的预测错误使用交叉熵作为损失函数，如式(3)所示。

$$L_{ed} = -\frac{1}{N} \sum_{j=1}^N \sum_{m=1}^M y_{j,m} \log y_{j,m}^* + (1 - y_{j,m}) \log(1 - y_{j,m}^*) \quad (3)$$

其中 $y_{j,m}^*$ 为第 j 个样本预测为第 m 类事件的预测值， $y_{j,m}$ 为第 j 个样本为第 m 类事件的真实值， N 为样本总数， M 为预定义事件类别总数，这里设定一个阈值为 α ，若 $y_{j,m}^*$ 的值大于等于 α ，则认为该样本包含 $y_{j,m}^*$ 对应的事件，否则认为该样本不包含 $y_{j,m}^*$ 对应的事件。

3.3 事件元素识别层

本文将事件元素识别建模为序列标注任务，使用BIO标签模式（Begin: 字段开头，Inside: 字段内部，Outside: 其他字段）为每个token赋予一个实体标签。

对于给定文档中的句子 s_j ，首先通过事件检测层预测得到对应的文档标题事件和文档内容事件，然后依次识别文档内容中每个事件的相关元素。为提高不同事件类型下实体的语义表示，本文在实体识别部分引入事件特征。具体做法如下：为每种事件类型定义 d 维（与token向量维度相同）的向量，通过查表的方式得到事件类型对应的向量 t_{vec} ，将句子中的每个token向量与事件类型向量 t_{vec} 进行拼接作为最终的特征向量，式(4)所示为计算某token的特征向量。

$$h_{vec,i} = [h_{i,j}; t_{vec}] \quad (4)$$

其中， $h_{i,j}$ 为第 j 个句子中第 i 个token的向量表示， $h_{vec,i}$ 为句子中第 i 个token最终的特征向量，“;”表示拼接，将特征向量 $H_{vec} = \{h_{vec,1}, h_{vec,2}, \dots, h_{vec,N_w}\}$ 作为全连接层的输入，使用softmax作为分类器预测每个token对应的标签，如式(5)所示。

$$P = softmax(W_{ner}H_{vec} + b_{ner}) \quad (5)$$

W_{ner} 为参数矩阵， b_{ner} 为偏置。使用交叉熵计算该部分的损失，如式(6)所示。

$$L_{ner} = -\frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K y_{i,k} \log P_{i,k} \quad (6)$$

N 为样本总数， K 为标签类别总数，第 i 个样本预测为第 K 个标签的概率为 $P_{i,k}$ ，第 i 个样本真实的标签为 $y_{i,k}$ 。

3.4 元素角色分类层

元素角色识别的目标是为确定事件类型下的实体赋予预定义的角色，本文将角色识别建模为多分类任务。为更好地区分实体扮演的角色，将利用注意力机制来增强文本的特征表示，依次判断每个实体在事件中扮演的角色。

实体往往包含多个 $token$ ，对于给定句子 s_j 中识别出的实体集 $E = \{e_1, e_2, \dots\}$ ，其中每个实体包含该句中的第 i 至第 k 个 $token$, $[tok_{i,j}, \dots, tok_{k,j}]$ ，本文取实体中的所有字符向量的均值作为该实体的嵌入表示 c ，维度为 d ，采用这种均值向量可以有效避免模型过拟合问题 (Ji al et., 2018)。计算方式如式(7)所示， $h_{i,j}$ 为第 j 个句子中的第 i 个 $token$ 。

$$c = \frac{1}{k} \sum_{i=1}^k h_{i,j} \quad (7)$$

然后，我们将 E_{vec} 中的第 j 个实体的向量表示 c_j 与包含该实体的事件类型对应的向量表示 t_{vec} 进行相加再求平均得到维度为 d 的注意力机制查询向量 q ，如式(8)所示。

$$q = \frac{t_{vec} + c_j}{2} \quad (8)$$

最终得到查询向量集合 $Q = q_1, q_2, \dots$ ，使用 q 与句子 s_j 的 $token$ 向量表示 H_{tok} 计算得到每个 $token$ 的注意力值 a_k ，计算方法如式(9)所示。

$$a_k = \frac{\exp(h_{k,j} q_{l,j}^T)}{\sum_{i=1}^{N_s} \exp(h_{i,j} q_{l,j}^T)} \quad (9)$$

其中， $q_{l,j}^T$ 为句子 s_j 第 l 查询向量， $h_{k,j}$ 表示句子 s_j 中第 k 个 $token$ 的对应的， a_k 为第 k 个 $token$ 的注意力值， $a = [a_1, a_2, \dots, a_{N_s}]$ 为所有 $token$ 的注意力值向量，由 a 与样本中每个 $token$ 的向量 H_{tok} 相乘得到融合实体信息和事件类型语义信息的特征向量 V ，如公式(10)所示， $*$ 表示点乘。

$$V = a * H_{tok} \quad (10)$$

将 V 作为全连接层的输入，使用 $softmax$ 分类器进行分类，如公式(11)所示。

$$y^* = softmax(W_{rt}V + b_{rt}) \quad (11)$$

其中 M 为类别总数， $y_{j,u}^m$ 为第 j 个样本中第 u 个事件元素属于第 j 类预定义角色类型的真实值， $y_{j,u}^{*(m)}$ 为模型预测值， N 为样本总数。此部分的损失函数如(12)所示：

$$L_{rt} = -\frac{1}{N} \sum_{j=1}^N \sum_{u=1}^M \sum_{m=1}^M y_{j,u}^m \log y_{j,u}^{*(m)} \quad (12)$$

3.5 模型训练

我们将事件检测、事件元素识别以及元素角色分类同时进行训练，模型的训练目标是综合3部分的损失达到最小，训练时分别计算事件检测层的二分类交叉熵损失 L_{ed} ，以及事件元素识别层与元素角色分类层的多分类交叉熵损失 L_{ner} 与 L_{rt} ，我们将3个损失求和作为模型最终的优化目标： $L_{final} = L_{ed} + L_{ner} + L_{rt}$

模型训练时采用Adam (Kingma al et., 2015)作为优化器，通过验证集选择最好的模型进行预测。

3.6 主从事件识别

文档中包含多个事件时, 将根据标题事件进行主从事件划分, 然后对同指事件元素进行融合, 从而得到篇章级事件抽取结果。文档标题往往能概括一篇文档的主要内容, 故本文将文档标题中的事件作为主事件, 其他事件作为从事件。

对于文档中的多个事件 $Events = \{e_0, e_1, e_2, \dots\}$, 其中 e_0 为文档标题预测出的事件, 其他的均为文档内容预测出的事件。事件类型以及事件元素使用从DLEMC模型中获得的嵌入式表示, 基于余弦相似度计算两个事件的相似程度, 如式(13)所示。

$$obj = sim(e_0, e_i) \quad (13)$$

obj 为相似度得分, 用于衡量 e_0 与 e_i 两个事件的共指程度, 我们取最高得分对应的那组事件作为文档的主事件, 其他事件为从事件。

主从事件融合的目的是对同一个文档里多个事件之间的共指事件元素进行合并, 从而得到规范的篇章级事件信息。本文通过计算不同事件中事件元素的语义相似度来衡量它们的共指程度, 具体规则为: 语义相似度超过设定阈值 γ 的事件元素作为共指元素, 否则为非共指元素。两个事件中不同元素的相似度基于余弦相似度计算, 如公式(14)所示。

$$score_{list} = sim(t_{1,i}, t_{2,j}) \quad i, j \in \{1, 2, \dots\} \quad t_{1,i}, t_{2,j} \in (T, O, F, D, N) \quad (14)$$

式(14)中 $t_{1,i}$ 与 $t_{2,j}$ 分别表示事件 e_1 的第 i 个元素与事件 e_2 的第 j 个元素, $score_{list}$ 为计算得到的语义相似度的得分集合, 若该集合中某一项值大于阈值 γ , 就将该项对应的两个事件元素合并。

4 实验

4.1 数据集

本文将从互联网上搜集的上市公司公告作为实验数据集。共有文档总数23067, 其中5056个文档中包含多个事件, 占比21.9%, 将总文档按照8: 1: 1划分成训练集、验证集和测试集, 数据集中标注的实体类别包括NUM (Number, 数值)、ORG (Organization, 组织机构)、FIN (Finance, 金融产品)、TIM (Time, 日期、时间)。事件分为11类: 上市交易、停牌、临时停牌、复牌、摘牌、名称变更、支付利息、债券转让、暂停上市、终止上市、到期兑付。各类样本数如表2所示。

事件类型	训练集	验证集	测试集	总数	多事件率(%)
上市交易	5575	902	897	7554	0
临时停牌	3061	266	284	3611	94.0
停牌	1671	189	236	2096	6.3
复牌	3122	260	321	3073	95.2
摘牌	1252	143	135	1530	100
债券转让	1082	154	166	1042	0
终止上市	589	128	102	819	0
到期兑付	1252	1143	135	1530	100
暂停上市	934	136	90	1160	0
名称变更	530	87	102	719	0
支付利息	822	91	167	1080	0
文档数量	18455	2306	2306	23067	21.9

表2.数据集样本统计

为了验证远程监督标注事件的质量, 我们从每类事件中随机选取20个样本进行人工标注, 作为真实值, 再用远程监督方法标注它们作为预测值, 依据4.2中的评价指标进行验证, 验证结果如表3所示。表3说明远程监督方法标注的语料具有较高的精确率, 以及不错的召回率和F1值。本文的实验中均采用远程监督方法生成训练集、验证集和测试集。

Precision	Recall	F1	多事件率(%)
97.5	90.0	93.3	26.3

表3.远程监督事件标注质量

4.2 模型评价

本文采用精确率(Precision, P)、召回率(Recall, R)和F1(F1-measure, F1)值作为评价指标,采用微平均计算F1值,一个事件类型与某一事件元素及其角色为一个统计项。在事件类型预测正确的前提下,若事件元素及其对应的角色均与标注相同则视为正确,否则视为预测错误,若事件类型预测错误则将所有的元素与角色均视为预测错误,具体计算方式如下。

$$P = \frac{\text{识别事件类型与元素和标注相同的数量}}{\text{识别出事件类型与元素总数量}}$$

$$R = \frac{\text{识别出事件类型与元素和标注相同的数量}}{\text{标注的事件类型与元素总数量}}$$

$$F1 = \frac{(2 * P * R)}{(P + R)}$$

4.3 参数设置

本文的实验基于BERT_base模型来初始化词向量,维度为786,dropout的比率为0.4,批次大小为16,模型学习率为3e-5,训练10个epoch,最大文本序列长度max_length为200,事件类型向量随机初始化生成,维度为768。

4.4 结果分析

我们实现了Yang (2018)提出的DCFEE模型,Zheng (2019)提出的Doc2EDAG模型,以及基于BERT的管道模型BERT-P,在BERT-P中事件检测部分与本文提出的方法DLEMC相同,但在实体识别任务中未增加事件特征,在角色分类任务中未利用事件和实体特征注意力,DLEMC-P为DLEMC的管道模式。我们将DCFEE、BERT-P和Doc2EDAG作为本文的baseline,在包含全部11类事件的测试集上进行各项测试。

1) 为验证本文的模型DLEMC在事件类型检测上的有效性,我们在测试集上对模型进行了评价,实验结果如表4所示。

Model	P(%)	R(%)	F1(%)
DCFEE	80.3	77.2	78.7
BERT-P/DLEMC-P	84.1	83.2	83.6
Doc2EDAG	86.1	84.3	85.1
DLEMC	85.8	84.7	85.3

表4.事件检测评价结果

如表4所示,利用序列标注识别事件触发词的DCFEE模型效果比较差。通过分类模型检测事件的BERT-P、DLEMC-P、DLEMC、Doc2EDAG模型在各项指标上均优于DCFEE,其中端到端联合学习模型DLEMC、Doc2EDAG在各项指标上均优于管道式模型。本文提出的DLEMC模型在准确率上略低于采用二分类策略的Doc2EDAG模型,但召回率和F1均优于Doc2EDAG,其中F1提高了0.2%,实验表明多标签分类模型在多事件检测中有较好的表现。

2) 为验证DLEMC模型在篇章事件元素识别和角色分类时的有效性,我们在测试集上对模型进行了评价。

由表5实验结果可以看出,基于预训练语言模型BERT词向量表征的事件抽取模型BERT-P明显优于使用word2vec (Mikolov et al., 2013)的DCFEE模型。本文提出的DLEMC模型则在P、R、F1等3个指标上都优于BERT-P,其中F1提升了4.1%。

DLEMC-P保留增强的文本嵌入表示,但采用管道式方法完成事件抽取的子任务,实验表明其在准确率上有提升,但是召回率上大幅下降,但F1值仍然比直接基于管道的BERT-P提升

Model	P(%)	R(%)	F1(%)
DCFEE	64.3	58.2	61.1
BERT-P	74.5	58.7	65.7
DLEMC-P	77.1	59.6	67.2
Doc2EDAG	76.1	64.0	69.5
DLEMC	76.2	64.3	69.8

表5.篇章级别事件抽取评价结果

了1.5%。端到端模型Doc2EDAG和DLEMC在准确率上略低于DLEMC-P，但召回率和F1较管道式方法均有大幅提升。得益于DLEMC在实体识别部分加入事件类型特征，在角色分类部分加入事件类型与实体注意力特征，本文的DLEMC在各项指标上均优于Doc2EDAG。

3) 为验证DLEMC在处理多事件时的有效性，我们将数据集划分为单事件(Single-event)与多事件(Multi-event)两个子集，并分别用这两个子集对模型进行评价，实验结果如表6所示。

Model	Single-event(%)	Multi-event(%)	Avg(%)
DCFEE	63.7	41.5	52.6
BERT-P	76.5	60.2	68.3
DLEMC-P	79.6	62.7	71.2
Doc2EDAG	81.2	63.5	72.3
DLEMC	81.4	63.8	72.6

表6.单一事件与多事件评测F1值与平均值(Avg)

表6中的实验结果表明DCFEE中的基于核心事件进行元素补全策略存在局限性，不能较好的处理多事件样本。而BERT-P以及DLEMC-P基于优秀的词嵌入方法，提高了特征表达能力，在单事件和多事件样本上性能均有较大改善，但由于管道模式不可避免地将前序任务中的错误信息传递至后序任务，模型的整体性能低于端到端的模型Doc2EDAG和DLEMC。实验表明，对于被分割的样本，DLEMC有效增强了文本中的事件特征表示，提高了在同一类型事件的多实例和不同类型事件的多实例情况下的性能，其在单事件和多事件评价上均优于Doc2EDAG。

4) 为得到完整的结构化篇章事件信息，我们在DLEMC事件抽取结果的基础上对包含多个事件的文档进行了主从事件划分和主从事件元素融合，实验结果如表7、表8所示。

P(%)	R(%)	F1(%)
83.4	82.1	82.7

表7.主从事件划分评价结果

由表7可看出，主从事件划分的效果是可接受的，由于主从事件划分依赖事件检测的结果，使得该部分仍具有较大提升空间。

P(%)	R(%)	F1(%)
70.4	67.1	68.7

表8.主从事件元素融合评价结果

表8给出了对正确识别且角色判定正确的事件元素进行同指事件元素融合实验的评价结果。事件元素融合的性能受到句子级事件抽取结果的影响，造成最终的性能指标偏低。

5 相关工作

目前事件抽取方法可以分为两类：基于模式匹配方法和基于统计学习方法。模式匹配方法在特定领域有较高的准确率，但是通常需要编写大量的人工模板，且普适性较差 (Yangarber and Grishman, 1997; Surdeanu and Harabagiu, 2002)。统计学习方法可以分为两类：传统的基于特征工程的机器学习方法以及基于深度学习的方法。传统特征工程主要依赖自然语言处理工具获取有效的特征（如句法、词汇、词性等），然后利用传统的分类模型（如最大熵、支持向量机）进行分类 (Ahn, 2006; Jungermann and Morik, 2008; Liao and Grishman, 2010)。基于深度学习的方法依靠神经网络自动提取特征，在事件抽取中取得了不错的效果。如WU (2019)使用一种混合神经网络模型，进行实体和事件的联合学习。Chen (2015)使用一种动态多池化卷积神经网络来捕获多个特征，提升了事件抽取的性能。Zeng (2016)使用一种卷积双向LSTM神经网络，分别从词级别和字级别进行触发词和实体的识别。

事件抽取任务依据是否具有预定义的事件框架（事件类型及每类事件对应的角色）可以分为开放域和限定域事件抽取 (Wei and Wang, 2019)，开放域事件抽取目标在于识别自然语言文本中的事件，一般没有领域限制，不需要预定义事件框架。限定域事件抽取会预先定义好要抽取的事件类型，如“袭击”事件、“审判”事件等。同时也会定义每类事件参与者的角色，如“审判”事件中包含“被审判人”、“审判时间”、“地点”等角色。

从文本粒度来看，目前事件抽取的相关研究主要集中在句子级别，即识别句子中的事件并提取相应的事件元素。句子级事件抽取主要有两种建模方式，管道模式和联合模式。管道式方法通常将事件识别和元素提取分为两个独立的任务，忽略了事件与元素之间的联系，导致效果不够理想。联合模型一般同时识别句子中的事件并提取相关元素，利用深度神经网络捕获事件与元素之间的语义联系，模型训练时能够互相影响并优化，性能一般要优于管道式模型。

现实中的文本信息往往是以篇章形式出现的，针对的篇章事件抽取能够获得更完整、规范的信息。篇章级别的事件抽取研究方法通常首先对给定文档中的句子进行处理，然后再对句子级别的事件信息进行合并，从而得到篇章级别的事件抽取结果。目前篇章级的事件抽取大多采用管道式模型，如ZHONG (2019)采用触发词和实体联合标注的方法同时抽取句子级别的触发词和实体，然后使用多层感知机对实体进行角色分类，并利用整数线性规划进行同指事件的融合，实现篇章级别的事件抽取。Yang (2016)通过采用联合因子图模型来联合学习每个事件内部的结构化信息、篇章内不同事件间的关系和实体信息，提高了篇章事件抽取的效果，但他们的工作依赖大量的人工特征。Yang (2018)基于句子抽取结果以及文本特征发现主事件描述，并利用上下文元素补齐策略得到篇章事件结构化信息。

总体来讲，目前篇章级别的事件抽取研究较少，且集中在特定领域，通常依赖大量人工规则，难以进行领域拓展。而句子级别的事件抽取日趋成熟，应用领域更广，但得到的结果无法提供较好的篇章级事件信息。

6 总结与展望

本文针对金融领域篇章级别事件抽取任务定义了事件表示框架，在该框架下提出基于深度学习的端到端模型抽取事件信息，模型采用3层，分别实现多标签分类的事件检测、基于融合事件类型特征的事件元素识别以及基于注意力机制的元素角色分类。对获取的多个事件，利用余弦相似度进行主从事件划分以及多事件的元素融合，得到篇章级事件结构化信息。我们构建了金融领域事件抽取语料对本文方法进行验证，实验证明本文方法明显优于基准方法。

然而，由于事件元素之间存在较强的相似性，如“上市交易”事件中的“股份总数160,000,000股，其中40,000,000股自上市之日起开始上市交易”，模型可能会将“股份总数”与“上市股数”错误分类。如何提高相似元素的特征表示及其分类效果，从而提升篇章级事件抽取的整体性能，是本文未来的改进方向。

参考文献

- David Ahn. 2006. *The stages of event extraction*, In The Workshop on Annotating and Reasoning about Time and Events. pages 1–8.
- Xiang, Wei, and Bang. Wang. 2019. *A Survey of Event Extraction From Text*, IEEE Access 7:173111-173137.

- J.Piskorski,H.Tanev,M.Atkinson,E.V.D.Goot,and V.Zavarella. 2011. *Online news event extraction for global crisis surveillance*, Transactions on computational collective intelligence, vol. 6910, no. 1, pp. 182–212.
- S.Ribeiro,O.Ferret,and X.Tannier. 2017. *Unsupervised event clustering and aggregation from news wire and web articles*, in Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism, pp. 62–67.
- S. Yu and B. Wu. 2018. *Exploiting structured news information to improve event detection via dual-level clustering*, in IEEE Third International Conference on Data Science in Cyberspace, pp. 873–880.
- Chen, Y., Xu, L., Liu, K., Zeng, D., Zhao, J. 2015. *Event extraction via dynamic multipooling convolutional neural networks*. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, vol. 1, pp. 167–176.
- Zeng, Y., Yang, H., Feng, Y., Wang, Z., Zhao, D. 2016. *A convolution BiLSTM neural network model for Chinese event extraction*. In Natural Language Understanding and Intelligent Applications (pp. 275-287). Springer, Cham.
- Qi Li, Heng Ji, and Liang Huang. 2013. *Joint event extraction via structured prediction with global features*. In ACL.
- Nguyen, Thien Huu and Cho, Kyunghyun and Grishman, Ralph. 2016. *Joint Event Extraction via Recurrent Neural Networks*. Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.
- Hang Yang, Yubo Chen, Kang Liu, Yang Xiao, and Jun Zhao. 2018. *DCFEE: A Document-level Chinese Financial Event Extraction System based on Automatically Labeled Training Data*, In Proceedings of ACL 2018, System Demonstrations.
- Zheng Shun, Cao Wei, Xu Wei, Bian Jiang. 2019. *Doc2EDAG: An End-to-End Document-level Framework for Chinese Financial Event Extraction*, Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP).
- ZHONG Weifeng, YANG Hang, CHEN Yubo, LIU Kang, ZHAO Jun. 2019. *Document-level Event Extraction Based on Joint Labeling and Global Reasoning*, 33(9): 88-95,106.
- Devlin, J., Chang, M., Lee, K., Toutanova, K. 2019. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. ArXiv, abs/1810.04805.
- Liu, Shulin & Li, Yang & Zhang, Feng & Yang, Tao & Zhou, Xinpeng. 2019. *Event Detection without Triggers*. 735-744. 10.18653/v1/N19-1080.
- Ji Xin, Yankai Lin, Zhiyuan Liu, Maosong Sun. 2018. *Improving Neural Fine-Grained Entity Typing with Knowledge Attention*. The 32th AAAI Conference on Artificial Intelligence.
- Kingma, Diederik P. and Ba, Jimmy. 2015. *Adam: A Method for Stochastic Optimization*. In ICLR.
- Mikolov T, Sutskever I, Chen K, et al. 2013. *Distributed representations of words and phrases and their compositionality*. Advances in Neural Information Processing Systems, 26, 3111-3119.
- Yangarber R, Grishman R. 1997. *Customization of information extraction systems*. [C]// Proceedings of International Workshop on Lexically Driven Information Extraction: 1-11.
- Surdeanu M, Harabagiu S M. 2002. *Infrastructure for open-domain information extraction*. [C]// Proceedings of the 2nd International Conference on Human Language Technology Research. Morgan Kaufmann Publishers Inc: 325-330.
- Felix Jungermann, Katharina Morik. 2008. *Enhanced Services for Targeted Information Retrieval by Event Extraction and Data Mining*. [M]// Natural Language and Information Systems. Springer Berlin Heidelberg.
- Liao S, Grishman R. 2010. *Using Document Level Cross-Event Inference to Improve Event Extraction*. [C]// ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden. DBLP.

- WU Wentao, LI Peifeng, ZHU Qiaoming. 2019. *Joint Extraction of Entities and Events by a Hybrid Neural Network*. 33(8): 77-83.
- Yang Bishan and Mitchell Tom M. 2016. *Joint extraction of events and entities within a document context*. // Proceedings of NAACL- HLT 2016. San Diego, California, USA: Association for Computational Linguistics: 289-299

JCL 2020