

CCL24-Eval 任务5系统报告：基于增量预训练与外部知识的古文历史事件检测

康文军^{1,3} 左家莉^{1,2} 胡益裕² 王明文^{1,2}

¹江西师范大学 计算机信息工程学院 江西 南昌 330022

²江西师范大学 数字产业学院 江西 上饶 334000

³南昌理工学院 计算机信息工程学院 江西 南昌 330044

Email: wenjun_kang@qq.com,zjl@jxnu.edu.cn,yiyu.hu@qq.com,mwwang@jxnu.edu.cn

摘要

古文历史事件检测任务旨在识别文本中的事件触发词和类型。为了解决传统pipeline方法容易产生级联错误传播，以及大多数事件检测方法仅依赖句子层面信息的问题，本文提出了一种结合外部信息和全局对应矩阵的联合抽取模型EIGC，以实现触发词和事件类型的精确抽取。此外，本文还整理了一个包含“二十四史”等古汉语文献的数据集，共计约97万条古汉语文本，并利用该文本对BERT-Ancient-Chinese进行增量预训练。最终，本文所提出的模型在三个任务上的总F1值达到了76.2%，验证了该方法的有效性。

关键词： 古文历史事件检测；增量预训练；外部知识

System Report for CCL24-Eval Task 5: Historical Event Detection in Ancient Texts Based on Incremental Pre-training and External Knowledge

Wenjun Kang^{1,3} Jiali Zuo^{1,2} YiYu Hu² Mingwen Wang^{1,2}

¹School of Computer and Information Engineering, Jiangxi Normal University, Nanchang, Jiangxi 330022, China

²School of Digital Industry, Jiangxi Normal University, Shangrao, Jiangxi 334000, China

³School of Computer and Information Engineering, Nanchang Institute of Technology, Nanchang, Jiangxi 330044, China

Email: wenjun_kang@qq.com,zjl@jxnu.edu.cn,yiyu.hu@qq.com,mwwang@jxnu.edu.cn

Abstract

The task of historical event detection in ancient texts is to identify event trigger words and types in the text. To solve the problem that traditional pipeline methods are prone to cascading error propagation and most event detection methods only rely on sentence-level information, we proposed a joint extraction model, EIGC, which combines external information and global correspondence matrix to achieve accurate extraction of trigger words and event types. In addition, we also compiled a dataset containing ancient Chinese literature, such as the Twenty-four Histories, with a total of about 970,000 ancient Chinese texts, and used the texts to incrementally pre-train BERT-Ancient-Chinese. Finally, the overall F1 value of the proposed model reaches 76.2% on the three tasks, which proves the effectiveness of the method.

Keywords: historical events test in ancient languages , incremental pre-training , external knowledge

©2024 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(61866018,62266023)

通讯作者：左家莉

1 引言

事件抽取 (Event Extraction) 是信息抽取的重要任务之一, 旨在从自然语言文本中识别和提取相关事件信息的过程, 它在知识图谱构建、阅读理解、问答系统等多个领域有着广泛的应用。具体而言, 事件抽取可以被分为两部分: 即事件检测任务 (触发词识别、事件类型分类) 和事件论元抽取任务 (事件论元识别和事件论元角色分类)。其中, 事件检测任务主要识别事件触发词并判断其所属的事件类型, 事件论元抽取则是识别出事件的参与者 (如行为主体、对象、时间、地点等) 以及它们之间的关系(吴旭等人, 2023)。

古汉语作为中华文明的重要载体, 其文献记载了丰富的历史和文化信息。然而, 由于古文句法、语义复杂, 使用范围小, 针对古汉语的信息抽取任务仍然面临着较大挑战。本文基于古文历史事件类型抽取评测任务提供的数据集, 对古文历史事件检测任务进行研究。针对传统pipeline方法容易产生级联错误传播, 以及大多数事件检测方法仅依赖句子层面信息的问题, 本文设计了一个结合外部信息和全局对应矩阵的古文历史事件触发词和类型联合抽取模型 (External Information and Global Corresponding Matrix Based Joint Trigger Words and Types of Ancient Historical Events, EIGC), 用于古文历史事件检测任务研究。

2 相关工作

经典的事件抽取方法大致包括基于模式匹配和基于机器学习。然而, 基于模式匹配需要领域专家设计模板, 从而使得该方法的泛化性较低, 基于机器学习则涉及复杂的特征工程。随着深度学习技术的不断进步, 基于深度学习的方法已经成为当前事件抽取研究的主流方法(王浩畅等人, 2023)。Chen等人(2015)提出一个动态多池化的卷积神经网络 (DMCNN), 以捕获句子级别的信息, 并且动态地保留多个重要信息。Feng等人(2016)提出了一种结合LSTM和CNN的混合模型, 该模型通过从特定上下文中提取序列和块级特征来增强语义信息, 并利用这些特征来训练多分类器, 从而提高了事件检测的性能。

目前, 基于深度学习的事件抽取方法可分为pipeline式抽取和联合抽取。在事件检测任务中, pipeline式抽取首先识别出事件的触发词, 随后依据这些触发词判断其相应的事件类型。然而, 这种方法容易导致错误的级联传播。联合抽取则将这两个子任务建模为一个统一的联合学习框架, 这种方法能够充分利用触发词与事件类型之间的潜在联系, 进而促进触发词和事件类型抽取效果的相互提升(李华昱等人, 2022)。Chen等人(2012)提出了一种联合抽取的方法, 将事件抽取任务分解为两个联合抽取任务, 并利用丰富的语言学特征来提取中文事件。贺瑞芳等人(2019)构建了一个基于CRF的多任务学习模型, 用于中文事件的联合抽取, 有效缓解了分类训练后的语料稀疏问题。Nguyen等人(2012)提出一种基于深度学习中共享的隐藏层表示的联合模型, 用以预测实体类型、触发词和事件论元角色, 显著提高了中文事件抽取的性能。本文采用全局对应矩阵的方法, 分别设计了触发词头尾对齐矩阵和触发词与事件类别对齐矩阵, 以实现触发词与事件类型的联合抽取。

大多数现有的事件检测方法主要依赖于句子层面的信息, 但在许多情况下, 句子层面的信息并不足以提供充分的信息来准确推断出事件类型(李华昱等人, 2022)。因此, 为了提高事件检测的准确性, 有必要考虑引入上下文信息, 以提供更丰富的语境。Lou等人(2021)提出一种多层双向网络 (MLBiNet), 以同时捕获篇章级的事件关联和语义信息。Veysseh等人(2021)提出一种策略, 动态地从文档中只选择与目标句子最相关的上下文句子, 然后将这些句子输入到BERT中进行事件检测, 以更有效地捕获长距离文档级上下文信息。为了增强模型对事件类型的理解和推断能力, 本文为每条输入文本选择了对应的上下文句子, 并设计信息融合模块, 以建模输入句子与其上下文之间关系。

自2018年来, Google、百度等公司发布了包括BERT(Devlin et.al, 2019)、ERNIE(Zhang et.al, 2019)在内的多种模型, 并在事件抽取等几乎所有自然语言处理任务中都表现出色。受基于领域自适应训练思想的启发, 一些研究人员考虑使用电子化古籍资源对BERT进行继续预训练, 以增强预训练语言模型在特殊领域的处理能力, 并取得了一系列优秀的成果, 例如GuwenBERT¹、SikuBERT(王东波等人, 2022)、BERT-Ancient-Chinese(Wang等人, 2022)等, 极大地推动了古籍智能化的发展。本文整理了包括“二十四史”在内的古汉语文献, 共计约97万条古汉语文本, 并利用该文本对BERT-Ancient-Chinese进行了增量预训练。

¹<https://github.com/Ethan-yt/guwenbert>

3 模型

3.1 任务定义

古文历史事件类型抽取评测任务的详细描述如下：针对给定的古汉语文本，本任务的目标是识别文本中所有的事件触发词及其对应的事件类型。以句子“进军建德，擒贼帅赵桑干。”为例，其中“进军”一词可视为“派兵到建德”事件的触发词，而“擒”则作为“抓住敌方的将领赵桑干”事件的触发词。因此，在此句子中，触发词分别为“进军”和“擒”，它们分别代表了“军事-备战-出兵”和“军事-作战-俘虏”这两种事件类型。

该评测任务包含两个子任务：（1）触发词识别：此子任务需识别文本中的事件触发词并标注其位置。触发词是指最能代表事件发生的词语，一般为句中的谓语动词（其他句子成分皆可）。（2）事件类型判别：此子任务需参考给定的事件类型体系，为每个触发词确定其所属的事件类型。子任务二进一步细分为粗粒度（9大类）事件类型判别和细粒度（67小类）事件类型判别两个部分。

3.2 EIGC模型

本文设计了一个结合外部信息和全局对应矩阵的古文历史事件触发词和类型联合抽取模型EIGC（图1），用以建模上述子任务。该模型采用全局对应矩阵的方法，分别设计了触发词头尾对齐矩阵和触发词与事件类别对齐矩阵，以实现触发词与事件类型的抽取。考虑到触发词通常是句中的谓语动词（或其他句子成分），本文在模型中引入了词性信息，并通过字与词性全局对应矩阵的方式来建模二者的关系。此外，通常情况下，由于单个句子的信息不够丰富，只考虑单个句子的信息不足以推断出事件类型，本文还引入了文本上下文信息，以提升模型对输入句子的理解能力。

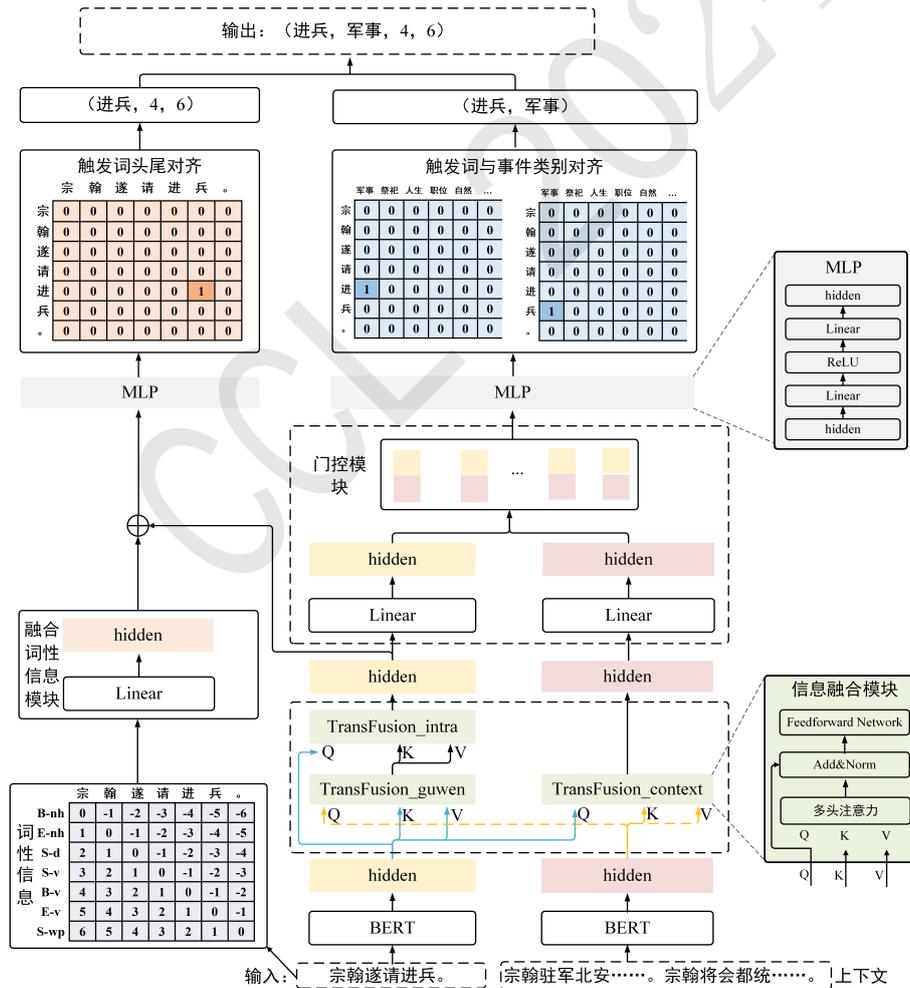


Figure 1: EIGC模型整体架构

3.2.1 编码层

对于给定的输入句子 S , $S = \{s_1, s_2, \dots, s_n\}$, s_n 表示句子 S 中第 n 个字。本文采用BERT作为编码器, 以获得 S 中每个字对应的向量表示。

$$\{h_1, h_2, \dots, h_n\} = BERT(\{s_1, s_2, \dots, s_n\}) \quad (1)$$

其中, $\{h_1, h_2, \dots, h_n\}$ 表示BERT最后一层输出的隐藏层状态, n 为输入文本的长度。

3.2.2 解码层

在本部分, 本文将详细介绍四个主要模块: 引入上下文信息模块、融合词性信息模块、触发词头尾对齐以及触发词与事件类别对齐模块。

(1) 引入上下文信息模块

对于给定的上下文文本 C , $C = \{c_1, c_2, \dots, c_l\}$, c_l 表示文本 C 中第 l 个字, 本模块同样利用BERT, 以获得上下文文本 C 中每个字对应的向量表示。

$$\{h_{-c_1}, h_{-c_2}, \dots, h_{-c_l}\} = BERT(\{c_1, c_2, \dots, c_l\}) \quad (2)$$

其中, $\{h_{-c_1}, h_{-c_2}, \dots, h_{-c_l}\}$ 表示BERT最后一层输出的隐藏层状态, l 为上下文句子文本的长度。

信息融合模块: 为了建模输入句子与其上下文之间, 以及输入句子中字与字之间的关系, 本文设计了三个信息融合模块 (Transformer-Based Information Fusion Module, TransFusion) (图2), 它主要由三个部分组成: 多头注意力层、残差网络和前馈神经网络。

$$Q_i, K_i, V_i = W_Q \cdot h_i^Q, W_K \cdot h_i^K, W_V \cdot h_i^V \quad (3)$$

$$Att_i = Softmax(Q_i \cdot K_i^T) \cdot V_i \quad (4)$$

$$LN(Att_i) = Norm(Att_i) + Q_i \quad (5)$$

$$FFN(LN(Att_i)) = W_2(ReLU(W_1 \cdot LN(Att_i) + b_1)) + b_2 \quad (6)$$

其中, W_Q 、 W_K 、 W_V 、 W_1 、 W_2 、 b_1 、 b_2 为可学习参数。在不同信息融合模块, 模型分别使用了不同来源的 Q (Query)、 K (Key)和 V (Value)。

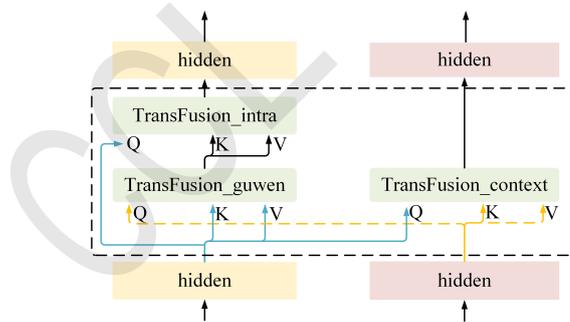


Figure 2: 信息融合层

在获取相应的 Q 、 K 、 V 之后, 模型通过以下过程生成对应的隐藏层表示。

$$O_n^1 = \text{TransFusion}_{\text{guwen}}(h_{-c_l}^Q, h_n^K, h_n^V) \quad (7)$$

$$O_n^2 = \text{TransFusion}_{\text{context}}(h_n^Q, h_{-c_l}^K, h_{-c_l}^V) \quad (8)$$

$$O_n^3 = \text{TransFusion}_{\text{intra}}(h_n^Q, h_{-c_l}^K, h_{-c_l}^V) \quad (9)$$

其中, $h_{-c_l}^Q$ 、 $h_{-c_l}^K$ 、 $h_{-c_l}^V$ 分别表示该上下文第 l 个字对应的 $hidden$ 作为 $TransFusion$ 模块中的 Q 、 K 、 V , h_n^Q 、 h_n^K 、 h_n^V 分别表示该上下文第 n 个字对应的 $hidden$ 作为 $TransFusion$ 模块中的 Q 、 K 、 V 。

(2) 融合词性信息

为了建模输入句子中每个 $token$ 和词性之间的关系，本文设计了输入 $token$ 和对应词性的对齐矩阵，并通过可学习参数让模型学习该矩阵所蕴含的 $token$ 和词性间的关系信息。

$$Pos_{i,j} = I_{i,j}W_{pos} + b_{pos} \quad (10)$$

$$h_{i,j}^{pos} = Concat(O_i^3, O_j^3) + Pos_{i,j} \quad (11)$$

其中， I 为图1中的词性信息矩阵， $I_{i,j}$ 表示输入句子中第 i 个 $token$ 和第 j 个词性的对应， O_i^3 和 O_j^3 为 $TransFusion_context$ 输出的句子中第 i 个字和第 j 个字对应的隐藏层表示， $Concat$ 表示拼接， W_{pos} 和 b_{pos} 为可训练参数。

(3) 触发词头尾对齐

为了建模触发词，本文设计了一个基于全局对应矩阵的触发词头尾对齐模块（图3）。该模块的目标是抽取出所有可能的触发词。具体过程如下：对于输入文本 S ， $S = \{s_1, s_2, \dots, s_n\}$ ，该矩阵的维度为 $n \times n$ ，其中每个位置表示一对（触发词开始 $token$ ，触发词结束 $token$ ），该位置上的分数表示存在该（触发词开始 $token$ ，触发词结束 $token$ ）组合的概率，分数越高，概率越大。为了获得所有可能的（触发词开始 $token$ ，触发词结束 $token$ ）组合，本文同样为每个位置上的分数设置了阈值，当该位置的分数大于该阈值时，则认为该位置对应的（触发词开始 $token$ ，触发词结束 $token$ ）组合存在，否则认为该组合不是（触发词开始 $token$ ，触发词结束 $token$ ）组合。

$$h_{i,j}^{pos} = MLP(h_{i,j}^{pos}) \quad (12)$$

$$P_{i,j}^{h-t} = \sigma(W_{h-t}h_{i,j}^{pos} + b_{h-t}) \quad (13)$$

其中， $h_{i,j}^{pos}$ 为融合词性信息模块输出的隐藏层表示， W_{h-t} 和 b_{h-t} 为可训练参数， σ 为sigmoid函数， MLP 为多层感知机。

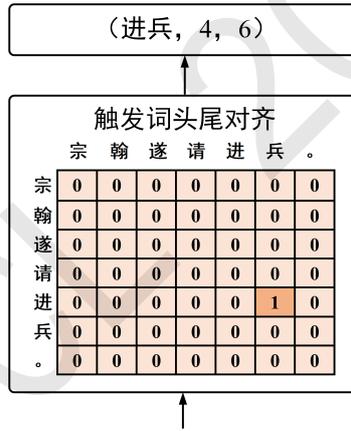


Figure 3: 触发词头尾对齐矩阵

(4) 触发词与事件类别对齐

本文将触发词与事件类别对齐细化为两个子任务：触发词开始 $token$ 和事件类别的全局对应，以及触发词结尾 $token$ 和事件类型的全局对应。为此，本文设置了触发词开始 $token$ 和事件类别的全局对应矩阵（图5左半部分）和触发词结尾 $token$ 和事件类型的全局对应矩阵（图5右半部分）。

通过 $TransFusion_context$ 和 $TransFusion_guwen$ ，模型分别获得了输入句子与上下文信息进行交互后的信息的表示和输入句子间进行交互的信息表示。由于上述两种信息对事件类别对齐的贡献不同，我们采用了门控机制（图4）对上述信息进行控制。

$$g = \sigma(W_3 \cdot O_i^3 + W_4 \cdot O_i^2 + b_{gate}) \quad (14)$$

$$u_i = [g \circ O_i^3] \oplus [(1 - g) \circ O_i^2] \quad (15)$$

其中， O_i^3 为输出的隐藏层表示， O_i^2 为 $TransFusion_guwen$ 输出的隐藏层表示， W_3 、 W_4 、 b_{gate} 为可学习参数， \circ 表示元素相乘。

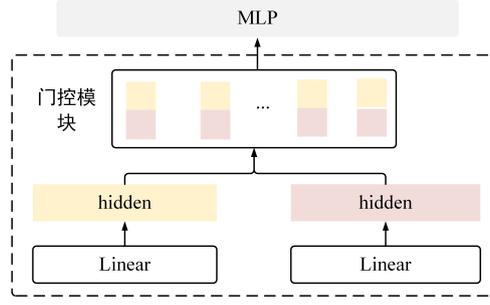


Figure 4: 门控模块

在获得融合后的 u_i 表示后，我们将其作为触发词与事件对齐矩阵解码部分（图5）的输入。

$$h_{i,j}^{et} = \text{MLP}(u_i^{3.2}) \quad (16)$$

$$p_{i,j}^{ht} = \sigma(h_{i,j}^{et}W_{ht} + b_{ht}) \quad (17)$$

$$p_{i,j}^{tt} = \sigma(h_{i,j}^{et}W_{tt} + b_{tt}) \quad (18)$$

其中， W_{ht} 、 W_{tt} 、 b_{ht} 、 b_{tt} 为可训练的参数， MLP 为多层感知机。

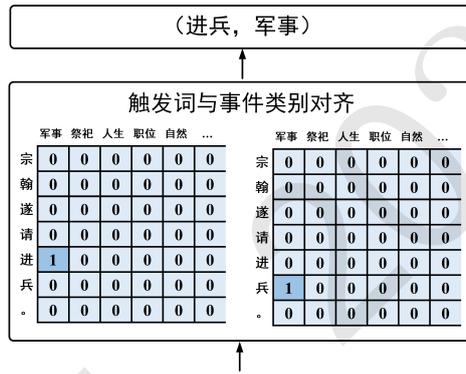


Figure 5: 触发词和事件类型全局对应

3.3 损失函数

该模型采用交叉熵损失函数作为模型的损失函数。模型最终的损失 l^{total} 包含了 l^{et} 和 l^{ht} 两部分。在模型训练过程中，本文将公式中的系数设置为 $\rho=\varphi=1$ 。

$$l^{total} = \rho l^{et} + \varphi l^{ht} \quad (19)$$

首先是 l^{et} ，其包括了 l^{eth} 、 l^{ett} ，两者具体计算过程如下：

$$l^{et} = \alpha l^{eth} + \beta l^{ett} \quad (20)$$

$$l^{eth} = \frac{-1}{n \times n^t} \sum_{i=1}^n \sum_{j=1}^n y_{i,j} \log p_{i,j}^{th} + (1 - y_{i,j}) \log(1 - p_{i,j}^{th}) \quad (21)$$

$$l^{ett} = \frac{-1}{n \times n^t} \sum_{i=1}^n \sum_{j=1}^n y_{i,j} \log p_{i,j}^{tt} + (1 - y_{i,j}) \log(1 - p_{i,j}^{tt}) \quad (22)$$

其中， $p_{i,j}^{th}$ 表示第 i 个 $token$ 为触发词开始 $token$ 和第 j 个事件类型是否存在联系的条件概率； $p_{i,j}^{tt}$ 表示第 i 个 $token$ 为触发词结束 $token$ 和第 j 个事件类型是否存在联系的条件概率。在训练过程中， α 和 β 被设置为0.5。

其次是 l^{ht} ，其为触发词头尾全局对应矩阵对应的损失。

$$l^{ht} = \frac{-1}{n \times n} \sum_{i=1}^n \sum_{j=1}^n y_{i,j} \log p_{i,j}^{ht} + (1 - y_{i,j}) \log(1 - p_{i,j}^{ht}) \quad (23)$$

其中， $p_{i,j}^{ht}$ 表示第 i 个 $token$ 为触发词开始 $token$ 以及第 j 个 $token$ 为触发词结尾 $token$ 的条件概率。在上述公式中， n 为输入序列的长度， n^t 为定义的事件类型数。

3.4 增量预训练

为确保模型能够充分理解古汉语的表达方式和文化内涵，本文基于东北大学²公布的“二十四史”等古汉语文献，共计约97万条古汉语文本，对BERT-Ancient-Chinese进行继续预训练。

在进行增量预训练时，本文沿用了BERT预训练方法中的掩码机制，随机掩码古汉语句子里15%的字符，其中80%的字符被替换为“[MASK]”，10%的字符被替换为词表中的任意一个词，剩余10%保持不变。此外，本文还设置了batch_size为64，文本最大长度为128，epoch为300，以确保训练过程的稳定性和有效性。增量预训练过程如图6所示。

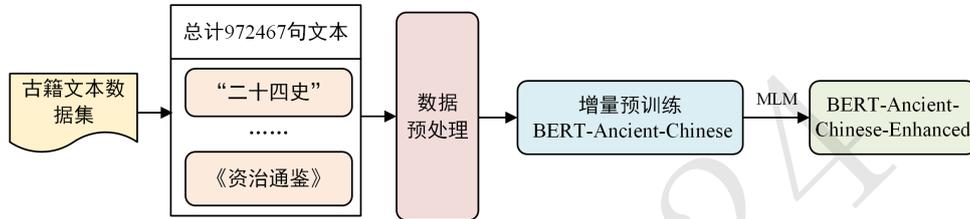


Figure 6: 增量预训练过程图

3.5 词性和上下文的获取与处理

(1) 上下文

本文依据“sen_id”和“doc_id”对语料进行了重新排序，以便构建篇章结构，从而获取古文文本的前后上下文。例如“宗翰遂请进兵。”，其前后上下文分别是“宗翰驻军北安，使希尹经略近地，获辽护卫耶律习泥烈，知辽主猎于鸳鸯泺。”和“宗翰将会都统杲于奚王岭。”。上下文的获取与处理如图7所示。

```
{ "sen_id": 7778, "doc_id": 32047, "text": "宗翰驻军北安，使希尹经略近地，获辽护卫耶律习泥烈，知辽主猎于鸳鸯泺。"}
{ "sen_id": 7779, "doc_id": 32047, "text": "宗翰遂请进兵。"}
{ "sen_id": 7780, "doc_id": 32047, "text": "宗翰将会都统杲于奚王岭。"}

```

Figure 7: 上下文的获取与处理（“sen_id”表示句子序号，“doc_id”表示文档编号）

(2) 词性

本文使用了jiayan工具³对古文文本进行词性标注，以获取相应的词性标签。具体而言，对于一个给定的古文文本 $S = \{w_1, w_2, w_3, \dots, w_n\}$ ，通过使用jiayan进行分词处理，可以得到 k 个词，即 $D = \{d_1, d_2, d_3, \dots, d_k\}$ ，其中 d_i 表示第 i 个词，且 D 中的每个元素 d_i 均对应一个词性标签(Yu 等人, 2023)。

例如“宗翰遂请进兵。”，经过分词后，我们得到了“宗翰”、“遂”、“请”、“进兵”和“。”这五个词或标点符号，它们各自的词性标签分别为“nh（人名）”、“d（介词）”、“v（动词）”、“v（动词）”和“wp（标点符号）”。详细的词性分类可见jiayan中的词性表。

在实验中，我们采用了“B/M/E/S-词性”的组合标签方式来表示词性，其中B（Begin）表示词汇的第一个汉字，M（Middle）表示词汇中间的汉字，E（End）表示词汇的最后一个汉字，而S（Single）则表示仅包含一个汉字。词性标注示例如图8所示。

²<https://github.com/NiuTrans/Classical-Modern>

³<https://github.com/jiaeyan/Jiayan>

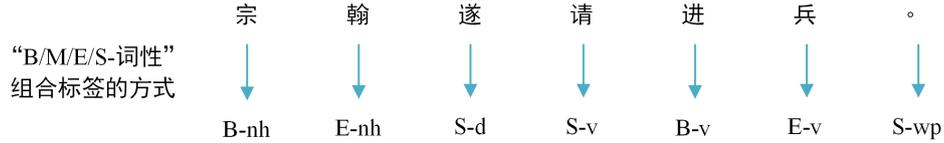


Figure 8: 词性的获取与处理

4 实验

4.1 数据集

本文主要使用古文历史事件类型抽取评测任务提供的数据集⁴对模型进行微调。该数据集共包含8122条古文文本，其中训练集5650条，验证集1218条，测试集1254条。事件类型体系涵盖9大类、67小类。

4.2 实验设置与评估指标

实验设置：本文实验运行环境采用ubuntu18.04系统、python3.7、CUDA版本11.3、以及显卡为RTX3090。此外，本文将epoch为100，触发词与事件类别对齐模块和触发词头尾对齐的阈值设置为0.5，训练和测试时batch size分别为8和6，输入文本最大长度为128。

评估指标：触发词识别任务的评估采用两个指标：Exact-match-score和Subset，这两个指标的平均值作为最终的评估结果。事件类型判别任务的评估采用宏平均（macro-average）和微平均（micro-average）两个指标，这两个指标的平均值作为最终的评估结果。

4.3 实验结果

(1) 在无答案测试集上的实验结果

如表1所示，本文所提出的模型在所有参赛队伍中排名第二，三个任务上的总F1值达到了76.2%。

排名	任务1	任务1*0.4	任务2	任务2*0.3	任务3	任务3*0.3	总分
1	0.763	0.305	0.842	0.253	0.779	0.234	0.791
2	0.703	0.281	0.825	0.248	0.777	0.233	0.762
3	0.700	0.280	0.824	0.247	0.771	0.231	0.758
4	0.672	0.269	0.810	0.243	0.769	0.231	0.742
5	0.690	0.276	0.799	0.240	0.752	0.226	0.741
6	0.674	0.270	0.808	0.242	0.739	0.222	0.734

Table 1: 评测总分排名

(2) 在验证集上的消融实验结果

为了验证本文所提出模型中各个模块的有效性，我们在验证集上进行了消融实验。在触发词识别、粗粒度事件类型判别和细粒度事件类型判别三个任务上，我们仅采用了微平均作为评估指标。

如表2所示，在去除增量预训练的消融实验中，三个任务的F1值分别下降了0.6%、1.4%和0.5%。这可能是由于通过在“二十四史”等古汉语文献的数据集上对BERT-Ancient-Chinese进行增量预训练，为模型提供了丰富的基于“二十四史”的领域知识，从而帮助模型能够更好地理解和处理古代汉语中的词汇、语法及表达方式，提升了模型在相关任务上的表现。

在去除词性信息的消融实验中，三个任务的F1值分别下降了0.8%、1.2%和0.5%。相较于在其他三个模块上的效果，触发词识别任务在该模块上下降的最多。这可能是由于触发词通常是句中的谓语动词（或其他句子成分），通过全局对应矩阵来建模输入文本中每个字与词性间的联系，能有效提升模型对古文文本中触发词识别的精度。

在去除门控模块或上下文信息的消融实验中，从粗粒度事件类型判别任务可以看出，当去除门控模块后，F1值下降了0.9%；当去除上下文信息后，F1值下降了0.2%。这可能是由于门控

⁴<https://github.com/NLPInBLCU/CHED2024>

模块能有效地筛选出输入文本和引入的上下文信息中与事件类别判别最相关的信息，从而提升模型对粗粒度事件类型判别的性能。然而，在触发词识别和细粒度事件类型判别任务中，当去除门控模块后，两个任务的F1值基本没发生变化；当去除上下文信息后，两个任务的F1值分别下降了0.6%和0.5%。这可能是由于在这两个任务中，模型已经通过其他模块（如上下文信息和词性信息）充分捕获了必要的信息，从而降低了门控模块对整体性能的影响。

	触发词识别			粗粒度事件类型判别			细粒度事件类型判别		
	P	R	F1	P	R	F1	P	R	F1
EIGC	0.816	0.817	0.817	0.784	0.813	0.798	0.767	0.768	0.768
-增量预训练	0.803	0.818	0.811	0.778	0.789	0.784	0.756	0.770	0.763
-词性信息	0.798	0.819	0.809	0.766	0.808	0.786	0.753	0.773	0.763
-门控模块	0.814	0.821	0.817	0.776	0.803	0.789	0.766	0.772	0.769
-上下文	0.796	0.826	0.811	0.792	0.801	0.796	0.749	0.778	0.763

Table 2: 在验证集上的消融实验结果

5 总结与展望

针对传统pipeline方法容易产生级联错误传播，以及大多数事件检测方法仅依赖句子层面信息而忽略上下文关联性的问题，本文提出了一种融合外部信息和全局对应矩阵的模型EIGC，旨在联合抽取古文历史事件的触发词及类型。实验结果表明，该模型在三个任务上均实现了显著的性能提升，总F1值得分高达76.2%。

未来，我们将探讨如何整合更多外部信息，例如现代文翻译，以进一步提升模型性能。此外，考虑到大型语言模型展现出的强大世界知识掌握能力和上下文理解能力，我们也计划研究如何利用ChatGPT⁵、荀子⁶等大语言模型，以进一步提升古文历史事件检测任务的性能。

参考文献

- 吴旭, 卞文强, 颀夏青, 孙利娟. 2023. 机器阅读理解式中文事件抽取方法. 计算机工程与应用, 59(16): 93-100.
- 李华昱, 毕经纶, 闫阳. 2022. 限定域中文事件抽取研究综述. 计算机工程与应用, 58(18): 43-58.
- 王浩畅, 周柳莲, Marius Gabriel PETRESCU. 2023. 基于深度学习的事件抽取研究综述. 软件学报, 34(08): 3905-3923.
- Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event Extraction via Dynamic Multi-Pooling 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*, pages 167-176.
- Xiaocheng Feng, Lifu Huang, Duyu Tang, Heng Ji, Bing Qin, and Ting Liu. 2016. A Language-Independent Neural Network for Event Detection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 66-71.
- Chen Chen and Vincent Ng. 2012. Joint Modeling for Chinese Event Extraction with Rich Linguistic Features. In *Proceedings of COLING 2012*, pages 529-544.
- 贺瑞芳, 段绍杨. 2019. 基于多任务学习的中文事件抽取联合模型. 软件学报, 30(4): 1015-1030.
- Nguyen, Trung Minh, and Thien Huu Nguyen. 2019. Joint Modeling for Chinese Event Extraction with Rich Linguistic Features. In *Proceedings of the AAAI conference on artificial intelligence*, pages 6851-6858.
- Dongfang Lou, Zhilin Liao, Shumin Deng, Ningyu Zhang, and Huajun Chen. 2021. MLBiNet: A Cross-Sentence Collective Event Detection Network. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 4829-4839.

⁵<https://chat.openai.com>

⁶<https://github.com/Xunzi-LLM-of-Chinese-classics/XunziALLM>

- Amir Pouran Ben Veyseh, Minh Van Nguyen, Nghia Ngo Trung, Bonan Min, and Thien Huu Nguyen. 2021. Modeling Document-Level Context for Event Detection via Important Context Selection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5403–5413.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 4171–4186.
- Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced Language Representation with Informative Entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451.
- 王东波,刘畅,朱子赫,刘江峰,胡昊天. 2022. SikuBERT与SikuRoBERTa: 面向数字人文的《四库全书》预训练模型构建及应用研究. *图书馆论坛*, 42(06): 31-43.
- Pengyu Wang and Zhichen Ren. 2022. The Uncertainty-based Retrieval Framework for Ancient Chinese CWS and POS. In *Proceedings of the Second Workshop on Language Technologies for Historical and Ancient Languages*, pages 164–168.
- Zijian Yu, Tong Zhu, and Wenliang Chen. 2023. 基于句法特征的事件要素抽取方法(Syntax-aware Event Argument Extraction). In *Proceedings of the 22nd Chinese National Conference on Computational Linguistics*, pages 196–207.