**北京航空航天大学计算机学院**

**硕士学位论文中期检查报告**

**论文题目**：面向程序设计语言课程

的个性化学习效果

建模研究

**专 业**：软件工程

**研究方向**：软件工程

**研 究 生**：王珊珊

**学 号**：SY1406112

**指导教师**：吴文峻

**北京航空航天大学计算机学院**

2016年 8 月30 日

目 录

1 论文工作计划 1

1.1 论文研究目标 1

1.2 论文主要研究内容 2

1.3 论文预期成果形式 4

1.4 论文研究进度 4

2 已经完成的工作 5

2.1 收集数据 5

2.2 利用IRT建立学习效果模型 5

2.3 建立个性化知识跟踪模型 9

2.4 程序错误定位模型 12

2.5 建立Topical HMM模型（模型设计） 15

3 关键技术或难点 18

3.1 建立概率图模型 18

3.2 提高模型预测准确率 18

3.3 模型的推理与学习 19

4 下一阶段工作计划 19

4.1 尚未完成的工作 19

4.2 解决问题的技术思路或措施 20

4.3 下一阶段计划 20

5 主要参考文献 21

**面向程序设计语言课程的个性化学习效果建模研究**

# 论文工作计划

## 论文研究目标

近年来在信息领域出现的先进计算技术和互联网络技术在教育信息化领域得到广泛应用，极大促进了新型大规模在线教育以及基于互联网学习模式的变革与发展。特别是近年来慕课（MOOC）教育的出现，使得大规模在线教育成为可能，并为推动高等教育改革、提高高等教育规模和受益面提供了全新思路。社交媒体技术和慕课（MOOC）教育正在深刻地改变教育的模式[1]，虽然大规模在线教育的规模化成效明显，但在如何兼顾海量学习主体的个性化学习[2]以及加强其实践能力培养方面面临着巨大的挑战。

在对学习者的个性感知和学习模式的研究方面，无论是经典的教育数据分析，还是目前正在开展中的慕课（MOOC）大数据分析[3]，都尚未把学习者的认知轨迹、教学内容的知识体系和软件项目实践活动进行时空维度的关联和融合。所以，我们需要突破目前慕课（MOOC）教学分析[4]的局限，拓展对学生学习行为观察的视野，从学生学习的完整轨迹入手来分析其学习与实践过程的起伏和变化。将学生的知识点掌握能力和实践能力进行比较分析，研究学生是否存在已经掌握知识点但却在实践方面运用能力不足的现象，最后建立学生学习程序设计语言的知识体系，并实现个性化学习效果的展示。基于以上情况，本文提出以下三点研究目标：

一、基于学生在学习程序设计语言课程上的选择题和填空题的答题表现数据，利用IRT建立学习效果模型和个性化知识跟踪模型，最终判定学生学习程序设计语言中对各个知识点的掌握水平。

二、基于学生在程序设计语言课程中的编程题的答题表现数据（即学生答题提交的代码），首先利用Tarantula、Ochiai算法结合“人工标注”做程序错误定位工作，然后对学生在知识点实践运用能力方面进行初步预测分析，然后结合Topical HMM模型[5]进行判定学生对各个知识点的实践能力水平。

三、将学生对程序设计语言课程中的知识点掌握能力水平模型和实践能力水平模型进行分析，并研究是否存在学生已经掌握知识点但在实践能力依然不足的现象，最后建立学生个性化知识体系，将结果进行可视化展示。

## 论文主要研究内容

本课题主要针对现有的学生学习效果研究中只考虑学生在知识点掌握能力方向的研究，几乎很少对与学生在运用知识点即实践能力方面进行研究，本文从知识点掌握能力和实践能力的双向建模分析，建立起最全面的个性化学习效果分析模型，提高个性化学习效果模型预测学生表现的准确率，及时发现学生在实践能力方面的不足并告知其加强实践。基于此目标，论文从以下三个方面展开研究：（系统总体设计如图1所示。）

1. 研究学生对程序设计语言各知识点的掌握能力水平

学生的知识点学习效果评判一直就是教育领域中一个非常重要的环节，准确评判学生的学习效果，不仅仅能够给学生更加公正的课程得分，而且还可以帮助教师找到知识难点和疑点，改善教学方式，甚至还可以针对不同的学生提供个性化的教学和练习。我们可以利用心理学统计性模型IRT理论和传统的智能教学系统（ITS）领域已经研究成熟的知识跟踪模型来对学生学习效果进行建模。本文的研究内容之一就是基于学生的答题表现数据结合IRT模型和对传统的知识跟踪模型进行改进，实现个性化知识跟踪模型，提高模型预测学生知识掌握能力水平的准确度。

1. 研究学生对程序设计语言在实践能力方面的水平

目前的学生学习效果模型中较少涉及在实践能力方面的研究，一般情况都是研究学生在答选择题和填空题方面的数据而建立起的学习效果模型，这只能体现学生在对知识点认知方面的能力掌握情况，即便已经熟知程序设计语言知识点，学生却不一定能将知识点实践运用在代码编写上，如何建立模型表示学生的实践

**图1 系统总体设计图**

能力水平，并且实践能力分析过程中，一道程序设计作业包含不止一个知识点，是我的第二个研究内容。首先利用gcov工具编写Tarantula、Ochiai算法结合“人工标注”做程序错误定位分析工作，让我们对学生在实践过程中的错误范围以及个人实践水平有一个初步分析，然后利用概率图模型知识以及吉布斯采样参数学习方法建立并学习Topical HMM模型进行判定学生对各个知识点的实践能力水平。

3.对知识点掌握能力模型和实践能力模型综合分析以及学生个性化知识体系的结果可视化

两个方向性的模型综合分析以及可视化展示是本文的最后一步，是对学生的个性化知识体系的一个结果性分析，及时反馈学生在实践方面的不足，促进其全面掌握知识。因此第三个研究内容是，将知识点掌握模型和实践能力做相关分析和配对t检验，去判断知识点掌握模型和实践能力模型的是否相关并且存在差异，如果实践水平不如知识点掌握水平则可以指示学生应该提高该知识点的实践能力应该多多训练这方面的编程题。可视化方面分布展示每个学生对程序设计语言课程中涉及的知识点体系的掌握水平和实践水平，并包括他们的知识水平随着学习时间和训练的增加所发生的变化时序图。

## 论文预期成果形式

论文最终的成果包含：

1.面向程序设计语言的个性化学习效果建模研究的小论文

2.介绍整个研究过程及模型设计和实现的毕业论文。

## 论文研究进度

|  |  |
| --- | --- |
| **时间** | **工作计划** |
| 2015.12 | 查阅资料，进一步了解课题关键技术的研究工作 |
| 2016.01―2016.04 | 依据学生的作业选择题和填空题分析情况，学习IRT模型，贝叶斯模型，知识跟踪模型以及学习模型参数的方法如吉布斯采样、最大后验概率估计等；并且提出新型个性化知识追踪模型。 |
| 2016.05—2016.06 | 依据学生提交的程序代码作业情况，学习使用错误定位方法工具（Tarantula、Ochiai算法结合“人工标注”思想）定位程序错误位置。设计Topical HMM模型来判定学生的实践能力水平。 |
| 2016.07—2016.08 | 收集实验数据，实现学生知识点掌握能力模型以及完成错误定位工作。准备中期答辩。 |
| 2015.09—2015.10 | 继续分析实验数据，验证模型有效性，并继续优化模型。设计可视化工作。 |
| 2015.10—2015.12 | 资料相关整理和撰写论文，准备答辩。 |

# 已经完成的工作

## 收集数据

程序设计语言（C语言）是所有学习计算机的学生最重要最基础的一门课，目前我校的学生都是在 程序设计类课程教学辅助系统上进行课后作业练习和考试答题。本文选取的是北航高等工程学院15级本科生的第一学期的C语言程序设计课程。我们利用爬虫工具收集学生有关于填空题和选择题的作业（后续统一成为普通作业）情况以及其编程题提交的程序（后统一称程序作业）数据集，收集到主要数据及其规模如表1所示。

**表1 数据种类及其规模**

|  |  |  |  |
| --- | --- | --- | --- |
| 数据种类 | 数据量 | 普通作业 | 程序作业 |
| 总人数 | 213人 |  |  |
| 作业题目总数 | 214道 | 156道 | 58道 |
| 提交作业总数 | 38104 | 26818 | 11286 |

人工专家统计知识点，将所有选择题和填空题这种普通作业进行人工知识点判定，由于题型并不复杂，因此一般一道普通作业只对应一个知识点，统计结果如图2所示。其中函数和指针这两个知识点占课程的比重最大，无论是教学还是学生学习需要在这两个知识点进行加强学习训练。

## 利用IRT建立学习效果模型

IRT理论（Item Response Theory）是一系列心理统计学模型的总称。IRT是用来分析考试成绩或者问卷调查数据的数学模型。这些模型的目标是来确定的潜在心理特征(latent trait）是否可以通过测试题被反应出来，以及测试题和被测试

**图2 普通作业知识点分布图**

者之间的互动关系，目前广泛应用在心理和教育测量领域。这里我们利用IRT模型训练得出学生的心理特征也即是学生对知识点的掌握水平和题目难度等试题特质。实现个性化判定学生的知识掌握程度，同时获得问题的难度水平。

1. 模型建立

IRT模型[6]本质是一个Logistic回归模型。在1PL（即是1参数logistic模型）模型中，项目反应（选择题和填空题的答题结果）的格式要么为正确要么不正确，用0或者1表示结果。假定每一道问题的区别只有区分度（一道题将把不同水平的人区分开来，也即题目的鉴别力）和难度水平。学生j答对问题i的概率如公式3所示，设学生总数为N，问题总数为M。

其中a表示问题的区分度，在1PL模型中不考虑区分度a的值对所有问题相等，表示问题i的难度，表示学生j的知识掌握水平。2参数logistic模型（简称2PL）与1PL模型相比，增加了问题i的区分度，于是学生j答对问题i的概率如公式4所示：

1. 参数学习

对IRT模型进行参数学习[11]，类比于对logistic回归模型学习，可以使用到的参数学习方法如最大似然估计、最大后验概率估计、马尔可夫链蒙特卡洛方法（MCMC）等方法。由于前两种方法对初始值较敏感，因此我们采用MCMC采样方法中的 Metropolis-Hastings 算法（表2）来进行IRT模型参数估计以期降低模型对初始值的敏感程度。

指定如下的独立的先验分布：

(5)

其中表示(log(x)-log(u))/为标准正态分布。

**表2 Metropolis-Hastings采样算法**

|  |
| --- |
| **算法：Metropolis-Hastings采样算法** |
| 1：初始化马氏链初始状态  2：对k=1,2,…,T循环以下过程进行采样  首先，从p()抽取 ()   1. 对各j=1,2,…,N独立抽取 2. 从均匀分布采样u Uniform[0,1] 3. 计算拒绝率R()=min{1,}，如果u < R() ，则接受转移，否则不接受转移   再从p() 抽取  (1)对各i=1,2,…,M独立抽取  (2)均匀分布采样u Uniform[0,1]  (3)计算拒绝R()=min{1,}，如果u< R()，则接受转移，否则  3：返回样本，则 |

1. 结果展示

将全部普通作业数据按3：1：1的比例随机划分成训练集、验证集和测试集。在训练集上训练模型参数，并采用交叉验证的方式确定最优参数，然后用测试集计算各项指标。最终得到的模型参数和各项指标如图3下：根据以下图形可以得出，2PL模型要比1PL模型模拟的效果更好。因此我们可以使用2PL模型中学生知识点掌握水平作为对学生知识点掌握能力的一个判定。

**图3 IRT模型测试结果**

**表3 1PL和2PL模型学习效果比较**

|  |  |  |
| --- | --- | --- |
| **评价指标** | **1PL** | **2PL** |
| Accuracy | 0.758 | 0.763 |
| Recall | 0.832 | 0.793 |
| Precision | 0.513 | 0.598 |
| AUC | 0.769 | 0.783 |

## 建立个性化知识跟踪模型

学生的答题数据则直接反映了学习效果，通过答题数据，可以准确判定学生的知识掌握水平。最简单也是最直接的方法是使用答题正确率来作为学习效果的评定标准，而答题正确率没有利用到学生在答题过程中的序列信息，例如某一章共有4道练习题，学生甲的答题序列为（1，1，0，0），表示前两道题回答正确，后两道题回答错误，学生乙的答题序列为（0，0，1，1），此时两名学生的答题正确率同为50%，但是其学习效果却不尽相同，尤其在MOOC环境下，学生回答错误时会得到相应的提示信息，帮助学生在练习过程中再次学习知识，所以虽然甲乙两名学生都回答正确了两道题，但是学生乙更像是通过练习逐渐掌握了知识，其学习效果应更好，而学生甲可能由于某些原因猜对了前两题，其学习效果应不如学生乙好。由此定性分析可见，答题的序列信息，对学习效果的评判有一定影响。而源于传统智能教学系统（ITS）的知识跟踪模型正是利用到了答题过程中的序列信息，来对学生的知识掌握水平进行准确评判。 因此我们需要建立起知识跟踪模型来判定学生的知识掌握程度，由于传统知识跟踪模型没有考虑到题目的难度差异与学生个性化问题，基于以上我们对传统知识跟踪模型做出改进，建立个性化知识跟踪模型。

1. 模型建立

传统知识跟踪模型（以下简称KT模型）基于学生在练习过程中的答题表现，跟踪其知识掌握情况：每次学生答题后，KT模型[7]会根据回答的正误来更新学生对相应知识的掌握水平。传统知识跟踪模型可以用2-状态的动态贝叶斯网络，也即隐马尔科夫模型来表达，如图4所示。学生在学习过程中的知识掌握情况是随着答题序列的变化而变化的。

具体来说，KT模型以知识点为实际的建模对象，每个知识点包含四个参数，分别为两个学习参数和两个表现参数。学习参数分别为和，为知识初始掌握概率：在答题之前，相应知识点已被掌握的概率（比如通过阅读学习材料等）；为知识转化概率：根据每次答题后的系统反馈，知识点由未掌握转为已掌握的概率，并且KT模型假设知识没有遗忘，即知识不能由已掌握状态转为未掌握状态；表现参数分别为和，为猜对概率：在知识点未掌握的情况下，学生仍然能回答正确的概率；为犯错概率：在知识点已掌握的情况下，学生不小心回答错误的概率。



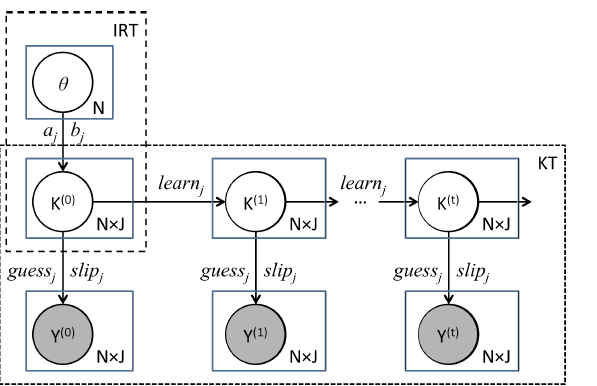
**图4 传统知识跟踪模型**

传统KT模型中的四个参数都是针对特定知识点的，即对于同一个知识点，所有学生都共享相同的初始掌握概率、学习概率、猜对概率和答错概率，尽管事实上并非如此，比如不同的学生初始知识掌握概率不同、细心程度不同或者各问题难度不同导致答错概率不同等情况。因此我们可以在传统KT模型中介入学生和问题的个性化因素，可以有效的提升预测准确性。所以本文基于传统KT模型和IRT模型 [8,10]，设计个性化知识追踪模型，模型如图5所示。

此个性化学习知识追踪模型使用IRT模型基于知识点的难度去估计学生的初始知识点掌握概率，然后再利用BKT去进行模型训练。

1. 模型学习

(1)利用大量的学生答题序列，训练模型参数。KT模型学习问题属于贝叶斯网络中结构已知、数据不完整的参数学习，本文采用期望最大化（Expectation Maximization, EM）算法进行模型训练。

(2)利用已训练好的模型推测学生对知识点的掌握情况。p(L0)表示学生对知识点掌握的初始情况，而后基于学生答题正误的序列，利用贝叶斯公式迭代地更新其对知识点的掌握情况。知识点掌握情况迭代更新的伪代码表4所示：

**图5 个性化知识跟踪模型**

**表4 知识掌握情况迭代算法**

|  |
| --- |
| **算法：知识掌握情况迭代算法** |
| 1：  2：  3：  4:  5: |

1. 结果展示

本文共选取了实验课程中的14个知识点，平均每个知识点对应11道习题；平均每个知识点有213名同学完成。利用每个知识点的前n-1道习题作为训练数据，分别训练传统的知识跟踪模型和改进的知识跟踪模型。利用最后一道题作为测试数据，检测模型的准确性，再与之前的2PL模型进行比较，得到的结果如图5所示，结合IRT模型的个性化知识跟踪模型估计效果最好，因此模型计算得出学生对知识点的掌握状态是我们要求的最终结果。

## 程序错误定位模型

程序错位定位是在我们做程序作业分析中一个初步的方法，一般程序发生编译错误，编译器可以很容易检测出来，但是当程序编译通过输出结果错误的时候就需要借助程序错误定位模型[9]可以通过错误定位方法对学生的程序作业找出行集合，其中TF={T1,…,Ts}表示测试运行结果错误的测试用例集

**图5 知识跟踪模型测试结果**

合共有s个，TP={Ts+1,…,Tm}表示测试执行结果正确的测试用例集合共有m-s个，则程序谱是一个二维矩阵，矩阵元素为：

(1)

一个形象化的基于程序谱的错误定位模型见表5，其中行s1,…,sn 表示程序中对应的语句，列T1,…,Tm 表示m个测试执行用例。

**表5 基于程序谱的错误定位模型**

|  |  |  |  |
| --- | --- | --- | --- |
| Program | Test cases | | Suspiciousness |
| Failed | Passed |
| T1 ... Ts | Ts+1 ... Tm |
| S1 | b1,1 ... b1,s | b1,s+1 ... b1,m | p1 |
| S2 | b2,1 ... b2,s | b2,s+1 ... b2,m | p2 |
| ... | ... ... ... | ... ... ... | ... |
| Sn | bn,1 ... bn,s | bn,s+1 ... bn,m | pn |

基于程序谱的错误定位模型主要通过分析成功测试用例运行和失败测试用例运行，以及二者之间在某种程序信息上的差异，并通过某种映射机制最终找到差异所在的最低抽象层次位置，这种最低抽象层次的差异就是缺陷所在的程序位置。程序位置粒度有大有小，典型的有语句、基本块、函 数、分支、路径等，本文的运用的程序位置粒度为语句。SFL方法使用不同的度量准则来评估一个程序位置包含错误的可能性大小，并且按包含错误的可能性大小对程序位置进行可疑度的排序，可疑性值越高就代表程序的该位置有缺陷的可能性越高。因此，度量方法的设计是SFL方法的关键问题之一。目前SFL使用多种不同的方法来计算程序单元的可疑度，典型的SFL方法有Tarantula和Ochiai等，表6显示语句的可疑度计算方法。（其中failed）

**表6 错误定位算法公式**

|  |  |
| --- | --- |
| 方法名 | 语句可疑度计算公式 |
| Tarantula() |  |
| Ochiai() |  |

程序错误定位模型实现概述如图6所示，首先我们使用KLEE工具通过符号执行产生一组测试用例，提供一个程序的正确实现产生的输出和错误版本（错误的程序作业）产生的输出进行比较，确定运行结果正确和错误的测试用例集合，在测试中使用gcov工具得到spectrum文件（程序执行的路径覆盖），最后利用怀疑度计算公式求出每一语句的怀疑度，按从大到小的顺序排序，排在前面的即是错误可能性最大的语句。

本文共有46道程序作业包含至少一个错误版本，平均每一道程序作业有15个错误版本，将所有程序作业分为1:2，其中前一部分程序作业使用“人工辅助”分派给线上多人不采用错误定位模型而使用人工调试定位错误，并将这个定位作为错误标准来检验错误定位模型的准确率，因此这一部分程序作业接着利用错误定位模型使用Tarantula和Ochiai算法，可以用定位代价来评估错误定位的准确度，定义为：使用选定的错误定位技术定位该错误时，错误存在于该错误定位技术输出可疑语句列表中的位置，n为程序错误总数，定位代价即是所有错误的定位代价的平均值，用如下公式进行计算：

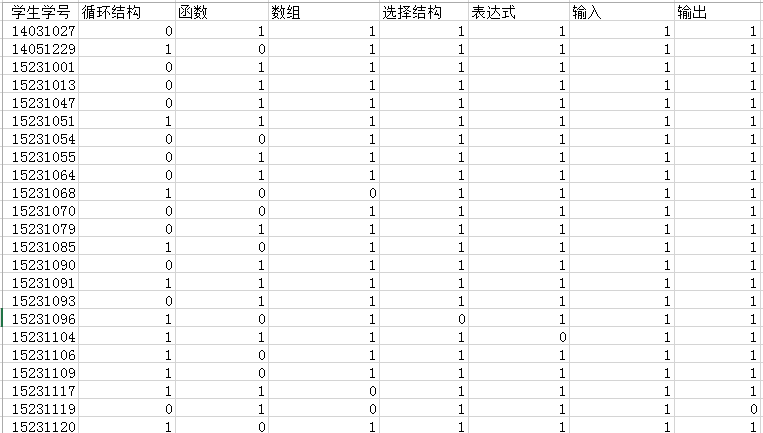


**图6 程序错误定位模型实现概述**

经过计算，使用Tarantula算法的错误定位代价为11.5，使用Ochiai算法的错误定位代价为11.3。在学生代码超过一百个语句的情况下，检测语句不超过10%，可以判定错误定位模型的有效性，因此采用Ochiai算法来进行错误定位，平均错误定位代价也就表示平均检查11.3个语句既可以实现定位该错误，因此剩下的程序作业就可以使用错误定位模型结合“人工标注”来定位，这样可以快速定位到错误的那个语句位置，然后人工标注错误是属于哪一个知识点，形如函数、数组等知识点。从而得出每一个学生对每一个知识点在实践能力水平的一个初步分析。如图7所示，可以得出学生对某一道作业的初始实践水平。

## 建立Topical HMM模型（模型设计）

对于判定学生对知识点的实践能力水平，由于一个问题不可能只与一个知识点有关，因此我们设计了Topical HMM模型（局部隐马尔科夫模型）。Topical HMM模型是基于一个主题模型（Latent Dirichlet Allocation），这个主题模型是将一个文档编码成一系列主题表示，而Topical HMM模型是将一个题目编码成一系列知识点表示，最后根据吉布斯采样推理求得学生对每个知识点的实践能力情况，Topical HMM概率图模型如图8表示。

**图7 学生初始实践水平**

1. 模型说明
2. 超参数

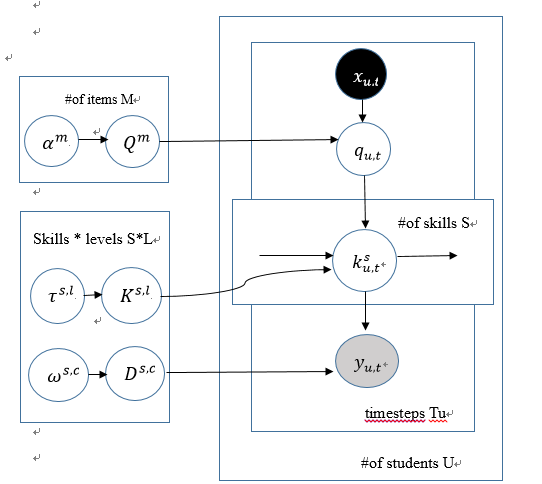
S表示模型中知识点的数目，U表示学生总数，Tu表示学生u练习的步数，M是问题的总数，L是知识点实践水平（本文L=2，要么掌握要么不掌握）

1. 离散变量

是学生u在时刻t练习的问题的编号，是必需的知识点，是学生u在时刻t有关于知识点s实践能力水平，表示答题表现。

1. 随机变量

是一个多项分布，从中选择需要的知识点类型，是当知识点为s，实践水平是l的转移参数，是发射矩阵。



**图8 Topical HMM 概率图模型（白色背景为隐藏变量，黑色为观察变量，灰色表示只在训练阶段为可观察）**

1. 模型生成算法

为设计方便，我们设定,为狄利克雷分布的参数，由于狄利克雷分布和多项式分布共轭，则意味后验分布也同为狄利克雷分布，因此便利了计算。模型生成算法如表7所示。

**表7 Topical HMM生成算法**

|  |
| --- |
| **算法：Topical HMM生成算法** |
| Function Topical HMM({…:u}),S,L,M  //Draw parameters from priors:  for each skill s from 1 to S do  for each knowledge state l from 1 to L do  Draw parameter Dirichlet()  Draw parameter Dirichlet()  for each item m from 1 to M do  Draw ~ Dirichlet()  // Draw variables from parameters:  for each student u from 1 to U do  for each timestep t from 1 to Tu do  Draw skill ~Multinomial()  for s from 0 to S do  if s= then  // knowledge state could change:  // previous knowledge  Draw ~Multinomial()  Else  // knowledge state can’t change:    // current skill  // current knowledge state  Draw performance Multinomial() |

# 关键技术或难点

## 建立概率图模型

知识跟踪模型和Topical HMM模型都利用到了概率图模型：知识跟踪模型用隐马尔科夫模型来表达，其中隐含节点表示学生对知识的掌握情况，观察节点表示学生在练习过程中答题的正误；Topical HMM模型是基于一个主题模型（Latent Dirichlet Allocation），这个主题模型是将一个文档编码成一系列主题表示，而Topical HMM模型是将一个题目编码成一系列知识点表示，最后根据吉布斯采样推理求得学生对每个知识点的实践能力掌握情况。因此，正确理解建立并推理概率图模型是本课题的一个关键技术。

## 提高模型预测准确率

如何提高IRT模型和知识追踪模型对学生知识点的掌握水平的预测能力是一个关键技术，基础知识跟踪模型在训练过程中没有考虑到个人的水平差异和题目的难度区分，因此加入个人差异和题目差异可以提高知识追踪模型的正确率。由于编程题不可能只含有一个知识点，因此Topical HMM模型可以同时训练判定学生两个以上知识点的实践能力水平，比起一次模型（HMM模型）训练只能判定一个知识点的实践能力水平提高了预测准确率。

## 模型的推理与学习

建立概率图模型只是其中一个技术点，但是后续最重要的过程就是概率图模型的推理和学习，在知识跟踪模型中推理过程运用到的是联合树算法，学习过程中由于KT模型属于贝叶斯网络中结构已知、数据不完整的参数学习，本文采用期望最大化（Expectation Maximization, EM）算法进行模型训练。Topical HMM模型由于联合分布难以计算因此采用Gibbs Sampling算法来学习参数，这些概率图模型的学习过程是本文一个非常关键的技术，直接影响模型的应用效果。

# 下一阶段工作计划

## 尚未完成的工作

1. 实现Topical HMM模型

由于编程题不可能只含有一个知识点，因此Topical HMM模型可以同时训练判定学生两个以上知识点的实践能力水平，比起一次模型（HMM模型）训练只能判定一个知识点的实践能力水平提高了预测准确率。因此，建立概率图模型Topical HMM 模型以及利用Gibbs Sampling算法学习模型参数并最终计算学生对知识点的实践能力水平是我们下一段的重要工作。

1. 双向模型相关性分析和可视化工作

得到学生对知识点的掌握能力水平和实践能力水平模型后，我们需要将这两个模型相关分析和配对t检验，去判断知识点掌握模型和实践能力模型的是否相关并且存在差异，去判断学生是否已经掌握了该知识点却在编程运用方面依然存在不足。接下来完成最后的可视化方面的工作，展示每个学生对程序设计语言课程中涉及的知识点体系的掌握水平和实践水平，并包括他们的知识水平随着学习时间和训练的增加所发生的变化时序图。

1. 学位论文的撰写

待完成所有工作后会整理资料，将完成的成果整理为文字提交，作为本人的毕业设计。

## 解决问题的技术思路或措施

1. Topical HMM模型

Topical HMM模型本质类比于一个研究方面相当成熟的模型LDA（隐含狄利克雷分布），以多项式分布编码一个题目对应于一系列知识点，题目所对应知识点的隐藏变量为先验分布为狄利克雷分布的多项分布（知识点分布）中抽取的一个知识点。再对Topical HMM进行参数估计，采用一种更优的方法即：吉布斯采样，当联合分布很难计算出来的时候，Gibbs Sampling利用条件概率产生符合分布的样本，用于估计分布的期望。

1. 双向模型相关性分析和可视化工作

对学生对知识点的掌握能力水平和实践能力水平模型做相关性分析，如用Pearson相关系数或者Spearman相关系数进行线性相关分析；距离分析两个模型之间的相似度。同时做配对t检验，检验两个模型的平均数是否存在显著性差异。可视化工作中利用echarts工具中的API结合搭建Django中的MVC模式实现动态显示学习效果数据。

## 下一阶段计划

1．2016.8-2016.9：完成Topical HMM分析学生的实践能力水平。

2．2016.9-2016.10：完善模型分析和实验验证

3．2016.10-2016.12：实现可视化，撰写学位论文

# 主要参考文献

1. Johnson L, Adams Becker S, Estrada V, et al. NMC horizon report: 2014 K[M]. 2014.
2. Liu J, Wong C K, Hui K K. An adaptive user interface based on personalized learning[J]. IEEE Intelligent Systems, 2003 (2): 52-57.
3. Gasevic D, Kovanovic V, Joksimovic S, et al. Where is research on massive open online courses headed? A data analysis of the MOOC Research Initiative[J]. The International Review Of Research In Open And Distributed Learning, 2014, 15(5).
4. Guo P J, Kim J, Rubin R. How video production affects student engagement: An empirical study of mooc videos[C]. Proceedings of the first ACM conference on Learning@ scale conference. ACM, 2014: 41-50.
5. Gonzalez-Brenes, J., & Mostow, J. (2013, July). What and when do students learn? Fully data-driven joint estimation of cognitive and student models[C]. In *Educational Data Mining 2013*.
6. Hirose, H., & Sakumura, T. (2012, August). Item response prediction for incomplete response matrix using the EM-type item response theory with application to adaptive online ability evaluation system[C]. In *Teaching, Assessment and Learning for Engineering (TALE), 2012 IEEE International Conference on* (pp. T1A-6). IEEE.
7. Corbett A T, Anderson J R. Knowledge tracing: Modeling the acquisition of procedural knowledge[J]. User modeling and user-adapted interaction, 1994, 4(4): 253-278.
8. Pardos Z A, Heffernan N T. KT-IDEM: Introducing item difficulty to the knowledge tracing model[C]. International Conference on User Modeling, Adaptation, and Personalization. Springer Berlin Heidelberg, 2011: 243-254.
9. Gong L. Auto-Grading Dynamic Programming Language Assignments[J].
10. Yudelson M V. Individualizing Bayesian Knowledge Tracing. Are Skill Parameters More Important Than Student Parameters?[J].
11. 涂冬波, 漆书青, 蔡艳, 等. IRT 模型参数估计的新方法——MCMC 算法[J]. 心理科学, 2008, 31(1): 177-180.