

基于情景记忆激活与再巩固的持续关系学习

本文基于 AAAI-2020 论文《Continual Relation Learning via Episodic Memory Activation and Reconsolidation》，论文由清华大学和腾讯微信 AI 团队合作完成。

导语：在智能搜索、自动问答等现实应用中，知识图谱正在扮演越来越重要的作用，图谱中丰富的结构化信息可以辅助实现更加精细的语义理解。为了不断地完善现有知识图谱的覆盖度，关系抽取被广泛研究，力图从无结构化的海量文本中挖掘出结构化知识。然而，在现实的开放文本环境下，实体间新的语义关系也在不断出现，如何持续地学习这些新关系就成为了一大难题，尤其在不大量使用历史数据反复重新训练模型的前提下。对此，作者提出一套持续关系学习模型，支持模型不断地在新关系的数据上进行训练，在掌握新关系的特性的同时，避免灾难性地忘记旧关系的特性。具体而言，我们采用了基于情景记忆和记忆回放机制的持续学习框架，并额外引入了记忆再巩固阶段。再巩固阶段中，我们采用了原型学习的思路，确保每次情景记忆模块激活与回放后，旧关系的记忆样本与记忆原型依然匹配，使得模型能够较好地保留对旧关系的认知。作者在当前主流的关系学习数据集上进行了系统评测，与已有持续学习算法相比效果提升显著，模型记忆保持能力得到了充分验证。

模型背景与简介

信息抽取旨在从大规模非结构或半结构的自然语言文本中抽取结构化信息，关系抽取是其中的重要子任务之一，用以从文本中检测实体之间的丰富语义关系。例如，给定句子“牛顿担任英国皇家学会主席”，可以从中提取出牛顿与英国皇家学会之间的会员从属关系。通

过关系抽取,我们可以极大地丰富知识图谱中的结构化信息,进而作为各类下游应用的外部资源。

传统的关系抽取方法主要侧重于识别预定义关系集中的关系。然而,这样的设定在实际工作场景下是非常受限的。在现实的开发文本场景下,新生的关系在源源不断地出现,作用于预定义关系集合的传统关系抽取方法无法解决这些新关系。为了在开放域下能够持续地处理新关系,最近的相关工作可以被归纳为两个方面:(1)开放关系抽取。学习提取句子中的词法、句法结构来构造特定的关系模式,通过聚类算法从中发现新的关系类型,并在大规模文本语料中扩展包含这些新关系语义的样本;(2)持续关系学习。不断使用新关系及其样本,来训练出一个高效的分类器。分类器将接受一系列任务的训练,每个任务都对应一个特定关系集合的,最终分类器能够同时处理现有和新出现关系的判别。虽然持续关系学习对于学习不断出现新关系至关重要,但相比于开放关系抽取,该领域仍然没有被充分探索。

对于持续关系学习,一种简单有效的方法是在每次出现新的关系及其对应样本时,都把它们存储进来一同参与训练,形成一种多任务学习的模式。然而,由于关系数量在持续增长,每个关系也拥有大量样本,频繁地混合新旧样本导致计算成本极为高昂。近期的一些工作提出了一些基于情景记忆和记忆重演的方法,即保存一些历史任务的训练样本,在未来重新训练,以维持分类器对历史数据的认知。这种记忆模型的主要问题在于易于过拟合少数的记忆样本,它们频繁改变旧关系的特征分布,逐渐过拟合记忆中的那些样本,在长期学习中逐渐发生混淆。因而提出一个能够在持续关系学习中较好地维持记忆、减少混淆,同时学习新出现关系的模型,就十分具有意义。

神经科学中有这样的发现,信息的重演、巩固有助于长期记忆的形成。已形成的长期记忆在被检索激活后,脑内的再巩固机制会对记忆进行再次存储。在再巩固阶段中,被激活的记忆十分脆弱,且易被更改。因而,在这个过程中,引入特殊的练习可以起到编辑记忆或强

化记忆的作用。出于此，作者提出一种基于情景记忆激活与再巩固的持续关系学习算法 (Episodic Memory Activation and Reconsolidation , EMAR) , 在已有记忆与重演的基础上，引入了额外的记忆再巩固阶段。在每次神经网络被激活以学习新关系前，EMAR 从历史关系的记忆样本中采样部分，用以计算历史关系的记忆原型。在神经网络被激活并完成一定的新关系学习后，EMAR 会启动再巩固机制，确保历史关系的所有记忆样本能与相应记忆原型匹配，以此提升模型在学习新关系后对旧关系认知的保持能力。

模型结构

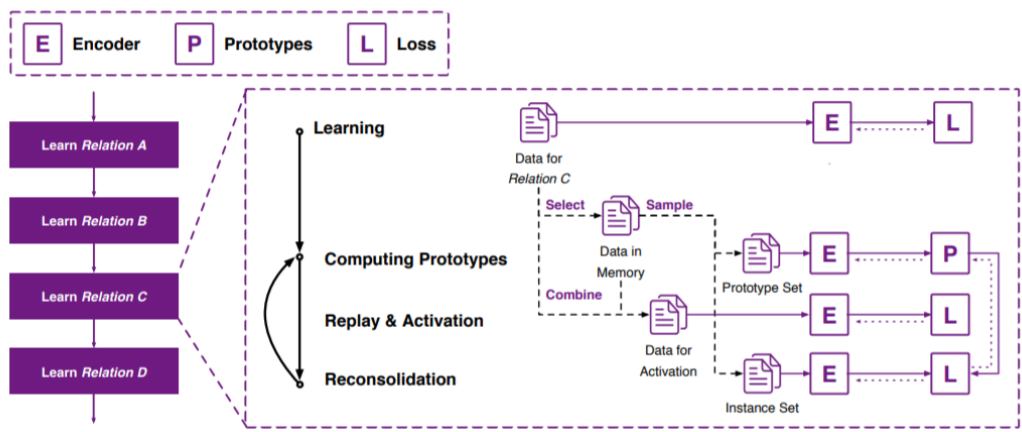


图 1：EMAR 的整体模型架构

图 1 给出了 EMAR 模型的整体架构。在开始一个新任务的学习时，模型通过编码器得出新关系样本的特征向量，给出涉及新关系的判断结果，并通过判断结果对编码器参数进行微调，从而最终学习新关系的特性。在一个新任务学习后，模型需要从这些已经学习的新关系样本中抽取一部分，以尽可能代表该任务中新关系在特征空间里的分布，保存在记忆中。这些记忆样本将在后续记忆重演、再巩固机制里发挥作用，对编码器与分类器进行进一步的参数优化，以确保既能够学习掌握新关系同时避免遗忘旧关系。

我们可以将持续关系抽取视作一系列任务上的训练,每个任务分为训练集 T_k ,验证集 V_k ,以及测试集 Q_k 。对于第 k 个任务,训练集用 $T_k = \{X^{T_k}, Y^{T_k}\}$ 表示,其包括一系列样本 $X^{T_k} = \{x_1^{T_k}, \dots, x_N^{T_k}\}$,以及对应的关系标记 $Y^{T_k} = \{y_1^{T_k}, \dots, y_N^{T_k}\}$, N 为 T_k 中样本容量。对于任一样本 $x_i^{T_k} \in X^{T_k}$ 以及其标记 $y_i^{T_k} \in Y^{T_k}$,表示样本 $x_i^{T_k}$ 蕴含着关系 $y_i^{T_k} \in R_k$ 的语义信息,这里 R_k 是第 k 个任务的新关系集合。当第 k 个任务出现时,模型会在训练集 T_k 上训练,来学习新的关系 R_k 。虽然新的关系在不断地出现,持续关系学习仍然需要模型在第 k 个任务以及前 $k-1$ 个任务上均有良好的表现。在利用 T_k 训练结束后,模型需要在 $\tilde{Q}_k = \bigcup_{i=1}^k Q_i$ 上测试,并且需要将所有测试样本在所有当前已知的关系集合 $\tilde{R}_k = \bigcup_{i=1}^k R_i$ 上进行分类预测。此外,为了应对记忆的丢失,模型中有情景记忆 $M = \{M_1, M_2, \dots\}$ 来保存一些过去任务中的样本,其中每个记忆单元 $M_k = \{X^{M_k}, Y^{M_k}\}$ 保存了若干样本,这些样本可以表示为 $X^{M_k} = \{x_i^{M_k}, \dots, x_B^{M_k} | x_i^{M_k} \in X^{T_k}\}$,以及它们的标记 $Y^{M_k} = \{y_i^{M_k}, \dots, y_B^{M_k} | y_i^{M_k} \in Y^{T_k}\}$, B 是记忆模块容量上限。

更具体地：

(1) 模型从样本中抽取句子语义信息、头尾实体位置等多种信息,解析成一个标记序列,再将标记输入神经网络,计算出相应的嵌入向量,用于进行实体关系预测。样本编码操作可以用如下方程表示：

$$x = f(x) \quad (1)$$

这里, $x \in \mathbb{R}^d$, 是 x 的语义嵌入向量, d 是向量维度。编码采用的具体模型并不是关键,作者在这里选取双向长短期记忆模型(BiLSTM)来作为对样本进行编码操作的编码器。

(2) 当第 k 个任务出现时,模型还尚未了解该任务中的新关系,模型利用训练集 $T_k = \{\{x_1^{T_k}, \dots, x_N^{T_k}\}, \{y_1^{T_k}, \dots, y_N^{T_k}\}\}$ 来掌握 R_k 中新关系的模式,对样本编码模块中的参数进行微调。

该模块中针对任务 k 的损失函数如下定义：

$$L(\theta) = -\sum_{i=1}^N \sum_{j=1}^{|\bar{R}_k|} \delta_{y_i^{T_k}=r_j} \log \frac{\exp(g(\mathbf{x}_i^{T_k}, \mathbf{r}_j))}{\sum_{l=1}^{|\bar{R}_k|} \exp(g(\mathbf{x}_i^{T_k}, \mathbf{r}_l))} \quad (2)$$

其中, $\mathbf{x}_i^{T_k}$ 是样本 $x_i^{T_k}$ 的嵌入向量。 \mathbf{r}_j 是第 j 个关系 $r_j \in R_k$ 的嵌入向量, $g(\cdot, \cdot)$ 是一个计算嵌入向量之间相似性的函数, 如余弦相似度函数。 θ 是所有可以被优化的参数, 包括样本编码模块中的参数以及关系嵌入向量。对于任一新关系, 首先随机初始化其嵌入向量, 随后使用(2)式进行优化。

(3) 在若干次利用公式(2)进行迭代后, 模型从 T_k 中提取若干样本, 将这些样本存入记忆单元 M_k 。更明确地, 模型从 T_k 中选取更有效的样本, 来尽可能涵盖新的关系模式。这样能够使得记忆有效接近关系的特征分布。模型对第 k 个任务的训练集 T_k 编码, 形成 $\{\mathbf{x}_1^{T_k}, \dots, \mathbf{x}_N^{T_k}\}$ 后, 使用 K 均值来对这些样本嵌入向量进行聚类, 而聚类的中心数等于记忆容量 B 。对于每个类, 选择其中最接近聚类中心的样本, 并记录其所属的关系类别。通过统计各个关系对应样本被选中的个数, 得出这个关系对于代表这个任务的重要程度, 被选中的个数越多, 这个关系便更具有重要性。对于重要的关系, 在记忆单元中其对应的样本个数至少为 $\lfloor \frac{B}{|R_k|} \rfloor$, 对于相对不重要的关系, 在记忆单元中其对应的样本个数至多为 $\lceil \frac{B}{|R_k|} \rceil$ 。如果一个关系所对应的样本个数不足以填充为其分配的内存容量, 那么空余的容量会重新分配给其他关系。在每个关系分配空间之后, 对于任一个关系, 仍然使用 K 均值来对其样本进行聚类, 聚类个数为其分配的样本个数, 对于每个类, 选择最接近类形中心的样本存入记忆单元 M_k 。

(4) 在利用训练集 T_k 对模型参数微调、选取样本填充记忆单元 M_k 后, 模型将通过记忆原型计算、记忆重演与激活以及记忆再巩固三个阶段, 力图在学习新关系的同时, 保留对旧关系的识别能力。

原型计算: 通过结合所有情景记忆中的样本, 模型获得了整体记忆集合 $\tilde{M}_k = \bigcup_{i=1}^k M_i$ 。

对于任意已知关系 $r_i \in \tilde{R}_k$,模型采样出一个原型集合 $P_i = \{x_1^{P_i}, \dots, x_{|P_i|}^{P_i}\}$,其中样本 $x_i^{P_i} \in P_i$ 均选自 \tilde{M}_k ,且对应的类别标签满足 $y_i^{P_i} = r_i$ 。以此 ,可计算出关系的记忆原型 :

$$p_i = \frac{\sum_{j=1}^{|P_i|} f(x_j^{P_i})}{|P_i|} \quad (3)$$

记忆重演和激活 : 将记忆集合 \tilde{M}_k 和第 k 个训练集 T_k 合并起来 ,生成激活样本集合 $A_k = \{\{x_1^{A_k}, \dots, x_M^{A_k}\}, \{y_1^{A_k}, \dots, y_M^{A_k}\}\}$ 来持续激活模型 ,以学习新的关系并一定程度地记住旧关系。在这里 , M 是 \tilde{M}_k 和 T_k 样本个数之和。损失函数定义为下式 :

$$L^A(\theta) = -\sum_{i=1}^M \sum_{j=1}^{|\tilde{R}_k|} \delta_{y_i^{A_k}=r_j} \log \frac{\exp(g(x_i^{A_k}, r_j))}{\sum_{l=1}^{|\tilde{R}_k|} \exp(g(x_i^{A_k}, r_l))} \quad (4)$$

这里 r_j 是第 j 个已知关系 $r_j \in \tilde{R}_k$ 的嵌入向量。

记忆再巩固 : 正如前文背景部分介绍到 ,仅仅通过情景记忆的重演和激活会导致过拟合 ,在长期训练后 ,模型最终只能记住若干被记住的样本。同时 ,学习关系的核心在于掌握关系原型而非死记硬背关系样本。所以 ,每次进行记忆重演和激活后 ,作者添加了额外的再巩固阶段 ,并采取如下方式对记忆进行增强 :

对于任一已知关系 $r_i \in \tilde{R}_k$,类似于之前对原型的采样 ,采样出它对应的实例集合 $I_k = \{x_1^{I_i}, \dots, x_{|I_i|}^{I_i}\}$,其中 $x_1^{I_i} \in I_i$ 同样来自于 \tilde{M}_k ,其对应关系标签仍然满足 $y_i^{I_i} = r_i$,损失函数被定义为下式 :

$$L^R(\theta) = \sum_{i=1}^{|\tilde{R}_k|} \sum_{j=1}^{I_i} \log \frac{\exp(g(x_j^{I_i}, p_i))}{\sum_{l=1}^{|\tilde{R}_k|} \exp(\exp(g(x_j^{I_i}, p_l)))} \quad (5)$$

其中 p_l 是关系 $r_l \in \tilde{R}_k$ 通过(3)式计算出的原型。

(5) 整体训练过程如下图所示。

Algorithm 1 Train EMAR for the k -th task

Require: The training set \mathcal{T}_k of the k -th task
Require: The emerging relation set \mathcal{R}_k of the k -th task
Require: The memory module \mathcal{M}_{k-1} before learning \mathcal{T}_k
Require: The known relation set $\tilde{\mathcal{R}}_{k-1}$ before learning \mathcal{T}_k

```
1: Initialize the relation embeddings for  $\mathcal{R}_k$ 
2:  $\tilde{\mathcal{R}}_k \leftarrow \tilde{\mathcal{R}}_{k-1} \cup \mathcal{R}_k$ 
3: for  $i \leftarrow 1$  to  $epoch_1$  do
4:   Update  $\theta$  with  $\nabla \mathcal{L}$  on  $\mathcal{T}_k$ 
5: end for
6: Select informative examples from  $\mathcal{T}_k$  to store into  $\mathcal{M}_k$ 
7:  $\mathcal{M}_k \leftarrow \mathcal{M}_{k-1} \cup \mathcal{M}_k$ 
8:  $\mathcal{A}_k \leftarrow \tilde{\mathcal{M}}_k \cup \mathcal{T}_k$ 
9: for  $i \leftarrow 1$  to  $epoch_2$  do
10:   for relation  $r_j \in \tilde{\mathcal{R}}_k$  do
11:     Sample  $\mathcal{P}_j$  from  $\tilde{\mathcal{M}}_k$  and compute its relation
       prototype embedding  $p_j$ 
12:   end for
13:   for  $j \leftarrow 1$  to  $iter_1$  do
14:     Update  $\theta$  with  $\nabla \mathcal{L}^{\mathcal{A}}$  on  $\mathcal{A}_k$ 
15:   end for
16:   for  $j \leftarrow 1$  to  $iter_2$  do
17:     Sample  $\mathcal{I}_i$  from  $\tilde{\mathcal{M}}_k$  for each known relation  $r_i$ 
18:     Update  $\theta$  with  $\nabla \mathcal{L}^{\tilde{\mathcal{R}}}$  on  $\{\mathcal{I}_1, \dots, \mathcal{I}_{|\tilde{\mathcal{R}}_k|}\}$ 
19:   end for
20: end for
```

在第 k 个任务结束后，对于任一已知的关系 $r_i \in \tilde{\mathcal{R}}_k$ ，模型收集其在记忆中的所有样本 $\{x_1, \dots, x_S\}$ ，其中 S 为该关系在记忆中的样本个数，然后计算最终的关系原型来进行推理和预测：

$$\tilde{p}_i = \frac{r_i + \sum_{j=1}^S f(x_j)}{1+S} \quad (6)$$

其中 r_i 是 r_i 这一关系的关系嵌入向量，通过式 (2) 与式 (4) 可学习得到。对于任一在 $\tilde{\mathcal{Q}}_k$ 中的测试样本 x ，作者定义其对于关系 r_i 的评分函数为

$$s(x, r_i) = g(f(x), \tilde{p}_i) \quad (7)$$

其中 \tilde{p}_i 是 r_i 最终计算出的原型。最终，测试样本的预测值 \tilde{y} 为

$$\tilde{y} = \arg \max_{r_i \in \tilde{\mathcal{R}}_k} s(x, r_i) \quad (8)$$

实验结果

作者在 FewRel、SimpleQA、TACRED 三个数据集上进行了测试。由于是持续关系学习，需要关注的是平均表现(Average)和整体表现(Whole)，前者是模型在某个训练阶段时在所有已知关系数据上的平均结果，主要能够表现模型对记忆的保持能力，后者是模型对所有关系测试数据的结果。测试结果如下：

| | FewRel | | SimpleQ | | TACRED | |
|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | W | A | W | A | W | A |
| Lower Bound | 18.9 | 20.8 | 63.2 | 56.9 | 12.3 | 9.5 |
| EWC | 27.1 | 30.2 | 67.2 | 59.0 | 14.5 | 14.5 |
| GEM | 49.2 | 59.8 | 84.1 | 79.6 | - | - |
| AGEM | 36.1 | 42.5 | 77.6 | 72.2 | 15.7 | 16.0 |
| EMR | 51.0 | 62.0 | 85.2 | 80.8 | 28.7 | 35.6 |
| EA-EMR | 56.6 | 67.3 | 87.8 | 82.4 | 30.5 | 40.5 |
| EMAR | 66.0 | 77.9 | 85.2 | 83.7 | 44.5 | 54.4 |
| Upper Bound | 81.9 | 85.8 | 88.9 | 84.1 | 74.3 | 77.0 |

图 2：持续学习表现测试结果

在两种指标下，EMAR 模型除了 SimpleQA 的整体表现之外，在各个数据集上都得到了最佳的效果，说明了 EMAR 对于旧关系知识有较好的保持能力。图 3 也显示，随着任务的增长，EMAR 预测结果的下降要更为缓慢，这也说明了 EMAR 更不容易遗忘历史知识。

实验通过 Upper Bound 指标说明 SimpleQA 数据集上已难产生显著提升。SimpleQA 表现较为薄弱的另一个原因在于，SimpleQA 的关系数量远大于记忆模块的容量，EMAR 需要记忆模块对于每个关系至少保留一个样本，因而再巩固阶段与测试阶段，EMAR 中关系的记忆原型无法运作，低记忆样本的启动问题将是我们下一步的研究点之一。

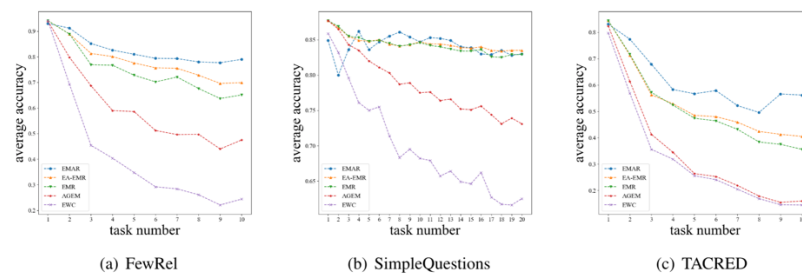


图 3：任务增长下的预测结果变化

除此之外，作者还进行了多个记忆容量大小下的对比实验。以图 4 为例，可以发现随着

记忆容量提高，各个模型的表现均有提升，说明记忆容量是决定持续关系学习模型表现的关键因素；对于 FewRel 和 TACRED 数据集，EMAR 模型在各个记忆容量下都具有最好的表现，甚至和具有更大记忆容量的其他模型相比，说明 EMAR 模型中的记忆再巩固机制更有效得利用了记忆。

| | FewRel | | | | | | SimpleQ | | | | | | TACRED | | | | | |
|--------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | 10 | | 25 | | 50 | | 10 | | 25 | | 50 | | 10 | | 25 | | 50 | |
| | W | A | W | A | W | A | W | A | W | A | W | A | W | A | W | A | W | A |
| EWC | - | - | - | - | 21.3 | 24.4 | - | - | - | - | 63.9 | 62.5 | - | - | - | - | 14.5 | 14.5 |
| AGEM | 29.0 | 34.0 | 33.8 | 39.0 | 41.2 | 47.5 | 69.1 | 66.1 | 72.2 | 69.2 | 76.2 | 73.1 | 14.7 | 14.5 | 15.0 | 15.5 | 15.7 | 16.0 |
| EMR | 42.0 | 54.1 | 49.0 | 60.5 | 53.6 | 65.1 | 81.5 | 77.4 | 84.9 | 81.0 | 86.9 | 82.9 | 21.8 | 26.5 | 25.7 | 31.6 | 28.7 | 35.6 |
| EA-EMR | 49.0 | 61.2 | 54.9 | 66.4 | 59.1 | 69.9 | 83.3 | 78.7 | 86.4 | 82.0 | 87.9 | 83.5 | 23.0 | 30.0 | 27.7 | 37.0 | 30.5 | 40.5 |
| EMAR | 53.8 | 69.1 | 62.5 | 74.9 | 66.0 | 77.9 | 80.9 | 78.7 | 84.6 | 81.4 | 85.2 | 83.7 | 31.0 | 36.3 | 37.8 | 48.5 | 44.5 | 54.4 |

图 4：不同记忆容量下的模型表现

总结

本文提出了一种新的持续关系学习方法 EMAR，能够较好地维持对历史关系的认知，同时支持学习新出现的关系。通过在原有情景记忆与记忆重演模型基础上引入额外的记忆再巩固模块，并采用原型学习的思想确保历史关系在模型参数激活变更前后原型匹配，最终实现了学习新知识与记忆旧知识的优秀平衡。EMAR 的有效性在现有的几个质量较高的数据集上也得到了验证。