

文章编号: 2095-2163(2020)06-0012-07

中图分类号: TP391.1

文献标志码: A

事件表示学习综述

廖 阔, 丁 效, 秦 兵, 刘 挺, 黄虎杰

(哈尔滨工业大学 计算机科学与技术学院, 哈尔滨 150001)

摘要: 事件是现实世界中信息的一种结构化的表示形式, 事件表示学习的目标是将这种结构化的信息表示为计算机可以理解的形式。事件表示学习是事件相似度度量、脚本事件预测、股市预测等任务的基础, 对人工智能的发展有着重要作用。早期研究大多采用离散的事件表示, 随着深度学习的发展, 人们开始尝试使用深度神经网络为事件学习稠密的向量表示, 同时逐步探索将多种形式的外部信息融入事件表示中。本文旨在总结事件表示学习的相关工作, 以期为后续研究提供参考。

关键词: 表示学习; 事件; 人工智能; 深度学习

Event Representation Learning: A Survey

LIAO Kuo, DING Xiao, QIN Bing, LIU Ting, HUANG Hujie

(School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China)

[Abstract] Events are structured representations of information in the real world. The goal of event representation learning is to represent this structured information in a form that can be understood by computers. Event representation learning is the basis of event similarity measurement, script event prediction and stock movement prediction, which plays an important role in the development of artificial intelligence. Early studies in this area use discrete event representations mostly. With the development of deep learning, researchers began to use neural networks to learn dense vector representations for events, and explored the integration of various form of external information into event representations. This paper aims to summarize the related work of event representation learning to provide reference for the follow-up research.

[Key words] representation learning; event; artificial intelligence; deep learning

0 引言

事件通常指包含参与者在内的某种动作或情况的发生, 或事件状态的改变。在粒度上, 事件介于词与句子之间, 与词相比, 事件通常包含多个词, 用来描述事件的发生及事件的组成要素, 是一种语义更完备的文本单元; 与句子相比, 事件更关注对现实世界中动作或变化的描述, 是对现实世界一种更细粒度的刻画。在形式上, 事件的组成要素通常包括事件的触发词或类型、事件的参与者、事件发生的时间或地点等, 与纯自然语言形式的文本相比, 事件是现实世界中信息的一种更为结构化的表示形式。事件在粒度上与形式上的特点使得对其进行表示时, 面临着与其他文本单元不同的问题, 由此引出了事件表示学习的概念。将结构化的事件信息表示为机器可以理解的形式, 对许多自然语言理解任务都十分必要, 例如: 脚本预测与故事生成。早期的研究大多采用离散的事件表示, 随着深度学习的发展, 人们开始尝试使用深度神经网络为事件学习稠密的向量表示, 同时逐步有研究探索将事件内信息、事件间信

息、外部知识等多种类型的信息融入事件表示中, 下面将分别对以上研究进行介绍。

1 离散的事件表示

早期的研究主要基于离散的事件表示, 通常将事件表示为由事件元素构成的元组。Kim 等将事件表示为三元组 (O_i, P, t) , 给定对象的集合 O , 其中 $O_i \subseteq O$ 为对象的谓词, P 为对象间的关系或属性, t 为事件发生的时间^[1]。Radinsky 等进一步将角色加入事件表示中, 每个事件包含标记事件发生的动作或状态 P , 一个或多个事件的实施者 O_1 , 一个或多个事件作用的对象 O_2 , 一个或多个使事件发生的工具 O_3 , 以及一或多个地点 O_4 , 以及时间戳 t , 记为六元组 $(P, O_1, O_2, O_3, O_4, t)$ ^[2]。Ding 等将其简化为 (O_1, P, O_2, T) 形式的元组, 其中 P 为事件动作, O_1 为施事者, O_2 为受事者, T 为时间戳, 一个事件只包含一个施事者和一个受事者^[3]。Chambers 与 Jurafsky 在脚本事件预测任务中, 提出了 Predicate-GR 事件表示方法, 该任务中与同一个角色相关的事件按照时间顺序整理成事件链, Predicate-GR 将

基金项目: 国家重点研发计划(SQ2018AAA0101901); 国家自然科学基金(61976073, 61702137)。

作者简介: 廖 阔(1996-), 男, 硕士研究生, 主要研究方向: 自然语言处理、社会计算。

收稿日期: 2020-03-05

每个事件表示为动作以及动作与角色之间的依存关系构成的二元组^[4],例如:(*arrest, obj*)表示一个逮捕事件,且事件链关联的角色在该事件中为宾语,即被逮捕的对象。因为同一个事件链中的角色是相同的,因此无需将角色加入事件表示中。

离散的事件表示面临稀疏性的问题。为了缓解稀疏性,一系列工作提出基于语义知识库对事件进行泛化。Ding等基于WordNet将各事件元素中的单词还原为词干,之后将事件动作词泛化为其在VerbNet中的类别名称,得到泛化事件^[3],例如:单词“adds”首先被还原为词干“add”,之后被泛化为VerbNet中的类别名“multiply_class”。Zhao等将事件元素中的名词泛化为其在WordNet中的上位词,将动词泛化为其在VerbNet的类别,之后在所有事件中的单词组成的词对(bi-gram)中筛选频率最高的词对,称为“高频共现词对(FCOPA)”,作为泛化后的事件^[5]。例如,“Hudson killed Andrew”首先被泛化为“people murder-42.1 people”,之后高频共现词对“murder-42.1 people”被作为一个泛化的事件。

2 稠密的事件表示

自2013年起,随着深度学习技术的发展,人们开始探索为文本学习分布式的语义表示。分布式语义表示将文本单元(如字、词等)嵌入到向量空间中,每个文本单元的语义信息由所有语义单元在向量空间中的位置共同决定。这种分布式的语义表示通常具有良好的性质。例如,相关性较强(语义相近)的文本单元具有相似的向量表示,并且在很大程度上缓解了文本单元的稀疏性。在这一系列工作的基础上,基于分布式语义的稠密的事件表示应运而生。稠密的事件表示通常以预训练的词向量为基础,根据事件结构对事件元素的词向量进行语义组合,为事件计算低维、稠密的向量表示。按照组合的方式,可以分为基于词向量参数化加法的事件表示与基于张量神经网络的事件表示。

2.1 基于词向量参数化加法的事件表示

基于词向量参数化加法的事件表示方法将事件元素的词向量进行相加或拼接后,输入一个参数化的函数,将相加或拼接后的向量映射到事件向量空间。Weber等对事件元素的词向量求均值,作为一种基线方法^[6]。Li等将事件元素的词向量进行拼接作为事件的向量表示^[7]。Granroth-Wilding等提出EventComp方法,将事件元素的词向量拼接后,输入多层全连接神经网络,对事件元素的词向量进行组合^[8]。作为词向量参数化加法的一种特例, Lee

等直接使用Predicate-GR的向量表示作为事件的向量表示,从而省略了将事件元素的词向量进行组合的步骤^[9]。

Tilk等与Hong等对事件元素向量求和或求均值作为事件表示,但额外考虑事件元素的角色,同一个词出现在事件的不同角色中时,使用不同的词向量^[10-11]。设词表大小为 $|V|$,角色数量为 $|R|$,词向量的维数为 H ,不同角色的词向量构成了三维的张量 $T \in \mathbb{R}^{|V| \times |R| \times H}$ 。为了减少模型的参数数量,进一步将该张量分解为 F 个一阶张量乘积的形式,使用三个矩阵 A, B, C 代替原本的三维张量 T ,公式(1):

$$T_{i,j,k} = \sum_{f=1}^F A_{i,f} B_{j,f} C_{f,k}. \quad (1)$$

设 r 为表示角色的独热(one-hot)向量,角色 r 对应的词向量矩阵为 T 的一个切片,表示为公式(2):

$$W_r = \text{Adiag}(rB)C. \quad (2)$$

最终,对于每个事件元素,分别在该事件元素对应角色的词向量矩阵中查找其词向量,并将所有事件元素的词向量进行组合作为事件向量。

2.2 基于张量神经网络的事件表示

基于词向量参数化加法的事件表示充分利用了事件元素的词向量信息,但在建模事件元素间的交互上较为薄弱。尽管这些方法均取得了一定的效果,但其加性本质使得难以对事件表面形式的细微差异进行建模。例如,在这些方法下,“She throw football(她扔足球)”与“she throw bomb(她扔炸弹)”会得到相近的向量表示,尽管两个事件语义上并不相似。为了解决这一问题,人们陆续提出了基于张量神经网络的事件表示方法。这一系列方法使用双线性张量运算组合事件的元素,设事件两个元素的向量分别为 $v_1, v_2 \in \mathbb{R}^d$,三维张量 $T \in \mathbb{R}^{k \times d \times d}$ 是张量神经网络的参数,则双线性张量运算的计算方式如公式(3):

$$v_{comp} = v_1^T T^{[1:k]} v_2. \quad (3)$$

其中,计算结果 v_{comp} 是一个 k 维向量,它的每一个维度 i 上的元素是由向量 v_1 ,矩阵 T_i 和向量 v_2 做矩阵乘法得到的。在双线性张量运算中,模型以乘性的方式捕获了事件论元的交互,使得即使事件论元只有细微的表面差异,也能够事件表示中体现出语义上的较大差别。

Ding等在2015年提出了Neural Tensor Network模型,简称NTN,模型结构如图1所示。该工作考虑

(O_1, P, O_2) 三元组形式的事件结构, 其中 O_1 为事件的施事者, P 为事件的动作或触发词, O_2 为事件的受事者, 此处用同样的符号表示三种事件元素的词向量。模型首先对施事者和动作词、动作词和受事者进行组合, 再对得到的两个向量进行组合, 得到最终的事件表示 E [12]。每次组合由一个双线性张量运算, 一个常规的线性运算与激活函数 f 组成, 计算方法如公式(4)~(6)所示:

$$R_1 = f(O_1^T T_1^{[1:k]} P + W_1 \begin{matrix} O_1 \\ P \\ P \end{matrix} + b_1), \quad (4)$$

$$R_2 = f(P^T T_2^{[1:k]} O_2 + W_2 \begin{matrix} P \\ O_2 \\ O_2 \end{matrix} + b_2), \quad (5)$$

$$E = f \begin{matrix} R_1 \\ R_2 \end{matrix}^T T_3^{[1:k]} R_2 + W_3 \begin{matrix} R_1 \\ R_2 \end{matrix} + b_3. \quad (6)$$

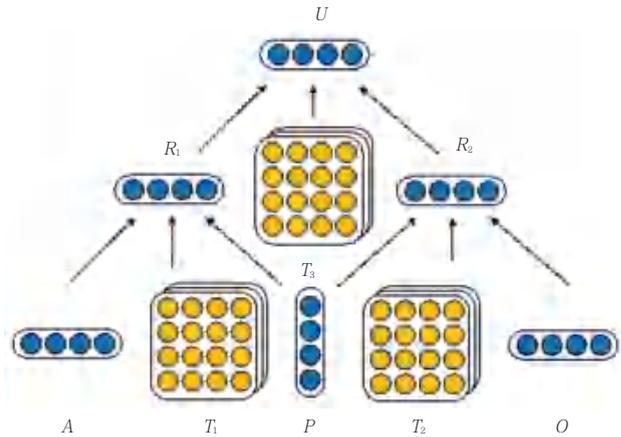


图 1 Neural Tensor Network 模型结构

Fig. 1 Architecture of Neural Tensor Network

Weber 等在 2018 年提出了 Predicate Tensor 模型与 Role-Factored Tensor 模型, 模型结构如图 2 所示。该工作同样考虑 (s, p, o) 三元组形式的事件, 其中 s 为主语, p 为谓语, o 为宾语, 并考虑使用三维张量 P 建模谓语 p , 并用该张量对主语 s 和宾语 o 进行语义组合得到事件向量 e [6], 其每个元素 e_i 的计算过程如公式(7):

$$e_i = \sum_{j,k} P_{ijk} s_j o_k. \quad (7)$$

该方法需要为每个谓语单词学习一个单独的三维张量, 但谓语的集合非常大, 在实践中不可行。Predicate Tensor 方法由谓语的词向量 p 动态地计算张量 P , 并用动态计算的张量对主语和宾语进行语义组合。其中, W 与 U 为模型参数, 设词向量的维数为 d , W 与 U 均为大小为 $d \times d \times d$ 的三维张量, 见公式(8)和公式(9):

$$P_{ijk} = W_{ijk} \sum_a p_a U_{ajk}, \quad (8)$$

$$e_i = \sum_{a,i,j,k} p_a s_j o_k W_{ijk} U_{ajk}. \quad (9)$$

Role-Factored Tensor 方法单独地对事件的主语及谓语、谓语及宾语进行语义组合, 组合后的两个向量通过线性变换后相加得到事件向量, 见公式(10)~(12):

$$vs_i = \sum_{j,k} T_{ijk} s_j p_k, \quad (10)$$

$$vo_i = \sum_{j,k} T_{ijk} o_j p_k, \quad (11)$$

$$e = W_s vs + W_o vo. \quad (12)$$

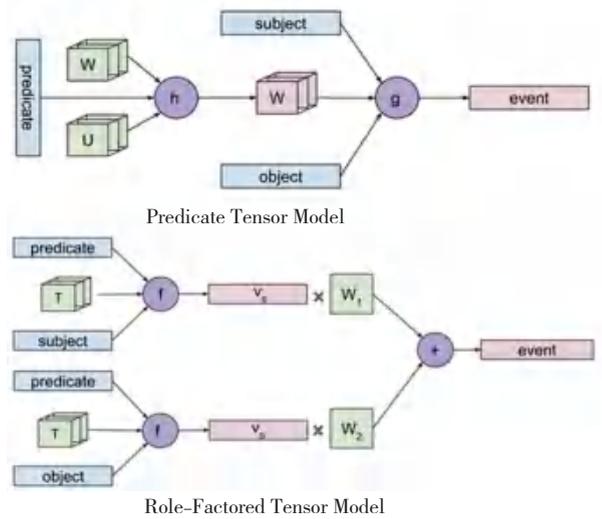


图 2 Predicate Tensor Model 与 Role-Factored Tensor Model 模型结构

Fig. 2 Architecture of Predicate Tensor Model and Role-Factored Tensor Model

基于张量神经网络的事件表示方法面临“维度灾难”的问题, 限制了该方法在许多领域的应用。Ding 等在 2019 年提出使用低秩张量分解 (Low-Rank Tensor Decomposition), 将张量神经网络中的三阶张量参数使用维度较小的张量进行近似, 以减少模型的参数 [13]。图 3 为低秩张量分解运算的示意图。具体地, 将模型中的三阶张量参数 T 替换为 $T_1 \in R^{k \times d \times r}$, $T_2 \in R^{k \times r \times d}$, $t \in R^{k \times d}$ 三个参数, 使用 T_{appr} 作为 T 的近似, 其每个切片 $T_{appr}^{[i]}$ 是通过公式(12)计算得到的:

$$T_{appr}^{[i]} = T_1^{[i]} \times T_2^{[i]} + \text{diag}(t^{[i]}). \quad (12)$$

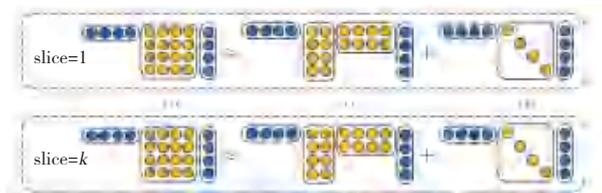


图 3 低秩张量分解示意图

Fig. 3 An illustration of low-rank tensor decomposition

Ding 等在事件相似度、脚本事件预测、股市预测多个任务上的实验结果表明:使用低秩张量分解,可以在减少模型参数的同时取得与原模型相当的效果,有些任务上甚至取得了比原模型更好的性能。

3 事件表示的学习方法

稠密的事件表示使用深度学习的方法,将事件嵌入到向量空间中,为了使嵌入后的向量保留丰富的语义信息,需要为事件表示设计合适的训练目标。一种学习事件表示的方式是直接使用下游任务作为训练目标,这种方式学习得到的事件表示往往只在特定的任务上具有较好的效果,而缺乏较好的泛化能力。其他学习事件表示的方法,包括基于事件本身的结构信息与事件在文本中的分布信息构造自监督的训练目标,在大量文本上进行预训练;以及使用知识库指导事件表示的学习,在事件表示中融入外部知识。下文具体介绍基于事件内信息的事件表示学习、基于事件间信息的事件表示学习与融合外部知识的事件表示学习。

3.1 基于事件内信息的事件表示学习

基于事件内信息的事件表示学习充分利用了事件的结构信息,通常采用自编码的思想,由事件表示恢复出事件元素,并由此构建训练目标,使得事件表示中尽可能地保留事件元素的信息。

Ding 等在 2015 年提出以区分正确的事件元组与被破坏的事件元组作为事件表示的训练目标。该方法首先使用开放事件抽取工具 ReVerb 从大量新闻文本中抽取包含施事者、事件动作、受事者的事件元组 $E = (O_1, P, O_2)$, 将事件论元中的单词随机替换为词表中的其他单词,构造被破坏的事件元组 $E' = (O'_1, P, O_2)$ 。事件元组经过 NTN 模型嵌入到向量空间中,并进一步由函数 f 映射一个分数。训练目标为使正确的事件元组分数高于被破坏的事件元组,具体使用最大边际损失进行计算^[12],其中, $\lambda \|\varphi\|_2^2$ 为 L2 正则项,公式(13):

$$\text{loss}(E, E') = \max(0, 1 - f(E) + f(E')) + \lambda \|\varphi\|_2^2. \quad (13)$$

Tilk 等在 2016 年提出 NNRF 方法,使用事件表示预测事件角色作为训练目标。该方法中,事件的结构被表示为包含谓词、论元的元组,其中每个事件论元被限制为一个单词,并被分配一个预先定义好的角色。事件元组首先由事件表示模型嵌入到向量空间中,得到事件向量 h 。之后,对于每种事件中的每种角色 t ,使用一个分类器对整个词表进行分类,以预测唯一正确的事件角色词作为训练目标。为了

缓解事件角色数较多带来的分类器参数量大的问题,该方法进一步将分类器中的参数进行分解,采用事件角色的独热向量 t 与各分类器共享的模型参数 A_c, B_c, C_c 动态地计算每种角色的预测结果^[10],见公式(14)和公式(15):

$$W_c = A_c \text{diag}(t B_c) C_c, \quad (14)$$

$$o_c = \text{Softmax}(h W_c + b_c). \quad (15)$$

其中, o_c 是该角色上每个单词的概率分布,使用该概率分布计算交叉熵损失,作为模型的训练目标。

Hong 等在 Tilk 等的基础上,提出 NNRF-MT 方法,进一步引入角色分类的训练目标,输入事件表示 h 与单词 w_r , 由分类器预测该单词在事件中的角色类别^[14]。类似地,该分类器的参数按照词表进行分解,由单词的独热向量 w_r 与共享的参数 A_r, B_r, C_r 动态地为每个单词计算预测结果^[11],公式(15)和公式(16):

$$W_r = A_r \text{diag}(w_r B_r) C_r, \quad (15)$$

$$o_r = \text{Softmax}(h W_r + b_r). \quad (16)$$

其中, o_r 是该单词属于每种角色的概率分布,使用该概率分布计算交叉熵损失作为角色分类的训练目标。最后,对角色预测与角色分类的损失进行加权平均得到最终的损失,采用多任务学习(Multi-Task)的形式同时优化两个训练目标,对事件表示进行学习。

3.2 基于事件间信息的事件表示学习

事件的发生并不是独立的,而是按照事件演化规律接连地发生,因此良好的事件表示,除了充分保留事件论元的信息外,还应该含有事件的演化规律信息。为了捕获事件间的演化规律,一系列研究提出考虑事件间的交互,利用事件在文本中的分布信息指导事件表示学习。开放事件抽取工作^[]的进展,使得挖掘事件在文本中的分布信息成为可能,为这一系列的工作提供了基础。具体地,这一系列工作可以分为基于事件对的方法、基于事件链条的方法与基于事件图的方法。

Granroth-Wilding 等在 2016 年提出了基于事件对的 EventComp 方法,利用事件对的顺承关系作为事件表示的训练目标。该方法首先从大规模文本中自动抽取事件链条,将事件链条拆分为多个满足顺承关系的事件对,并通过随机采样的方法构造不满足顺承关系的事件对。之后,使用全连接网络对事件对是否满足顺承关系进行分类,使用分类器的输出计算交叉熵损失,作为事件表示的训练目标。具

体地,对于第 i 个训练样本中的事件对 (e_{0i}, e_{1i}) , 使用全连接网络计算两个事件的相关性分数 $coh(e_{0i}, e_{1i})$, 并与事件对的真实类别 $p_i \in \{0, 1\}$ 计算交叉熵损失^[8], 公式(17):

$$L = \frac{1}{m} \sum_{i=1}^m -\log(p_i \times coh(e_{0i}, e_{1i}) + (1 - p_i) \times (1 - coh(e_{0i}, e_{1i}))) + \lambda L(\theta). \quad (17)$$

Weber 等在该方法的基础上,在事件链条的窗口中采样满足顺承关系的事件对,并以余弦相似度度量事件向量的相关性分数,采用最大边际损失作为训练目标。具体地,对于一个输入事件 e_i , e_i 为事件链条中 e_i 前后窗口中的一个事件, e_n 为从整个语料中随机采样的一个事件, $sim(e_i, e_i)$ 为两个事件的余弦相似度,使用 (e_i, e_i) 与 (e_i, e_n) 两个事件对的相似度计算最大边际损失^[6], 公式(18):

$$L = \frac{1}{N} \sum_{i=1}^N \max(m + sim(e_i, e_n) - sim(e_i, e_i)) + \lambda L(\theta). \quad (18)$$

Wang 等采用事件链条信息指导事件表示的学习,其训练方式为给定事件链条中的上下文事件,预测下一个会发生的事件。上下文事件和候选结尾事件首先被拼接为完整的事件链条,之后每个事件的向量表示被输入长短时记忆网络(LSTM),得到上下文相关的事件表示,基于上下文相关的事件表示计算,计算候选事件与每个上下文事件的相关性得分,并由该得分计算交叉熵损失作为训练目标^[14]。具体地,对于上下文事件 $e_1, \dots, e_i, \dots, e_{n-1}$ 与候选事件 e_c , 采用公式(19)~公式(22)计算其上下文相关的向量表示:

$$h_i = LSTM(e(e_i), h_{i-1}), \quad (19)$$

$$h_c = LSTM(e(e_c), h_{n-1}), \quad (20)$$

$$s_i = \text{sigmoid}(W_{si} h_i + W_{sc} h_c + b_s), \quad (21)$$

$$s = \frac{\sum_{i=1}^{n-1} s_i}{n-1} L = \sum_{i=1}^N (s_i - y_i)^2 + \frac{\lambda}{2} \|\theta\|^2. \quad (22)$$

其中, $e(e_i), e(e_c)$ 为事件 e_i, e_c 的初始事件表示, h_i, h_c 为其上下文相关的向量表示, h_0 为随机初始化的向量。候选事件 e_c 与上下文事件 e_i 的相关性分数 s_i 由一个全连接网络计算, W_{si}, W_{sc}, b_s 为该网络的参数, e_c 作为候选事件的概率 s 为其与所有上下文事件相关性的均值。最终由该概率值计算交叉熵损失作为模型的训练目标,公式(23):

$$L = \sum_{i=1}^N (s_i - y_i)^2 + \frac{\lambda}{2} \|\theta\|^2. \quad (23)$$

Zhao 等由自动抽取的因果事件对构造抽象的因果事件图,并提出基于图嵌入的事件表示学习方法 Dual-CET,在因果事件图上学习事件表示^[7]。这一方法为图中每个事件节点以及原因关系、结果关系学习一个向量表示^[5], 对于一个因果事件对 (c, e) , 定义其能量函数为公式(24):

$$f(c, e) = \|c + t - e\|_1 + \|e + \tau - c\|_1. \quad (24)$$

其中, c 为原因事件向量, e 为结果事件向量, t 为原因关系向量, τ 为结果关系向量。训练目标为使正确因果事件对的能量低于随机构造的错误因果事件对,公式(25):

$$L = \sum_{(c, e) \in P^+} \sum_{(c', e') \in P^-} [\gamma + f(c, e) - f(c', e')] + \frac{\alpha}{\|t + \tau\|_2}. \quad (25)$$

其中, P^+ 为所有正确因果事件对的集合, P^- 为所有错误因果事件对的集合, γ 为最大边际损失的超参数, α 为正则项系数。

Li 等进一步使用图神经网络在事件图上学习事件表示。该工作考虑事件间的顺承关系,从大规模文本中抽取事件链后构建叙事事件图,并使用门控图神经网络(GGNN)在事件图上学习事件表示,该模型的输入为事件节点的初始向量表示,输出为融合图结构信息的事件向量。与 Granroth-Wilding 等相同,该方法使用预测事件链条的后续事件作为训练任务,但将训练目标替换为最大边际损失^[7], 公式(26):

$$L = \sum_{I=1}^N \sum_{j=1}^k \max(0, \text{margin} - s_{Ij} + s_{Ij'}) + \frac{\lambda}{2} \|\theta\|^2. \quad (26)$$

其中, N 为训练样本数, s_{Ij} 为第 I 个训练样本中正确候选事件的得分, $s_{Ij'}$ 为该样本中错误候选事件的得分。

3.3 融合外部知识的事件表示学习

基于事件内与事件间信息的事件表示学习方法,考虑了事件的结构信息以及事件在文本中的分布信息,但忽略了文本中未显式提及的常识知识。例如,“Steve Jobs quits Apple”与“John leaves Starbucks”尽管具有相似的结构,但“Steve Jobs”是“Apple”公司的 CEO,而“John”与“Starbucks”并无特殊关系,因此两个事件会对它们的客体产生不同的影响。为了解决这一问题,一系列工作提出将外部知识融入事件表示学习中,为事件表示补充文本中没有显示提及的信息。

Ding 等在 2016 年提出将实体关系知识融入事件表示, 这种知识可以从知识图谱如 YAGO, Freebase 中获取。该方法的示意图如图 4 所示, 对于一个事件元组 (A, P, O) , 首先在知识图谱中找到包含事件施事者 A 或受事者 O 的三元组 (e_1, R, e_2) , 其中 e_1 为头实体, e_2 为尾实体, R 为两个实体间的关系。之后, 采用公式 (27) 的张量神经网络为 (e_1, R, e_2) 三元组计算得分^[15]:

$$g(e_1, R, e_2) = \mu_R^T f(e_1^T H_R^{[1:k]} e_2 + V_R \begin{matrix} e_1 \\ R \\ e_2 \end{matrix}) + b_R. \quad (27)$$

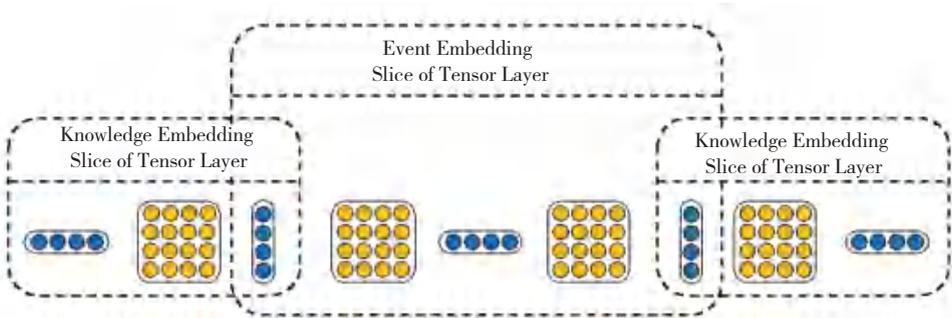


图 4 融合实体关系知识的事件表示学习框架结构

Fig. 4 Architecture of event representation learning framework enhanced with entity relation knowledge

Lee 等在 2018 年提出将情感以及事件主体的有生性信息融入事件表示中。事件的情感极性会影响后续事件的发生。例如: 带有积极情感的事件“Jenny liked the food”后, 往往不会发生带有消极情感的事件“*She scolded the server*”。事件元素的有生性也是很有价值的信息, 有些事件的主体只能是有生的, 还有些事件的含义在主体是无生命事物时会发生变化。例如: “*This song is sick*”与“*This person is sick*”。该工作将情感极性划分为“消极”、“中性”和“积极”3 个类别, 将有生性划分为“有生命”、“无生命”和“未知”3 个类别, 并将每个类别映射为一个嵌入向量, 训练事件表示与其情感类别、有生性类别的嵌入向量尽可能相似^[9]。

Ding 等在 2019 年提出将意图、情感等有关参与者心理状态的常识知识融入事件表示中, 以帮助更好地建模事件语义。例如: “*PersonX threw basketball*”与“*PersonY threw bomb*”两个事件尽管字面上相近, 但考虑两个事件的意图, “扔篮球”可能是为了锻炼身体, “扔炸弹”可能是为了杀伤敌人, 因此考虑意图信息可以较好地区分两个事件; 再如, “*PersonX broke record*”与“*PersonY broke vase*”两个事件也在字面上相近, 但“打破花瓶”带有消极的情感, “打破纪录”带有积极的情感, 因此考虑情感极

其中, μ_R, H_R, V_R, b_R 为特定于关系 R 的参数。给定一个来自知识图谱中的正确三元组与随机替换头实体或尾实体后的错误三元组, 使用最大边际损失训练正确三元组的得分高于错误三元组。训练过程中, 头实体或尾实体的词向量在反向传播时被更新, 使其含有来自知识图谱的实体关系信息, 进而使事件中的施事者或受事者通过共享词向量的形式融入这一信息。

性信息可以较好地地区分两个事件。该工作提出了一种多任务 (Multi-Task) 学习方法, 加入了两个额外的训练目标, 在事件表示中融入意图和情感的信息: 对于意图, 使用长短时记忆网络, 将意图文本编码为一个向量, 使用最大边际损失训练事件向量与意图向量尽可能相似; 对于情感信息, 使用全连接网络对事件向量进行情感分类, 并与真实的情感极性标签计算交叉熵损失。两个额外的训练目标与基于事件内或事件间信息的训练目标进行加权平均, 得到最终的损失。该方法的示意图如图 5 所示。事件的意图和情感信息可由 ATOMIC 事件常识数据集与 SenticNet 情感字典得到。实验结果表明意图、情感信息在事件相似度、脚本事件预测、股市预测任务上都带来了有效的提升^[13]。

4 结束语

本文对事件表示方法的发展进行了概述。早期的研究大多基于离散的事件表示, 随着深度学习的发展, 人们开始探索为事件学习稠密的向量表示, 其中基于张量神经网络的方法取得了令人瞩目的效果。在事件表示的学习方法上, 在融合事件内信息和事件间信息的基础之上, 融合外部知识的方法带来了新的突破。本文对各种事件表示方法的特点进行了梳理和总结, 以期为后续工作提供参考。

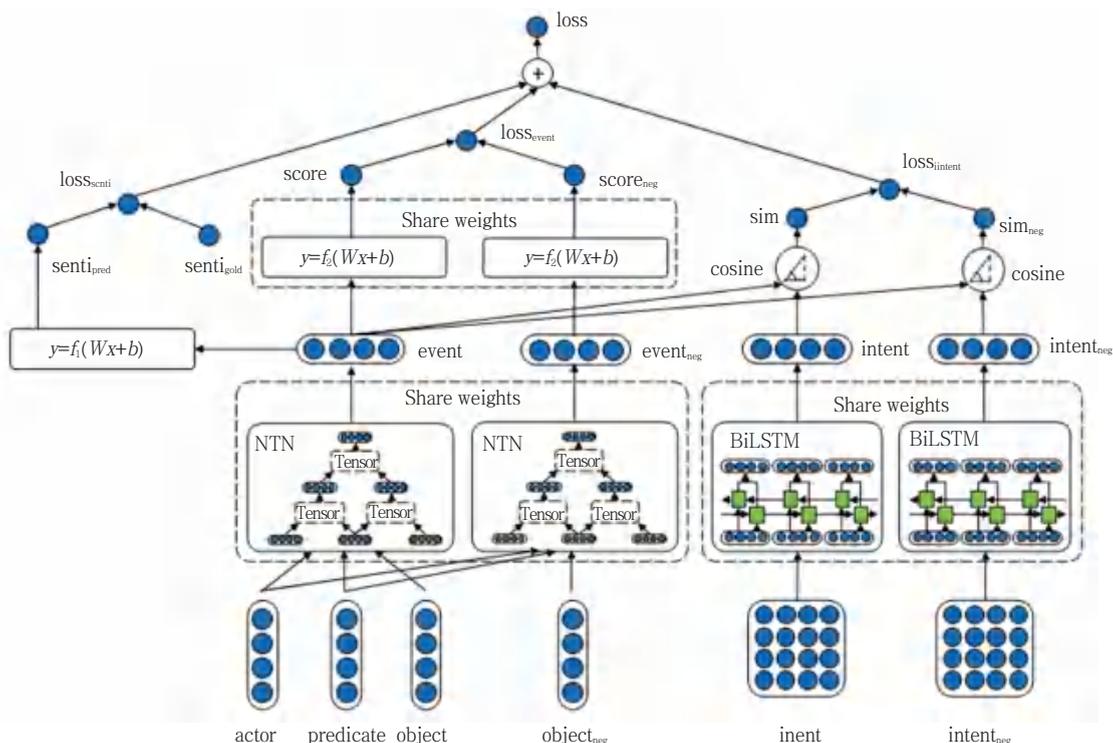


图5 融合意图、情感信息的事件表示学习框架结构

Fig. 5 Architecture of event representation learning framework enhanced with intent and sentiment information

参考文献

- [1] KIM J. Supervenience and mind: Selected philosophical essays [M]. Cambridge University Press, 1993.
- [2] RADINSKY K, DAVIDOVICH S, MARKOVITCH S. Learning causality for news events prediction [C]//Proceedings of the 21st international conference on World Wide Web. 2012: 909-918.
- [3] DING X, ZHANG Y, LIU T, et al. Using structured events to predict stock price movement: An empirical investigation [C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014: 1415-1425.
- [4] CHAMBERS N, JURAFSKY D. Unsupervised learning of narrative event chains [C]//Proceedings of ACL - 08: HLT. 2008: 789-797.
- [5] ZHAO S, WANG Q, MASSUNG S, et al. Constructing and embedding abstract event causality networks from text snippets [C]//Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. 2017: 335-344.
- [6] WEBER N, BALASUBRAMANIAN N, CHAMBERS N. Event representations with tensor-based compositions [C]//Thirty - Second AAAI Conference on Artificial Intelligence. 2018.
- [7] LI Z, DING X, LIU T. Constructing narrative event evolutionary graph for script event prediction [J]. arXiv preprint arXiv:1805.05081, 2018.
- [8] GRANROTH - WILDING M, CLARK S. What happens next? event prediction using a compositional neural network model [C]//Thirtieth AAAI Conference on Artificial Intelligence. 2016.
- [9] LEE I-T, GOLDWASSER D. Feel: Featured event embedding learning [C]//Thirty - Second AAAI Conference on Artificial Intelligence. 2018.
- [10] TILK O, DEMBERG V, SAYEED A, et al. Event participant modelling with neural networks [C]//Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. 2016: 171-182.
- [11] HONG X, SAYEED A, DEMBERG V. Learning distributed event representations with a multi - task approach [C]//Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics. 2018: 11-21.
- [12] DING X, ZHANG Y, LIU T, et al. Deep learning for event-driven stock prediction [C]//Twenty - fourth international joint conference on artificial intelligence. 2015.
- [13] DING X, LIAO K, LIU T, et al. Event Representation Learning Enhanced with External Commonsense Knowledge [J]. arXiv preprint arXiv:1909.05190, 2019.
- [14] WANG Z, ZHANG Y, CHANG C Y. Integrating order information and event relation for script event prediction [C]//Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. 2017: 57-67.
- [15] DING X, ZHANG Y, LIU T, et al. Knowledge-driven event embedding for stock prediction [C]//COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016; Technical Papers. 2016.