# TransE算法及应用示例

## 一、内容简介

TransE算法是知识图谱表示学习算法中的首个基于Translation的方法，也可以说TransE算法是基于Translation的后续方法的起始。TransE算法来源于论文《Translating Embeddings for Modeling Multi-relational Data》。我们对知识图谱进行简单的定义，知识图谱是一种通用的用图结构的方式对真实世界的万事万物进行建模的工具，因此，其具备的直观强大的表示能力使得人们能够很容易理解并进行应用，但是计算机只能处理0和1这样的数字，因此，知识图谱表示学习的目的就是将知识图谱中的元素，也即节点和边表示成可计算的向量形式，从而可供计算机进行计算。

TransE算法首次采用了基于Translation的方法，作为该类知识图谱表示学习方法的开山之作，也在后续许多工作中进行了应用。此外，后续许多针对于TransE算法的改进工作，如TransH、TransR等，共同构成了知识图谱表示学习的Trans系列算法。

## 二、理论基础

### 2.1 知识点一 知识图谱基本理论

2.1.1 什么是知识图谱

知识图谱是一种用图模型来描述知识和建模世界万物之间的关联关系的技术方法。知识图谱由节点和边组成，其中节点可以是任意实体如一个人、一本书等，也可以是抽象的概念，例如人工智能、知识图谱等；边可以是实体之间的关系，也可以是实体的属性。

知识图谱旨在从数据中识别、发现和推断事物与概念之间的复杂关系，是事物关系的可计算模型。知识图谱的构建涉及知识建模、关系抽取、图存储、关系推理、实体融合等多方面的技术，如图1所示展示了更加细分的知识图谱技术要素。此外，知识图谱的利用又涉及自然语言理解和处理、语义搜索、知识问答、自动推理、机器学习、数据库等多个技术方面，大数据时代促进了人工智能算法的进步，也为规模化知识图谱的构建提供了新的技术基础和发展条件。

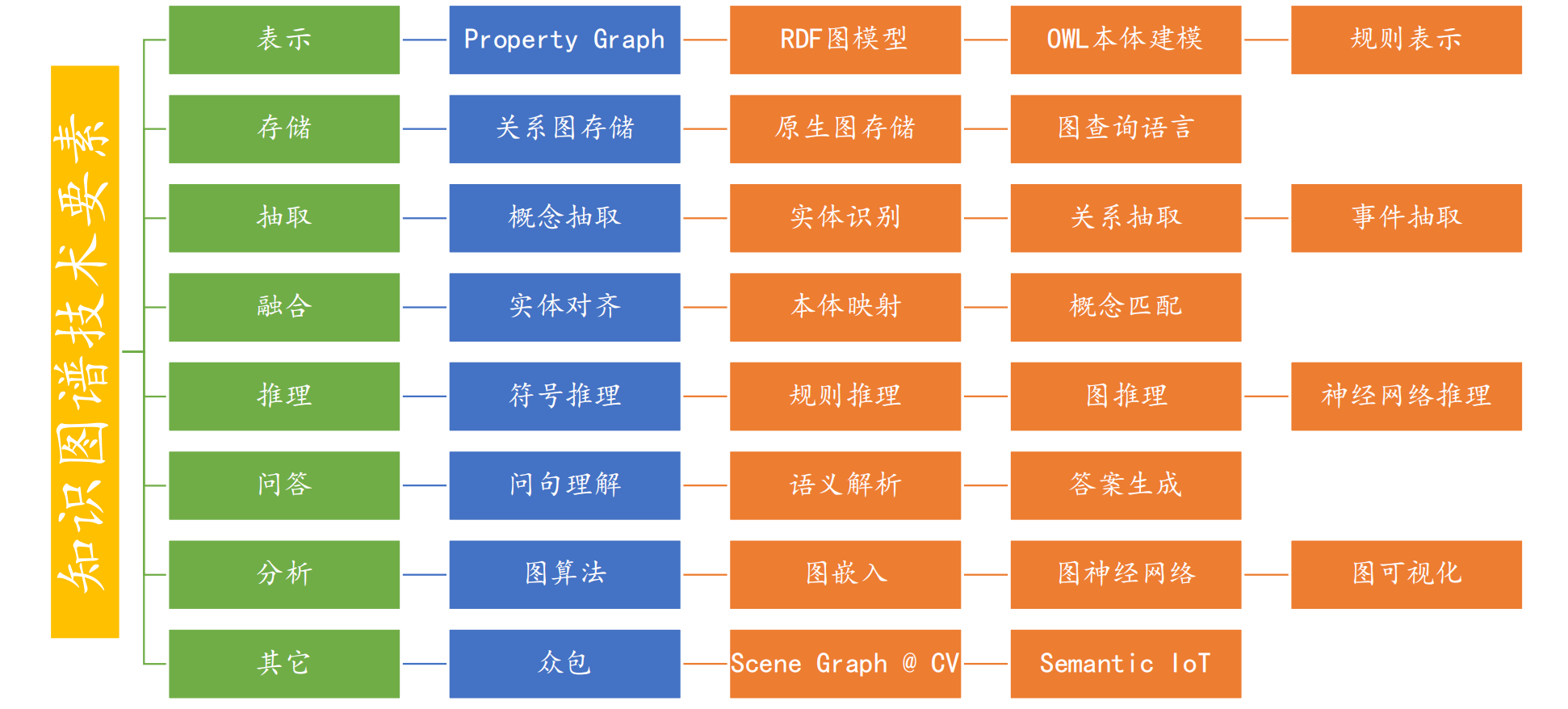


图 1 知识图谱技术要素

2.1.2 知识图谱的发展简史

二十世纪六十年代提出的语义网络，可以认为是知识图谱的雏形。如图2所示是一个简单的发展时间轴。1989年Lee提出构建全球化的以链接为中心的信息系统，以链接为中心和基于图的组织方式更加适合互联网这类开放系统。1998年，Lee提出Semantic Web的概念，其中节点不单单只表示网页，还表示客观世界中的实体，超链接也被增加了语义描述。在此之后，涌现了一大批新兴的语义知识库，如谷歌知识图谱后端的Freebase，IBM的Waston后端的DBpedia和YAGO，Wikidata等。2010年谷歌收购了早期语义网公司MetaWeb，以Freebase为基础于2012年正式推出了称为知识图谱的搜索引擎服务，随后知识图谱逐步在语义搜索、智能问答等领域发挥越来越重要的作用。

概括来讲，知识的表示形式从最初的简单的图结构化的只是表示方式，逐渐演变成了如今具有识别、发现、推断事物与概念的复杂关系的可计算的模型，也即是现在的知识图谱。



图 2 知识图谱发展简史

### 2.2 知识点二 表示学习与知识图谱的表示学习

2.2.1 表示学习

表示学习，英文为Representation Learning，又被称为特征学习或表征学习。在深度学习领域，表示是指通过模型的参数，采用何种形式、何种方式来表示模型的输入观测样本，其能将原始数据转换成为能够被机器学习有效利用的一种形式，最常见的就是把一个原始数据样本表示为一个低维、连续空间中的向量。表示学习是学习一个特征的技术的集合，将原始数据转换成为能够被机器学习来有效利用的形式，避免了手动提取特征的麻烦，也允许计算机学习使用特征的同时学习到图和提取特征。表示的选择通常取决于随后的具体任务，总的原则就是一个好的表示应该使随后的任务的学习变得更加容易。

从统计学习的狭义的角度来看，表示学习得到的低维向量表示是一种数据特征的分布式表示(distributed representation)。之所以如此命名，是因为孤立地看向量中的每一维，都没有明确对应的含义；而综合各维形成一个向量，则能够表示对象的语义信息。表示学习有很多种形式，比如CNN参数的有监督训练是一种有监督的表示学习形式，对自动编码器和限制玻尔兹曼机参数的无监督预训练是一种无监督的表示学习形式，对DBN参数-先进行无监督预训练，再进行有监督fine-tuning-是一种半监督的共享表示学习形式。

2.2.2 知识图谱的表示学习

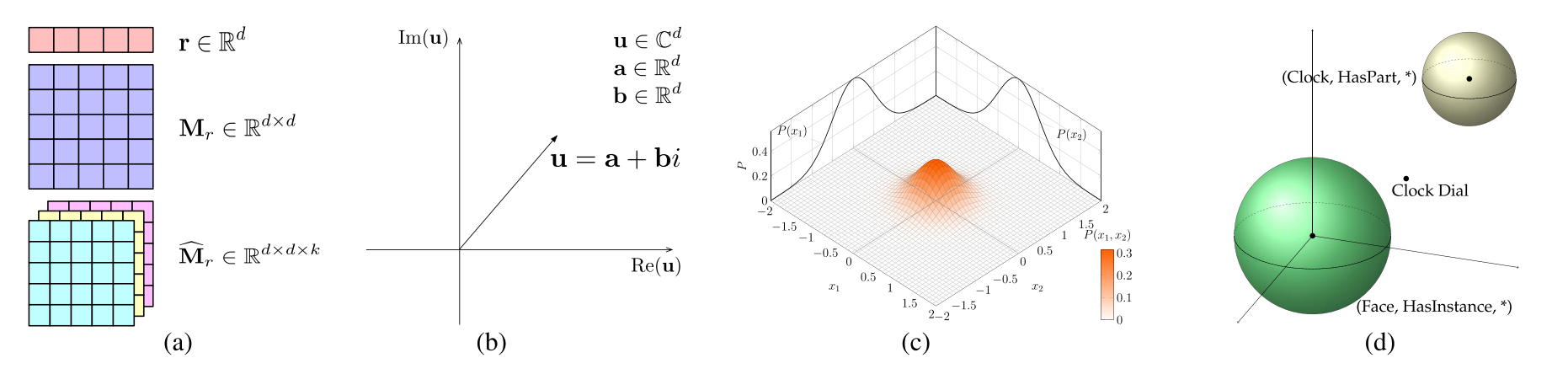
前面提到，知识图谱这种离散化的符号化表达方式可以有效将数据结构化，但是这些符号并不能在计算机中表达相应的语义层面的信息，也不能进行语义计算，这对具体的基于知识图谱的下游应用并不友好。此外，由于图结构的特殊性，应用算法的使用和图算法有关，相关算法也具有很高的复杂度，面对大规模的知识库时也很难有效扩展。因此，对于知识图谱的表示学习对于变化各异的应用来说是非常重要的。

在自然语言的词向量的启发下，考虑将知识图谱中的实体和关系映射到连续的向量空间，并且包含一些语义层面的信息，并且对于计算机来说，连续向量的表达可以蕴含更多的语义，更容易被计算机理解和操作。

因此，我们给出定义：知识图谱的表示学习就是对图谱中所包含的实体和关系投影到连续向量空间中，实现对实体和关系的语义信息的表示，可以高效计算实体、关系之间的语义关联。

进一步地，对于向量型的知识图谱表示学习方法，其优点在于：（1）可以提高应用时的计算效率，基于数值计算的算法的计算效率也会更高；（2）增加了下游应用设计的多样性，向量化表示更加适应于机器学习算法，因为输入不再是符号，因此也就不再局限于图算法；（3）将知识图谱表示学习的结果作为下游应用的预训练向量输入，使得输入的信息不再是孤立的不包含语义信息的符号，而是已经经过一次训练，并且包含一定信息的向量。

综上所述，对于知识图谱进行表示学习，可以显著提高计算效率，增加下游应用的多样性，并且可以作为预训练为下游模型提供语义支持。因此，现在已经与许多类型的知识图谱表示学习算法，例如在不同的几何空间层面，就有基于网格空间（point-wise）、基于向量空间（vector-space）、基于高斯分布（Gaussian distribution）和基于流型空间（Manifold space）的方法。



### 2.3 知识点三 TransE核心思想

前面已经提到知识图谱表示学习，简单来讲就是把图谱中的实体和关系表示成低维、连续向量空间中的向量的形式，从而表示其语义并进行计算。

简单来讲，TransE算法就是将知识图谱中的每个三元组（head，relation，tail）表示为对应的向量***h，r，t***，但是需要满足一个约束条件，即***h*** + ***r*** ≈ ***t*。**也即头实体向量和关系向量的和要尽可能与尾实体向量相等，如果我们用一个二维平面坐标系来进行说明的话，即嵌入向量之间的关系应当如下图所示：

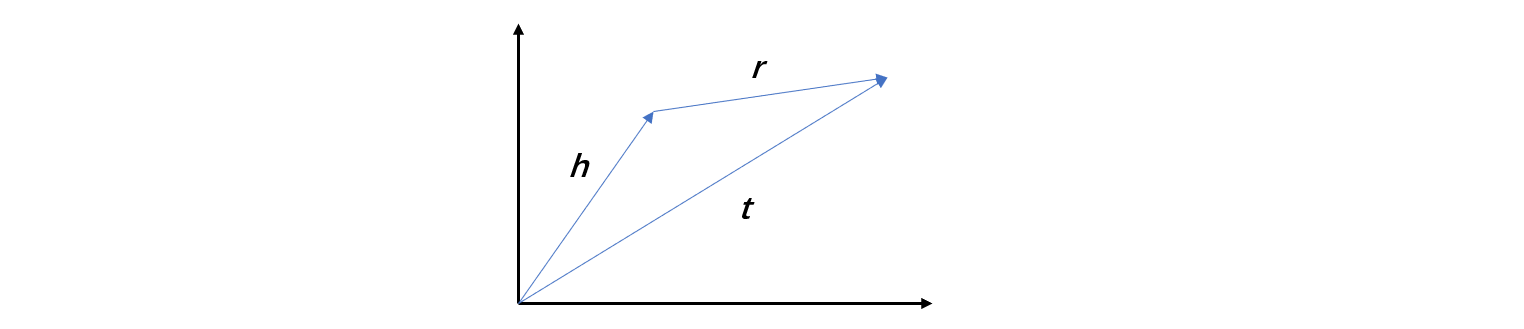
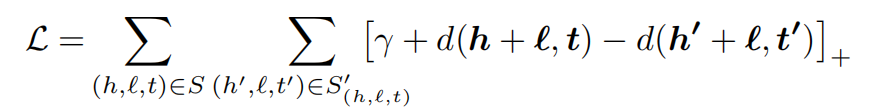


图 3 TransE算法示意图

如果将实体和关系看作两个矩阵，实体矩阵为n×d，表示n个实体，每个实体的embedding维度为d维；关系矩阵维r×d，表示r个关系，每个关系的embedding维度为d维；因此，在TransE训练完成后，理想情况下，从实体矩阵和关系矩阵中抽取一对对应的向量，其相加得到的结果为该三元组的尾实体的向量，也即实体矩阵中的某一列向量。因此，在这里我们可知，算法的Translation的含义，可以理解为翻译、即实体之间通过关系向量的翻译建立连接，从而翻译语义；也可以理解为转移，即实体之间的关系通过关系向量进行转移，从而建立语义上的连接。因此，TransE算法所要优化的目标函数如下所示：

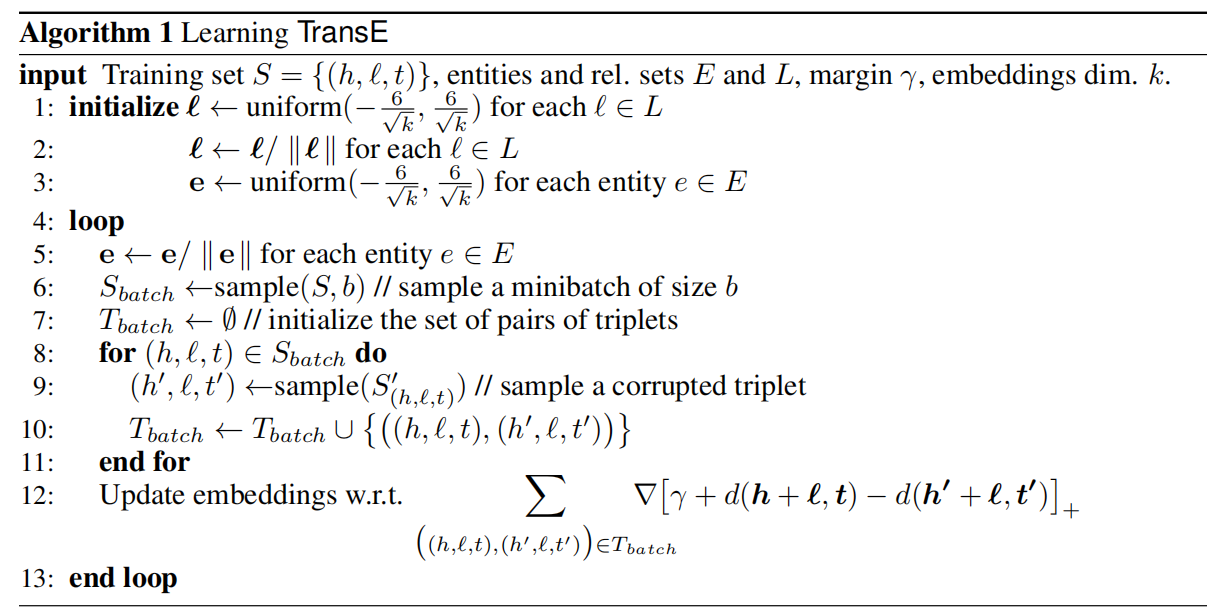


### 2.4 知识点四 TransE算法流程

2.4.1 总体算法流程

TransE算法的总体流程如下所示：

1. 首先为实体和关系进行初始化；
2. 对实体和关系的初始化向量进行规范化；
3. 从所有三元组样本中采样得到训练样本三元组集合；
4. 基于训练三元组，替换头实体或尾实体构造负样本三元组集合；
5. 将正负样本三元组集合进行合并，作为训练集；
6. 通过目标函数（得分函数/距离函数），最大化正负三元组样本之间的距离；
7. 通过SGD算法优化损失函数从而更新embedding表示；
8. 输出得到实体的embedding结果和关系的embedding结果；



2.4.2 算法输入输出

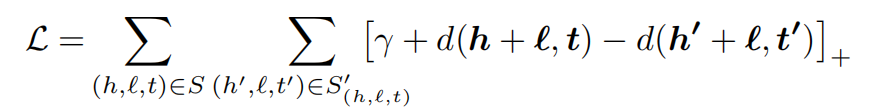
算法输入：三元组集合S；实体集合E、关系集合L；嵌入维度k；

算法输出：实体和关系的embedding向量表示；

2.4.3 负样本构造

由于负样本的构造是基于已有的三元组集合来进行头实体或尾实体的替换，从而构造出图谱中并不存在的三元组，并最大化正负三元组之间的差异；因此，实际应用中可以选择随机头实体或尾实体进行替换、也可以设置两个参数tph或hpt来选择更换头实体还是更换尾实体。

2.4.4 损失函数及优化



TransE算法要优化的损失函数如上所示，d(·)表示的即是三元组的头实体加关系与尾实体向量的关系，而h’和t’是替换后的负样本三元组中的头实体或尾实体。损失函数的优化的理想情况为对于正确的三元组，其d尽可能小，负样本三元组其d要尽可能大，这样整体的Loss趋近于0。因此，在SGD算法的参数更新过程中，正样本中的h和r逐渐减小，t逐渐增大；而负样本中h’和r逐渐增大，而t’则逐渐减小。

### 2.5 知识点五 TransE算法优缺点

2.5.1 TransE算法优点

（1）首次提出采用基于Translation的知识表示学习算法；

（2）容易训练，可以扩展到比较大的图谱当中；

（3）相较于以往模型，参数较少、计算复杂度低、能直接建立实体和关系之间的语义联系；

2.5.2 TransE算法缺点

（1）适用于处理一对一的单一关系型数据，对于复杂关系的表示学习效果不太理想；

（2）仅利用图谱结构信息，忽视了多源信息的融合；

## 三、演示验证

### 3.1 运行环境搭建

Anaconda, pytorch, python, cuda, streamlit

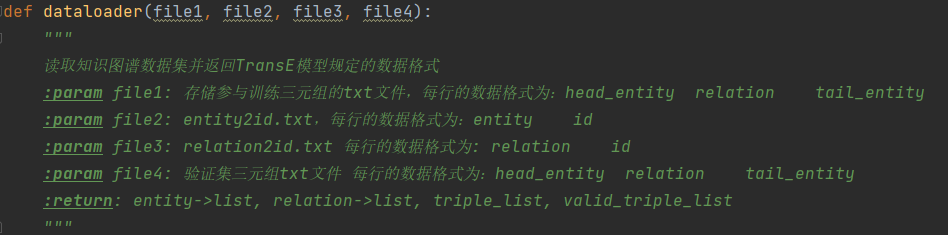
### 3.2 实验任务一 数据集加载

步骤1.1 读取数据集

读取数据集的triples.txt（三元组文件）、entity2id.txt（实体文件）、relation2id.txt（关系文件）；

步骤1.2 提取数据信息

提取三元组信息、实体信息和关系信息、分别存储到三个list中；



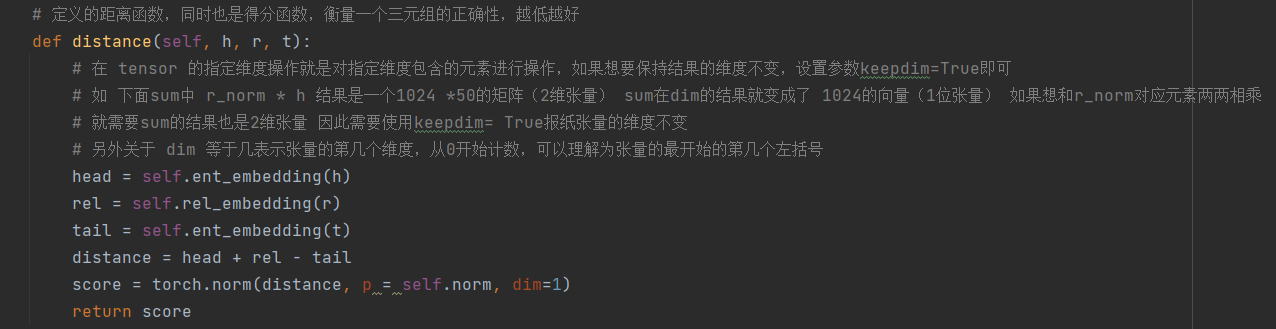
### 3.3 实验任务二 TransE核心代码

步骤2.1 初始化实体和关系的embedding



步骤2.2 对实体和关系的embedding进行规范化

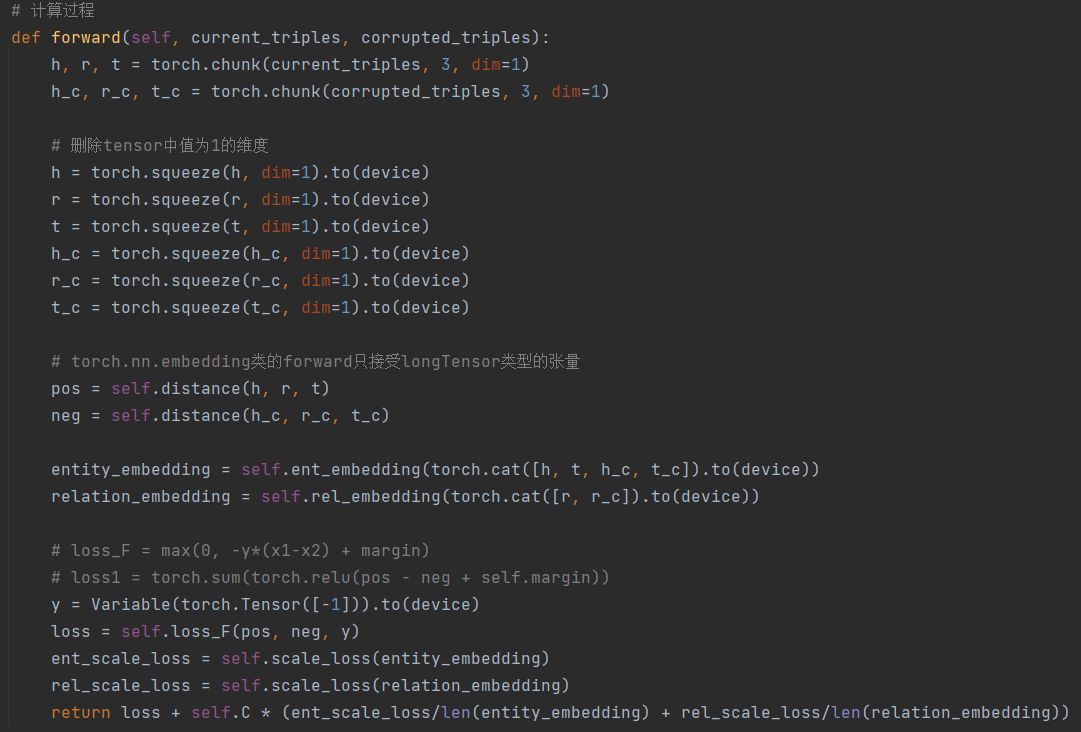
步骤2.3 定义目标函数



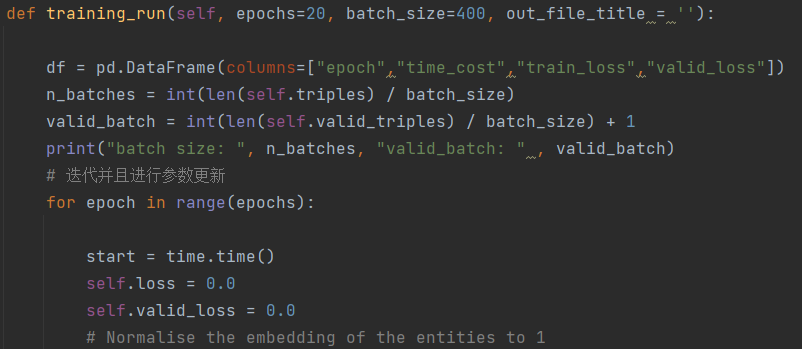
步骤2.4 构造负样本三元组

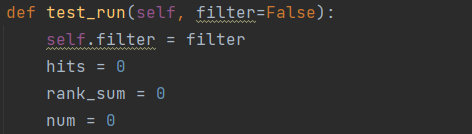


步骤2.5 定义模型的计算过程



### 3.4 实验任务三 训练及测试





## 四、实验环境

以下仅供参考：

操作系统：windows10

平台软件：Anaconda、torch、Python

## 五、界面设计

