计 算 机 研 究 与 发 展
Journal of Computer Research and Development
DOI: 10.7544/issn1000-1239.202220447
61(1): 184−195, 2024

基于分层潜在语义驱动网络的事件检测

肖梦南　贺瑞芳　马劲松
（天津大学智能与计算学部　天津　300350）
（天津市认知计算与应用重点实验室（天津大学）　天津　300350）
（mnxiao@tju.edu.cn）

Event Detection Based on Hierarchical Latent Semantic-Driven Network

Xiao Mengnan, He Ruifang, and Ma Jinsong
（*College of Intelligence and Computing*, *Tianjin University*, *Tianjin* 300350）
（*Tianjin Key Laboratory of Cognitive Computing and Application* (*Tianjin University*), *Tianjin* 300350）

Abstract　Event detection aims to detect triggers in sentences and classify them into pre-defined event types. The key factors lie in appropriately representing triggers. Existing representation-based methods learn the semantic representation of candidate triggers through complex deep neural networks to improve the performance of models. However, these methods ignore two important problems: 1) affected by sentence context, the same trigger can trigger different event types; 2) due to the diversity of natural language expression, different triggers can trigger the same event type. Inspired by hidden variables in the variational auto-encoder (VAE) and hierarchical structure in other natural language processing (NLP) tasks, we propose a hierarchical latent semantic-driven network (HLSD) for event detection to address the above two problems through latent semantic information of sentences and words. The model reduces the dimension from the text representation space to the new latent semantic space and explores the more essential influence information in the macro and micro context of events. Firstly, we get the representation of a sentence and the words through BERT. Secondly, a dual latent semantic mechanism is designed, and VAE is used to mine the latent semantic information at the sentence and word levels. Finally, from the perspective of different granularity contexts, a hierarchical structure from coarse to fine is proposed to make full use of the latent semantic information of sentences and words, to improve the performance of the model. The experimental results on ACE2005 corpus show that the *F*1 performance of the proposed method achieves 77.9%. In addition, we quantitatively analyze the above two problems in the experiment, which proves the effectiveness of our method.

Key words　latent semantic; hierarchical structure; variational auto-encoder; representation learning; event detection

摘　要　事件检测旨在检测句子中的触发词并将其分类为预定义的事件类型.如何有效地表示触发词是实现该任务的核心要素.目前基于表示的方法通过复杂的深度神经网络来学习候选触发词的语义表示,以提升模型性能.然而,其忽略了2个问题:1)受句子语境的影响,同一个触发词会触发不同的事件类型;2)受自然语言表达多样性的影响,不同的触发词会触发同一个事件类型.受变分自编码器中隐变量及其他自然语言处理(natural language processing, NLP)任务中分层结构的启发,提出基于分层潜在语义驱动

收稿日期: 2022-05-28; **修回日期**: 2023-01-04
基金项目: 国家自然科学基金项目(61976154); 国家重点研发计划项目(2019YFC1521200)
This work was supported by the National Natural Science Foundation of China (61976154) and the National Key Research and Development Program of China (2019YFC1521200).
通信作者: 贺瑞芳(rfhe@tju.edu.cn)

网络（hierarchical latent semantic-driven network, HLSD）的事件检测方法，通过句子和单词的潜在语义信息来辅助缓解以上2个问题. 模型从文本表示空间中分层降维到新的潜在语义空间，探索事件宏微观语境中更本质的影响信息. 首先，通过BERT对候选句子进行编码，得到句子的表示和句子中单词的表示；其次，设计一个双重的潜在语义机制，并采用VAE挖掘句子和单词级潜在语义；最后，从不同粒度的上下文角度，提出采用一个由粗到细的分层结构来充分使用句子和单词的潜在信息，从而提升模型的性能.ACE2005英文语料库上的实验结果表明，所提方法的$F1$值在事件检测任务上达到了77.9%. 此外，在实验部分对以上2个问题进行了定量分析，证明了所提方法的有效性.

关键词 潜在语义；分层结构；变分自编码器；表示学习；事件检测

中图法分类号 TP391

作为信息抽取的子任务，事件抽取旨在从大量非结构化的文本中抽取出用户感兴趣的事件，并以结构化的形式呈现，其可以进一步应用到事理图谱[1]、信息检索、自动文摘[2]、对话系统及故事生成[3]等下游任务中. 根据自动内容抽取（automatic content extraction, ACE）评测组织的定义，事件抽取包括4个任务：1）触发词识别. 二分类任务，判断候选词是否触发事件. 2）触发词分类. 多分类任务，已知候选词是触发词的情况下，判断其触发的事件类型. 3）事件论元识别. 二分类问题，判断候选实体是否是事件的参与者. 4）事件论元分类. 多分类任务，已知候选实体是事件参与者的情况下，判断其在事件中扮演的角色. 其中前2个任务可以合并为事件检测，后2个任务可以合并为事件论元抽取. 事件检测可形式化为一个单词级的多分类问题，旨在为句子中的每个单词分配1个所属的事件类型或非事件类型. 考虑到事件检测仍然面临着严峻的挑战，本文聚焦该任务并采用ACE2005英文数据集进行探索. 例1是一个标准的ACE事件标注样例，它包含一个由触发词“lost”触发的End-Position事件，该事件涉及一个事件元素Person（“two million Americans”）.

例1. In all, two million Americans（A1）have lost（E1）their jobs under President Bush so far, not to mention three of them being the top three leaders of his economic team.

传统基于特征的方法[4-9]主要是将单词的分类线索（如词法、句法、文档等信息）转换为特征向量，然后输入分类器中进行事件检测. 尽管此类方法在一定程度上有效，但是特征工程需要专家知识和丰富的外部资源，并且人为设计特征模板通常耗时耗力. 近几年，基于表示的神经网络方法在事件抽取中也取得了不错的成绩. 此类方法[10-23]主要将候选事件提及用词嵌入表示，通过神经网络学习更抽象的特征表示，并进行事件分类. 然而该类方法忽略了2个很重要的问题：

1）受句子语境的影响，同一个事件触发词会触发不同的事件类型. 以例1中的事件触发词“lost”为例，表1展示了ACE2005英文训练集中“lost”触发的事件类型、频数和上下文. 从表1可以看出，“lost”一共触发了8个事件，共包括4个事件类型. “lost”在这8个事件中的意思均为“失去”，但由于句子语境的不同，“lost”作为“失去”的意思触发的事件类型也不同. 由此可见，事件触发词触发的事件类型不仅和触发词本身有关，同样受其所在句子的语境的影响.

Table 1 Context of “lost” and the Event Type and Frequency It Triggers

表1 “lost”的上下文及其触发的事件类型、频数

事件类型	频数	触发词以及上下文
End-Position	1	lost their jobs
Injure	1	lost both his arms, his home
Die	5	lost their lives/lost your header/lost their lives/lost 13 more than/lost such a big part of her life
Elect	1	lost her seat in the 1997 election

2）受自然语言表达多样性的影响，不同的事件触发词会触发同一个事件类型. 在日常表达中，每个句子都对应一个潜在语义. 由于自然语言表达的多样性，使得相同的潜在语义有不同的外在表达. 对应到事件检测中，就是触发同一个事件可能对应的事件触发词有多个，将其定义为多词触发同一个事件类型问题. 以Transport事件类型为例，其在ACE2005英文数据集中一共出现603次，由233个不同的事件触发词触发，事件触发词包括“land”“got”“move”“arrive”等. 图1展示了ACE2005英文训练语料中每种事件出现的次数、触发事件的触发词的种类数以及二者的比值. 图1中“count”表示每种事件类型出现的次数，“unique”表示触发事件的触发词的种类数（相同的触发词看作一类），“ratio”表示“unique”和“count”的比值. 图1中，在ACE2005英文训练语料中，

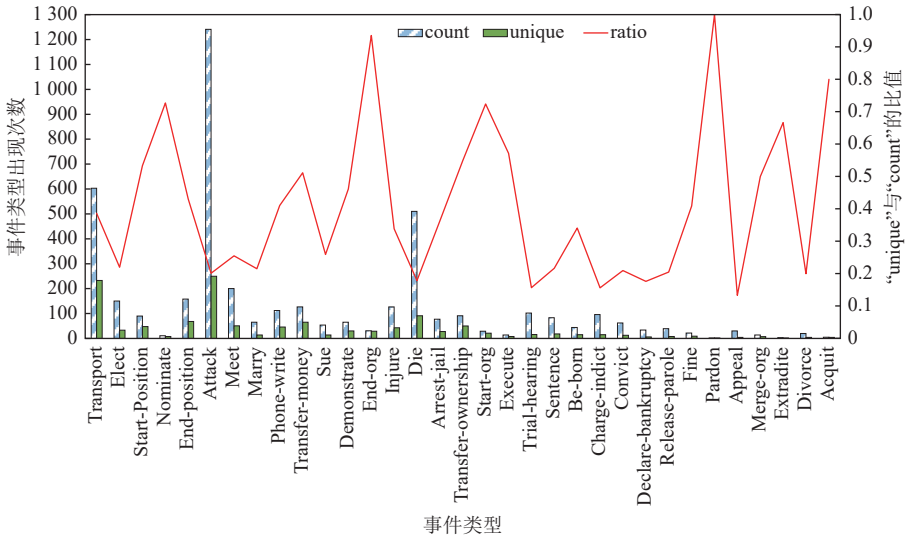


Fig. 1 Distribution of various events in ACE2005 English training dataset

图 1 ACE2005 英文训练数据集中各类事件的分布

多个词触发同一个事件类型的现象非常频繁, 平均(“unique”之和与“count”之和的比值)占比为 29.1%.

前人基于表示的模型在处理 1 词触发多个事件类型问题时, 将其当作歧义问题对待. 其中 Liu 等人[16]通过知识蒸馏的方式提高模型识别句子中实体的能力, 利用实体来辅助解决歧义问题, 但是需要对数据进行实体标注; Zhao 等人[19]通过有监督的方法来获取全局文档表示, 以此来增强文档中所有单词的表示; Chen 等人[20]通过注意力机制为每个句子选择不同的文档形式, 从而使得选择的句子和当前句子更相关. 从例 1 和表 1 可以看出, “lost”在句子中的意思是确定的, 其触发的事件类型不仅受触发词本身的影响, 还受到句子语境的影响. 如果我们考虑显式地将句子的语义作为特征来辅助进行分类, 那么就能为触发词增加更多的判别知识. 这对于只触发 1 次的事件类型是有帮助的, 对触发多次的事件类型虽然也能起到促进作用, 但不可避免地会带来一定的噪声, 这源于触发多次的事件的上下文不完全相同, 句子的语义也就不完全相同. 表 1 中“lost”一共触发了 5 次“Die”事件, 而这 5 个“Die”事件的上下文不尽相同, 因此句子的语义信息也不完全相同, 但它们触发的事件类型都是“Die”. 此种情况下, 只考虑浅层的句子语义信息, 将给触发词识别带来噪声. 因此我们更需要句子的潜在语义信息, 这样对于不同上下文的触发词触发同一个事件类型来说, 既能有效利用句子信息, 又可以减少不同上下文带来的噪声.

同样地, 对于多个事件触发词触发同一个事件类型的情况, 我们可以通过单词的潜在语义信息来缓解. 对应同一事件类型的不同的事件触发词经过上下文编码后得到的表示可能是不同的, 但若挖掘到单词的潜在语义信息, 并将其作为分类特征来辅助事件检测, 则具有相似潜在语义的单词, 被识别为同一个事件类型的概率就会变大. 为解决如上问题, 受变分自编码器(variational auto-encoder, VAE)中隐变量以及其他自然语言处理(natural language processing, NLP)任务中分层结构的启发, 本文提出了一种基于分层潜在语义驱动网络(hierarchical latent semantic-driven network, HLSD)的方法进行事件检测. 具体来讲, 我们首先利用 BERT 来对候选的事件提及进行编码, 得到句子的表示和句子中单词的表示, 并分别采用 VAE 建模得到不同粒度的潜在语义表示, 最后设计一个由粗到细的分层结构, 先利用句子的潜在语义信息再利用单词的潜在语义信息来完成事件检测. 本文的主要贡献有 3 个方面:

1)提出一种新的由粗到细的分层潜在语义驱动网络模型进行事件检测, 模型从文本表示空间中分层降维到新的潜在语义空间, 探索事件宏微观语境中更本质的影响信息.

2)利用变分自编码器捕获句子和单词的潜在语义信息. 据我们所知, 这是第 1 个显式考虑不同粒度潜在语义信息的事件检测, 且不需要对数据进行额外的标注.

3)在 ACE2005 英文数据集上的实验表明我们的模型是有效的, 在 1 词触发多个事件类型上, 模型的正确率有所提高, 在多个词触发同一个事件类型上, 模型识别的触发词的种类数有所提高.

1 相关工作

目前，事件检测主要归纳为3类：基于特征、基于表示和基于外部资源的方法.

1.1 基于特征的事件检测方法

该类工作将单词的分类线索如词法、句法、文档等信息转换为特征向量，然后输入分类器中进行事件检测. Ahn[4]使用词性特征（如单词全拼、词性标记）、句法特征（如依赖关系）以及额外的知识特征（如WordNet）. Ji等人[5]将事件抽取的范围从单个文档扩展到与其相关的多个文档，结合来自相关文档的全局信息和局部决策来进行事件检测. 为了从文本中捕获更多的线索，Liao等人[6]利用跨事件的方法以捕获事件之间的共现信息. Hong等人[7]则采用跨实体的方法，通过查询大规模语言资源来形成对实体的丰富描述. 此外，还有一些方法考虑词对之间的特征，Li等人[8]利用结构化感知机模型融入全局特征以显式地捕获触发词和事件论元之间的依赖关系；Liu等人[9]利用概率软逻辑方法编码事件之间的联系. 然而人力设计特征模板通常耗时耗力，且特征工程还需要专家知识和丰富的外部资源.

1.2 基于表示的事件检测方法

深度学习的进步极大地推动了事件检测的发展. 基于表示的方法将候选事件提及用词嵌入表示，再采用各种神经网络模型自动学习更抽象的特征表示，最后通过分类层来进行事件检测. 首先，Chen等人[10]和Nguyen等人[11-12]成功地将卷积神经网络（convolutional neural network, CNN）应用到事件检测中，并获得了较好的性能. 随后，Nguyen等人[13]用循环神经网络（recurrent neural network, RNN）来进行事件检测和事件论元抽取的联合学习. 此后Feng等人[14]利用CNN和RNN分别捕获与语言无关的短语块信息和文本序列信息，在中英文事件抽取任务上均获得了良好的效果. 此外，为了解决触发词的歧义性问题，Duan等人[15]采用文档级信息来改进句子级的事件检测；Liu等人[16]提出采用基于知识蒸馏的对抗模拟方法进行事件检测，提高模型从原始文本中获取抽象事件知识的能力，从而解决事件触发词的歧义问题，并且减少对外部自然语言处理工具的依赖.Yang等人[17]利用预训练语言模型进行事件抽取和生成新的训练数据，在事件检测任务上取得了不错的性能.

为提高事件检测模型的泛化能力和鲁棒性，一些工作引入知识增强的方法. Lu等人[18]提出一种Delta表示学习方法，力图将事件表示解耦为词汇具体部分和词汇无关部分，而不是仅学一个单独完整的表示，以蒸馏出事件的判别和泛化知识. Nguyen等人[21]在句子的依存分析图上使用图卷积网络，并提出一种新的池化方式来聚合卷积向量，用于事件类型预测，取得了良好的性能. Liu等人[22]将事件抽取转化为机器阅读理解任务，将需要抽取的事件类型和论元通过无监督方法转化为问题，借助阅读理解模型来促进事件抽取. Lai等人[23]基于图卷积网络引入门控机制来整合语法信息，提升事件检测性能.

整体上，基于表示的事件检测方法很难解决数据稀疏和分布不平衡的问题.

1.3 引入外部资源的事件检测方法

FrameNet中定义的框架（frame）与ACE事件抽取任务中的事件存在高度相似的结构，其包含的语法单元与框架元素分别对应触发词和事件论元. 因此，Liu等人[24]首先建立FrameNet中的框架与ACE2005中的事件间的匹配对应关系，进而得到FrameNet事件数据集，然后结合ACE2005数据集对模型进行联合训练；Chen等人[25]利用远程距离监督的方法结合Freebase, FrameNet, Wikipedia生成一批高质量的标注数据以辅助事件检测. Liu等人[26]利用文本注意力机制捕获多语数据的一致性信息以缓解数据的稀疏性，同时采用门控跨语言注意力机制捕获多语数据的补充性信息以应对单语数据的歧义性. Wang等人[27]基于Freebase建立一个覆盖率良好的大型事件相关候选集，然后利用对抗训练机制从候选数据集识别出信息实例，并过滤掉有噪声的实例；Tong等人[28]借助WordNet开放域触发词知识，设计一种教师-学生网络模型，实现对事件触发词的知识感知表示，从而提高模型性能. Wang等人[29]采用对比学习在大规模无标记语料上基于抽象语义表示（abstract meaning representation, AMR）进行预训练，利用无标注数据中的事件知识来提高事件抽取性能.

然而，引入外部资源得到的数据，质量很难保证，并带来一定的噪声. 本文尝试在不引入外部资源、不使用额外标注信息的情况下，从不同的潜在语义空间角度，探索事件宏微观语境中更本质的影响信息.

1.4 变分自编码器和分层结构

变分自编码器[30]是将变分分布引入自编码器的一种生成模型，其从重构的角度挖掘出数据的代表性潜在信息，在自然语言处理中取得了不错的效果. 例如Miao等人[31]以及Srivastava等人[32]将变分自编

码器运用于主题模型中, 获得一致性更佳的主题.Xu 等人[33] 在中文隐式篇章关系识别任务中考虑变分自编码器中的潜在语义变量, 获得了更好的性能. 在本文中我们试图通过 VAE 中的潜在语义变量捕获句子和单词的潜在语义. 如何合理地利用不同粒度的语境进行事件检测成为一个关键问题.

分层结构在许多自然语言处理任务上取得了比较好的结果. 例如, Yang 等人[34] 通过层次化的注意力网络在文本分类任务上取得了良好的性能; Xiong 等人[35] 在 Yang 等人[34] 的基础上使用层次输出层, 取得了更好的性能. 考虑语言的多样性, Pappas 等人[36] 通过多任务学习训练多语言层次注意力网络, 用以学习文档结构, 提高模型的迁移能力. 受分层结构的启发, 本文将从 VAE 中捕获的不同粒度语境信息, 采用分层结构进行事件检测; 不同于前人由细粒度到粗粒度的分层结构, 本文采用一个由粗粒度到细粒度的策略. 这源于事件检测是一个细粒度的单词级分类任务, 而文本分类是粗粒度的句子级/篇章级分类任务.

2 任务描述

与前人工作[16] 一致, 我们将事件检测建模为一个单词级分类任务, 即我们将句子中的每个单词都看作候选触发词, 判断其触发的事件类型, 对于未触发事件的单词, 其类别设置为 None.下面对 ACE2005 语料中出现的概念进行简要描述.

1)实体(entity). 包含某种特定语义类别的对象或对象集合, 例如人(PER)、组织机构(ORG)等.

2)事件提及(event mention). 发生事件的短语或句子, 包含 1 个触发词和任意数量的论元.

3)触发词(trigger). 代表事件发生的核心词, 大多数为动词或名词.

4)事件类型(event type).ACE2005 语料中共包含 8 种事件类型和 33 种事件子类型.

5)论元(argument). 在事件中扮演一个角色, 可能是实体提及、时间表达式或值.

6)论元角色(argument role). 每一种事件子类型均对应着一定数量的角色, ACE2005 语料共涉及 35 种角色, 但没有一种事件类型可以有 35 种角色.

3 HLSD 研究框架

图 2 展示了本文提出的 HLSD 模型框架, 其主要

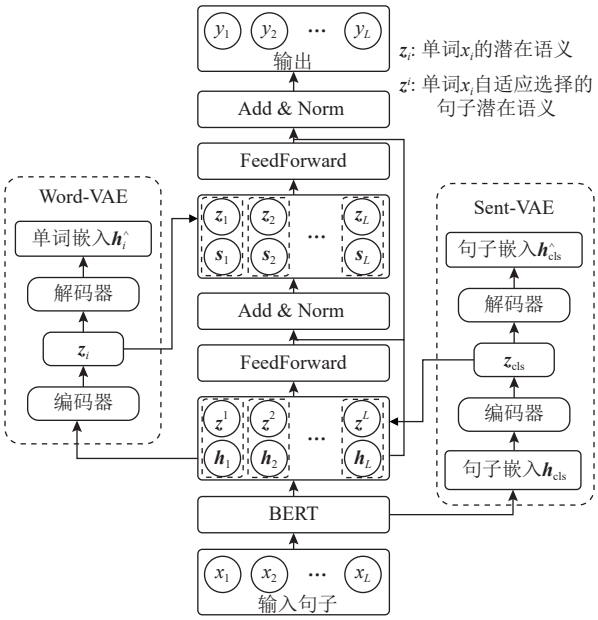


Fig. 2 The architecture of HLSD

图 2 HLSD 架构

包含 4 个模块: 1)词嵌入与序列编码. 将句子中的每个单词都转换为连续向量, 同时对句子进行编码. 2)双重潜在语义挖掘. 利用句子级变分自编码器挖掘句子的潜在语义信息, 利用单词级变分自编码器挖掘单词的潜在语义信息. 3)分层结构. 采用由粗到细的分层结构, 充分利用句子和单词的潜在语义信息. 4)事件类型预测. 计算每个候选触发词对应每个子类的概率. 下面将详细介绍模型的各个部分.

3.1 词嵌入与序列编码

近年来, 单词表示学习有了新进展[37-39], 有效的单词表示能够为模型带来性能上的提升. 为了充分挖掘单词表示带来的潜在性能提升, 本文模型的单词嵌入和序列编码使用 BERT[39].

BERT 是一种由多层双向 Transformer[40] 堆叠、经掩码语言模型(masked language model, MLM)预训练的语言表示模型, 在特定语料上进行微调后编码句子, 可以得到高质量的语境化单词表示. 给定事件提及句子, x_i是句子中第i个位置上的单词, L是句子中单词的个数, 在句子的开头和结尾处分别加上“[CLS]”和“[SEP]”符号, 然后将句子中每个单词的字块嵌入(wordPiece embedding)、位置嵌入(position embedding)和段嵌入(segment embedding)加和, 加和后的句子作为 BERT 模型的输入, 经编码得到句子的表示$\boldsymbol{h}_{\text{cls}} \in \mathbb{R}^d$和句子中单词的表示$\boldsymbol{h}=(\boldsymbol{h}_1,\boldsymbol{h}_2,\cdots,\boldsymbol{h}_L),\boldsymbol{h}_i \in \mathbb{R}^d$表示第$i$个单词的表示, 其中$d$表示 BERT 隐层向量维度.

3.2 双重潜在语义挖掘

针对同一个触发词可触发多个事件类型，以及同一个事件类型可由多个不同的事件触发词触发的问题，为了探索事件宏微观语境中更本质的影响信息，考虑变分自编码器可以从重构角度捕捉数据中潜在的表征，本文依据事件提及所在上下文范围的不同粒度（即句子、单词），采用变分自编码器进行双重潜在语义挖掘，并用来验证如何辅助事件检测.

变分自编码器是 Kingma 等人[30]提出的一种生成模型，它由神经网络参数化的变分分布来近似后验分布，通过最小化 2 个分布间的 KL 散度（Kullback–Leibler divergence），使得模型更好地重构原始输入，在本文的 HLSD 模型中指 BERT 编码后的句子或单词表示.

VAE 由编码器（encoder）和解码器（decoder）组成，编码器通过神经网络得到变分分布的均值$\boldsymbol{\mu}$和方差$\boldsymbol{\sigma}^2$，在此基础上通过重参数化方法得到潜在语义向量，解码器通过对潜在语义向量进行解码，对原始输入进行重构，以辅助挖掘宏微观语境中更本质的事件判别信息.

3.2.1 句子潜在语义挖掘

为了解决同一个触发词触发多个事件类型的问题，我们采用变分自编码器来挖掘句子的潜在语义. 给定句子经过 BERT 编码后的语义表示$\boldsymbol{h}_{\text{cls}}$，我们将$\boldsymbol{h}_{\text{cls}}$送入编码器，将其映射到隐空间$\boldsymbol{h}_{\text{cls}}^{\text{enc}}$中.

$$\boldsymbol{h}_{\text{cls}}^{\text{enc}} = relu(\boldsymbol{W}_{\text{cls}}^{\text{enc}}\boldsymbol{h}_{\text{cls}} + \boldsymbol{b}_{\text{cls}}^{\text{enc}}), \tag{1}$$

其中$\boldsymbol{W}_{\text{cls}}^{\text{enc}}$和$\boldsymbol{b}_{\text{cls}}^{\text{enc}}$均是编码器的参数，我们将函数 *relu* 作为激活函数. 与文献 [30] 类似，本文假设 BERT 编码后的句子表示$\boldsymbol{h}_{\text{cls}}$的先验分布和后验分布为高斯分布. 高斯分布均值$\boldsymbol{\mu}_{\text{cls}}$和方差$\boldsymbol{\sigma}_{\text{cls}}{}^2$通过线性变换获得，如式（2）（3）所示：

$$\boldsymbol{\mu}_{\text{cls}} = \boldsymbol{W}_{\text{cls}}^{\mu}\boldsymbol{h}_{\text{cls}}^{\text{enc}} + \boldsymbol{b}_{\text{cls}}^{\mu}, \tag{2}$$

$$\log(\boldsymbol{\sigma}_{\text{cls}}{}^2) = \boldsymbol{W}_{\text{cls}}^{\sigma}\boldsymbol{h}_{\text{cls}}^{\text{enc}} + \boldsymbol{b}_{\text{cls}}^{\sigma}, \tag{3}$$

其中$\boldsymbol{W}_{\text{cls}}^{\mu}$，$\boldsymbol{b}_{\text{cls}}^{\mu}$，$\boldsymbol{W}_{\text{cls}}^{\sigma}$，$\boldsymbol{b}_{\text{cls}}^{\sigma}$均是线性变换层的参数. 句子的潜在语义通过重参数化技巧得到，形式化如式（4）所示：

$$\boldsymbol{z}_{\text{cls}} = \boldsymbol{\mu}_{\text{cls}} + \varepsilon \times \boldsymbol{\sigma}_{\text{cls}}, \tag{4}$$

其中ε服从标准高斯分布. 至此我们就得到了句子的潜在语义，但是为了保证句子的潜在语义能够尽可能地包含句子的主要信息，需要通过解码器来对得到的潜在语义进行解码，重构原始的输入.

我们首先利用softmax函数将句子的潜在语义向量归一化，得到$\boldsymbol{\theta}_{\text{cls}}$，例如向量$\boldsymbol{\theta}_{\text{cls}}$的第$i$个元素$\theta_{\text{cls}}^{i}$计算如式（5）所示：

$$\theta_{\text{cls}}^{i} = \frac{\exp(z_{\text{cls}}^{i})}{\sum_{t=1}^{m}\exp(z_{\text{cls}}^{t})}, \tag{5}$$

其中m表示$\boldsymbol{z}_{\text{cls}}$的维度，然后将$\boldsymbol{\theta}_{\text{cls}}$送入解码器，得到重构之后的输出$\boldsymbol{h}_{\text{cls}}^{\wedge}$，形式化如式（6）所示，其中$\boldsymbol{W}_{\text{cls}}^{\text{dec}}$是解码器的参数.

$$\boldsymbol{h}_{\text{cls}}^{\wedge} = softmax(\boldsymbol{W}_{\text{cls}}^{\text{dec}}\boldsymbol{\theta}_{\text{cls}} + \boldsymbol{b}_{\text{cls}}^{\text{dec}}). \tag{6}$$

通过让$\boldsymbol{h}_{\text{cls}}^{\wedge}$尽可能接近$\boldsymbol{h}_{\text{cls}}$来实现句子的潜在语义$\boldsymbol{z}_{\text{cls}}$，尽可能包括$\boldsymbol{h}_{\text{cls}}$的主要信息.

根据变分自编码器的损失函数，句子潜在语义部分的损失函数如式（7）所示：

$$\begin{aligned} L_{\text{sent}} = & D_{\text{KL}}[q(\boldsymbol{\theta}_{\text{cls}}, \boldsymbol{z}_{\text{cls}}|\boldsymbol{h}_{\text{cls}})\|p(\boldsymbol{\theta}_{\text{cls}}|\boldsymbol{\mu}_{\text{cls}}, \boldsymbol{\sigma}_{\text{cls}}{}^2)] - \\ & E_{q(\boldsymbol{\theta}_{\text{cls}}, \boldsymbol{z}_{\text{cls}}|\boldsymbol{h}_{\text{cls}})}[\log p(\boldsymbol{h}_{\text{cls}}|\boldsymbol{z}_{\text{cls}}, \boldsymbol{\theta}_{\text{cls}}, \boldsymbol{W}_{\text{cls}}^{\text{dec}})] \quad . \end{aligned} \tag{7}$$

式（7）中等号右侧的第 1 项表示 KL 散度，用来衡量变分分布$q(\boldsymbol{\theta}_{\text{cls}}, \boldsymbol{z}_{\text{cls}}|\boldsymbol{h}_{\text{cls}})$和真实后验分布$p(\boldsymbol{\theta}_{\text{cls}}|\boldsymbol{\mu}_{\text{cls}}, \boldsymbol{\sigma}_{\text{cls}}{}^2)$的接近程度，该距离越小越好，即隐变量能更好地表示输入数据；等号右侧第 2 项中的$E_{q(\boldsymbol{\theta}_{\text{cls}}, \boldsymbol{z}_{\text{cls}}|\boldsymbol{h}_{\text{cls}})}[\log p(\boldsymbol{h}_{\text{cls}}|\boldsymbol{z}_{\text{cls}}, \boldsymbol{\theta}_{\text{cls}}, \boldsymbol{W}_{\text{cls}}^{\text{dec}})]$表示重构期望，通过从变分分布中采样隐变量的值，结合重参数化的方法，使得对数似然函数越大越好，即隐变量的值能更好地重构原始输入. 综上，在最小化损失函数的约束下，模型会产生更好的潜在语义变量和重构原始输入数据.

3.2.2 单词潜在语义挖掘

多个事件触发词可以触发同一个事件类型，说明具有多样性的触发词存在一些深层的共性语义联系. 在宏观语境信息的约束下，使用变分自编码器来挖掘句子中每个单词的潜在语义，我们通过捕捉这些事件触发词的共同特性，为事件检测提供精准的语境信息. 对比而言，句子的潜在语义表示一句话的主要信息，单词的潜在语义则是单词在某个特定上下文的主要语义信息，这样当不同的上下文表达相似的语义时，单词的潜在语义也相似. 具体的实现过程与 3.2.1 节相同，唯一的不同点在于 1 个句子只需要挖掘 1 个潜在语义$\boldsymbol{z}_{\text{cls}}$，而在挖掘单词的潜在语义时，需要对句子中每个单词都挖掘 1 个潜在语义.

3.3 分层结构

得到句子和单词的潜在语义后，我们设计一个由粗到细的分层结构进行不同粒度语境的融合. 首先利用句子的潜在语义信息进行整体把握，然后利用单词的潜在语义信息以促进事件检测.

1）句子级. 通过 3.2.1 节得到句子的潜在语义信息$\boldsymbol{z}_{\text{cls}}$，考虑句子中的大部分单词不触发事件，然而散在句子中的核心事件元素对触发事件的判断起到重

要的支撑作用. 因此我们设计一种注意力机制来自动选择句子潜在语义的相关部分, 以削弱对事件判断的无用信息:

$$\alpha_i = sigmod(MLP(\boldsymbol{h}_{\text{cls}}, \boldsymbol{h}_i)), \tag{8}$$

其中MLP代表多层感知机, $sigmod$代表激活函数, α_i表示第i个单词的注意力得分. 第i个单词选择的句子潜在语义$\boldsymbol{z}^i$如式(9)所示:

$$\boldsymbol{z}^i = \alpha_i \boldsymbol{z}_{\text{cls}}. \tag{9}$$

然后将$\boldsymbol{h}$和单词选择的句子潜在语义进行拼接得到句子级表示$\boldsymbol{h}_{\text{sen}} = ((\boldsymbol{h}_1, \boldsymbol{z}^1), (\boldsymbol{h}_2, \boldsymbol{z}^2), \cdots, (\boldsymbol{h}_L, \boldsymbol{z}^L))$.

对于得到的句子级表示, 我们采用类似于 Transformer[40] 的 Layer Normalization, 得到单词语义和句子潜在语义的深度交互表示$\boldsymbol{s} = (\boldsymbol{s}_1, \boldsymbol{s}_2, \cdots, \boldsymbol{s}_L), \boldsymbol{s}_i \in \mathbb{R}^d$表示第$i$个单词的中间表示, 形式化如式(10)所示:

$$\boldsymbol{o} = LayerNorm(FFN(\boldsymbol{h}_{\text{sen}}) + \boldsymbol{h}), \tag{10}$$

其中FFN表示 FeedForward Network.

2)单词级. 通过 3.2.2 节得到单词的潜在语义$\boldsymbol{z}_{\text{word}} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_L)$, 然后将单词的潜在语义和$\boldsymbol{s}$进行拼接, 得到单词级表示$\boldsymbol{h}_{\text{word}} = ((\boldsymbol{s}_1, \boldsymbol{z}_1), (\boldsymbol{s}_2, \boldsymbol{z}_2), \cdots, (\boldsymbol{s}_L, \boldsymbol{z}_L))$.

对于得到的单词级表示, 我们同样采用 Layer Normalization, 得到单词语义和单词潜在语义的深度交互表示$\boldsymbol{o} = (\boldsymbol{o}_1, \boldsymbol{o}_2, \cdots, \boldsymbol{o}_L)$, $\boldsymbol{o}_i \in \mathbb{R}^d$表示第$i$个单词的最终表示, 形式化如式(11)所示:

$$\boldsymbol{o} = LayerNorm(FFN(\boldsymbol{h}_{\text{word}}) + \boldsymbol{h}). \tag{11}$$

3.4 事件类型预测

我们将$\boldsymbol{o}_i$输入到具有softmax层的前馈神经网络中, 获得r维归一化标签概率向量$\boldsymbol{p}_i = (p_i^1, p_i^2, \cdots, p_i^j, \cdots, p_i^r)$, 计算形式如式(12)(13)所示:

$$\boldsymbol{q}_i = \boldsymbol{W}_y \boldsymbol{o}_i + \boldsymbol{b}_y, \tag{12}$$

$$p_i^j = \frac{\exp(q_i^j)}{\sum_{j=1}^{r} \exp(q_i^j)}, \tag{13}$$

其中r为预定义的事件类型的个数(本文中r=34), p_i^j是将单词x_i分类为第j个事件类型的概率值. 如果p_i^j的概率值最大, 则将第j个事件类型作为该候选触发词x_i的事件类型.

给定所有的H个训练样例$((x_1, y_1), (x_2, y_2), \cdots, (x_H, y_H))$, 我们采用平均负对数似然函数作为该模型的损失函数, 如式(14)所示:

$$J(\boldsymbol{y}, \boldsymbol{p}) = -\frac{1}{H} \sum_{i=1}^{H} \sum_{j=1}^{r} y_i^j \log(p_i^j). \tag{14}$$

结合式(7)(14)以及 3.2.2 节的损失, 我们计算本文模型 HLSD 的联合损失, 如式(15)所示:

$$J'(\boldsymbol{\theta}) = J(\boldsymbol{y}, \boldsymbol{p}) + \lambda_1 L_{\text{sent}} + \lambda_2 L_{\text{word}}, \tag{15}$$

其中L_{word}为 3.2.2 节单词潜在语义挖掘部分的损失, λ_1, λ_2为平衡J, L_{sent}, L_{word}的超参数.

4 实验与结果分析

4.1 实验数据

本文采用 ACE2005 的英文数据集, 该语料由美国宾夕法尼亚大学的语言数据联盟(linguistics data consortium, LDC)提供, 包含实体、关系和事件注释等, 主要支持英文、中文和阿拉伯文.

ACE2005 英文语料包括 599 篇文档, 定义了 8 种事件类型和 33 种事件子类型. 为了公平比较, 本文遵循前人[7-9] 的语料划分方法, 即随机选择 40 篇新闻文档作为测试集, 来自不同类别的 30 篇文档作为验证集, 剩下的 529 篇文档作为训练集, 预处理后的训练集、验证集、测试集包含的句子数、事件类型数如表 2 所示.

Table 2 ACE2005 English Dataset Statistics

表 2 ACE2005 英文数据集统计

数据集	文档数	句子数	触发词数
训练集	529	14 724	4 311
验证集	30	875	492
测试集	40	713	422

4.2 评估方法

目前, 事件抽取主要用 3 个评价指标来衡量事件抽取性能的好坏: 精确率(precision, P)、召回率(recall, R)、$F1$ 值($F1$−measure).

4.3 基线方法与超参数设置

本文方法没有引入大规模外部资源, 选择相关的基线模型包括:

1)HBTNGMA. Chen 等人[20] 通过门控多级注意力自动抽取并动态融合了句子级信息和文档级信息.

2)DEEB-RNN. Zhao 等人[19] 采用有监督的分层注意力机制来学习文档嵌入, 并使用它来增强事件触发词检测.

3)Delta. Lu 等人[18] 提出一种 Delta 表示学习方法, 力图将事件表示解耦为词汇具体部分和词汇无关部分, 而不是仅学一个单独完整的表示, 以蒸馏出事件的判别和泛化知识.

4)GCN-ED. Nguyen 等人[21] 使用图卷积网络在句子的依存分析图上进行卷积, 并提出一种新的池化方式来聚合卷积向量, 增强单词的表示.

5）TS-distill. Liu 等人[16]采用基于知识蒸馏的对抗模拟方法进行事件检测，提高模型从原始文本中获取抽象事件知识的能力.

6）PLMEE. Yang 等人[17]提出利用预训练语言模型进行事件检测和自动数据增强.

7）DMBERT. Wang 等人[27]提出了用一种对抗训练机制来迭代识别信息实例并过滤掉噪声的实例.

8）CSMG. Liu 等人[41]提出一个新的预训练机制，即上下文选择性掩码，以提升事件检测模型的鲁棒性，从而减小词汇的微小扰动对神经模型的影响.

9）CDSIGM. 陈佳丽等人[42]采用门控机制融合依存与语义信息来进行事件检测.

10）MSBERT. 王捷等人[43]采用基于共享 BERT 和门控多任务学习的方法来提高事件检测性能.

11）RCEE_ER. Liu 等人[22]将事件检测转化为阅读理解任务，借助阅读理解模型来促进事件检测.

12）GatedGCN. Lai 等人[23]基于图卷积神经网络引入门控机制和语法信息来提高事件检测性能.

我们在 ACE2005 数据集上使用 PyTorch 微调了一个 bert-base-cased 模型. 该模型具有 12 层，每层包括 12 个 head attention，句子和单词经 BERT 编码后的维度均为 768. 本文使用验证集进行调参，最终设置句子的潜在语义和单词的潜在语义的维度均为 50，批次大小为 48，dropout 率为 0.2，学习率为 0.000 02，优化器采用 Adam，epoch 设置为 50，early stop 设置为 10，即在连续 10 个 epoch 中模型在验证集上的性能没有提升，则结束训练过程，λ_1和λ_2均设置为 0.5.

4.4 总体性能比较

表 3 为 HLSD 在相同测试集下与基准方法的性能对比. 为了进行公平的比较，基线方法的结果除了 PLMEE 均来自原始论文. 考虑 PLMEE 的事件检测部分与我们的 HLSD 基线方法均属基本的 BERT 模型，因此我们重构实现. 从表 3 可得到 3 点观察：

1）在不引入外部资源的情况下，HLSD 的性能优于所有的基准模型. 具体地，和基于 LSTM 作为 Encoder 的模型 DEEB-RNN 相比，HLSD 在 $F1$ 上至少提高了 3.9 个百分点；和基于 ELMo 的 Delta 方法相比，HLSD 在 $F1$ 上提高了 3.9 个百分点；和基于 GCN 的 GCN-ED 方法相比，HLSD 在 $F1$ 上提高了 4.8 个百分点；和基于 GAN 的 TS-distill 方法相比，HLSD 在 $F1$ 上提高了 3.1 个百分点；和基于 BERT 的模型 MSBERT 相比，HLSD 在 $F1$ 上至少提高了 1.4 个百分点.

2）基于 BERT 的模型 DMBERT, CSMG, MSBERT, HLSD 的性能超过了大部分基于 LSTM, ELMo, GCN 的模型的性能，这源于大规模的预训练语言模型改善了单词的表示能力，使得 BERT 能更好地捕获上下文语境信息，从而提升了事件检测的性能.

Table 3 Overall Performance Comparison

表 3 总体性能比较 %

方法	Encoder	P	R	$F1$
HBTNGMA	LSTM	77.9	69.1	73.3
DEEB-RNN	LSTM	72.3	75.8	74.0
Delta	ELMo	76.3	71.9	74.0
GCN-ED	GCN	77.9	68.8	73.1
CDSIGM	GCN	75.5	72.4	73.9
TS-distill	GAN	76.8	72.9	74.8
PLMEE	BERT	72.2	73.9	73.0
DMBERT	BERT	77.6	71.8	74.6
CSMG	BERT	75.2	74.4	74.8
RCEE_ER	BERT	75.6	74.2	74.9
MSBERT	BERT	75.1	77.9	76.5
GatedGCN	BERT	78.8	76.3	77.6
HLSD（本文）	BERT	75.1	80.9	77.9

3）和基于 BERT 的最新模型 MSBERT, GatedGCN 相比，HLSD 在 $F1$ 上取得了更好的性能，这表明 HLSD 的有效性不仅仅源于 BERT 预训练语言模型. 对由 BERT 得到的表示空间进一步降维可辅助挖掘更加本质的潜在语义信息.HLSD 通过 VAE 挖掘句子和单词不同粒度语境的潜在语义信息为触发词带来了更多的事件判别知识. 同时，通过分层结构动态地整合不同粒度语境有助于探索事件检测中更本质的宏微观影响要素.

4.5 退化实验

4.5.1 句子和单词潜在语义的作用

为了验证 HLSD 中潜在语义挖掘的有效性，我们选择 4 个消融模型用于比较：

1）BASE.基本模型，仅使用 BERT 进行事件检测，不考虑句子和单词的潜在语义.

2）BASE+SENT.在 1）的基础上，考虑句子的潜在语义.

3）BASE+WORD.与 2）类似，在 1）的基础上，考虑单词的潜在语义.

4）BASE+SENT+WORD.考虑句子和单词的潜在语义，但不采用分层结构，只是将句子和单词的潜在语义进行拼接.

表 4 展示了基准模型和退化模型的性能；由于只考虑句子或单词的潜在语义时，无法再使用分层结构，

Table 4 Performance Comparison of Integrating Different Latent Semantic Modules

表 4 整合不同潜在语义模块的性能比较 %

方法	*P*	*R*	*F*1
BASE	72.2	73.9	73.0
BASE+SENT	73.1	74.1	73.6
BASE+WORD	73.6	74.0	73.8
BASE+SENT+WORD	74.2	75.7	74.9
HLSD（本文）	75.1	80.9	77.9

因此 BASE+SENT 和 BASE+WORD 模型均将 BERT 编码后的表示和句子或单词的潜在语义进行拼接. 从表 4 可以看出:

1)相比于 BASE, 仅考虑句子的潜在语义的模型无论是在精确率、召回率还是 *F*1 值上都有一定程度的提高, 精确率最高提高为 0.9 个百分点, 召回率最低提高为 0.2 个百分点. 这表明考虑句子的潜在语义可以带来有效的宏观语境信息.

2)相比于 BASE, 仅考虑单词潜在语义的模型在精确率、召回率和 *F*1 值上也都有提升, 但和 BASE+SENT 模型相比, BASE+WORD 的精确率提升更明显, 召回率略有下降(下降了 0.1 个百分点), 这是因为 BASE+WORD 仅考虑了单词的潜在语义, 对于潜在语义不明显的触发词直接当作非触发词, 同时减少了把非触发词识别为触发词的情况.

3)BASE + SENT + WORD 模型在不使用分层结构的情况下性能有提升, 但没有 HLSD 模型高, 这充分说明融合句子潜在语义和单词潜在语义均对事件检测有帮助.

4.5.2 层级结构的作用

为了更加直接地验证 HLSD 分层结构的有效性, 我们选择下面的消融模型用于比较:

1)HLSD-1. 见 4.5.1 节的 BASE + SENT + WORD.

2)HLSD-2.采用由细到粗的分层结构, 即先利用单词的潜在语义, 然后再利用句子的潜在语义.

3)HLSD. 本文提出的模型, 先利用句子的潜在语义, 再利用单词的潜在语义.

不同模型结果如表 5 所示. 从表 5 可以看出:

1)HLSD-2 和 HLSD 相比于 HLSD-1, *F*1 值均有提高, 说明了采用分层结构比直接拼接不同粒度的潜在语义信息更能有效提高模型的性能. HLSD-2 在 *F*1 上不如 HLSD, 可能是由于由细到粗的分层结构对句子的整体把握不如由粗到细的分层结构.

2)HLSD 模型没能在全部指标上超过 HLSD-2 模型, 在精确率上下降了 1.5 个百分点. 这可能由于句子的潜在语义不仅改善了触发词的表示, 而且改善了非触发词的表示; 再经过由粗到细的分层结构, 结果倾向于把非触发词识别为触发词, 同时减少把触发词识别为非触发词的情形. 但可以看到 HLSD 模型兼顾宏微观语境, 进而提高模型的整体性能.

Table 5 Performance Comparison of Different Hierarchical Structures

表 5 不同分层结构的性能比较 %

方法	*P*	*R*	*F*1
HLSD-1	74.2	75.7	74.9
HLSD-2	76.6	77.3	76.9
HLSD（本文）	75.1	80.9	77.9

4.6 1 词触发不同事件类型的影响

为了更加具体分析模型的性能, 我们对模型在测试集上的结果进行了统计. 测试集中一共包括 422 个事件, 其中有 75 个事件是由 1 词触发多个事件类型而引发的, 占比为 17.8%.

计算 BASE, BASE+WORD, BASE+SENT, HLSD 正确识别的事件数量以及正确率(正确识别的事件数量与总的由 1 词触发多个事件类型而引发的事件总数之比), 如表 6 所示.

Table 6 Performance of Different Models on One Word Triggering Different Event Types

表 6 不同模型在 1 词触发不同事件类型上的性能

方法	正确识别的事件数量	正确率/%
BASE	32	42.7
BASE+WORD	37	49.3
BASE+SENT	44	58.7
HLSD（本文）	51	68.0

从表 6 可以观察到:

1)相比于 BASE, BASE+WORD 正确率有小幅度提升, 而 BASE+SENT 提升在 10 个百分点以上. 这源于在 1 词触发不同的事件类型情况下, 考虑宏观句子的潜在语义可以为触发词分类提供额外的判别信息, 因此挖掘句子的潜在语义有助于正确率的提升.

2)相比于 BASE+SENT 模型, HLSD 模型性能有额外的提升, 这也从侧面说明了分层结构和单词潜在语义的有效性.

4.7 多个词触发同一个事件类型的影响

考虑不同的事件触发词触发同一个事件类型的情况, 测试集中一共包括 422 个事件, 对每个事件类型的事件触发词进行去重, 然后求和可以得到不同

的事件触发词数量共计 239 个，占比为 56.7%.

我们考虑触发词去重后由 2 种及以上不同的触发词触发的事件类型，然后统计 BASE, BASE+SENT, BASE+WORD, HLSD 模型在这些事件类型上识别出的触发词类别数，如图 3 所示，其中 ALL 表示每种事件类型的所有触发词类别数. 从图 3 可以看到，对于每种事件类型，BASE+SENT 识别的触发词类别数均大于等于 BASE 识别的触发词类别数；BASE+WORD 识别的触发词类别数大于等于 BASE+SENT 识别的触发词类别数. 这表明，对于多个事件触发词触发的事件类型，考虑句子的潜在语义可以提升识别的触发词类别数，但效果不如单词的潜在语义，这源于考虑微观的单词潜在语义可以为不同的事件触发词提供更多的相互促进作用. 对于不同模型识别触发词类别数相等的情况，是源于同一事件类型下不同种类触发词数量分布不均衡，未识别出来的触发词类别在数据集中出现的数量很少. HLSD 识别的触发词类别数大于等于 BASE+WORD 识别的触发词类别数，这进一步表明，综合考虑句子和单词的潜在语义信息可以更加有效地进行触发词的识别.

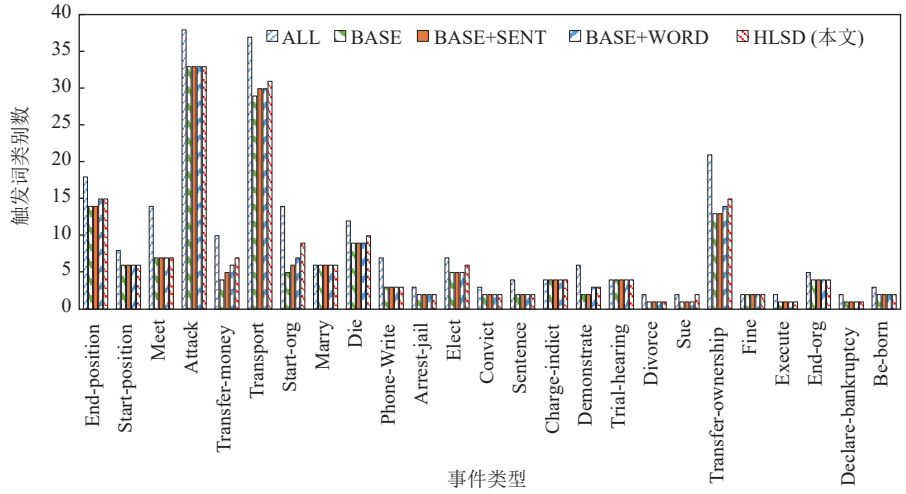


Fig. 3 Number of event trigger types identified by different models

图 3 不同模型识别出的事件触发词种类数

5 结 论

抓住事件检测面临 1 词触发多个事件类型和多词触发同一个事件类型的挑战，受变分自编码器中隐变量和其他 NLP 任务中分层结构的启发，提出了一种新的分层潜在语义驱动网络(HLSD)的事件检测方法，以探究事件宏微观语境中更本质的影响要素.1)通过句子级变分自编码器来捕获句子的潜在语义信息，为触发词识别提供显式的句子级别辅助信息，同时尽可能地去除句子中的无关信息，保留句子中的主要信息；2)通过单词级变分自编码器来捕获单词的潜在语义信息，为触发词识别提供显式的单词级别辅助信息. 在文本表示的基础上，句子和单词均被映射到更低维的语义空间，使得模型更易于捕捉最核心的潜在语义，也等效于间接利用了跨文档语境信息. 为了有效利用句子和单词的潜在语义信息，本文采用一个由粗到细的分层结构来充分融合不同层级的信息，以提高事件检测的性能. 在 ACE2005 英文数据集上的实验结果验证了本文提出方法的有效性，特别地，在 1 词触发多个事件类型问题上，显式考虑并自适应选择句子的潜在语义信息，模型可以为事件触发词提供更有效的判别信息；在多个词触发同一个事件类型上，显式考虑单词的潜在语义可以为每种事件类型的不同触发词提供相互促进的作用. 通过对比模型各个模块的表现，验证了潜在语义和分层结构对事件检测的促进作用. 具体分析对比各个消融模型在测试集上的表现，验证了 2 点：1)句子的潜在语义对 1 词触发多个事件类型问题具有指导作用；2)单词潜在语义对不同触发词触发同一个事件类型具有促进作用.

作者贡献声明：肖梦南提出了算法思路、完成实验和撰写论文；贺瑞芳提出指导意见并修改论文；马劲松修改论文.

参 考 文 献

[1] Zhang Hongming, Liu Xin, Pan Haojie, et al. ASER: A large-scale eventuality knowledge graph [C] //Proc of the 29th Int World Wide

Web Conf. New York: ACM, 2020: 201−211

[2] Glavas G, Snajder J. Event graphs for information retrieval and multi-document summarization [J]. Journal of Expert Systems with Applications, 2014, 41(15): 6904−6916

[3] Eisenberg J, Sheriff M. Automatic extraction of personal events from dialogue [C] //Proc of the 1st Joint Workshop on Narrative Understanding, Storylines, and Events. Stroudsburg, PA: ACL, 2020: 63−71

[4] Ahn D. The stages of event extraction [C] //Proc of the Workshop on Annotating and Reasoning about Time and Events. Stroudsburg, PA: ACL, 2006: 1–8

[5] Ji Heng, Grishman R. Refining event extraction through cross-document inference [C] //Proc of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA: ACL, 2008: 254–262

[6] Liao Shasha, Grishman R. Using document level cross-event inference to improve event extraction [C] //Proc of the 48th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2010: 789–797

[7] Hong Yu, Zhang Jianfeng, Ma Bin, et al. Using cross-entity inference to improve event extraction [C] //Proc of the 49th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2011: 1127–1136

[8] Li Qi, Ji Heng, Huang Liang. Joint event extraction via structured prediction with global features [C] //Proc of the 51st Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2013: 73–82

[9] Liu Shulin, Liu Kang, He Shizhu, et al. A probabilistic soft logic based approach to exploiting latent and global information in event classification [C] //Proc of the 30th AAAI Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2016: 2993–2999

[10] Chen Yubo, Xu Liheng, Liu Kang, et al. Event extraction via dynamic multi-pooling convolutional neural networks [C] //Proc of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th Int Joint Conf on Natural Language Processing. Stroudsburg, PA: ACL, 2015: 167–176

[11] Nguyen T H, Grishman R. Modeling skip-grams for event detection with convolution neural networks [C] //Proc of the 2016 Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2016: 886–891

[12] Nguyen T H, Grishman R. Event detection and domain adaptation with convolution neural networks [C] //Proc of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th Int Joint Conf on Natural Language Processing. Stroudsburg, PA: ACL, 2015: 365–371

[13] Nguyen T H, Cho K, Grishman R. Joint event extraction via recurrent neural networks [C] //Proc of the 15th Annual Conf of the North American Chapter of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2016: 300–309

[14] Feng Xiaocheng, Huang Lifu, Tang Duyu, et al. A language-independent neural network for event extraction [C] //Proc of the 54th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2016: 66–71

[15] Duan Shaoyang, He Ruifang, Zhao Wenli. Exploiting document level information to improve event detection via recurrent neural networks [C] //Proc of the 8th Int Joint Conf on Natural Language Processing. Stroudsburg, PA: ACL, 2017: 351–361

[16] Liu Jian, Chen Yubo, Liu Kang. Exploiting the ground-truth: An adversarial imitation based knowledge distillation approach for event detection [C] //Proc of the 33rd AAAI Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2019: 6754−6761

[17] Yang Sen, Feng Dawei, Qiao Linbo, et al. Exploring pre-trained language models for event extraction and generation [C] //Proc of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2019: 5284−5294

[18] Lu Yaojie, Lin Hongyu, Han Xianpei, et al. Distilling discrimination and generalization knowledge for event detection via delta-representation learning [C] //Proc of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2019: 4366−4376

[19] Zhao Yue, Jin Xiaolong, Wang Yuanzhuo, et al. Document embedding enhanced event detection with hierarchical and supervised attention [C] //Proc of the 56th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2018: 414−419

[20] Chen Yubo, Yang Hang, Liu Kang, et al. Collective event detection via a hierarchical and bias tagging networks with gated multi-level attention mechanisms [C] //Proc of the 2018 Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2018: 1267−1276

[21] Nguyen T H, Grishman R. Graph convolutional networks with argument-aware pooling for event detection [C] //Proc of the 32nd AAAI Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2018: 5900–5907

[22] Liu Jian, Chen Yubo, Liu Kang, et al. Event extraction as machine reading comprehension [C] //Proc of the 2020 Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2020: 1641−1651

[23] Lai V D, Nguyen T N, Nguyen T H. Event detection: Gate diversity and syntactic importance scores for graph convolution neural networks [C] //Proc of the 2020 Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2020: 5405−5411

[24] Liu Shulin, Chen Yubo, He Shizhu, et al. Leveraging FrameNet to improve automatic event detection [C] //Proc of the 54th Annual Meeting of the Association for Computational Linguistic. Stroudsburg, PA: ACL, 2016: 2134–2143

[25] Chen Yubo, Liu Shulin, Zhang Xiang, et al. Automatically labeled data generation for large scale event extraction [C] //Proc of the 55th Annual Meeting of the Association for Computational Linguistic. Stroudsburg, PA: ACL, 2017: 409–419

[26] Liu Jian, Chen Yubo, Liu Kang, et al. Event detection via gated multilingual attention mechanism [C] //Proc of the 32nd AAAI Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2018: 4865–4872

[27] Wang Xiaozhi, Han Xu, Liu Zhiyuan, et al. Adversarial training for weakly supervised event detection [C] //Proc of the 17th Annual Conf of the North American Chapter of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2019: 998−1008

[28] Tong Meihan, Xu Bin, Wang Shuai, et al. Improving event detection via open-domain trigger knowledge [C] //Proc of the 58th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2020: 5887−5897

[29] Wang Ziqi, Wang Xiaozhi, Han Xu, et al. CLEVE: Contrastive pre-training for event extraction [C] //Proc of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int Joint Conf on Natural Language Processing. Stroudsburg, PA: ACL, 2021: 6283−6297

[30] Kingma D P , Welling M . Auto-encoding variational Bayes [C/OL] // Proc of the 2nd Int Conf of Learning Representation. Ithaca, NY: Cornell University, 2014[2022-12-16].https://openreview.net/forum?id=33X9fd2−9FyZd

[31] Miao Yishu, Grefenstette E, Blunsom P. Discovering discrete latent topics with neural variational inference [C] //Proc of the 34th Int Conf on Machine Learning. New York: ACM, 2017: 2410−2419

[32] Srivastava A, Sutton C. Autoencoding variational inference for topic models [C/OL] //Proc of the 5th Int Conf of Learning Representation. Ithaca, NY: Cornell University, 2017[2022-12-16].https://openreview.net/forum?id=BybtVK9lg

[33] Xu Sheng, Li Peifeng, Kong Fang, et al. Topic tensor network for implicit discourse relation recognition in Chinese [C] //Proc of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2019: 608−618

[34] Yang Zichao, Yang Diyi, Dyer C, et al. Hierarchical attention networks for document classification [C] //Proc of the 15th Annual Conf of the North American Chapter of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2016: 1480−1489

[35] Xiong Tengke, Manggala P. Hierarchical classification with hierarchical attention networks [C/OL] //Proc of the 24th ACM SIGKDD Conf on Knowledge Discovery and Data Mining. New York: ACM, 2018[2022-12-16].https://www.kdd.org/kdd2018/files/deep-learning-day/DLDay18_paper_47.pdf

[36] Pappas N, Popescu-Belis A. Multilingual hierarchical attention networks for document classification [C] //Proc of the 8th Int Joint Conf on Natural Language Processing. Stroudsburg, PA: ACL, 2017: 1015−1025

[37] Mikolov T, Sutskever I, Chen Kai, et al. Distributed representations of words and phrases and their compositionality [C] //Proc of the 27th Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2013: 3111–3119

[38] Peters M, Neumann M, Iyyer M, et al. Deep contextualized word representations [C] //Proc of the 16th Annual Conf of the North American Chapter of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2018: 2227−2237

[39] Devlin J, Chang Mingwei, Lee K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding [C] //Proc of the 17th Annual Conf of the North American Chapter of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2019: 4171−4186

[40] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [C] // Proc of the 31st Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2017: 5998−6008

[41] Liu Jian, Chen Yubo, Liu Kang, et al. How does context matter? On the robustness of event detection with context-selective mask generalization [C] //Proc of the 2020 Conf on Empirical Methods in Natural Language Processing: Findings. Stroudsburg, PA: ACL, 2020: 2523−2532

[42] Chen Jiali, Hong Yu, Wang Jie, et al. Combination of dependency and semantic information via gated mechanism for event detection [J]. Journal of Chinese Information Processing, 2020, 34(8): 51−60 (in Chinese)
(陈佳丽, 洪宇, 王捷, 等. 利用门控机制融合依存与语义信息的事件检测方法[J], 中文信息学报, 2020, 34(8): 51−60)

[43] Wang Jie, Hong Yu, Chen Jiali, et al. Event detection by shared BERT and gated multi-task learning [J]. Journal of Chinese Information Processing, 2021, 35(10): 101−109 (in Chinese)
(王捷, 洪宇, 陈佳丽, 等. 基于共享BERT和门控多任务学习的事件检测方法[J], 中文信息学报, 2021, 35(10): 101−109)



Xiao Mengnan, born in 1997. Master. His main research interests include natural language processing and event extraction.
肖梦南, 1997 年生. 硕士. 主要研究方向为自然语言处理、事件抽取.



He Ruifang, born in 1979. PhD, professor, PhD supervisor. Senior member of CCF. Her main research interests include natural language processing, social media mining, and machine learning.
贺瑞芳, 1979 年生. 博士, 教授, 博士生导师. CCF 高级会员. 主要研究方向为自然语言处理、社会媒体挖掘、机器学习.



Ma Jinsong, born in 1997. Master. His main research interests include natural language processing and event extraction.
马劲松, 1997 年生. 硕士. 主要研究方向为自然语言处理、事件抽取.