硕 士 学 位 论 文

基于习题表征和学生能力表征的学生知识追踪算法研究

Research on Knowledge Tracing Algorithm Based on Exercise Representation and Students' Ability Representation

作者：王珍珠

导师：陈一帅

北京交通大学

2021 年 5 月

学位论文版权使用授权书

本学位论文作者完全了解北京交通大学有关保留、使用学位论文的规定。特授权北京交通大学可以将学位论文的全部或部分内容编入有关数据库进行检索，提供阅览服务，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。同意学校向国家有关部门或机构送交论文的复印件和磁盘。学校可以为存在馆际合作关系的兄弟高校用户提供文献传递服务和交换服务。

（保密的学位论文在解密后适用本授权说明）

学位论文作者签名：　　导师签名：

签字日期：　　年　　月　　日　　签字日期：　　年　　月　　日

学校代码：10004 密级：公开

北京交通大学

硕士学位论文

基于习题表征和学生能力表征的学生知识追踪算法研究

Research on Knowledge Tracing Algorithm Based on Exercise Representation and Students' Ability Representation

作者姓名：王珍珠 学　　号：18120144

导师姓名：陈一帅 职　　称：副教授

学位类别：工学 学位级别：硕士

学科专业：通信与信息系统 研究方向：信息网络

北京交通大学

2021 年 5 月

致谢

本论文的研究工作是在我的导师陈一帅老师的悉心指导下完成的。感谢陈一帅老师在我读研期间帮助我面对科研和学习上的困难，对我学术上和学习上一直耐心的指导和鼓励。精益求精的学术风范、恪尽职守的工作作风，也深深地激励着我不断进取。陈一帅老师对待学术的热情和对待生活的态度都让我颇受感染。在此由衷感谢陈老师三年以来对我的关心和指导。除陈一帅老师之外，衷心感谢郭宇春老师、赵永祥老师和实验室里的所有老师在我研究生学习阶段对我的无私帮助和关怀使我能顺利完成研究生阶段的学习。

另外，在实验室工作和撰写论文期间，感谢唐伟康师兄，艾方哲师兄，苏建师兄和冯梦菲师姐等师兄师姐给予的热情帮助。也感谢戚余航，孙欢，曹中，李想等同学在学习和生活中的帮助和陪伴。

最后，特别感谢一直给予我无尽的付出和支持的家人，感谢他们在我困难时给予我精神上的鼓励，在我生病时给予我生活上无微不至的照顾，在我想要退缩时给我坚持下去的勇气，正是他们多年来默默关心和奉献，才使得我不断克服困难，顺利完成学业，成为有用之人。

摘要

随着互联网的发展，智能教育成为教育发展的迫切需求。学生知识追踪模型能够根据学生的历史学习记录获得习题表征和学生能力表征，追踪学生对知识的掌握情况。智能教育需要准确的学生知识追踪模型。

目前，制约学生知识追踪模型效果的两个原因是：(1)习题表征不准确。以知识点为粒度的习题表征难以刻画习题的精确信息；以习题为粒度的表征，由于数据稀疏导致模型参数学习不准确；已有的综合知识点和习题的混合模型采用加性模型进行习题表征，不符合实际数据的分布。(2)学生能力表征不准确。传统模型基于知识点进行学生能力表征，忽略了同一知识点下不同难度习题对学生能力的提升不同，而深度模型在对学生学习能力进行度量时，没有考虑习题难度，并且通过累积学生记录进行学习能力计算的方法不能准确捕捉学生即时的能力变化，也没有考虑学生的长期能力特征。

为了解决上述问题，本文基于真实智能教育系统的学生练习数据集，细致测量与分析了学生做题行为和习题难度特征，针对传统模型，提出能够准确进行习题表征和学生能力表征的改进模型，针对深度模型，提出长期和动态相结合的个性化深度知识追踪模型。具体贡献如下：

(1) 对于传统学生知识追踪模型，本文首先基于真实学生做题数据分析结果，设计了综合考虑习题和知识点的习题难度乘性模型，以改进习题表征。实验表明：该方法将目前主流的知识追踪模型 DAS3H、AFM 和 PFA 在 Assistment12 数据集上的 AUC 分别提高了 0.7%、1.3%和 3.5%，在 Geometry 数据集上分别提高了 2.1%、3.5%和 3.6%。其次，本文又设计了新的基于习题难度的学生能力表征模型，将各模型在 Assistment12 数据集上的 AUC 进一步分别提高了 0.2%、5.6%和 3.2%，在 Geometry 数据集上分别提高了 0.1%、0.6%和 1.2%。最终，本文设计的基于习题难度增强的知识追踪模型 DAS3H-DW 比当前最新的 DAS3H 模型，在 Geometry 和 Assistment12 数据集上 AUC 分别提高 2.2%和 0.9%，证明了本文算法的通用性和有效性。

(2) 对于深度学生知识追踪模型，本文首先也在模型中引入习题难度信息，然后基于真实学生做题数据分析结果，设计了新的学生学习能力分段表示方法，以实现准确的学生知识状态动态追踪。在真实数据上的实验结果表明：该方法将现有深度知识追踪模型的 AUC 提高了 0.3%。其次，本文引入班级类型对学生进行长期能力特征的区分，AUC 提高了 0.4%。最终，本文设计的综合上述两种方法的新的深度知识追踪模型 DIDKT-CL 相比于现有个性化深度学生模型 IDKT，AUC 提高了 0.7%，证明了本文算法的有效性。

图 19 幅，表 5 个，参考文献 52 篇。

关键词：学生知识追踪；习题表征；学生能力表征；学生学习能力

ABSTRACT

With the development of the Internet, intelligent education has become an urgent need for the development of education. Knowledge tracing model is to get accurate students' ability representation and exercise representation based on the students' historical learning records, and then to predict the students' mastery of skill. Therefore, intelligent education requires an accurate knowledge tracing model.

At present, there are two main reasons that restrict the effect of the knowledge tracing model: (1) The representation of exercises is inaccurate. It is difficult to describe the precise information of the exercises with skills as granularity; Taking exercises as granularity, the model parameters learning is not accurate due to sparse data; The existing model of comprehensive skills and exercises uses additive model to represent exercises, which does not conform to the distribution of actual data. (2) The representation of students' ability is inaccurate. The traditional model is based on skills to characterize students' abilities, ignoring that different difficulty exercises under the same skill can improve students' abilities differently. However, the current dynamic personalized deep knowledge tracing model does not consider the difficulty of exercises when measuring students' learning ability. Besides, it calculates the learning ability by accumulating student records, so it can’t accurately capture the students' real-time ability changes. In addition, the long-term personalization of students is not considered.

Therefore, the paper is based on a student practice data set of real intelligent education system, and carefully measured and analyzed the students' behavior and difficulty of exercises. For traditional knowledge tracing models, the paper proposes an improved model which can accurately represent the exercise and the students' ability. For deep knowledge tracing model, the paper proposes a personalized deep knowledge tracing model based on the combination of long-term personalization and dynamic personalization. The specific contributions are as follows:

(1) For the traditional knowledge tracing models, firstly, in order to improve the representation of exercises, the paper designs a multiplicative model based the exercises and skills according to the data analysis results of real students. The AUCs of DAS3H, AFM and PFA in Assistment12 dataset were increased by 0.7%, 1.3% and 3.5% respectively, and 2.1%, 3.5% and 3.6% in Geometry data set respectively. Secondly, the paper improves the representation of students' ability based on difficulty of the exercise. The AUCs of this method were improved by 0.2%, 5.6% and 3.2% on Assistant12 dataset,

and by 0.1%, 0.6% and 1.2% on Geometry dataset respectively. Finally, the new knowledge tracing model DAS3H-DW based on the difficulty enhancement of exercises designed in this paper, which integrated the above two methods, had an AUC increase of 2.2% and 0.9% on Geometry and Assistant12 datasets respectively, compared with the latest DAS3H model. It proved the generality and effectiveness of the algorithm designed in this paper.

(2) For the deep knowledge tracing model. Firstly, this paper introduces the difficulty information of exercises, and then represents the students' learning ability in segments based on the data analysis results of real students, to trace students' knowledge state accurately. Experimental results on real data show that the proposed method improves the AUC of the existing dynamic personalized deep knowledge tracing model by 0.3%. Secondly, this paper introduces the class type characteristic into distinguishing the long-term ability of students, AUC increased by 0.4%. Finally, the new deep knowledge tracing model DIDKT-CL designed in this paper, which integrates the above two methods, had an AUC increase of 0.7% on DIDKT, compared with IDKT model, which proved the effectiveness of the algorithm.

19 figures, 5 tables, 52 references.

KEYWORDS: Knowledge tracing; Exercise representation; Students' ability representation; Students' learning ability

目录

1 引言

1.1 研究背景及意义

教育是迫切的民生要求。互联网的发展促进了在线教育系统取得了很大的进步[1]，如慕课（Massive Open Online Courses, MOOCs），ASSISTments 等在线教育系统使更多学生享受到了好的教育资源[2]，但是目前的智能教育发展仍不成熟，仍然是优质的教师资源为主导。

智能教育是学生进行有意义学习的必然要求。因为教育资源有限，没有足够的师资进行“1 对 1”的教育，教师在教学过程中只能按最典型同学的接受能力和学习方法，给所有学生布置同样的习题。但是每个学生都有自己的个性化学习方式，就导致中小学生存在作业负担重，学习效率低的问题。这种现象已经引起社会和学术界的广泛重视。近日，教育部就在线教育发布了指导意见，推动人工智能和大数据技术在在线教育中的应用，建立以学习者为中心的学习型社会。此外，在学生和智能教育系统交互的过程中积累的数据也能更好地促进智能教育的发展[3]。

智能教育需要准确的学生知识状态追踪模型。智能教育系统的主要目标是为学生提供个性化教育服务，而个性化教育服务要求智能教育系统必须掌握学生个体的知识状态，这就需要学生知识状态追踪模型。学生知识状态追踪模型跟据学生的做题记录追踪学生个体对知识的掌握程度，同时学习关于习题难易程度的信息，并依据学生个体的知识状态与习题的难易度信息进一步预测学生个体在未来时间段的答题表现。

但是，目前学生知识追踪模型仍存在设计上的缺陷，导致模型构建的用于表示习题难易度的习题表征和用于表示学生知识状态的学生能力表征不够准确。具体来说：(1)习题表征不准确。以知识点为粒度的习题表征难以刻画习题的精确信息；以习题为粒度的表征，由于数据稀疏导致模型参数学习不准确；已有的综合知识点和习题的混合模型采用加性模型进行习题表征，不符合实际数据的分布。(2)学生能力表征不准确。传统模型基于知识点进行学生能力表征，忽略了同一知识点下不同难度习题对学生能力的提升不同，而深度模型在对学生学习能力进行度量时，没有考虑习题难度，并且通过累积学生记录进行学习能力计算的方法不能准确捕捉学生即时的能力变化，也没有考虑学生的长期能力特征。

因此，本研究的目标是：基于大规模实际系统的数据集，对真实系统中的习题和学生特性进行测量、分析和研究，设计准确的学生能力表征和习题表征模型，进

而准确地进行学生知识追踪。具体来说，本文对传统学生知识追踪模型和深度学生知识追踪模型都进行了研究，设计了改进算法。这些算法充分利用了智能教育系统学生做题记录资源，结合发现的规律，改进现有学生知识追踪的模型性能，提高模型预测的准确度。本文的工作对设计智能的个性化教育，减少学生的题海负担，实现因材施教，降低家长的课外辅导经济负担，具有重要的经济价值和社会意义[4-6]。

1.2 国内外研究现状

知识追踪任务是基于学生练习的历史，追踪学生知识水平变化。由 Corberrt 和 Anderson 教授于 1995 年提出该任务[7-8]。基于智能教育系统得到的数据，进行学生知识追踪是目前的研究趋势。目前国内外有大量的工作研究学生知识追踪问题，主要为了解决习题表征不准确以及学生能力表征不准确，导致学生答题预测不准确的问题。下面将从传统学生知识追踪和深度学生知识追踪两个方面进行介绍。

1.2.1 传统学生知识追踪技术及其问题

传统学生知识追踪模型具有可解释性强的优点，在智能教育系统中应用广泛。本小节我们将介绍传统学生知识追踪模型的研究以及习题难度的研究，简要解释其在习题表征和学生能力表征方面的不足。

传统学生知识追踪方面的研究有以下几个方面：

(1) 贝叶斯知识追踪模型(Bayesian knowledge tracing, BKT)[9]

BKT 是基于贝叶斯网络构建的时序模型，它基于一个二元变量表示学生对特定知识点的掌握情况，使用二元隐马尔可夫模型更新该变量。BKT 模型的问题在于它是以二元变量表示知识点的掌握程度不细致，也没有考虑到知识点下的题目之间的区别。后续也有大量基于 BKT 改进的模型，比如 Multi-Grained-BKT 和 Historical-BKT 模型[10]考虑了概念之间的关系，将知识点的先后关系以及层级架构引入到模型里，提高了模型的预测性能。贝叶斯诊断模型(BDT , Bayesian diagnosis tracing)模型相对于 BKT 模型将学生答错习题作为负反馈引入模型，提高模型的预测准确度[11]。TD-BKT 模型基于 BKT 进行改进，融合时间维度的特征，提高了预测的准确度[12]。还有融合了学生行为和遗忘因素的 BF-BKT 模型[13]。以及添加了教学干扰改进 BKT[14]。还有一些改进的模型将学生个性化因素融合到模型里面[15-16]。虽然这些扩展模型相比于 BKT 都提高了预测准确度，但是仍然沿用二元变量表示学生知识点的掌握情况。目前有研究者提出了三状态的学生知识追踪模型，增加了一种过度状态[17]。但是仍然无法准确跟踪知识点的掌握水平。

BKT 以及基于 BKT 进行改进的模型在习题表征和学生表征方面的不足是：它们在习题表征方面都以知识点为粒度，粒度太粗，忽略了同一知识点下的题目难易度不同。而在学生能力表征方面以二状态或者三状态，无法准确跟踪知识点的掌握水平。

(2) 基于习题进行研究的模型

项目反应函数（Item Response Theory, IRT）[18-20]是线性模型，仅仅使用拟合的习题难度作为习题表征，并且认为学生能力是一直不变的，不是动态更新的。三参数项目反应函数（Three-Parameter Logistic, 3PL-IRT）[21]在原始 IRT 上添加了区分度，猜测度参数。埃罗预测（ELO）模型原本是博弈领域的模型，作者认为学生和习题应该是对手的关系，因此应用在教育领域进行学生知识追踪。根据制定的规则，当学生做完一道题目之后根据做题结果更新学生的状态[22-23]。DASH[24-25]模型认为学生的能力是随着时间变化的，基于此思想改进 IRT，将习题的答错和答对数量在时间窗口内进行累加，然后按照指数衰减的方式进行遗忘。

基于习题进行研究的模型在习题表征和学生表征方面的不足是：对于学生做题记录少的习题，习题关系学习不准确，导致习题表征不准确。此外，因为学生学习通常以知识点为单位，所以模型仅仅以习题为粒度进行学生能力的表征不准确。

(3) 基于知识点进行研究的模型

加性因素模型（Additive Factor Mode, AFM）[26]和学习效果因素分析（Performance Factors Analysis, PFA）[27]都是动态的概率模型，都是用知识点的容易度表示习题表征，认为知识点越容易，那么学生答对的概率越大。但是我们通过分析数据集里面的题目发现，即使一个知识点下的题目，习题难易度差别也很大，因此，仅仅利用知识点容易度进行题目表征，粒度太粗。在动态学生能力表征的过程中，AFM 模型认为在同一个知识点上学生每做一次题，就增加相同的能力。PFA 认为在同一个知识点上答对一道题目和答错一道题目，学生的能力增长是不一样的，基于此思想在 AFM 模型的基础上进行改进，提高了预测准确度。学习因素分析(LFA, Learning Factors Analysis）[28]以及 AFM+Slip 和 PFA+Slip 模型等扩展模型也都是基于知识点进行研究的模型[29]，这些模型在习题表征和学生表征方面都不准确。

现有的基于知识点进行研究的模型，在习题表征和学生能力表征方面存在的不足是：其习题表征仅考虑了题目相关知识点的难度，没有区分知识点下的题目难度，所以习题表征不准确。在学生能力表征过程中也没有考虑不同习题难度带来的能力增益不同，所以学生能力表征不准确。

(4) DAS3H（item Difficulty, student Ability, Skill, and Student Skill practice History）[30]模型

DAS3H 模型是目前效果最好的传统学生知识追踪模型，知识点容易度和习题难度都是参数学习的方式得到，然后进行加性结合得到最终的习题表征，基于此思想改进 DASH 模型，然而我们通过测量数据集，发现知识点难度和习题难度是对数正态分布的，所以这种结合方式并不合理。另外对于学生能力水平表征，此模型参考了基于知识点进行研究的模型，对于学生能力表征在知识点粒度上进行积累或者遗忘。但是此模型对于一个知识点的能力增长或者遗忘也是通过题目答对次数以及尝试次数累加的方式，并没有考虑一个知识点下不同习题难度带来不同的能力增益，因此学生学习能力增益模型设计不合理，导致学生能力表征不准确。

(5) 关于习题难度在学生知识追踪方面的研究

目前加拿大蒙特利尔理工学院的研究者根据学生的做题记录得到习题难度，然后可视化学生的学习过程，发现习题难度在学生学习的过程中起到很重要的作用，对于学生是否能答对题目有很大的影响，但是他们未将习题难度融合到学生知识追踪模型进行模型的改进[31]。[32]将习题难度特征融入到深度学生知识追踪模型进行改进，但是并未有习题难度作为特征进行传统学生知识追踪的改进。

综合上述分析，目前传统模型在习题表征和学生表征方面都存在不足：以知识点为粒度的习题表征难以刻画习题的精确信息；以习题为粒度的表征，由于数据稀疏导致模型参数学习不准确；已有的综合知识点和习题的混合模型采用加性模型进行习题表征，不符合实际数据的分布。学生能力表征也不准确。学生学习以知识点进行，传统模型基于知识点进行学生能力表征，忽略了同一知识点下不同难度习题对学生能力的提升不同。

基于上述分析，由于学生学习是基于知识点进行的，所以本文采用考虑了知识点的模型 AFM，PFA 和最新的效果最好的 DAS3H 模型，然后根据测量结果，基于学生做题历史得到习题难度特征结合乘性模型进行习题表征的改进，利用习题答错率特征进行学生能力表征的改进。

1.2.2 深度学生知识追踪技术以及问题

针对学生学习过程，传统的模型虽然可解释性强，但是参数较少，无法准确表达复杂的非线性的学生学习过程。为此，深度学习被引入学生知识追踪领域。虽然它在教学中可解释性不强，但是由于准确度较高，也取得了较好效果，本小节进行现有工作总结。简单解释其在学生表征方面的不足。

对于深度知识追踪模型个性化问题将从以下三个方面进行介绍：

(1) 经典深度知识追踪模型

斯坦福大学的研究者提出深度知识追踪（Deep Knowledge Tracing，DKT）[33]，

首次使用深度学习解决知识追踪问题。他们采用循环神经网络，模型取得了较好的效果。随后，香港科技大学的研究者们提出 DKT 模型存在的问题：(1)学生能力达到了一定程度，但是模型却认为没有达到；(2)学生对一个知识点的掌握是比较稳定的，但是模型预测不稳定等问题。通过引入正则化改进了模型的预测性能[34]。美国布兰迪斯大学的研究者们通过堆叠网络层克服单层循环神经网络长期依赖问题，并且利用残差网络克服深度网络训练难的问题[35]。日本的研究学者考虑学生学习过程中的遗忘，改进 DKT 模型中，取得了较好的效果[36]。香港中文大学的研究者们提出动态键值对网络（Dynamic Key-Value Memory Networks, DKVMN）改进神经元网络的结构、损失函数、训练方法等，改进模型的跟踪性能[37]。用记忆网络(Memory Network) 增强原来的循环神经元网络模型，在其中清楚地加入概念记忆模块，改进模型性能。他们在数据集上进行测试，取得了比 DKT 更好的效果。北京交通大学的研究者们基于 DKVMN 提出 DKVMN-CA，利用显示的知识点代替隐知识点取得了超过 DKVMN 的效果[38]。剑桥大学的研究者们将学生的答题序列作为输入，运用神经网络学习学生在每一个知识点上的掌握情况作为学生的能力表征。神经网络是由 embedding 矩阵组成。每一行代表一个学生，每一列代表每一个学生在这个知识点上的表现，神经网络反向传播来更新每一个 embedding 层参数[39]。

现有的深度知识追踪模型在学生表征方面的不足是：没有考虑学生的个性化特征。这些模型把每一个学生当成一个样本，学习到的是学生的总体特征，然后根据学生的历史记录学习到学生的目前的知识状态。没有考虑到学生自身的个性化特征。

(2) 改进习题表征的深度学生知识追踪模型

好未来公司的研究者们将图嵌入引入到知识追踪领域，用图嵌入模型学习习题之间的关系，他们认为具有相似难度以及相似的概念的两个习题，学生回答这两个题的正确性应该是相似的[40]。纽约州立大学布法罗分校的研究者们提出了一个 DHKT 模型[41]。他们引入习题的概念信息来补充由学生答题行为序列获得了习题关系表征，引入了一个 mapping-matrix 图来表达概念-习题图。科大讯飞公司的研究者们认为习题文本带来了很多习题自身的信息，添加了习题文本，通过 word2vec 和双向长短期记忆网络模型得到习题表征向量，然后送到学生知识追踪模型里面进行学生能力追踪，取得了较好的效果[42-44]。因此对于那些学生做题数量少的题目，模型学习不到题目与其他题目之间的关系。因为文本里面本身就带有语义信息，所以引入题目文本表征。这些模型是通过改进习题表征提升预测的准确度但是都没有考虑学生个性化。

现有的改进习题表征的深度知识追踪模型在习题表征方面进行了改进，但是

依旧是把每一个学生当成一个样本，学习到的是学生的总体特征，没有考虑学生自身的个性化特征。

(3) 引入个性化的深度学生知识追踪模型

蒙特利尔工学院和西北大学的研究者们为了得到更准确的学生能力表征引入了学生学习能力的特征，基于学生能力对学生进行聚类，认为具有相似能力的学生学习能力相似。通过实验验证个性化对学生知识追踪模型有一定的效果提升[45-46]。但是对学生的个性化还不够具体，学生的学习能力只考虑了每一个知识点上的答对比率，并没有考虑到习题的难度。并且通过积累学生记录进行学习能力定义，在累积多了之后会无法捕捉到学生的即时变化。

事实上习题难度也会影响学生的学习能力的定义。同一个知识点相同正确率的条件下，答对难度大的习题比较多的学生比答对难度小的习题多的学生的学习能力强。蒙特利尔工学院的研究者们认为不同难度的习题对学生有不同的影响[31]。如果学生都学习同一个知识点，学习成绩差的做了几次简单的练习之后再做简单的还是做错，但是学习能力强的做了几个简单的就能答对难度大的习题，当能答对难度大的题时说明学生能力有了很大的提升。所以习题难度对于学生能力的定义有重要的影响。

现有的引入个性化的深度知识追踪模型在学生表征方面的不足是：在学生能力定义的时候没有考虑知识点下的题目的区别，并且通过累积学生记录进行学习能力计算的方法不能准确捕捉学生即时的能力变化，所以在学生能力向量的编码过程中不准确。因此，基于该编码结果的学生分类也会存在偏差，影响模型性能。也没有考虑学生的长期能力特征而追踪到表示学生能力水平的学生能力表征。

基于上述分析，本文为了得到更准确的学生能力表征，在深度学生模型上进一步改进个性化，将个性化分为长期个性化和动态个性化，对于动态个性化引入习题难度并对学生学习能力进行分段化表示改进现有动态个性化方法，使学生学习能力定义更准确。

1.3 主要研究内容

目前的学生知识追踪模型，不管是传统学生知识追踪模型还是深度知识追踪模型都需要得到**准确的学生能力表征和习题表征**，才能进行准确的预测。本文针对传统模型在习题表征和学生能力表征存在的问题进行改进，针对深度学生知识追踪在学生能力表征方面进行改进。

本文分传统学生知识追踪和深度知识追踪进行研究。具体有以下两方面的工作：(1)传统学生知识追踪，本文基于数据测量结果，就习题表征和学生能力表征不

准确进行了改进，提高了模型预测准确度。(2)深度学生知识追踪，本文基于个性化改进学生能力表征，进而提高了预测准确度。

1.3.1 传统知识追踪模型

因为传统的知识追踪模型具有很好的可解释性，所以在商业智能教育系统普遍应用。**我们通过分析模型存在的缺陷和数据的规律，合理的改进传统知识追踪模型中的习题表征和学生能力表征，提升模型预测学生答题结果的准确性**。基于在线教育系统学生答题日志分析数据中存在的规律，进行学生知识追踪模型中习题表征和学生能力表征的改进，提升模型的预测性能。

具体来说，基于 ASSISTments 学习网站公开的真实的数学科目学生答题记录数据集进行分析，数据集包括了答题知识点，习题编号，做题时间，习题作答结果等详细字段信息。通过对知识点粒度和习题粒度的学生答错率的分布，数量分布，习题难度分布等习题相关的测量，结合模型本身的特点，对习题表征进行改进。通过观察学生的答题过程，以合理的方式对学生能力表征进行改进。在 ASSISTments 公开数据集 Assistment12 和 Pittsburgh Science of Learning Center DataShop 数据中心的 Geometry 数据集进行实验，分析改进方法对于模型性能的影响。

1.3.2 深度知识追踪模型

深度知识追踪模型利用深度神经网络进行模型的构建，虽然可解释性差，但是由于其性能好，也受到了人们广泛的研究。**我们通过研究学生的长期个性化和短期动态个性化，得到准确的学生能力表征，对模型的效果进行改进。**

具体来说对于短期动态个性化，现有模型基于学生历史动态地得到学生学习能力表征向量，利用向量进行聚类对学生进行分组，实现动态个性化，但是学生学习能力表征向量定义不准确。(1)我们通过分析聚类的效果，发现规律，然后结合第三章测量习题难度，发现不同习题难度给学生学习能力带来不同的提升，基于习题难度特征改进学生学习能力向量的定义方式。(2)通过分析发现学生学习能力编码单元的选取不合理，通过对学生学习能力进行分段化表示改进学生学习能力向量的定义。(3)对于长期个性化，我们分析数据，发现不同班级的学生学习能力不一样，基于此特征进行学生长期个性化研究。最后通过在真实大规模系统数据进行实验，通过实验结果分析动态个性化的改进以及长期个性化对于模型性能的影响。

1.4 贡献

我们将本文贡献分两部分进行介绍:

(1) 对于传统学生知识追踪模型，具体贡献如下：

1) 为了解决传统学生知识追踪习题表征不准确的问题，本文通过测量真实大规模在线教育系统的学生做题数据，发现习题答错率和知识点答错率都服从对数正态分布，并且基于知识点粒度的做题数据，稠密而均匀，基于习题粒度的做题数据稀疏并且不均匀。为了知识点和习题进行结合得到习题表征。本文提出知识点难度和习题难度以乘性模型的形式结合进行习题表征。知识点难度是模型学习出来的参数，习题难度特征是通过习题答错率和知识点答错率相除得到。在目前主流的知识追踪模型 DAS3H、AFM 和 PFA 模型上进行实验：在 Assistment12 数据集上的 AUC 分别提高了 0.7%、1.3%和 3.5%，在几何数据集上 AUC 分别提高了 2.1%、3.5%和 3.6%。证明了该方法的有效性和通用性。

2) 为了解决学生能力表征不准确的问题。本文通过分析学生做题数据，发现习题答错率对于学生能力的表征有很大的影响，传统模型基于知识点进行学生能力表征，忽略了同一知识点下不同难度习题对学生能力提升不同。本文引入习题答错率作为习题难度的表征，对学生能力表征进行改进。接着在 1）改进的基础上进行实验，在 Assistment12 数据集上 AUC 分别提高了 0.2%、5.6%和 3.2%，在几何数据集上分别提高了 0.1%、0.6%和 1.2%。证明了在学生能力表征中引入习题答错率的有效性和通用性。

3) 最终，本文设计的综合上述两种方法的新的知识追踪模型 DAS3H-DW 模型，比当前最新的 DAS3H 模型，在几何数据集上 AUC 提高 2.2%，在 Assistment12 数据集上提高 0.9%。

(2) 为了解决深度学生知识追踪模型中学生能力表征不准确的问题。我们对现有动态个性化进行改进，并提出长期个性化和动态个性化结合进行学生个性化。对于动态个性化我们通过分析现有模型的聚类结果，发现对学习能力计算时，时间单元选取不合理，因为累积学生记录进行，选取的学生记录无法准确捕捉学生即时的状态变化，也没有考虑习题难度。本文通过引入习题难度信息并对学生学习能力进行分段化表示，改进了现有的学生动态个性化模型，在真实数据上的实验结果表明，AUC 指标效果提高了 0.3%。基于学生班级进行学生长期的区分，实现长期个性化，AUC 指标效果提高了 0.4%。最后将长期个性化特征和动态个性化结合提出 DIDKT-CL 模型相比于添加现有个性化特征模型 IDKT，AUC 指标效果提高了 0.7%，相比于 DKT 模型效果提高了 1.1%。

1.5 论文的组织结构

本文的组织结构如下：

第二章为本文的相关工作的技术背景，包括所使用的机器学习，自然语言处理技术，假设检验方法，模型的评估指标和开发平台。

第三章详细介绍了基于数据测量进行传统学生知识追踪的改进，分析现有的智能导学系统中的学生做题记录，根据分析结果对现有的模型在习题表征和学生能力表征两方面进行改进。并通过对比实验，评估不同改进对模型的影响。

第四章详细介绍了基于个性化进行研究的深度学生知识追踪模型。分析数据，基于学生班级进行学生长期的区分，实现长期个性化。分析现有动态个性化存在的问题，基于习题难度和编码单元进行改进，实现动态短期个性化。通过实验评估不同改进对模型的影响。

第五章对全文进行总结，并提出对未来工作的展望。

2 技术背景

本章主要介绍研究工作相关的技术背景，包括使用的相关机器学习算法，相关的学生知识追踪模型以及数据分布检验方法，最后介绍了模型评估指标，实验开发平台以及所需要的算法库。

2.1 机器学习

机器学习是一门涉及多个领域的学科，主要研究机器如何按照人的方式去学习，用机器模拟人类学习，属于一门人工智能科学。机器学习主要是根据之前的历史数据或者历史经验，调节模型，使模型像人一样基于知识进行学习，然后进行预测，包括分类，回归，生成等任务。本小节将针对本文研究相关的 K 均值聚类和循环神经网络进行介绍。

2.1.1 K-means 聚类

聚类方法在数据挖掘，数据分析，模式识别等领域被广泛使用，属于无监督学习，主要是通过聚类将数据进行分组，然后发现每一组数据的特点。聚类的方法有很多，为了解决学生个性化的问题，寻找具有相同学习率的学生，K 均值聚类方法被引入到学生知识追踪领域。

K 均值聚类是将所有样本分成 k 个子集，每一个子集都有一个中心点，每一个样本都需要算出和 k 个中心点的距离，选出离样本最近的中心点，就是样本所属的类别。首先我们介绍 K 均值聚类的距离计算方法，在很多开源库都有 K 均值聚类的实现方法，开源库使用的距离计算方法都是欧式距离，具体定义公式如下（2-1）所示：

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{m} (x_{ki} - x_{kj})^2} \tag{2-1}$$

欧式距离是根据两个样本对应维度之差的平方和，然后求平方根得到。测量的两个样本在同一个欧式空间中的距离。这就要求样本点数据都在同一个欧式空间，所以 K 均值聚类对于样本数据有较高的要求。

K 均值聚类算法是一个包含两个步骤的迭代过程。首先对于 k 个中心点，也即是 k 个类别的质心。根据 k 个质心，进行类别划分，算出样本和每一个质心的距离，每一个样本属于离它最近距离的质心所属的类别，如下公式（2-2）所示：

$$\min\sum_{l=1}^{k}\sum_{C(i)=l} d(x_i, m_l) \tag{2-2}$$

其中 m_l 为 k 个质心，x_i 为样本 i，l 为类别。目标是使得依据的划分方式得到的所有样本到所属类别的质心的距离加一起数值最小。也即是使得每一个样本划分到离自己最近的质心。

然后基于每一类样本的综合，对样本向量的每一维度平均，得出来的平均向量，就是每一个类别的新质心，也即是更新 k 个中心点，如下公式（2-3）所示。

$$m_l = \frac{1}{n_l}\sum_{C(i)=l} x_i, \qquad l = 1,\ldots,k \tag{2-3}$$

其中n_l为划分到类别 l 的样本的数量。

K 均值聚类就是重复上述两个步骤，直到最后每一类别的中心点不再变化，K 均值聚类算法完成。对于 k 个质心的初始化，通常选择从样本集里面随机选择 k 个样本作为初始质心。本文利用 K 均值聚类算法实现学生学习能力的动态分组。

对于聚类类别的选择，目前也有一些方法。肘部法是常用的方法，但是肘部法会存在“肘点”不清晰的情况。因此本文选择间隔统计量 gap-statistic 进行最优 k 值的选择，gap-statistic 是 Robert 教授提出的[47]。间隔统计量使用的是样本之间的类内样本之间的欧式距离，距离越近，也就说明类内的样本越紧密，效果越好。类别之间的距离定义如下公式（2-4）所示：

$$D_k = \sum_{x_i\in C_k}\sum_{x_j\in C_k}\left\|x_i - x_j\right\|^2 \tag{2-4}$$

其中 x_i 和 x_j 属于类别为 k 的任意两个样本。D_k 为类别为 k 的任意两点的距离和。标准化后得到的 w_k 定义如下公式（2-5）所示：

$$w_k = \frac{1}{2n_k}D_k \tag{2-5}$$

其中 n_k 为类别 k 内的样本数量。那么间隔统计量定义如下公式（2-6）所示：

$$Gap(k) = E_n^*(\log(w_k)) - \log(w_k) \tag{2-6}$$

$E_n^*(\log(w_k))$ 表示参考样本数据集的标准化后的平均值。定义如下公式（2-7）所示：

$$E_n^*(\log(w_k)) = (1/B)\sum_{b=1}^{B}\log(w_{kb}^*) \tag{2-7}$$

其中 B 为生成的参考数据集的个数，参考数据集是由蒙特卡洛采样方法获得。那么 E_n^* 也就是参考样本数据集的标准化后的平均值。

最后通过标准差来矫正蒙特卡洛抽样算出来的间隔统计量，在计算标准差之前我们定义 $sd(k)$也即是抽样样本的标准差为如下（2-8）所示：

$$sd(k) = \sqrt{(1/B)\sum_{b}(\log(w_{kb}^*) - E_n^*(\log(w_k)))^2} \tag{2-8}$$

最终的标准差 s_k 定义如下公式（2-9）所示：

$$s_k = sd(k)\sqrt{1+1/B} \tag{2-9}$$

那么满足以下公式（2-10）的最小的类别值就是我们选择的最优类别值。

$$Gap(k) \geq Gap(k+1) + s_{k+1} \tag{2-10}$$

本文利用 Gap-statistic 方法进行 K 均值聚类算法中类别个数的寻找。

2.1.2 循环神经网络

循环神经网络添加了记忆功能，受启发于人类大脑在学习新东西的时候，大脑存储的有之前学习到的基础知识，所以在原始前向神经网络的基础上添加了循环结构达到了记忆功能。在自然语言处理，天气任务预测，语音识别等带有先后顺序的时间序列数据任务中广泛使用[48-49]。

循环神经网络 RNN 具体模型结构如下图 2-1 所示，*x* 为模型的输入，*h* 为模型的隐藏层单元，*o* 为 RNN 单元结构的输出。模型的当前时刻的隐藏节点包含了，当前时刻以及之前的所有时刻的序列的输入信息的总结，并且还和下一时刻的输入一起作为输入送入到下一时刻的隐藏节点。此模型结构由于此记忆功能的存在，能发现输入序列之间的关联性，因此也能解决哪些需要长期依赖关系的任务。正如人类进行学习，已经学习了这个知识点的很多题目，那么下一时刻遇到这个知识点的新题目，就会根据之前学习到的知识积累来解决这道题目。

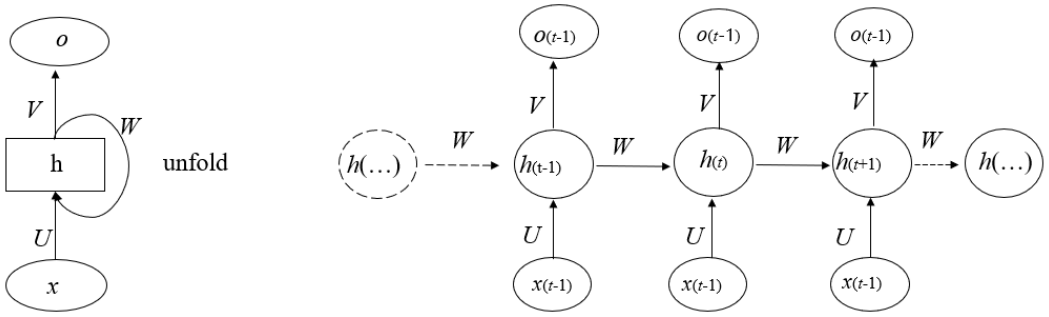

图 2-1 RNN 模型结构

Figure 2-1 Recurrent Neural Network Structure

如上图 2-1 所示，为完整的 RNN 模型结构。左边是未展开的 RNN 结构，右边为展开的 RNN 结构。其中 *U*，*V* 和 *W* 都是模型的权重，同一类型的权值共享。其中 *U* 为模型输入时学习到的权重，*W* 为上一时刻的隐藏节点连接到下一时刻的隐藏节点时需要学习到的权重，*V* 为模型由隐藏节点到输出时需要学习到的权重。那么 *t* 时刻，隐藏层 *h* 的状态为如下公式（2-11）所示：

$$h_t = \phi(Ux_t + Wh_{t-1} + b) \tag{2-11}$$

其中 *b* 为偏置，∅为非线性的激活神经函数，一般选择 Sigmoid 或者 Tanh。*U* 和 *W* 作为此刻输入和上一时刻隐藏节点的权重，主要起到筛选作用，*b* 作为偏置主要起

到调节作用。

最后隐藏层通过权重 V 过滤和偏置 c 的调节，得到 RNN 的输出 o，具体公式如下（2-12）所示：

$$o_t = Wh_t + c \tag{2-12}$$

那么最终的模型输出会根据下游任务进行激活函数的选择，一般 RNN 进行的是分类任务，因此通常会选择 Sigmoid 激活函数。模型里面的权重和偏置会在模型训练反向传播的时候基于损失函数最小原则进行调节。

RNN 结构虽然添加了记忆功能，但是在较长的时间序列任务中，会存在梯度爆炸和梯度消失的问题。长短期记忆神经网络属于循环神经网络的一种，受启发于大脑随着时间的推移，在不复习某些基础的时候会遗忘，在循环神经网络的基础上添加了门进行控制，一定程度上解决了循环神经网络梯度消失和梯度爆炸的问题。

LSTM[50]单元结构在 RNN 的基础上添加了门控装置，包括输入门，输出门和遗忘门。具体单元结构如下图 2-2 所示。通过门控装置让信息选择性地流动到细胞。

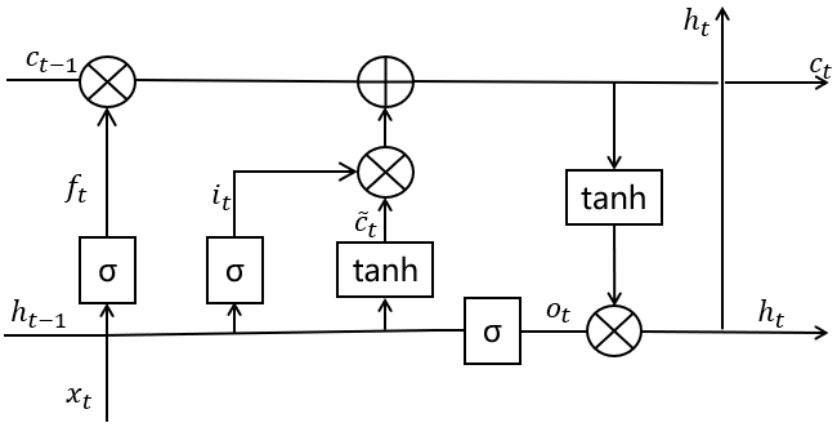

图 2-2 LSTM 单元结构

Figure 2-2 LSTM Neural Network Structure

图 2-2 中，x_t 表示 t 时刻的输入，h_{t-1} 表示 t-1 时刻的隐藏层信息，C_{t-1} 表示在 t-1 时刻的细胞状态。σ 表示 Sigmoid 激活函数， tanh 表示 Tanh 激活函数。

遗忘门具体计算如下公式（2-13）所示：

$$f_t = \sigma(U_f x_t + W_f h_{t-1} + b_f) \tag{2-13}$$

输入门具体计算如下公式（2-14）所示：

$$i_t = \sigma(U_i x_t + W_i h_{t-1} + b_i) \tag{2-14}$$

输出门具体计算如下公式（2-15）所示：

$$o(t) = \sigma(U_o x_t + W_o h_{t-1} + b_0) \tag{2-15}$$

记忆单元 $\tilde{c}_t$ 定义如下公式（2-16）所示：

$$\tilde{c}_t = \mathrm{Tanh}(W_o x_t + U_o h_{t-1} + b_o) \tag{2-16}$$

那么最终得到的当前细胞状态 c_t 如公式（2-17）所示：

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{2-17}$$

其中⊙表示的是哈达马乘积。从公式可以看出，最终的状态是由前一时刻的细胞状态经过遗忘和当前输入的新信息经过记忆单元更新得到。隐藏层 h_t 也是迭代更新，通过输出门和当前细胞状态进行更新，计算公式如下所示：

$$h_t = o_t \odot \mathrm{Tanh}(c_t) \tag{2-18}$$

模型结构里的权重 W 和 U 以及偏置 b 也会在模型训练反向传播的时候基于损失函数最小原则进行调节。

后续 GRU 基于 LSTM 进行了改进，将输入门和遗忘门进行结合，GRU 在训练速度上较 LSTM 更快，但是在某些实验结果并没有比 LSTM 好。因此对于不同的数据需要进行一定的尝试，确定哪一个模型更适合。

本文引入长短期记忆网络用于深度学生知识追踪，基于学生以前的做题历史，追踪到学生目前所处的知识状态水平。

2.2 学生知识追踪模型

本小节就学生知识追踪模型进行介绍。分传统学生知识追踪和深度学生知识追踪进行介绍。传统学生知识追踪主要是模型可解释性强，所以具有使用价值高，一直被研究。而深度学生知识追踪主要是模型效果好，所以近年来也被大家广泛研究。本章主要介绍基于心理统计学模型的传统学生知识追踪模型，和基于深度神经网络的深度学生知识追踪模型。

2.2.1 传统学生知识追踪

本小节主要介绍 IRT 模型，以及基于 IRT 模型改进的 DASH 模型。第三章所用的 DAS3H 模型就是在 DASH 模型的基础上进行改进的。

(1) IRT 模型

IRT 模型主要是用来研究学生是否能通过测试的一种模型。最初的模型由两个参数表示，一个参数表示用户的能力，一个参数表示习题的难度。最后通过 Sigmoid 函数决策学生答对的概率。其中 Sigmoid 函数定义如下公式（2-19）所示。

$$\sigma(x) = 1/(1+e^{-x}) \tag{2-19}$$

IRT 定义公式如下（2-20）所示：

$$P(Y_{s,j} = 1) = \sigma(\alpha_s - \delta_j) \tag{2-20}$$

其中 s 表示学生，j 表示题目序号，α_s 表示学生 s 的能力，δ_j 表示习题 j 的难

度。σ 表示 Sigmoid 函数。最后输出的就是学生答对的概率。通常概率超过 0.5 就认为预测学生答对，小于 0.5 就是预测学生答错。也即是当学生能力超过习题难度一定的程度，学生就能答对。

IRT 模型认为学生能力不是动态变化的，所以不符合常理。所以后续 DASH 模型在 IRT 的基础上进行改进。

(2) DASH 模型

DASH 模型是将学生能力，习题难度以及学生做题历史融合在一起。在 IRT 的基础上添加了学生做题历史，也即是学生能力是动态变化的。具体公式如下（2-21）所示：

$$P(Y_{s,j,t}=1)=\sigma(\alpha_s-\delta_j+h_\theta(t_{s,j,1:l},y_{s,j,1:l-1})) \tag{2-21}$$

其中 h_θ 是关于参数 θ 的函数，表示的是学生在 t 此刻之前的学生做题历史总结，具体定义如下（2-22）所示：

$$h_\theta(t_{s,j,1:l},y_{s,j,1:l-1})=\sum_{w=0}^{W-1}\theta_{2w+1}\log(1+c_{s,j,w})-\theta_{2w+2}\log(1+a_{s,j,w}) \tag{2-22}$$

其中 W 是时间窗口的个数，时间是以天为单位时间窗口也是$\{1/24,1,7,30,+\infty\}$，$c_{s,j,w}$ 为学生 s 做习题 j 之前在时间窗口 w 内，关于习题j答对的次数，$a_{s,j,w}$ 为学生 s 做习题j之前在时间窗口 w 内，关于习题 j 尝试的次数。θ_{2w+1} 参数表示的是学生在时间单元 w 内答对习题带来的能力的提升，θ_{2w+2} 需要学习的是学生在时间单元 w 内每次尝试带来的遗忘，也即是每做一道题目带来的是一次遗忘，每答对一道题目带来的是学生能力的提升，所以研究的目标是最短的时间内带来最多的掌握。

此模型基于学生做题历史通过时间窗口，表示学生的遗忘和提升，基于习题进行的，但是学生学习是以知识点为单位进行的，所以学生能力表征不合理。对于习题表征，DASH 模型用习题难度表示，没有考虑知识点的影响，因此也不合理。后续我们第三章研究的 DAS3H 模型在此模型的基础上改进了基于知识点改进了学生能力的表征和习题表征。

2.2.2 深度学生知识追踪

本小节介绍深度学生知识追踪领域的经典模型 DKT。DKT 模型为深度知识追踪领域的经典模型，目前很多研究也是基于 DKT 模型进行的。本文也是在 DKT 模型的基础上进行研究。具体模型结构如下图 2-3 所示：

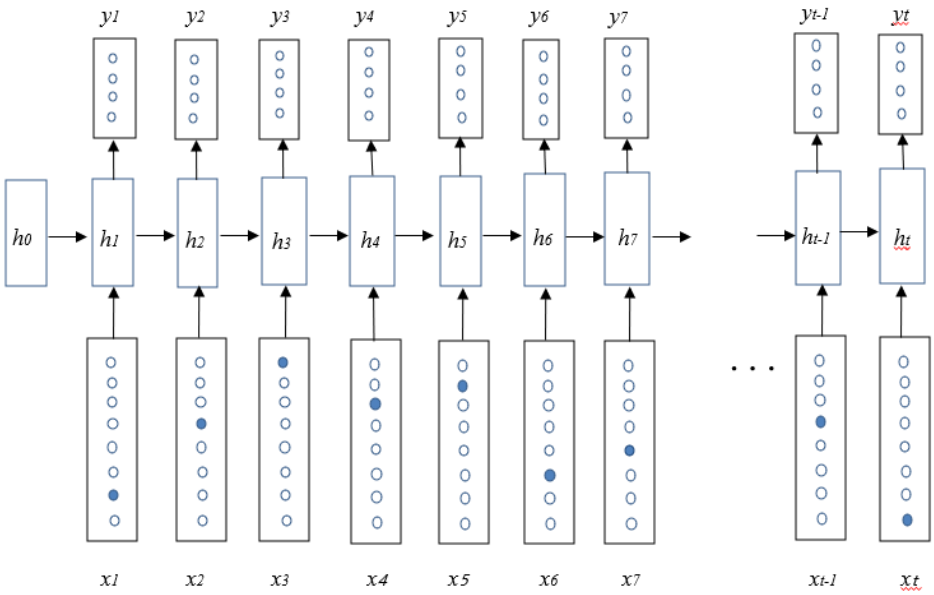

图 2-3 DKT 模型

Figure 2-3 DKT Model

如上图 2-3 所示，为 DKT 模型，以长短期记忆网络为基础模型。LSTM 模型结构的输入为特征向量的学生做题时间序列。是将每一个时刻的学生答题序号（习题序号或者知识点序号）和答题状态以独热编码的格式形成特征向量。以 LSTM 模型结构根据输入的学生特征向量序列，进行学生知识状态的追踪和遗忘，学习到学生目前所处的知识状态，最后根据学生做出的知识状态进行预测。输出维度是题目的数量，每一维度代表每一个题目，输出的概率就是预测学生在每一道题目上答对的概率。本文第四章研究学生个性化，以 DKT 模型为基础模型。

2.3 柯尔莫可洛夫-斯米洛夫检验

柯尔莫可洛夫-斯米洛夫检验（K-S test，Kolmogorov–Smirnov test）[51]是基于积累分布函数的检验方法，用于验证单一经验分布是否服从某一理论分布函数，或者两个经验分布是否是同一分布，也即是是否有显著性差异，是一种非参数检验方法。

本文使用的是单样本 K-S 检验，K-S 检验使用的是最小距离估计样本分布，首先将样本从小到大进行排序，然后比较样本的累加频率分布和理论分布在每一个点上的差值，取出差值的最大值，最大值越小说明检验样本和理论分布越接近。

对于假设验证，检验样本是否服从某一理论分布，需要一个指标来说明问题，检验标准就是 p-value 值。那么对于单一样本假设，原假设就是样本服从某一分布。在假设检验中，p-value 值通常与 0.05 进行比较，期望 p-value 值越大越好，p-value

值大于 0.05 就说明有 95%的可能性服从原假设。本文利用 K-S test 实现第三章习题分布的假设验证。

2.4 模型评估

对模型的性能进行评估是必不可少的。本节我们对模型的评估指标 AUC 和负对数似然（Negative Log-Likelihood，NLL）进行介绍。

2.4.1 AUC 指标

本小节结合知识追踪任务对 AUC 指标进行介绍。

知识追踪任务是预测学生是否能答对某道题目，也即是预测为 1 的概率，然后通过阈值进行决策，当预测概率大于阈值就是 1，当预测概率小于阈值就是 0。根据真实学生做题记录中是学生答对或者答错，我们将样本标签修改为 1 或者 0。也即是学生答对为 1，答错为 0。样本标签有两类，明显就是一个分类任务。对于分类问题，ROC-AUC 是度量模型性能必不可少的指标。AUC 是 ROC 曲线下的面积。下面我们基于表 2-1 所示的混淆矩阵进行介绍：

表 2-1 混淆矩阵

Table 2-1 Confusion Matrix

	预测为 1	预测为 0
真实为 1	TP	FN
真实为 0	FP	TN

其中 TP (True Positive) 为真实的样本为 1，模型预测也为 1，也即是真实记录学生答对了，模型预测学生也答对了的数量。FN (False Negative) 为真实记录学生答对了，模型预测学生答错了的数量。FP (False Positive)为真实记录学生答错了，模型预测学生答对了的数量。TN (True Negative)为真实记录学生答错了，模型预测学生也答错了的数量。这四个指标加一起也就是样本总和。

分类任务常用的指标有准确率，召回率和 F1，F1 为调和平均数，主要是为了平衡精确率和召回率。计算公式如下（2-23），（2-24）和（2-25）所示：

$$precision = \frac{TP}{TP + FP} \tag{2-23}$$

$$recall = \frac{TP}{TP + FN} \tag{2-24}$$

$$F1 = 2\frac{precision * recall}{precision + recall} \tag{2-25}$$

真阳率（True Positive Rate, TPR）和假阳率（False Positive Rate, FPR）是由混淆矩阵计算出来的两个指标，分类器在一个特定阈值下能计算出来一对 TPR 和 FPR 值， TPR 和 FPR 定义公式如下所示：

$$TPR = \frac{TP}{TP + FN} \tag{2-26}$$

$$FPR = \frac{FP}{FP + TN} \tag{2-27}$$

AUC-ROC 就是 FPR 和 TPR 在不同阈值下得出来不同的值，然后作为横纵坐标，画出来的曲线。我们希望分类器进行样本分类时，区分错的概率越少越好，也即是 FPR 越小越好，区分对的概率越多越好，也即是 TPR 越大越好。理想中的分类器是 AUC 为 1，曲线是经过（0,1）点的折线。也即是当 FPR 为 0，TPR 范围[0，1]，为纵坐标上 0 到 1 的一条直线，当 TPR 为 1，FPR 范围 [0，1]，是横坐标为 0 到 1，纵坐标为 1 的一条直线。当 AUC 为 0.5 时，说明分类器没有作用，随机分。当 AUC 为 0，说明分类器将正样本全部分成负样本，将负样本全部分成正样本。

2.4.2 NLL 指标

本小节结合学生知识追踪任务对负对数似然指标进行介绍。似然函数表示的是在已知观察数据（标签）时，关于模型参数的一个函数。对于我们的知识追踪模型，标签为二元变量 y_t（0 或者 1），似然函数表示的就是在已知标签的情况下，模型输出为 1 的概率。我们假设模型的输出为 P，P 是关于模型参数的一个函数，也即是模型预测为 1 的概率。那么似然函数如下公式（2-28）所示：

$$p(P / y_t) = P^{y_t} * (1-P)^{(1-y_t)} \tag{2-28}$$

极大似然的思想就是极大化似然，也即是标签已知的条件下，如何调节模型的参数，让输出的概率最大。为了计算方便，通常对似然函数求对数，也即是最大化对数似然，对数似然的定义如下公式（2-29）所示。

$$\ln p(P / y_t) = y_t \log(P) + (1-y_t)\log(1-P) \tag{2-29}$$

极大化对数似然就是极小化负对数似然。通常对模型训练的时候，期望损失函数最小，也就是模型中的参数使得模型预测和真实的标签非常的贴近。所以通常用的是负对数似然，如下公式（2-30）所示。值越小越好，越小说明模型性能越好。

$$-\ln(P / y_t) = -(y_t \log(P) + (1-y_t)\log(1-P)) \tag{2-30}$$

本文第三章使用样本的负对数似然的均值作为衡量指标。

2.5 开发平台

本节主要介绍实验过程中使用到的开发工具和框架。包括用于搭建深度神经网络模型的 PyTorch 框架，用于构建传统学生知识追踪模型的 pywFM 库, 用于假设检验的 Scipy 库和用于指标计算的 Scikit-learn 库。

2.5.1 PywFM 库

PywFM 是运用 python 编程语言包装的因子分解机库(libFM, Factorization Machine library)。libFM 的源码是 c++编程语言，PywFM 运用 numpy， scipy，sklearn 和 pandas 这些 python 工具库，基于 python 语言重新进行了包装，提供了 python 接口，提高了 PywFM 库的简便性。

libFM 最初适用于推荐系统中的问题，但是最近也被应用于很多领域的数据挖掘问题。libFM 受启发于经典的因子分解模型（SVD），但是 SVD 只有用户和标签两种类型的特征，为了解决特征增强的问题，就需要多特征结合，特征结合就会存在特征交叉的问题，常用的解决方法就是 one-hot 编码，one-hot 编码会导致特征稀疏， FM 就是解决多种特征结合下的稀疏性问题而提出的，而 libFM 就是一个通用的多种特征结合的库。本文利用 PywFM 库进行多种特征的结合，构建传统知识追踪模型。

2.5.2 PyTorch 框架

PyTorch 神经网络框架是由 Facebook 人工智能研究院基于 Torch 开发出来的。2017 年开源以来，受到人们广泛关注。PyTorch 底层和 Torch 结构一样，但是它们上层语言包装不一样，Torch 使用的是小众的 Lua，而 PyTorch 运用了 Python 编程语言重写了很多内容，并且提供了 Python 接口，便于学习和使用。PyTorch 和 Google 推出的 TensorFlow 类似，都是基于计算图进行完整的计算任务。但是 TensorFlow 在 2.0 版本之前都是基于静态图，2.0 之后的版本才改进为动态图。而 PyTorch 一直是基于动态图的，便于调试。由于简单易学易用等优点，PyTorch 神经网络框架目前已经成为流行的搭建神经网络框架的工具。

PyTorch 能够灵活支持 GPU 运行加速，对于多个 GPU 并行加速操作简单。PyTorch 中的张量(Tensor)，类似 NumPy 库的 ndarrays 的格式。PyTorch 对于张量的处理，类似 NumPy 对 ndarrays 的操作方式，方便简单快速。PyTorch 对 Tensor 采用的是自动求导机制，可以通过设置属性为 True 跟踪 Tensor 的相关计算。也可以利用.detach 将计算图中的某一个张量剥离出来，停止此张量的自动跟踪求导。PyTorch 在神经网络构造中可以通过多次重用同一个模块达到共享权重的目的。本

论文的神经网络基于 PyTorch 框架完成搭建和训练。

2.5.3 SciPy 工具库

SciPy 库是一个开源的基于 python 编写语言的高级数据科学库。SciPy 专门为科学计算提供统计分析，微积分，线性代数，信号处理等各种计算方法的一个工具库。主要应用于数学，工程学，科学等学科领域。SciPy 库可以有效的利用 NumPy 进行各种高性能的计算和数组变换。

本论文主要是利用 SciPy 进行假设检验。本文利用 SciPy 库实现 Kolmogorov-Smirnov 方法验证数据是否服从对数正态分布以及 p-value 指标的计算。

2.5.4 Scikit-learn 工具库

Scikit-learn 库是基于 python 编程语言写的用于数据挖掘和数据分析的开源框架。依赖 python 工具库中的 NumPy，matplotlib 和 SciPy 库，利用 NumPy 进行各种高性能的计算和数组变换。Scikit-learn 实现了多种有监督，半监督和无监督的机器学习算法，模型的选择和评估方法，数据集的多种转换方式和数据集的多种加载工具。

关于机器学习方法，Scikit-learn 包含有决策树，逻辑回归，支持向量机等常用分类算法；K-means 聚类，EM 最大期望聚类，层次聚类等常用聚类算法；线性回归，多项式回归，Lasso 回归和 Ridge 回归等常用回归算法；PCA，SVD 等常用降维算法。关于模型选择和评估方法，Scikit-learn 包含了网络追踪法进行模型选择，AUC，F1，MSE 等各种评价指标。关于数据集转换方式，Scikit-learn 实现了特征联合，缺失值插补和特征值处理等常用数据处理方式。关于数据集加载工具，Scikit-learn 包含通用数据集 API 和样本生成器等。本论文用 Scikit-learn 工具库实现数据分析和模型评估指标的计算。

2.6 本章小结

本章主要介绍了论文工作相关的技术背景。首先介绍了用于学生聚类的 K 均值聚类算法和用于跟踪学生能力的循环神经网络。然后介绍了与第三章和第四章学生知识追踪算法相关的传统学生知识追踪和深度学生知识追踪。最后介绍了分布的假设检验方法，模型评估指标以及实验的开发平台。

3 改进习题表征和学生表征的传统知识追踪模型

传统学生知识追踪模型的效果，主要和习题表征和学生能力表征这两方面因素有关。为了解决传统知识追踪模型中习题表征和学生能力表征不准确，进而导致预测准确度不高的问题，本章首先对现有模型进行介绍，然后基于大规模学生做题记录数据，对真实系统中学生答题记录进行了统计分析，最后对现有模型存在的问题进行改进，完成实验验证。

3.1 现有模型介绍

现有传统模型可以分解成习题表征和学生能力表征。本节分析了现有传统知识追踪模型的两种类型，并就各种类型模型分习题表征和学生能力表征两部分进行问题分析。

3.1.1 基于知识点进行研究的模型

AFM，PFA 是基于知识点进行学生知识追踪的模型，在目前的智能导学系统中被广泛地使用。下面将对模型进行介绍

AFM 模型如公式（3-1）所示：

$$P(Y_{s,j}=1)=\sigma(\sum_{k\in KC(j)}\beta_k+\gamma_k a_{s,k}) \quad (3\text{-}1)$$

其中，$KC(j)$表示题目 j 包含的知识点，β_k 和 γ_k 是模型需要学习的超参数，β_k 为知识点 k 的容易度，γ_k 为知识点 k 的学习率，$a_{s,k}$ 为学生 s 在做 j 这道题前在知识点 k 尝试的次数。

我们对公式（3-1）所示的 AFM 模型分解成两部分：

(1) 习题表征为：$\sum_{k\in KC(j)}\beta_k$

(2) 动态学生能力表征为：$\sum_{k\in KC(j)}\gamma_k a_{s,k}$

AFM 的问题是：在 AFM 模型中对于习题表征，仅仅考虑了题目涉及到的知识点容易度进行习题表征，并没有考虑到知识点下的题目容易度不同。对于学生动态能力表征也是基于知识点进行的，但是认为这个知识点下的题目无论简单还是难，每做一次能力增长都是一样的。

PFA 模型如公式（3-2）所示:

$$P(Y_{s,j}=1)=\sigma(\sum_{k\in KC(j)}\beta_k+\gamma_k c_{s,k}+\rho_k f_{s,k}) \quad (3\text{-}2)$$

其中 β_k 为知识点 k 的容易度，$c_{s,k}$ 为学生s在做j这道题前在知识点 k 尝试的正确的次数，γ_k 是知识点 k 答对的学习率，$f_{s,k}$ 为学生在做j这道题前在知识点 k 尝试的错误的次数，ρ_k 是知识点 k 答错的学习率。PFA 模型是基于 AFM 模型进行的改进，认为学生答错和答对对于学生能力的增长不一样。

我们对公式（3-2）所示的 PFA 模型分解成两部分：

(1) 习题表征：$\sum_{k\in KC(j)}\beta_k$

(2) 动态学生能力表征：$\sum_{k\in KC(j)}\gamma_k c_{s,k}+\rho_k f_{s,k}$

PFA 的问题是：在 AFM 的基础上，认为做题结果不同，学生的学习率也不同，能力增长也不同，以此观点对学生能力表征进行改进。但是和 AFM 也是一样的问题，没有考虑同一知识点下的题目难度不同给学生带来能力增长不同。

总之，这两种模型是基于知识点的，在习题表征和学生能力表征中并没有考虑到一个知识点下的习题之间的区别，认为所有习题中仅仅存在于知识点之间的区别，一个知识点下的习题难度都是一样的。因此习题表征和学生能力表征都不准确。

3.1.2　基于知识点和习题进行研究的模型

针对上述仅考虑了知识点的模型的问题，已有模型综合考虑了知识点和习题，如 DAS3H。本节介绍该模型。

DAS3H 模型如下（3-3）公式所示：

$$P(Y_{s,j}=1)=\sigma(\alpha_s-\delta_j+\sum_{k\in KC(j)}\beta_k+h_\theta(t_{s,j,1:l},y_{s,j,1:l-1})) \tag{3-3}$$

$h_\theta(t_{s,j,1:l},y_{s,j,1:l-1})$ 如下公式（3-4）所示：

$$h_\theta(t_{s,j,1:l},y_{s,j,1:l-1})=\sum_{k\in KC(j)}\sum_{w=0}^{W-1}\theta_{k,2w+1}\log(1+c_{s,j,w})-\theta_{k,2w+2}\log(1+a_{s,j,w}) \tag{3-4}$$

其中 α_s 为学生初始能力，δ_j 为习题难度，W 是时间窗口的个数，时间是以天为单位，时间窗口设置方式和 DASH 一样，时间窗口也是{1/24,1,7,30,+∞}，β_k 习题 j 相关的知识点 k 的容易度，$\theta_{k,2w+2}$ 是关于知识点 k 需要学习的学生在时间单元 w 内答对习题带来的能力的提升，$\theta_{k,2w+2}$ 需要学习的是学生在时间单元 w 内每次尝试带来的遗忘，也即是每做一道题目带来的是一次遗忘，每答对一道题目带来的是学生能力的提升，所以研究的目标是最短的时间内带来最多的掌握。

DAS3H 模型是在 DASH 模型的基础上综合考虑了习题因素和知识点因素，DASH 模型是以习题为粒度的模型，在习题表征和学生能力表征都是以习题为单位进行研究的。DASH 借鉴了基于知识点进行研究的模型进行改进。

我们对公式（3-3）所示的 DAS3H 模型分解成两部分，分别进行介绍。

(1) 习题表征：$\delta_j-\sum_{k\in KC(j)}\beta_k$，基于 DASH 模型进行改进，在习题表征的时

候，考虑了知识点因素，以加性的方式和习题难度相结合，也即是知识点容易度对习题表征进行了修正。

(2) 动态学生能力表征：$\alpha_s + \sum_{k \in KC(j)} \sum_{w=0}^{W-1} \theta_{k,2w+1} \log(1+c_{s,j,w}) - \theta_{k,2w+2} \log(1+a_{s,j,w})$，考率了知识点因素，在 DASH 模型的基础上认为学生能力的增长是基于知识点，所以做题次数以及答对的次数都是基于一个知识点进行计算的，基于此思想对于学生能力进行了修正。

因此，DAS3H 考虑了知识点和习题，是目前传统知识追踪领域最好的模型，但依旧有如下不足：

(1) 此模型仅仅在习题表征中考虑了知识点和习题结合，但是结合的方式需要改进。它对知识点容易度和习题的难度是以一种相减的形式进行处理的，但是我们通过下面章节中介绍的测量结果发现学生做题的答错率和知识点答错率都是 log-normal 分布的，所以应该以乘性模型进行习题表征。

(2) 它在动态学生能力表征的时候，认为一个知识点下的题目都一样，表征不准确。追踪学生能力的过程中，认为一个知识点下的题目，学生每答对一次或者尝试一次带来的影响都是相同的，但是我们通过下面的测量结果发现，在一个知识点下学生做不同习题难度的题目带来的增益是不同的。

所以，本文在以下的分析中主要致力于解决上述这些问题，改善习题表征和学生能力表征。

3.2 问题分析

上文介绍的传统学生知识追踪模型存在两个问题：习题表征不准确和动态学生能力表征不准确。本节分析学生在线做题数据，就习题表征存在的问题分析知识点和习题答错率、习题难度等特征，就学生能力表征存在的问题分析学生做题过程。

3.2.1 数据集介绍

本章工作中主要使用：Geometry 和 Assistment12 两个公开数据集。Geometry 是 Pittsburgh Science of Learning Center DataShop 数据中心的几何数据集。Assistment12 是 ASSISTments 智能在线教育系统的学生做题记录数据。这两个数据集是传统学生知识追踪常用的数据集。本节将基于 Assistment12 公开数据集进行测量分析。2622657 条学生和系统交互数据，其中有 37259 道题目，22557 位学生，265 个知识点。

3.2.2 习题表征问题的分析

本小节对习题答错率以及知识点答错率，习题难度和习题做题人数进行了统计分析。

首先对下文统计分析中用到的一些特征符号进行说明，表 3-1 列出了本节定义的符号。

表 3-1 符号说明

Table3-1 Symbol description

符号	说明
d	习题答错率
α	知识点答错率
φ	习题难度（except 知识点）

最终的习题表征是考虑了知识点，也考虑了知识点下习题的难度。学生学习是基于知识点进行学习的，因此对于一个知识点下的题目，我们以知识点为基准，习题难度来区分同一知识点下的不同难度的习题。

(1) 答错率分析

本节基于学生做题历史分别对全部习题的答错率，一个知识点下习题的答错率，全部知识点的答错率进行了统计分析。

习题答错率的定义，我们分三步进行，首先统计某一道题目所有学生答错的数量如公式（3-5）所示：

$$fail_{(j)} = \sum_{i=1}^{s} count(answer\_state_{i,j} == 0) \tag{3-5}$$

然后统计所有学生回答习题 j 的次数，如公式（3-6）所示：

$$total_{(j)} = \sum_{i=1}^{S} count(answer\_state_{i,j}) \tag{3-6}$$

最终的习题答错率定义如公式（3-7）所示

$$d_j = fail_{(j)} \Big/ total_{(j)} \tag{3-7}$$

其中j为题目的序号，i 为学生序号，S 为学生总数量,$answer_state_{i,j}$为学生 i 在 j 这道题上的答题的状态，当为 0 的时候表示题目做错了。d_j是习题 j 的答错概率。同样的方法得到知识点的答错率。

下面分别就全部知识点下的答错率以及一个知识点下的答错率进行分析，结果如下图 3-1，图 3-2 所示：

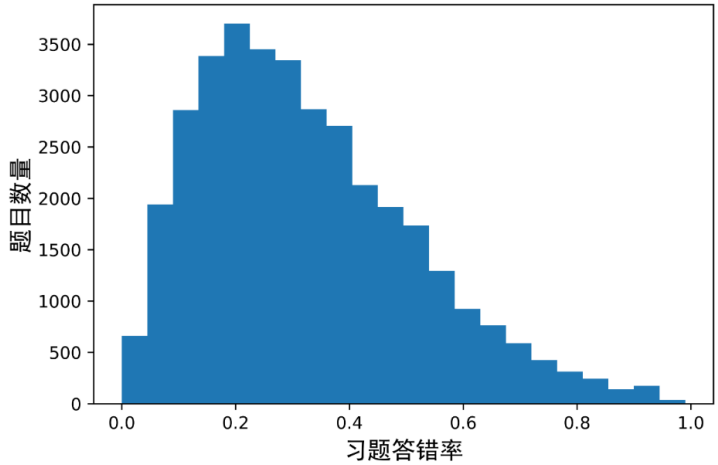

图 3-1 习题答错率的分布

Figure 3-1 The distribution of wrong rate in exercises

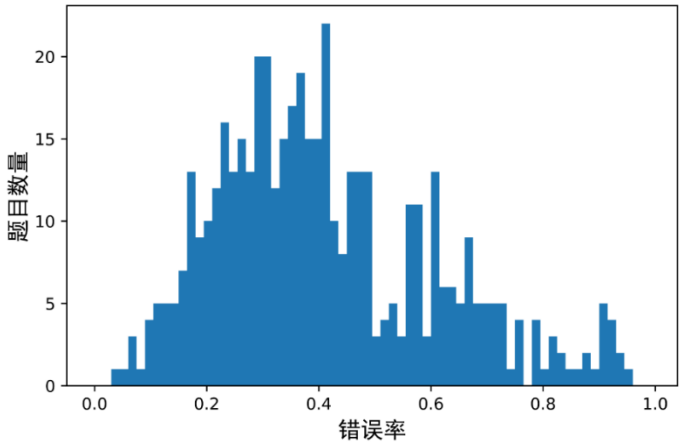

图 3-2 固定知识点下题目错误率分布

Figure 3-2 The distribution of wrong rate in a skill

图 3-1 画出了所有习题的答错率分布，横坐标是习题的答错率，纵坐标是此答错率下的习题的数量，从图 1 中可以看出学生做题错误率和题目数量关系是对数正态分布，我们利用 KS-test 进行对数正态分布和正态分布的假设检验，计算出 p-value 值，对数正态分布分布是 0.118，大于 0.05，说明此统计分布为对数正态分布。

图 3-2 画出了单个知识点下的习题答错率分布。该知识点包括 19051 条学生做题记录，一共有 481 道题目，此知识点的平均答错率为 0.37，如图 2 所示，我们可以发现即使是在同一个知识点下，答错率也具有一定的统计特征，确实在 0.37 附近出现的题目比较多，但是题目答错率极值差也很大。所以，在知识追踪时，一个

知识点下的题目也应该区分。

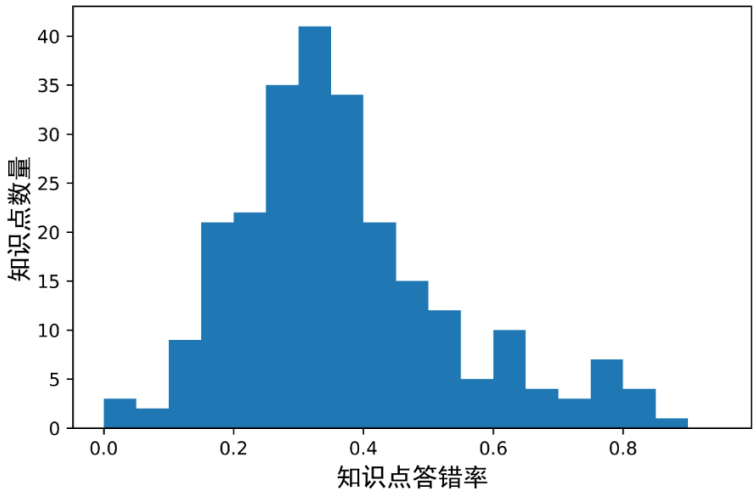

图 3-3 知识点答错率分布

Figure 3-3 The distribution of wrong rate in skills

图 3-3 画出了所有知识点的平均学生答错率分布。一共包括 265 个知识点。如图 3-3 所示，我们可以发现不同知识点答错率也具有一定的统计特征，利用 ks-test 进行对数正态分布和正态分布的假设检验，计算出 p-value 值，对数正态分布是 0.59，大于 0.05，说明此统计分布为对数正态分布分布。在 0.35 附近出现的题目比较多，但是知识点答错率的极值差也很大。所以不同知识点具有不同难度。

(2) 习题难度分析

从图 3-1 和图 3-3 测量的习题答错率和知识点答错率发现：它们都是服从对数正态分布分布。首先我们对对数正态分布进行如下介绍：

假设 x 服从对数正态分布，则 $\ln(x)$服从正态分布 $N(\mu,\sigma^2)$。那么 x 的概率密度如公式（3-8）所示：

$$f(x;\mu;\sigma)=\frac{1}{\sqrt{2\pi}x\sigma}e^{-\frac{(\ln x-\mu)^2}{2\sigma^2}} \tag{3-8}$$

其中 μ 和 σ^2 为 $\ln(x)$的均值和方差。

对于对数正态分布的数据，log 处理后是正态的，所以对数正态分布通常是几种因素相乘得到的结果，由于习题答错率为对数正态分布，为了得到每一个知识点下的习题难度分布，我们使用习题答错率除以知识点答错率然后归一化得到真正的习题难度，其中 n 为每一个题涉及到的知识点数量。具体公式如下：

首先得到习题答错率除以知识点答错率的最大值如公式（3-9）所示

$$\max\_diff=\max((\sum_{k\in KC(j)}\frac{d_j}{\alpha_k})/n) \tag{3-9}$$

得到的习题难度特征如公式（3-10）所示：

$$\varphi_j = \frac{(\sum_{k \in KC(j)} \frac{d_j}{\alpha_k}) / n}{max\_diff} \tag{3-10}$$

根据习题答错率和知识点答错率相除，归一化得到的真正的习题难度特征分布如下图所示：

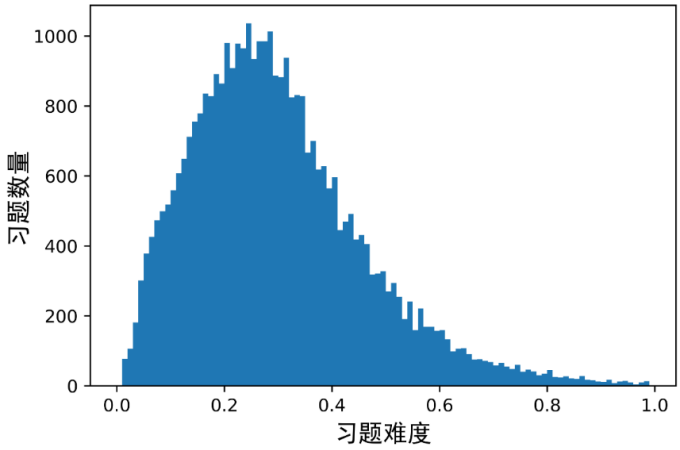

图 3-4 习题难度分布

Figure 3-4 The difficulty distribution of exercises

图 3-4 是习题难度分布，横坐标是习题难度，纵表征是此难度区间内的习题数量，发现确实也是 log-normal 分布。通过 KS-test 进行了 log-normal 分布的验证，得到 p-value 值是 0.051，大于 0.05，说明难度统计特征也是服从 log-normal 分布的。

(3) 习题做题人数分析

下面分别对全部习题的以及各知识点下习题的答错率进行了统计分析。

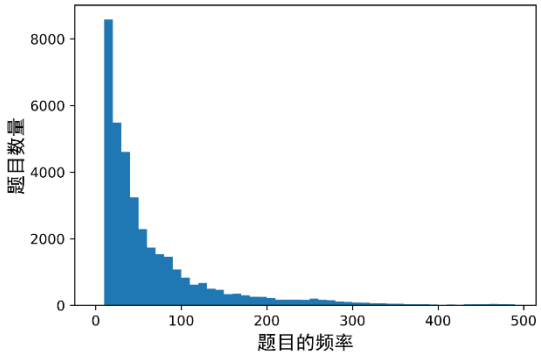

图 3-5 习题的做题次数分布

Figure 3-5 Frequency distribution of exercises

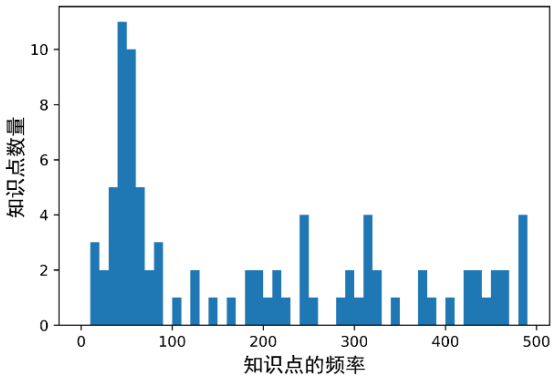

图 3-6 知识点的做题次数分布

Figure 3-6 Frequency distribution of skills

图 3-5 画出了习题的做题人数分布，横坐标是题目被学生所做的次数，纵坐标是在此次数段内的题目的数量，如图 3-5 所示，题目被学生做的频率呈指数衰减的，题目被做的次数越多，题目的数量越少。

图 3-6 画出了知识点的做题次数分布，横坐标是学生所做的习题对应的知识点被学生所做的次数，纵坐标是在此次数段内的题目数量。但是从图 3-6 可以看出知识点大部分是在 50 以上，并且知识点次数相对于题目次数均匀很多，所以习题比知识点稀疏很多，并且不均匀。

(4) 模型设计的启发

上面的分析在知识追踪的粒度选择上，给我们如下启发：

1) 一般来说稠密且均匀的数据才能得到好的拟合效果，知识点的做题记录比习题的做题记录更加稠密更加均匀，因此基于知识点难度进行拟合是更加合理的。
2) 在知识追踪的时候要得到准确的习题表征，应该考虑题目的难易度以及知识点的难易度，以提高追踪的准确率。学生学习被认为是基于知识点进行的，因为同一个知识点的题目具有相似性，但是即使在同一个知识点，学生学习也是由易到难，因此对于习题表征，同一个知识点下区分不同习题的难易度是很有必要的。

总之，同一个知识点的题目具有相似性，所以学生学习是一个知识点一个知识点下进行学习的，但知识点之间差别很大，所以我们需要知识点特征，用来区分每类知识点与其他知识点的差别。但是即使同一个知识点内的题目，难易度差别也很大，所以准确的习题表征需要知识点和题目进行融合，得到最终的题目表征。

3.2.3 学生表征问题的分析

本小节对两个学生在同一个知识点下的学习过程进行可视化，分析了不同难度的习题对于学生能力提升的影响，发现其规则，为知识追踪模型的设计提供启发。

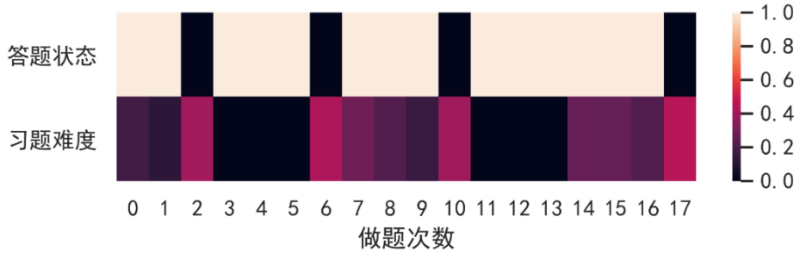

图 3-7 一号学生的学习过程

Figure 3-7 The learning process of No.1 student

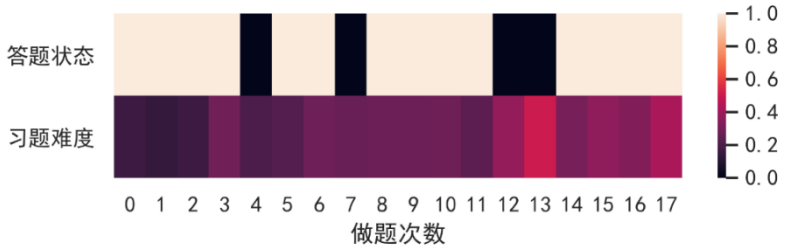

图 3-8 二号学生的学习过程

Figure 3-8 The learning process of No.2 student

图 3-7 和图 3-8 画出了两个学生在一个知识点下的学习过程，横坐标是学生的做题次数，其中 0 表示做的第一道题，最大值是 17 表示学生一共做了 18 道题。纵坐标上面第一行表示答题状态。其中答题状态分为 0 和 1，黑色就是 0 表示答错了，1 表示答对了。第二行表示习题难度，习题难度也是从 0 到 1，0 表示简单题，也即是做这道题的学生全部都答对了，1 表示难题，即是做这道题的学生全部都答错了。

图 3-7 是一号学生的学习过程，从图中我们可以看出，这个学生做了两道容易题后尝试了做一道明显更难的题，结果做错了，之后开始做特别简单的题目，答对了几次之后，尝试做一道很难的题，结果又错了。又开始回答相对简单的题目，答对了几道之后开始回答难题又错了，之后又开始做简单的题目。总之，这个学生选择的题目难度跳跃性很大，要不太难，要不太简单，做了很多题之后，还是在做基础题，最后还是做不出中等难度的题，意味着他在练习中得到的收益并不大。

图 3-8 是二号学生的学习过程，从图中我们可以看出，这个学生首先也是容易题，然后难度慢慢增长，而且难度变化不是特别大，逐渐从简单往难，最后能答对中等难度的题目。意味着他能够选择适合自己难度的题目，能力一直在提升中。

上面的观察说明：应该在知识追踪模型中考虑习题难度，以准确理解学生的做题结果，感知学生状态，得到更准确的追踪结果。这两个学生都是从简单的题目开始做，但是做题的习题难度顺序不一样，所以学生能力的增长不一样。一号学生和二号学生遇到难一些的题都会做错，但是不同的是，一号学生能答对的一直是简单题，但是二号学生做的习题难度是循序渐进的，所以锻炼了一段时间之后，可以看出来是逐渐的能力提升。学生做有难度的题，并且答对了，说明学生能力达到了一定水平，如果仅仅是一直做容易题，遇到难的就错了，那么能力并不能提高太多。

因此从上述图中说明学生学习能力的提升不仅仅和学生在这个知识点下所做题的数量，答对的题的数量以及答错的题的数量有关，也和所做的习题难度有关，所以学生知识追踪应该考虑习题难度。

3.3 改进的模型

本节基于习题难度特征对现有模型进行习题表征和学生能力表征的改进。对于习题表征，本文采用知识点难度参数和习题难度特征相乘的方式，知识点越难，并且这个知识点下的习题难度越大，那么两个值相乘就越大，学生答对的概率也就越小。对于学生能力表征，我们引入习题答错率作为习题难度进行学生能力增长的改进。

3.3.1 基于知识点进行研究的模型

本小节利用习题容易度和习题答错率分别对基于知识点进行研究的模型 AFM，PFA 进行习题表征和学生能力表征的改进。

下面分习题表征和动态学生能力表征两部分分别进行介绍。

(1) 引入习题难度特征，利用乘性模型，改进模型中的习题表征

对于习题表征，我们对 AFM 模型和 PFA 模型的改进是一样的，引入习题难度特征利用乘性模型得到最终的习题表征为$\sum_{k\in KC(j)}(-\delta_k * \varphi_j)$。

那么最终的 AFM-D 模型如下（3-13）公式所示：

$$\sigma(\sum_{k\in KC(j)}(-\delta_k * \varphi_j)+\gamma_k a_{s,k}) \quad (3\text{-}13)$$

其中δ_k为知识点 k 的难度，是需要学习的参数，φ_j为习题 j 的难度，为知识点下的习题难度特征，γ_k为知识点 k 的学习率，$a_{s,k}$为尝试次数。也即是习题越难，答对的概率越小。

那么最终的 PFA-D 模型如下（3-14）公式所示：

$$\sigma(\sum_{k\in KC(j)}(-\delta_k * \varphi_j)+\gamma_k c_{s,k}+\rho_k f_{s,k}) \quad (3\text{-}14)$$

其中δ_k为知识点 k 的难度，是需要学习的参数，φ_j为习题 j 的难度，为知识点下的习题难度特征，$c_{s,k}$为学生 s 在做 j 这道题前在知识点 k 尝试的正确的次数，γ_k是知识点 k 答对的学习率，$f_{s,k}$为学生在做 j 这道题前在知识点 k 尝试的错误的次数，ρ_k是知识点 k 答错的学习率。

(2) 引入习题答错率特征作为习题难度，改进学生能力表征

对于学生能力表征，分别就 AFM-D 和 PFA-D 进行改进。

1） 对于 AFM-D，学生能力增长仅仅考虑了学生尝试次数，由于习题难度不同，对于学生能力的增长不一样，所以我们进行以下改进。原始 AFM-D 学生能力为$\sum_{k\in KC(j)}\gamma_k a_{s,k}$，假设$a_{s,k}$等于$a$，也即是学生在做$j$这道题时，关于$k$这个知识点已经做了$a$道题目。那么学生能力就变成了$\sum_{k\in KC(j)}\gamma_k(1_1+1_2+1_3+...+1_a)$。我们添加习题答错率特征进行改进，如果此道题目包含 n 个知识点，那么我们将习题答错率拆分到 n 个知识点上，用

此道题目的答错率除以 n 得到的习题答错率作为输入，进行以下修改：$\sum_{k\in KC(j)}\gamma_k(1_1*d_{(1)}+1_2*d_{(2)}+1_3*d_{(3)}+...+1_a*d_{(a)})$。

那么最终的 AFM-DW 模型如以下（3-15）公式所示：

$$\sigma(\sum_{k\in KC(j)}(-\delta_k*\varphi_j)+\gamma_k(1_1*d_{(1)}+1_2*d_{(2)}+1_3*d_{(3)}+...+1_a*d_{(a)})) \quad (3\text{-}15)$$

其中 $d_{(j)}$表示学生做的涉及知识点 k 的第 j 到题目的习题答错率，1_j 表示学生所做的涉及知识点 k 的第 j 道习题。

2） 对于 PFA-D，学生能力增长考虑了学生答对和答错习题对于能力的增长是不一样的，因此将尝试次数拆分为答对的次数 $c_{s,k}$ 和答错的次数 $f_{s,k}$，假设 $c_{s,k}$ 等于 c, $f_{s,k}$ 等于 f 。那么 AFM-D 的学生能力为 $\sum_{k\in KC(j)}\gamma_k c_{s,k}+\rho_k f_{s,k}=\sum_{k\in KC(j)}\gamma_k(1_1+1_2+1_3+...+1_c)+\rho_k(1_1+1_2+1_3+...+1_f)$。我们添加习题答错率特征进行以下修改：$\sum_{k\in KC(j)}\gamma_k(1_1*d_{(1)}+1_2*d_{(2)}+1_3*d_{(3)}+...+1_c*d_{(c)})+\rho_k(1_1*d_{(1)}+1_2*d_{(2)}+1_3*d_{(3)}+...+1_f*d_{(f)})$。

最终的 PFA-DW 模型如以下（3-16）公式所示：

$$\sigma(\sum_{k\in KC(j)}(-\delta_k*\varphi_j)+\gamma_k(1_1*d_{(1)}+1_2*d_{(2)}+1_3*d_{(3)}+...+1_c*d_{(c)})+\rho_k(1_1*d_{(1)}+1_2*d_{(2)}+1_3*d_{(3)}+...+1_f*d_{(f)})) \quad (3\text{-}16)$$

其中 $d_{(j)}$表示学生做的涉及知识点 k 答对或者答错的第 j 到题目的习题答错率，1_j 表示学生所做的涉及知识点 k 答对或者答错的第 j 道习题。

3.3.2 基于知识和习题进行研究的模型

本小节利用习题难度和习题答错率分别对基于知识点难度和习题难度进行研究的模型 DAS3H 进行改进。我们将分习题表征和学生能力表征进行介绍两部分进行介绍

(1) 利用乘性模型改进模型中的习题表征

原始 DAS3H 模型中的习题表征是通过模型学习的知识点难度参数和习题难度参数，本文修改为知识点难度参数是进行学习得到，习题难度是通过训练集中的学生做题数据得到的习题特征。习题表征就修改为 $\sum_{k\in KC(j)}-\delta_k*\varphi_j$ ，那么最终的 DAS3H-D 模型如下（3-17）公式所示

$$\sigma(\alpha_s-\sum_{k\in KC(j)}\delta_k*\varphi_j+h_\theta(t_{s,j,1:l},y_{s,j,1:l-1})) \quad (3\text{-}17)$$

其中 α_s 为学生的初始能力，δ_k 为知识点 k 的难度，是需要学习的参数，φ_j 为习题 j 的难度，为知识点下的习题难度特征，$h_\theta(t_{s,j,1:l},y_{s,j,1:l-1})$ 为学生能力的动态变化如上述（3-4）公式所示。

(2) 引入习题答错率特征作为习题难度，改进学生能力表征

本文认为学生能力不仅和学生做的次数有关，还和所做过的题的习题答错率有关。而 DAS3H-D 模型中的学生能力特征中的学生能力动态变化仅仅考虑了每一个知识点下的每个时间窗口内的学生答对和尝试次数，因此本文引入习题答错率对公式（3-4）进行修改。

原始公式中$c_{s,j,w}$为学生 s 做习题 j 之前在时间窗口 w 内，关于知识点 k 答对的次数，$a_{s,j,w}$为学生 s 做习题 j 之前在时间窗口 w 内，关于知识点 k 尝试的次数。本文假设原始公式中的$c_{s,j,w}$的数值为 c，$a_{s,j,w}$的数值为 a，修改为如下（3-18）公式所示：

$$h_\theta(t_{s,j,1:l}, y_{s,j,1:l-1}) = \sum_{k\in KC(j)} \sum_{w=0}^{W-1} \theta_{k,2w+1} \log(1+(1_{s,1,w} * d_{(s,1,w)} + 1_{s,2,w} * d_{(s,2,w)} + 1_{s,3,w} * d_{(s,3,w)} + ... + 1_{s,c,w} * d_{(s,c,w)})) - \theta_{k,2w+2} \log(1+(1_{s,1,w} * d_{(s,1,w)} + 1_{s,2,w} * d_{(s,2,w)} + 1_{s,3,w} * d_{(s,3,w)} + ... + 1_{s,a,w} * d_{(s,a,w)})) \quad (3\text{-}18)$$

其中$d_{(s,j,w)}$表示学生所做涉及知识点 k，在时间窗口 w 内，第 j 道题目的习题答错率，$1_{s,j,w}$表示学生所做涉及知识点 k，在时间窗口 w 内，第 j 道题目。

得到的最终的 DAS3H-DW 模型如下（3-19）公式所示：

$$P(Y_{s,j}=1) = \sigma(\alpha_s - \sum_{k\in KC(j)} \delta_k * \varphi_j + h_\theta(t_{s,j,1:l}, y_{s,j,1:l-1})) \quad (3\text{-}19)$$

其中h_θ为上述（3-18）公式所示。

3.4　实验验证

本节主要是介绍实验以及结果评估。主要包括实验的参数处理，实验结果的比较和分析。通过实验，我们通过习题表征和学生能力表征结果实验数据，验证了习题难度特征的有效性。

3.4.1　数据预处理

本章进行实验的数据采用的是两个数据集：Geometry 和 Assistment12 两个公开数据集。Geometry 是在 Pittsburgh Science of Learning Center DataShop 数据中心的几何数据集。Assistment12 是 ASSISTments 智能在线教育系统的学生做题记录数据。这两个数据集都是传统学生知识追踪常用的数据集。数据集中包括学生 ID，习题 ID，知识点，学生答题对错(0, 1)，学生做题开始时间等字段信息。

结合我们具体的使用场景，为了追踪的准确性，我们参考文献[52]，保证每个学生的做题路径足够，对学生数据进行了过滤，将学生做题记录少于 10 次的过滤掉。

3.4.2 实验细节

模型是通过 Q-Matrix 矩阵实现各个特征的参数学习。以改进的 DAS3H 模型举例，用户特征是 one-hot 编码，如果一共 3 个用户，那么 one-hot 编码就是三维，说明模型的用户参数α_s就是一个三维的向量，值为 1 的位置，就是用户的编号，其余都是 0。知识点，首先是以 one-hot 编码的格式，如果也是 3 维，那么就说明数据集涉及到的知识点一共就 3 个，但是由于我们添加了统计的习题难度特征，所以值为 1 的位置，我们赋值为此记录的习题难度。同时答对和尝试也是基于知识点进行统计的，所以也是三维，但是由于添加了 5 个时间窗口，每一个知识点对应 5 维，所以答对和尝试每一个是 15 维的向量，每一维度输入是在此时间窗口内涉及该知识点的题目的次数，但是由于我们添加了答错率特征，所以需要每一次记录都是 1*diff(记录的题目答错率)，因此题目次数就赋值为对应记录的难度累加。所以对应 3 个用户，3 个知识点的数据集，输入的维度是 3+3+3*5*2=36。如果三次记录，那么构造的就是 3*36 维的 Q-Matrix。

针对每一组实验，我们将学生做题记录数据集按照 0.8：0.2 的比例进行训练集和测试集的划分。其中习题难度特征和习题答错率特征，根据训练集的数据进行统计处理得到，在测试集上沿用训练集得到的统计标签。训练集进行模型参数的学习，和超参数调参。以 AUC 和 NLL 为模型的评价指标，测试集评估模型的性能。通过添加习题难度特征和习题答错率进行模型的改进。其中 D 为加入习题难度特征利用乘性模型改进习题表征，W 为加入习题答错率作为习题难度改进学生能力表征。测试结果如下表 3-2 和表 3-3 所示。

表 3-2 几何数据集不同模型评估

Table3-2 Evaluation of Different Model on Geometry

模型	AUC	NLL
AFM	0.652	0.575
AFM-D	0.687	0.558
AFM-DW	0.693	0.556
PFA	0.676	0.569
PFA-D	0.712	0.552
PFA-DW	0.724	0.544
DAS3H	0.751	0.448
DAS3H-D	0.772	0.438
DAS3H-DW	0.773	0.437

表 3-3 Assistment12 数据集不同模型评估
Table3-3 Evaluation of Different Model on Assistment12

模型	AUC	NLL
AFM	0.611	0.604
AFM-D	0.624	0.601
AFM-DW	0.680	0.584
PFA	0.677	0.583
PFA-D	0.712	0.570
PFA-DW	0.744	0.545
DAS3H	0.773	0.51
DAS3H-D	0.780	0.495
DAS3H-DW	0.782	0.484

3.4.3 实验结果

对于本章的测量和模型改进，我们主要关注以下两个问题：

(1) 通过训练集统计得到的习题难度特征，结合模型学习到的知识点难度参数，以乘性处理的方式进行题目表征是否能提升学生知识追踪模型的预测效果。

我们将基于知识点难度参数和习题难度特征进行乘性处理得到的习题表征用于现有模型中并与现有模型性能进行比较。在选定的现有模型中，AFM 和 PFA 是基于知识点进行研究的经典算法，DAS3H 是习题难度特征和知识点容易度进行加性处理的目前效果最好的传统知识追踪模型。表 3-2 和表 3-3 中为模型在两个数据集上的测试指标效果，包括 AUC 和 NLL：

1) 表 3-2 和表 3-3 是两个数据集在模型上的测试结果，我们可以看出在几何数据集上，AFM-D 的 AUC 为 0.687，NLL 为 0.575，相对于 AFM 的 AUC 为 0.652，NLL 为 0.558，AUC 和 NLL 指标效果分别提升了 3.5%和 1.7%，同理在 Assistment12 数据集上 AUC 指标和 NLL 指标效果分别提升了 1.3%和 0.3%。在几何数据集上，PFA-D 相对于 PFA，AUC 指标和 NLL 指标效果分别提升了 3.6%和 1.7%，在 Assistment12 数据集上 AUC 指标和 NLL 指标效果分别提升了 3.5%和 1.3%。这意味着：对于习题表征，引入习题容易度基于乘性模型进行习题表征，相比于只用知识点进行习题表征，表征效果更好，模型预测准确度更高。
2) 对于 DAS3H 模型，知识点容易度和习题难度加性处理进行习题表征的模型，我们引入习题难度特征，与知识点难度参数进行乘性处理得到习

题表征，在几何数据集上 DAS3H-D 相对于 DAS3H，AUC 指标和 NLL 指标效果分别提升了 2.1%和 1.0%。在 Assistment12 数据集上 DAS3H-D 相对于 DAS3H，AUC 指标和 NLL 指标效果分别提升了 0.7%和 1.5%。这意味着：乘性模型进行习题表征相比于加性模型，表征效果更好，模型预测准确度更高。

因此基于训练集得到习题难度特征和模型学习到的知识点难度以乘性处理的方式结合得到的习题表征可以提高模型的预测效果

(2) 基于训练集统计得到的习题答错率作为习题难度对学生能力表征进行改进，是否能提升学生知识追踪模型的预测效果。

将习题答错率引入到学生能力表征中进行表征的改进，在改进了习题表征的 AFM-D, PFA-D 和 DAS3H-D 模型上进行了实验，在两个数据集上的测试结果如表 3-2 和 3-3 所示。

从几何数据集上我们可以看出，AFM-DW, PFA-DW 和 DAS3H-DW 相比于 AFM-D, PFA-D 和 DAS3H-D 模型，AUC 指标效果分别提高了 0.6%，1.2%和 0.1%。NLL 指标效果分别提升了 0.2%，0.8%和 0.1%。

从 Assistment12 数据集上我们可以看出，AFM-DW, PFA-DW 和 DAS3H-DW 相比于 AFM-D, PFA-D 和 DAS3H-D 模型，AUC 指标效果分别提高了 5.6%，3.2%和 0.2%。NLL 指标效果分别提升了 1.7%，2.5%和 1.1%。

因此可以说明习题答错率可以有效的提升学生能力表征，进而提升模型的预测效果。

3.5 本章小结

本章首先基于学生在导学系统中的做题日志进行分析，分别就知识点粒度和习题粒度分析答错率分布和答题人数分布，以及习题难度和学生答题过程。然后基于分析结果，结合学生知识追踪模型在习题表征和学生能力表征方面存在的问题进行改进，基于改进的模型进行实验验证。本章完成了如下工作：

(1) 对本文研究的传统知识追踪模型依据习题的表征方式进行分类，然后基于模型存在的问题进行分析，分成了两类：

1) 对习题是通过其所属的知识点容易度进行的表征，包括（AFM,PFA）

2) 对习题是通过其所属的知识点容易度和习题难度进行的表征，DAS3H

这两类模型对于习题表征和学生能力表征都不够准确。

(2) 分别在知识点和习题粒度上进行分析，得出如下发现：

1) 不同知识点答错率极值差很大，一个知识点下的题目答错率极值差也很

大，所以为了准确的习题表征，应该考虑知识点和习题相结合。

2) 分别在习题和知识点粒度上进行分析，发现知识点答错率和习题答错率分布都服从对数正态分布，根据分布特点，用习题答错率除以知识点答错率，得到习题难度特征。

3) 分别进行习题和知识点做题人数的统计，发现习题是随着做题人数的增加快速衰减，数据不仅稀疏而且不均匀，但是知识点相对均匀而稠密。

(3) 通过对不同学生在同一个知识点上的题目作答情况分布，发现学生的能力增长情况和习题答错率有关系。

(4) 基于上述(2)和(3)的测量结果，对(1)中的模型进行改进，提出基于统计特征进行改进的学生知识追踪模型具体包括以下工作：

1) 对于习题表征，基于统计的习题难度特征和模型学习的知识点难度参数利用乘性模型得到习题表征。

2) 对于学生能力表征，将统计的习题答错率作为习题难度特征引入模型，进行学生能力表征的改进。

(5) 最后通过进行实验验证，统计得到的习题难度特征和模型学习到的知识点难度进行乘性模型处理可以改进习题表征，AUC 指标相比于原始模型最高提升了 3.6%。基于习题答错率改进学生能力表征，AUC 指标最高提升 5.6%。

4 个性化的深度学生知识追踪模型

本章介绍本文提出的基于学生个性化的深度学生知识追踪模型，包括动态学生个性化和长期学生个性化两个部分。对于动态学生个性化，本章在现有个性化模型的基础上，提出基于习题难度特征改进学生学习能力编码方式和基于学生前一时间单元做题记录改进学生学习能力编码方式。对于长期个性化，本章通过分析数据，提出班级类型特征作为学生长期个性化特征。最后提出长期个性化特征和动态个性化结合进行学生个性化，通过实验验证。

4.1 基本思路

本章主要研究基于个性化的深度学生知识追踪模型。个性化是智能教育的关键，准确了解到学生的学习能力，才能在学生做完一道题目之后得到学生目前所处的知识状态，也即是准确的学生能力表征，进而才能准确地进行预测学生答对题目的概率。DKT 模型认为所有学生的学习能力是一样的，只跟踪学生的做题技能序列，每一个学生都是一个样本，没有考虑学生的个性化差异。

我们基于个性化进行研究，提出将个性化分为长期个性化和动态个性化。长期的个性化如班级类型差异，这是由于基础不同和教学模式不同导致的学生长期的差异。动态的个性化如动态变化的学习能力，因为学生擅长的知识点可能不一样，所以对于某些学生，当遇到自己擅长得知识点上表现很好，当遇到自己不擅长而且不感兴趣的知识点上表现可能会变差，所以他们的学习能力是动态变化的。

我们首先介绍现有的动态个性化研究，动态地基于学生学习能力进行编码，然后基于编码结果对学生进行分类实现个性化。然后分析数据，基于班级类型提出长期个性化，最后通过实验验证效果，具体步骤如下：

(1) 我们分析现有编码方式以及编码结果，发现问题，提出基于习题难度特征改进学生学习能力编码方式。

(2) 分析学生学习能力编码结果以及编码包含的时间单元，对所包含的时间单元进行修改，使学生学习能力编码显示学生即时学习能力，而不是累加学习能力，改进学生学习能力编码。

(3) 测量数据，发现不同班级类型特征对学生答题结果有影响，基于此特征进行长期个性化的研究。

(4) 最后通过实验进行验证。

4.2 数据集介绍

本章利用一个智能在线教育系统的五年级学生答题记录数据集。此数据集包含数千位学生在近千道习题上的做题数据，学生身份进行了匿名化处理。习题涉及98 个知识点。每一条交互数据包含习题文本，习题难度（1-6 一共 6 个等级），习题 ID，学生答题对错（0, 1），学生班级类型，学生所在区域，学期 ID，科目等字段。我们基于习题难度特征和班级类型进行学生个性化的改进。

4.3 学生学习能力分类

本小节研究基于学生做题记录对学生进行学习能力分类的方法。首先介绍现有动态学生学习能力编码方法，分析不足，然后提出考虑习题难度改进现有学生编码方法和基于学生前一时间单元做题记录改进学生编码方式。

4.3.1 现有动态学生学习能力编码

目前有研究者提出基于学生历史答题记录动态地对学生进行学习能力编码[45]，然后根据编码得到的学习能力向量进行聚类，将学生根据聚类的结果进行分组，进行知识追踪，使知识追踪模型能够根据学生的特点追踪到更准确的学生知识状态，实现个性化。他们认为学生学习能力不一样并且是动态变化的，根据学生做题历史得到学生学习能力，然后根据学生学习能力进行聚类，将学生进行分组，实现了动态的个性化。

动态个性化思想是学生学习能力是不断变化的，并且不同学生之间存在差别。模型根据学生在系统中的不同表现，将学习能力相似的学生分到一个小组。而大多数学生的学习能力是会变化的，所以模型分时间单元对学生做题历史进行编码。得到所有编码向量之后通过 K-means 聚类的方式对编码进行分类，动态评估学生学习能力所属的小组。下面我们分时间单元，学生尝试次数的分段，学生能力的编码进行介绍：

(1) 对于时间单元，将学生在系统里面的一个答题记录为一个时间单位，也即是学生做一道题为一个时间单位，那么时间单元就是学生做的连续的几个题目。如果时间单元为 k，就是学生每做 k 道题为一个时间单元，如果学生做了 2*k 道题，就是 2 个时间单元。如下图 4-1 所示，假设时间单元为 10，学生做了 30 道题目，就是 3 个时间单元。

图 4-1 学生尝试次数单元划分

Figure4-1 Time Unit of students' attempts

(2) 对于学生的做题记录进行分段，在每个时间单元的最后添加上此段时间学生的做题记录对学生的学习能力进行重新评估，目的是捕捉学生的学习能力变化。在下一个时间单元内将学生按照此次的评估结果进行分组。

(3) 对学生学习能力进行编码，基于学生做题历史，在每一个时间单元最后将学生的能力编码为知识点长度的向量，并随着学生做题过程进行更新。编码向量的长度为知识点长度，向量的每一维度的值是，学生在历史时间单元内对相应知识点尝试的答对率和答错率的一个差值。答对率和答错率的分别定义为如下公式（4-1）和公式（4-2）所示：

$$Correct(x_j)_{1:z}=\sum_{t=1}^{z}\frac{count(x_{jt}==1)}{\left|N_{jt}\right|} \tag{4-1}$$

$$Incorrect(x_j)_{1:z}=\sum_{t=1}^{z}\frac{count(x_{jt}==0)}{\left|N_{jt}\right|} \tag{4-2}$$

向量的第j维度的值如下公式（4-3）所示：

$$R(x_j)_{1:z}=Correct(x_j)_{1:z}-Incorrect(x_j)_{1:z} \tag{4-3}$$

最终学生i在时间单元 Z 的学习能力向量如公式（4-4）所示：

$$R^{i}_{1:z}=(R(x_1)_{1:z},R(x_2)_{1:z},...,R(x_n)_{1:z}) \tag{4-4}$$

其中 t 为做题间隔，Z 当前时间单元，N_{jt} 为在时间单元 1 到时间单元 Z 内，学生回答知识点j的次数，i 为学生序号。

根据得到的学生学习能力向量此向量利用 K 均值聚类方法进行聚类，根据聚类结果对学生进行分组。

4.3.2 问题分析

本小节基于现有学生学习能力编码的聚类结果进行如下分析。(1)分析习题难度是否对学生聚类有影响。(2)分析学生学习能力的变化情况，以设计能够得到学生即时学习能力的编码方法。

我们首先设置学生学习能力的时间单元。通过分析数据集，发现每一个课次设定的题目有 20 道，也即是学生需要做的题目有 20 道，并且课次的先后顺序对应知识点存在递进关系，即：下一个课次的知识点，以上一个课次的知识点为基础。

因此我们将学生学习过程的时间单元设定为 20，也即是以 20 道题为一个单位，重新计算学生的学习能力向量。

基于获得的学生学习能力向量，我们对学生进行聚类。然后将新的学生学习能力聚类结果，作为后面的学生能力追踪模型的输入。对于 K-means 聚类的类别数目选择，使用 gap-statistics 方法寻找最优的 k 值为 7，也即是所有学生被分成了 7 种类型。

下面我们具体分析聚类的结果。因为有 98 个知识点，所以学生的学习能力向量是 98 维，各聚类的中心点也为 98 维，为了比较不同类别学生的差别，我们将 7 个类别的中心点向量进行了可视化，如下图 4-2 所示：

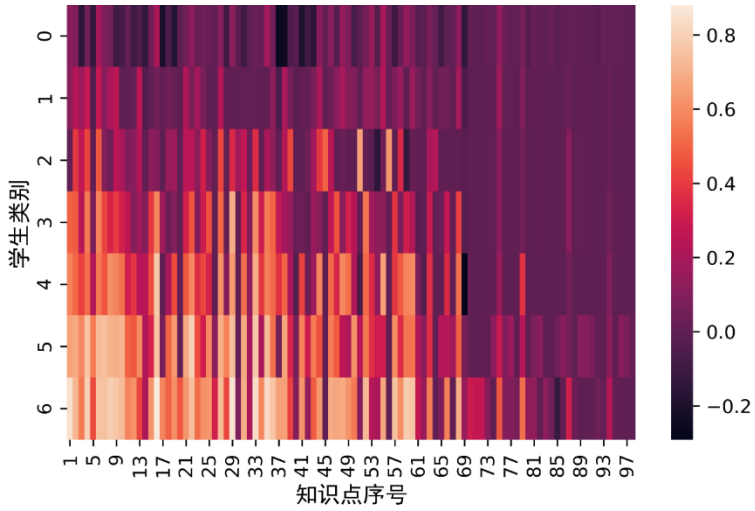

图 4-2 七类学生的学习能力向量

Figure 4-2 Learning capacity vectors of 7 types of students

图 4-2 画出七类学生的学习能力中心点向量，我们对每一类的中心点向量的 98 个维度值求均值，然后对均值按照从小到大的顺序排列依此为上图 4-2 中的 0-6 顺序，0 表示这一类学生的平均学习能力最差，6 表示这一类学生的平均学习能力最强。每一行表示一类学生的学习能力向量。横坐标为知识点序号所对应的维度。每一列对应 7 类学生在特定知识点上的学习能力。

基于图 4-2，我们有如下观察：平均学习能力强的学生，在各个知识点上的能力都强。具体来说，如图 4-2 所示，比较不同类型学生的结果可以发现：如果该类学生的能力均值比另一类的大（比如第 6 类的均值比第 0 类的大），那么该类学生在每一维上的能力通常也大于另一类学生对应维度的值。

我们基于学生做题记录，分析了产生这个现象的原因，发现：

(1) 同一个类别中的学生具有相似的学习能力，比如一部分学生在当前时间单元内学习能力极强，因此，他们被分为了一类；

(2) 一个时间单元内的习题，一般来说，关联性较强。因此，一类学生，在当前时间段内所做习题涉及的知识点上都有相似的表现。

所以表现好的这一类学生在习题涉及的知识点对应的维度上的值都要大于其他类别的学生。

以上聚类结果给我们如下启发：

(1) 通过基于时间单元的学生做题结果的学生能力聚类，能分辨出不同的学习能力的学生。

(2) 学生学习能力的准确定义非常重要，而目前存在的学生学习能力编码方法存在如下不足：

1) 学生学习能力向量的定义过程中没有考虑到习题难度。现有的方法认为：同一个知识点下的习题，不论难度是否相同，每次给学生带来的能力提高是一样的，这就导致不同学生做不同难度的习题，由于答对和答错的数量在总数量中的比例相似，那么他们将会被聚到一个类别中。但是依据第三章的测量结果我们了解到不同习题难度给学生能力带来的增益是不一样的。增益不一样那么学生学习能力的定义就不准确，就会导致学生聚类的结果有变化。

2) 累积型学生学习能力编码在实际中存在重大问题。现有的动态学生能力编码，是在每个时间单元最后对学生的历史累加进行编码，但是，如上面的分析所示，学生的能力编码单元的选择，对学生的分类效果，有着非常重要的影响，具体来说：

① 如果单位选得太长，学生的做题记录累加太多之后，此种编码方式能够反映的学生能力的变化就不明显。如图 4-2 所示，当学生在很多知识点上表现好的时候，即使碰到某些不擅长的知识点，之前的累积记录也会使学生被分到好学生类别中。此时，对于学生学习能力的变化，就可能捕捉不到。

② 对于学生知识追踪模型来说，当我们要预测学生做下一道题的表现时，最重要的是得到学生在这道题目相关的知识点上的能力类型。既然，根据学生做题序号和答题状态已经学习到学生目前所处的知识状态，那么，学生能力聚类应该表示的是当前习题相关的学生学习能力，而且这种学生学习能力应该是即时的。

③ 我们通过分析学生数据发现一个学生在连续的几个特定的时间单元内他完成的题目所涉及的知识点基本是具有相似性或具有依赖关系的。所以遇到的可能是他擅长的知识点（数论）以及递进或者相似知识点所对应的习题，但是在另外一个特定的时间单元内可能是他不擅长的知识点（几何）

所对应的习题。那么这个学生在这数论相关的时间单元内和几何相关的事件单元内将会反映出不同的能力，因此应该被划分到不同的类别中去，在涉及数论以及相关知识点的时间单元内被化分为能力较强的学生所对应的类别，但在另一个时间单元内应该会被划分为能力较弱的学生所对应的类别。这种变化应该是即时的。所以我们考虑用学生的短时学习能力，也即是在时间单元最后用的是当前时间段的学生记录，而不是累加学生所学得到的学生学习能力。

4.3.3 改进学生学习能力编码

基于上节的分析结果，本小节提出学生能力编码的解决方案。分两个方面：(1)基于时间单元的动态学生学习能力编码；(2)引入习题难度特征，改进学生学习能力编码。

(1) 改变编码时间单元，改进编码方式

本小节修改学生在编码时候所选取的学生记录时间段。如图 4-3 和 4-4 所示。4-3 为原始编码选取的时间单元，原始学生能力编码选取此时间单元最后时刻之前的所有记录进行学生学习能力的编码。但是由于长时间累积，捕捉不到学生的即时学习能力变化。所以我们修改为在时间单元的最后，如图 4-4 所示选取一个时间单元的学生答题记录进行学生学习能力的编码。

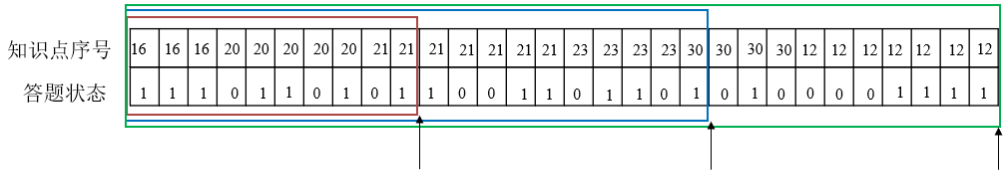

图 4-3 原始时间单元的选取方法

Figure 4-3 Original selection of time unit

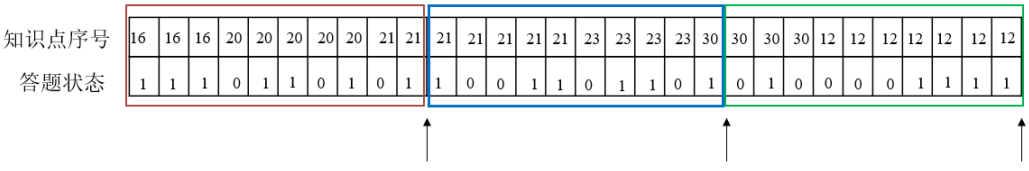

图 4-4 改进的时间单元的选取方法

Figure 4-4 Improved selection method of time unit

下面我们以公式形式进行详细说明。修改学生记录时间单元进行改进，习题答错率和答对率如下（4-5）和（4-6）所示：

$$Correct(x_j)_z = \frac{count(x_{j,z} == 1)}{N_{j,z}} \quad (4\text{-}5)$$

$$Incorrect(x_j)_z = \frac{count(x_{j,z} == 0)}{N_{j,z}} \quad (4\text{-}6)$$

其中 $N_{j,z}$ 为在时间单元 Z 内，学生回答知识点 j 下的习题的数量。$x_{j,z}$ 为在时间单元 Z，学生回答知识点 j 的习题状态。

那么向量的第j维度的值如下公式（4-7）所示：

$$R(x_j)_z = Correct(x_j)_z - Incorrect(x_j)_z \quad (4\text{-}7)$$

那么如果题库里面一共有 n 个知识点，在 z 时刻学生的学习能力可以表达为如公式（4-8）所示：

$$R_z^i = (R(x_1)_z, R(x_2)_z, ..., R(x_n)_z) \quad (4\text{-}8)$$

其中 i 为学生序号，学生学习能力最终表示为，时间单元 z 内计算得到的每一个知识点上答对率减去答错率得到的值组成的向量。

(2) 引入习题难度特征，改进编码方式

本小节引入习题难度特征，就学生学习能力的编码方式进行改进。因为学生能力不仅和学生在知识点下答对的次数和答错的次数有关，还和做的习题的难度有关。就同一个知识点来说，如果两个学生做同样数量的题，并且答对的数量也一样，但是一个学生回答对的都是难题，另一个学生回答对的都是简单题，那么第一个学生的要比第二个学生的能力强。具体改进方式我们以公式形式进行详细说明。添加习题难度特征加权的答对率和答错率定义如下公式（4-9）和（4-10）所示：

$$Correct\_d(x_j)_{1:z} = \sum_{t=1}^{Z} \frac{\sum_{d=1}^{m} d/m * count(x_{j,d,t} == 1)}{\left|N_{j,t}\right|} \quad (4\text{-}9)$$

$$Incorrect\_d(x_j)_{1:z} = \sum_{t=1}^{Z} \frac{\sum_{d=1}^{m} d/m * count(x_{j,d,t} == 0)}{\left|N_{j,t}\right|} \quad (4\text{-}10)$$

其中 d 为难度水平，m 为最高难度，Z 为当前时间单元，$N_{j,t}$ 为在间隔 1 到 Z 内，知识点 j 下题目回答的次数。$x_{j,d,z}$ 为在 1 到 Z 时间单元，学生回答知识点 j 下习题难度为 d 等级的习题状态。

那么向量的第j维度的值如下公式（4-11）所示：

$$R(x_j)_{1:z} = Correct\_d(x_j)_{1:z} - Incorrect\_d(x_j)_{1:z} \quad (4\text{-}11)$$

那么如果题库里面一共有 n 个知识点，在 z 时刻学生的学习能力可以表达为如公式（4-12）所示：

$$R_{1:z}^i = (R(x_1)_{1:z}, R(x_2)_{1:z}, ..., R(x_n)_{1:z}) \quad (4\text{-}12)$$

其中 i 为学生序号，学生学习能力最终表示为由每一个知识点上的答对率和答错率组成的向量。

4.4 班级类型测量

我们数据涉及到的班级一共 10 个。包括超常班，尖子班，启航班，创新班，提高班等 10 个班级类型。因为学生的基础能力不一样，经过班级的划分，教学模型也不一样，所以最终学生的学习能力也不一样，这是由班级导致的学生的长期的不同，所以我们称之为长期的个性化差异。

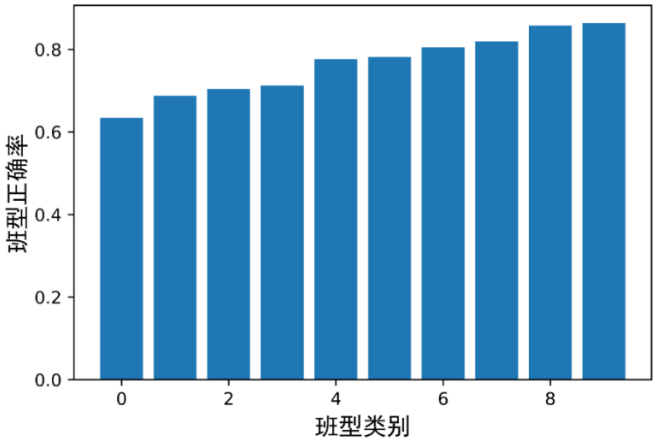

图 4-5 各个班级类型正确率

Figure 4-5 Correct rate of each class type

如图 4-5 所示，我们统计了数据集中 10 个班级类型的答对率。图中我们可以看出最小的答对率在 0.6 左右，最大的答对率接近 0.9。其中答对率最大的为超常班，答错率最小的为启航班。所以答错率差别很大，说明班级类型确实能在一定程度上反应出学生的学习能力。因此我们根据班级类别对学生进行划分，一个班级类型的学生属于一组。然后作为特征送到追踪模型。我们就一个班级类型所包含的各个小组的学生数量进行分析，如下图 4-6 所示。

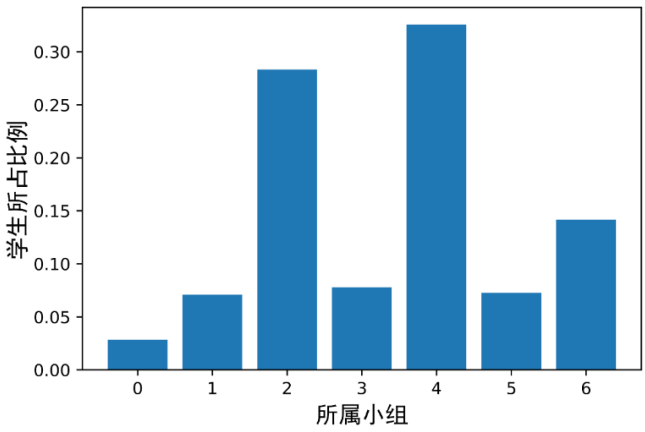

图 4-6 一个班级类型下的每个小组的学生所占的比例

Figure 4-6 The proportion of students in each group of the class type

如上图 4-6 所示为一个班级类型下的每个小组的学生所占的比例。此班级类型

为表现较差的启航班。其中时间单元为第一个时间单元，我们选出第一个时间单元的学生，并将他们根据 K 均值聚类的结果进行分组。其中类别 0 为表现最差的学生小组，类别 6 为表现最好的学生小组。其余的都是中间小组。我们可以看出即使是表现最差的起航班，属于学习能力最差的小组的也是最少的。并且此班级的学生属于每一个小组的都有。

因此，我们提出班级类型应该作为长期的个性化特征改进模型的效果。

4.5 模型结构

本小节介绍基于 LSTM 结构的个性化深度学生知识追踪模型，如下图 4-7 所示。

我们基于模型的输入进行改进：(1)添加了长期的个性化特征，班级类型，为下图 4-7 蓝色虚线框圈出部分。(2)基于习题难度和时间单元改进学生学习能力向量，为下图 4-7 所示红色虚线框圈出部分。然后将学生能力向量基于 K 均值聚类算法进行聚类，依据聚类结果对学生进行分组，将分组结果作为输入。

其中模型的输入为 one-hot 编码的拼接组成。输入包含习题对于输入特征主要有以下三部分，对于不同的模型，特征的组合不同。其中图 4-7 中的 x 就是特征的组合结果。

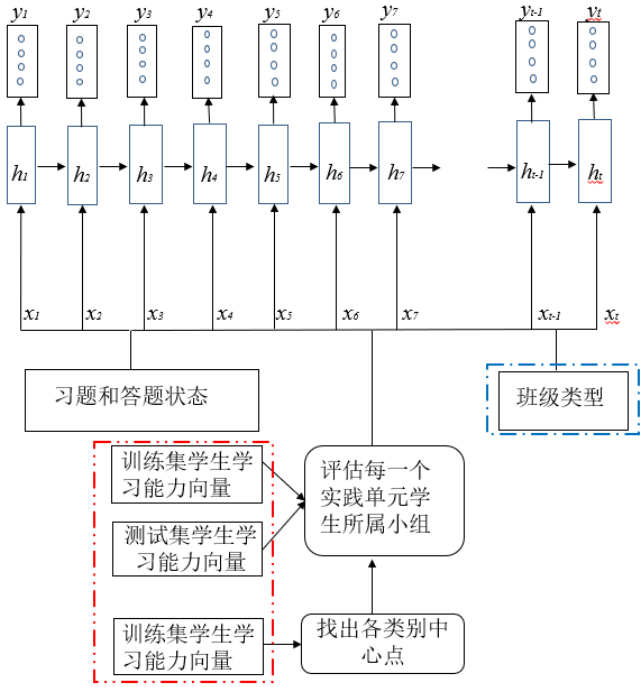

图 4-7 模型结构

Figure 4-7 Illustration of the model

下面我们将详细介绍三部分的 one-hot 编码：

(1) 学生所做的题目编号和做题结果组合进行 one-hot 编码，具体来说习题 1024 道，学生答题状态是对还是错（0 或者 1），那么为了表示学生做的是哪道题目，并且是答对还是答错，one-hot 编码就是一个 2048 维的向量，其中前 1024 维为学生在题目答错的时候为 1，后 1024 维为学生在题目答对的时候为 1。具体来说就是如果学生题目编号为 0 的题目答错了，那么 2048 维的向量为第 0 维为 1，其他全部是 0，如果学生题目编号为 0 的题目答对了，那么 2018 维的向量为第 0+1024=1024 维度为 1，其他全部为 0。

(2) 得到学生学生能力编码向量，然后依据此向量聚类得到学生所属小组信息进行 one-hot 编码。具体来说学生在第一个时间单元内，初始类别我们设置为类别 0，所有学生的初始类别都一样。然后依据 gap-statistic 算法寻找到最优的 k 值，设定 one-hot 编码的维度。k 值为 7，那么总的 one-hot 编码维度就是 8。在 8 维度的向量上根据学生所属的类别将向量的对应位置设置为 1，其他设置为 0。

(3) 学生的长期个性化特征班级类型的 one-hot 编码。具体来说就是班级类型有 10 种，one-hot 编码就是一个 10 维度的向量。如果学生所在的班级为编号为 0 的班级，那么就是第 0 维度为 1，其他维度都是 0。

基于长短期记忆网络 LSTM 进行学生状态的学习。h 为学生某一时刻所处的知识掌握程度隐状态。因为 LSTM 能根据学生的历史记录和此刻的输入进行遗忘和学习，所以 h 就是 LSTM 模型基于此刻的输入和历史得到的学生做完此刻的题目之后的知识隐状态。

输出是基于学生此刻的隐性状态预测的下一时刻每一个题目学生每一道题目答对的概率。输出的维度就是题目的总数量，其中维度对应一道题目的答对概率。

4.6 实验验证

本小节主要通过实验验证提出方法的有效性。分模型训练和实验结果两方面进行介绍。

4.6.1 模型训练

模型使用的标签是下一时刻学生答题的状态（0 或者 1）。但是由于输入没有提供下一时刻的题目编号，所以模型并不知道下一时刻学生答对的是哪道题目，因此模型的输出定义为每一道题目答对概率的向量。所以训练的时候根据提供的下一时刻的题目编号选出模型预测的下一时刻的这道题目的答对概率，然后和模型

标签一起计算损失函数。由于是两状态 0 和 1，所以我们把这个问题当成一个分类问题，使用的目标函数为二元交叉熵损失定义如公式（4-13）所示：

$$loss = -(\sum_t (y_t \log(p_t) + (1 - y_t) \log(1 - p_t))) \quad (4\text{-}13)$$

其中 y_t 为模型在 t 时刻的预测值， p_t 为 t+1 时刻所作题目的答题状态（0 或者 1），为 t 时刻的标签。

采用 Adam 优化器进行模型的优化，梯度剪裁方法避免长短期记忆模型结构梯度爆炸，梯度剪裁阈值设置为 4，也即是模型参数的 L2 范数大于 4 的设置为 4。利用 dropout 方法避免模型过拟合。

4.6.2 模型结果

本小节通过分析学生知识追踪模型呈现的效果，验证改进学生个性化的作用，包括长期个性化和短期动态个性化。

DKT 和 IDKT 是我们的基线模型。DKT 模型为没有添加学生个性化特征。个性化的深度知识追踪(IDKT, Individual Deep Knowledge Tracing)是根据学生做题记录中答对和答错的数量进行的学生学习能力的编码，将能力编码聚类得到的学生分组的结果作为特征，添加了动态个性化特征。IDKT-seg 是在 IDKT 的基础上修改学生编码的历史记录时间段，修改了学生学习能力编码。添加习题难度的个性化的深度知识追踪 DIDKT 是在 IDKT 学生能力的编码的基础上，添加了习题难度特征改进了学生学习能力编码，然后将能力编码聚类得到的学生分组结果作为特征。DKT-CL 添加了学生班级类型作为特征，进行了学生长期个性化。DIDKT-CL 是将学生的长期个性化特征和短期个性化特征结合进行的学生个性化改进。针对每一组实验，我们将学生做题记录数据集按照 0.8：0.2 的比例进行训练集和测试集的划分，对于每一组实验，我们取十次结果的平均值。实验结果如下表 4-1 所示。

表 4-1 实验结果对比

Table 4-1 Performance of different models

模型	AUC
DKT	0.843
DKT-CL	**0.847**
IDKT	0.847
IDKT-seg	0.849
DIDKT	**0.850**
DIDKT-CL	**0.854**

从表 4-1 我们可以看出：

(1) DKT-CL 为添加了班级类型作为长期的个性化特征，相比于原始 DKT，AUC 指标提高了 0.4%。说明我们添加长期的个性化对于模型效果有一定的改进。

(2) IDKT 添加现有的动态个性化特征，效果相比于原始 DKT，AUC 指标提高了 0.4%。IDKT-seg 在 IDKT 的基础上修改学生能力编码单元，效果相比于 IDKT，AUC 指标提高了 0.2%。说明修改编码单元对学生学习能力有一定提升。DIDKT 通过添加习题难度特征改进现有的动态个性化特征，相比于 IDKT，AUC 指标提高了 0.3%，相比于 IDKT-seg，AUC 指标提高了 0.1%，说明我们添加习题难度对于模型的预测是有一定的效果提升的。

(3) DIDKT-CL 是长期个性化特征和效果最好的动态个性化特征进行结合的个性化方法。比 DIDKT 和 DKT-CL 的 AUC 指标分别高出 0.4%和 0.7%。说明长期个性化和动态个性化结合的个性化方法对于模型有一定的提升。

4.7 本章小结

本章主要研究基于个性化的深度学生知识追踪模型。首先基于现有动态学生学习能力编码进行介绍，然后分析根据编码向量得到的聚类结果，发现问题，提出基于习题难度改进学生学习能力编码和基于编码时间单元改进学生学习能力编码的方法。然后通过分析班级类型，将班级类型特征作为学生长期个性化特征。最后通过实验验证。

添加长期个性化特征相比于原始没添加个性化的模型 AUC 指标效果提高 0.4%。基于习题难度和编码时间单元改进相比于添加现有动态个性化方法 AUC 具有不同程度的提高，AUC 指标效果最好提高了 0.3%。添加长期个性化和动态个性化结合特征相比于添加现有动态个性化特征 AUC 指标效果提高了 0.7%。相比于没有添加个性化特征的模型 AUC 效果提高了 1.1%。

5 结论

5.1 本文工作总结

本文从传统学生知识追踪和深度学生知识追踪两方面进行知识追踪的研究：

(1) 针对传统学生知识追踪学生能力表征和习题表征不准确的问题。本文基于学生做题数据，测量习题答错率，知识点答错率，习题数量，知识点数量以及习题难度的分布，设计合理的习题表征和学生能力表征。

(2) 针对深度知识追踪模型无法准确追踪到学生知识能力水平的问题，也即是能力表征不准确的问题，提出长期个性化和动态个性化结合进行学生学习能力个性化。针对长期学习能力差异，本文结合数据，测量数据特征，找出班级类型特征作为长期个性化特征。对于动态个性化，本文在模型中引入习题难度信息，并在训练的过程中对学生学习能力进行分段化表示，实现准确的学生知识状态的动态个性化追踪。最终，本文综合上述两种方法实现长期和动态结合的学生知识追踪模型。并通过实验进行评估。

具体贡献如下：

1) 对于传统学生知识追踪，本文通过测量学生做题数据，发现习题答错率和知识点答错率都服从对数正态分布，并且基于知识点粒度的做题数据，稠密而均匀，基于习题粒度的做题数据稀疏并且不均匀。为了得到知识点粒度和习题粒度进行结合得到习题表征。本文提出知识点难度和习题难度以乘性模型的形式结合进行习题表征。知识点难度是模型学习出来的参数，习题难度特征是通过习题答错率和知识点答错率相除得到的。在真实数据上的实验结果表明：该方法将目前主流的知识追踪模型 DAS3H、AFM 和 PFA 在 Assistment12 数据集上的 AUC 分别提高了 0.7%、1.3%和 3.5%，在 Geometry 数据集上的 AUC 分别提高了 2.1%、3.5%和 3.6%，证明了该方法的有效性和通用性。
2) 通过分析学生做题数据，发现同一知识点下不同难度的习题对学生能力的提升不同，现有模型在学生能力表征中都是以知识点为粒度进行的，忽略了这个问题。所以为了得到准确的学生能力表征，本文引入习题答错率作为习题难度表征，对学生能力表征进行改进。接着在 1）改进的基础上进行实验，实验结果表明该方法将上述模型在 Assistment12 数据集上分别提高 0.2%、5.6%和 3.2%，在 Geometry 数据集上分别提高了 0.1%、0.6%和 1.2%，证明了在学

生能力表征中引入习题答错率的有效性和通用性。

3)对于深度学生知识追踪模型，首先，为了准确跟踪学生能力水平，本文在模型中引入习题难度信息，并对学生学习能力进行分段化表示，实现准确的学生知识状态的动态个性化追踪。在真实数据上的实验结果表明，该方法将 AUC 提高了 0.3%。其次，本文引入班级类型对学生进行基于长期能力特征的区分实现了长期个性化，将 AUC 提高了 0.4%。最终，本文设计的综合上述两种方法的新的深度知识追踪模型 DIDKT-CL 相比于添加现有个性化的深度学生模型 IDKT 的 AUC 提高了 0.7%，相比于没添加个性化的模型 DKT，AUC 指标提高了 1.1%，证明了本文算法的有效性。

5.2 未来工作展望

本文的工作从传统学生知识追踪和深度学生知识追踪两方面对知识追踪模型进行研究。对于传统学生知识追踪本文基于知识点和习题结合进行学生表征和习题表征的改进。对于深度学生知识追踪本文基于学生个性化进行改进。所以未来的工作希望：(1)对于传统模型引入学生个性化，现有的模型对于所有学生在同一个知识点上的学习率都一样，可以通过区分学生学习率，引入学生个性化。(2)对于深度学生知识追踪模型引入习题文本，然后将知识点和习题进行结合，利用知识点进行习题文本表征向量的改进。

参考文献

[1] Huang Y, Yudelson M, Han S, et al. A Framework for Dynamic Knowledge Modeling in Textbook-based Learning[C]. //Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization. Halifax, NS, Canada, 2016: 141-150.

[2] Pandey S, Karypis G. A Self-Attentive Model for Knowledge Tracing[J]. arXiv preprint arXiv:1907.06837, 2019.

[3] Mei H, Bansal M, Walter M R. Coherent Dialogue with Attention-based Language Models[C]// Proceeding of the Thirty-First AAAI Conference on Artificial Intelligence. New Orleans, Louisiana, USA, 2017: 3252-3258.

[4] Le C V, Pardos Z A, Meyer S D, et al. Communication at Scale in a MOOC Using Predictive Engagement Analytics[J]. Proceedings of International Conference on Artificial Intelligence in Education, 2018: 239-252.

[5] Vie J J. Deep Factorization Machines for Knowledge Tracing[C]. //Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications, New Orleans, Louisiana, 2018: 370-373.

[6] Vie J J, Kashima H. Knowledge Tracing Machines: Factorization Machines for Knowledge Tracing[C]. //Proceeding of the AAAI Conference on Artificial Intelligence. New York, USA, 2019: 750-757.

[7] Xu Y, Mostow J. Using Logistic Regression to Trace Multiple Sub-skills in a Dynamic Bayes Net[C]. // Proceedings of Educational Data Mining 2011, Eindhoven, The Netherlands, 2011: 241-246.

[8] 张明心. 基于认知诊断的贝叶斯知识追踪模型改进与应用[D].华东师范大学, 2019: 3.

[9] Corbett, A T, Anderson, J.R.: Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge[J]. UMUAI 4(4), 1995: 253–278.

[10] Wang Z, Zhu J, Li X, et al. Structured Knowledge Tracing Models for Student Assessment on Coursera[C]. // Proceedings of ACM Conference on Learning. ACM, 2016: 209-212.

[11] Baker R S, Corbett A T, Aleven V. More Accurate Student Modeling through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing[J]. Lecture Notes in Computer Science, 2008, 5091: 406-415.

[12] Zhu J, Zang Y, Qiu H, et al. Integrating Temporal Information into Knowledge Tracing: A Temporal Difference Approach[J]. IEEE Access, 2018: 1-1.

[13] 黄诗雯, 刘朝晖, 罗凌云, 等. 融合行为和遗忘因素的贝叶斯知识追踪模型研究[J].计算机应用研究, 2021, 38(07): 1-5.

[14] Lin C, Chi M. Intervention-BKT: Incorporating Instructional Interventions into Bayesian Knowledge Tracing[J]. in Intelligent Tutoring Systems, ITS 2016. Lecture Notes in Computer Science, vol 9684. Springer, Cham, 2016: 208-218.

[15] Khajah M, Wing R M, Lindsey R V, et al. Incorporating Latent Factors into Knowledge Tracing to Predict Individual Differences In Learning[C]. //Proceedings of the 7th International Conference on Educational Data Mining, London, UK, 2014: 99-106.

[16] Yudelson M V, Koedinger K R, Gordon G J. Individualized Bayesian Knowledge Tracing Models[C]. //Proceedings of 16th International Conference on Artificial Intelligence in Education (AIED 2013). Springer, Berlin, Heidelberg, 2013: 171-180.

[17] Zhang K, Yao Y. A Three Learning States Bayesian Knowledge Tracing Model[J]. Knowledge-Based Systems, 2018, 148(MAY15): 189-201.

[18] Embretson S E, Reise S P. Item response theory[M]. Psychology Press, 2013.

[19] Pardos Z, Heffernan N. Kt-idem: Introducing Item Difficulty to the Knowledge Tracing Model[J]. User Modeling, Adaption and Personalization, 2011: 243-254.

[20] Johnson M S, et al. Marginal Maximum Likelihood Estimation of Item Response Models in R[J]. Journal of Statistical Software, 2007, 20(10): 1–24.

[21] Dibellol V, Roussos L A, Stout W. A Review of Cognitively Diagnostic Assessment and A Summary of Psychometric Models[J]. Handbook of statistic,2006, 26: 979-1030.

[22] Abdi S, Khosravi H, Sadiq S, et al. A Multivariate ELO-based Learner Model for Adaptive Educational Systems[C]. //Proceedings of EDM 2019, Montreal, Canada, July 2-5, 2019: 228-233.

[23] Rachel R. Using a Glicko-based Algorithm to Measure In-Course Learning[C]. // Proceedings of EDM 2019, Montréal, Canada, 2019: 754-759.

[24] Lindsey R V, Shroyer J D, Pashler H, et al. Improving Students' Long-term Knowledge Retention through Personalized Review[J]. Psychological Science, 2014, 25(3): 639–647.

[25] Mozer M C, Lindsey R V. Predicting and Improving Memory Retention: Psychological Theory Matters in the Big Data Era[J]. In Big Data in Cognitive Science, Psychology Press, 2016: 43–73.

[26] Cen H. Generalized Learning Factors Analysis: Improving Cognitive Models with Machine Learning[D]. Carnegie Mellon University, 2009: 5-6.

[27] Pavlik P I, Cen H, Koedinger K R. Koedinger. Performance Factors Analysis –A New Alternative to Knowledge Tracing. In V. Dimitrova and R. Mizoguchi[C]. //Proceeding of International Conference on Artificial Intelligence in Educational, Brighton, UK, 2009: 531–538.

[28] Cen H, Koedinger K, Junker B. Learning Factors Analysis – A General Method for Cognitive Model Evaluation and Improvement[C]. // Proceedings of International Conference on Intelligent Tutoring Systems, Springer-Verlag, Berlin, Heidelberg, 2006: 164–175.

[29] Maclellan C J, Ran L, Koedinger K R. Accounting for Slipping and Other False Negatives in Logistic Models of Student Learning[C]. // Proceedings of the 8th International Conference on Educational Data Mining. Madrid, Spain, 2015: 53-60.

[30] Choffin B, Popineau F, Bourda Y, et al. DAS3H: Modeling Student Learning and Forgetting for Optimally Scheduling Distributed Practice of Skills[C]. // Proceedings of the 12th International Conference on Educational Data Mining. Montréal, Canada, 2019: 29-38.

[31] Minn S, Zhu F, Desmarais M C. Improving Knowledge Tracing Model by Integrating Problem Difficulty[C]. // Proceedings of 2018 IEEE International Conference on Data Mining Workshops (ICDMW). Singapore, 2018: 1505-1506.

[32] 艾方哲. 基于知识追踪的智能导学算法设计[D].北京交通大学,2019.

[33] Chris P, Jonathan S, Jonathan H, et al. Deep Knowledge Tracing[C]. // Proceedings of NIPS 2015, Montreal, QC, Canada, 2015: 505-513.

[34] Chunkit Y, Dityan Y. Addressing Two Problems in Deep Knowledge Tracing via Prediction-Consistent Regularization[C]. // Proceedings of L@S 2018, London, UK, 2018, Article No. 5.

[35] Long S, Pengyu H, Neural knowledge tracing[C]. // Proceedings of BFAL 2017, Patras, Greece, 2017: 108-117.

[36] Koki N, Qian Z, Masahiro S, et al. Augmenting Knowledge Tracing by Considering Forgetting Behavior[C]. // Proceedings of WWW 2019, San Francisco, CA, USA, 2019: 3101-3107.

[37] Jiani Z, Xingjian S, King I, Dit-Yan Y, Dynamic Key-Value Memory Networks for Knowledge Tracing[C]. // Proceedings of WWW 2017, Perth, Australia, 2017: 765-774.

[38] Fangzhe A, Yishuai C, Yuchun G, et al. Concept-Aware Deep Knowledge Tracing and Exercise Recommendation in an Online Learning System[C]. // Proceedings of EDM 2019, Montreal, Canada, 2019, 240: 245.

[39] Moore R, Caines A, Elliott M, et al. Skills Embeddings: a Neural Approach to Multicomponent Representations of Students and Tasks[C]. // Proceedings of EDM 2019, Montreal, Canada, 2019: 360-365.

[40] Wang Z, Feng X, Tang J, et al. Deep Knowledge Tracing with Side Information[C]. // Proceedings of AIED 2019, Chicago, Illinois, USA, 2019: 303-308

[41] Wang T, Ma F, Gao J. Deep Hierarchical Knowledge Tracing[C]. // Proceedings of EDM 2019, Montreal, Canada, 2019: 671-674.

[42] Yu S, Qingwen L, Qi L, et al. Exercise-Enhanced Sequential Modeling for Student Performance Prediction[C]. // Proceedings of The Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, Louisiana, USA, 2018: 2435-2443.

[43] Liu Q. EKT: Exercise-Aware Knowledge Tracing for Student Performance Prediction[J]. in IEEE Transactions on Knowledge and Data Engineering, 2021,33(1): 100-115.

[44] 黄振亚. 面向个性化学习的数据挖掘方法与应用研究[D].中国科学技术大学,2020:72-78.

[45] Minn S, Yu Y, Desmarais M C, et al. Deep knowledge tracing and dynamic student classification for knowledge tracing[C]. // Proceedings of ICDM 2018, Singapore, 2018: 1182-1187.

[46] 赵旭. 基于动态键值记忆网络的知识追踪算法研究[D]. 西北大学, 2020:25-28.

[47] Tibshirani R, Walther G, Hastie T. Estimating the Number of Clusters in a Data Set via the Gap Statistic[J]. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 2001, 63(2): 411-423.

[48] Https://zh.wikipedia.org/zh-hans/循环神经网络[EB/OL].

[49] Sutskever I, Vinyals O, Le Q. Sequence to Sequence Learning with Neural Networks[C]. // Proceedings of Advances in Neural Information Processing Systems. 2014: 3104-3112.

[50] Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. Neural Computation, 1997,9(8): 1735-1780.

[51] Kirch W. K-S Test[M]. Springer Netherlands, 2008: 823-824.

[52] Li Y, Du N, Bengio S. Time-Dependent Representation for Neural Event Sequence Prediction[C]. \\ Proceedings of ICLR, 2018.

作者简历及攻读硕士学位期间取得的研究成果

一、作者简历

王珍珠，女，1995 年 1 月生。2014 年 9 月至 2018 年 6 月就读于哈尔滨理工大学电气与电子工程学院电子信息工程专业，取得工学学士学位。2018 年 9 月至 2021 年 6 月就读于北京交通大学电子与信息工程学院通信与信息系统专业，研究方向是信息网络，取得工学硕士学位。攻读硕士学位期间，主要从事学生知识追踪方面的工作。

二、发表论文

[1] **Wang Z**, Chen Y, Su J, et al. Measurement and Prediction of Regional Traffic Volume in Holidays[C]. // proceeding of Intelligent Transportation Systems Conference (ITSC), 2019, Auckland, New Zealand, October 27-30.

[2] Ai F, Chen Y, Guo Y, Zhao Y, **Wang Z**. Concept-Aware Deep Knowledge Tracing and Exercise Recommendation in an Online Learning System[C]. //Proceedings of EDM 2019, Montreal, Canada, Jul 2-5.

三、参与科研项目

[1] 区域交通流量的预测

[2] 基于习题表征和学生能力表征的学生知识追踪

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作和取得的研究成果，除了文中特别加以标注和致谢之处外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京交通大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文作者签名：　　签字日期：　　年　　月　　日

学位论文数据集

表 1.1： 数据集页

关键词*	密级*	中图分类号	UDC	论文资助
学生知识追踪；习题表征；学生能力表征；学生学习能力	公开			
学位授予单位名称*		学位授予单位代码*	学位类别*	学位级别*
北京交通大学		10004	工学	硕士
论文题名*		并列题名		论文语种*
基于习题表征和学生能力表征的学生知识追踪算法研究				中文
作者姓名*	王珍珠		学号*	18120144
培养单位名称*		培养单位代码*	培养单位地址	邮编
北京交通大学		10004	北京市海淀区西直门外上园村 3 号	100044
学科专业*		研究方向*	学制*	学位授予年*
通信与信息系统		信息网络	3	2021
论文提交日期*	2021.05.07			
导师姓名*	陈一帅		职称*	副教授
评阅人	答辩委员会主席*		答辩委员会成员	
电子版论文提交格式 文本（ ） 图像（ ） 视频（ ） 音频（ ） 多媒体（ ） 其他（ ） 推荐格式：application/msword；application/pdf				
电子版论文出版（发布）者		电子版论文出版（发布）地		权限声明
论文总页数*	53			
共 33 项，其中带*为必填数据，为 21 项。				