**Task2项目第三次报告**

1. **整个项目实验完成的主要工作**

**1. 仔细研读论文，深入理解DeepProblog涉及的相关知识整理每个部分的相关思路。**

**2. 理解DeepProblog的相关代码和实现逻辑。**

**3. 根据论文给出的代码进行DeepProblog模型训练。**

**4. 运行代码中给出的CoinUrn、Forth和MNIST三个示例，得到结果并进行分析。**

**5. 根据MNIST示例改写数字图像加法为减法。**

1. **项目完成内容**

**第一部分**

**论文的主要内容：**

**1. 论文提出了新的概率逻辑编程框架：DeepProbLog**

**DeepProbLog是一个集成了通用神经网络和表现性概率逻辑的建模推理的框架，能够通过示例进行端到端的训练学习。**

**DeepProbLog保留了ProbLog的语义、推理机制和实现，核心的改进是基于模型的实例训练。**

**DeepProbLog使用神经网络，采用损失梯度反向传播的方法进行训练。输入数据为通过模型得到的特征向量和概率事实，原子表达式上的输出为概率，总输出封装为神经谓词。**

**2. 论文介绍了逻辑编程概念**

**Atoms：形如q(t1, ..., tn)的表达式（q是谓词，ti是条件，t可以为常量、变量或函数）**

**literal：q(t1, ..., tn)或¬q(t1, ..., tn)。**

**ground：不含有任何变量的表达式。**

**rule：形如h :— b1, ..., bn的表达式，其中h为atom，bi为literals。rule的含义为：每当 bi 的连词成立时，h 就成立。当n=0时，rule被称为facts.**

**3. 论文给出了DeepProbLog的推理机制**

**第一步根据查询的逻辑程序，生成查询所依赖的程序中子句的所有基本实例。**

**第二步将ground logic program改写成命题逻辑中的公式，根据概率事实的真值定义查询的真值。此过程中遇到神经谓词时，都会执行对神经网络组件的正向传递。当这种情况发生时，所需的输入（例如图像）会被输入到神经网络中，然后将Softmax 输出层的结果作ground AD的概率。**

**第三步将逻辑公式编译成决策图(SDD)。**

**自下而上评估SDD，计算给定查询的成功概率。**

**第二部分：**

**在学习完论文之后，我通过源码查看DeepProblog的设计思路和逻辑：**

**1. DeepProblog代码主要包括逻辑程序构建，模型构建，神经网络构建，梯度计算，模型优化处理和模型训练这几个部分。**

1. **DeepProblog的过程为：首先构建DeepProblog模型、逻辑程序和神经网络模型，之后进行半环梯度计算，返回SDD的查询概率。最后在模型训练的过程中使用优化器对DeepProblog模型进行优化。**

**第三部分**

**根据MNIST示例改写数字图像加法为减法**

**在MNIST实验中，DeepProbLog实现了个位数图像加法和多位数图像加法，我们在运行并分析过该实验后，想仿造该示例，将加法改写为减法。**

**个位数图像减法**

**思路过程：**

**1. 首先分析个位数图像加法的处理过程：加载训练测试数据，获取编写好的逻辑表达式，接着构建神经网络、模型和优化器，最后训练测试。**

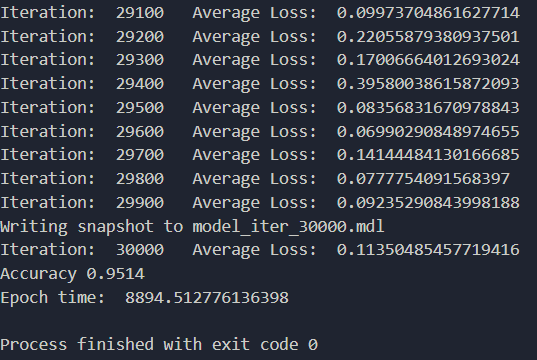
**2. 想要将加法改写为减法，最重要的是修改逻辑表达式，之后用修改后的表达式来构建模型。**

**改写后的逻辑表达式：nn(mnist\_net,[X],Y,[0,1,2,3,4,5,6,7,8,9]) :: digit(X,Y).**

**subtract(X,Y,Z) :- digit(X,X2), digit(Y,Y2), Z is X2-Y2.**

**3. 修改generate\_data.py文件中的generate\_examples函数用于处理数据集，使得其适用于减法模型。**

**个位数图像减法模型训练测试效果：**



**准确率达到95.14%**

**多位数图像减法**

**思路过程：**

**1. 与个位数减法相同，最重要的是修改逻辑表达式。**

**改写后的逻辑表达式：**

**nn(mnist\_net,[X],Y,[0,1,2,3,4,5,6,7,8,9]) :: digit(X,Y).**

**//多张图像转为对应的多位数**

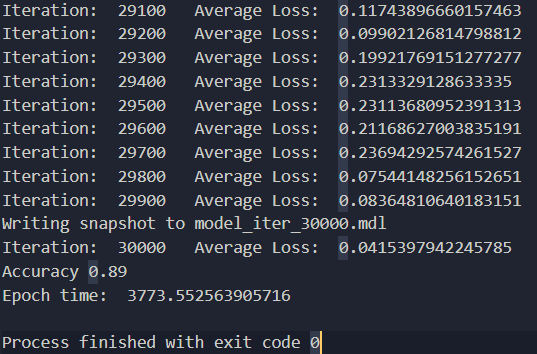
**number([],Result,Result).**

**number([H|T],Acc,Result) :- digit(H,Nr), Acc2 is Nr+10\*Acc,number(T,Acc2,Result).**

**number(X,Y) :- number(X,0,Y).**

**subtract(X,Y,Z) :- number(X,X2), number(Y,Y2), Z is X2-Y2.**

**多位数图像减法模型训练测试效果：**



**准确率达到89%**

1. **总结**

**在本学期的项目实验过程中，我们认真学习了解了论文中提出的DeepProbLog工具，并通过实验深入了解了DeepProbLog在深度学习中解决逻辑编程的高效性，并自己尝试使用DeepProbLog解决了一些简单的逻辑编程问题。**