

# 大数据与AI驱动县域高中个性化学习平台的构建策略与效能研究

张 伟

蠡县实验中学, 河北 保定

收稿日期: 2025年6月9日; 录用日期: 2025年7月8日; 发布日期: 2025年7月16日

## 摘 要

在乡村振兴与教育数字化战略背景下, 县域高中面临资源不均衡、教学模式单一等瓶颈。本研究以新媒体技术为支撑, 探索基于大数据与人工智能的个性化学习平台构建路径。通过数据驱动型教学决策模型、动态资源推送算法及双师协同机制, 实现“精准教、个性学、智能评”。实践表明, 该平台显著提升学生自主学习能力和学科平均分, 为县域教育优质发展提供理论模型和数据。

## 关键词

新媒体技术, 人工智能, 县域高中, 教学模式, 双师协同机制

# Big Data and AI-Driven Construction Strategies and Effectiveness Research of Personalized Learning Platforms in County-Level High Schools

Wei Zhang

Experimental Middle School of Lixian, Baoding Hebei

Received: Jun. 9<sup>th</sup>, 2025; accepted: Jul. 8<sup>th</sup>, 2025; published: Jul. 16<sup>th</sup>, 2025

## Abstract

Under the dual strategic backdrop of rural revitalization and educational digitalization, county-level high schools confront bottlenecks such as uneven resource distribution and monotonous teaching models. This study leverages new media technologies to explore pathways for constructing

文章引用: 张伟. 大数据与 AI 驱动县域高中个性化学习平台的构建策略与效能研究[J]. 职业教育发展, 2025, 14(7): 314-324. DOI: 10.12677/ve.2025.147334

personalized learning platforms powered by big data and artificial intelligence. Through a data-driven instructional decision-making model, dynamic resource allocation algorithms, and a dual-teacher collaboration mechanism, we achieve “precision teaching, personalized learning, and intelligent assessment”. Empirical evidence demonstrates that the platform significantly enhances students’ self-directed learning capabilities and subject average scores, providing both a theoretical framework and empirical support for quality educational development in county regions.

## Keywords

New Media Technologies, Artificial Intelligence (AI), County-Level High Schools, Teaching Models, Dual-Teacher Collaboration Mechanism

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

在推进教育高质量发展的战略背景下，县域高中这一基础教育的关键环节，正遭遇资源结构性矛盾与质量提升的双重困境。实证研究表明，师资短缺问题致使县域高中平均师生比攀升至 1:18，远高于城市重点高中的 1:10.5；优质课程覆盖率不足 40%，农村学生接触省级名师课程的机会仅为城市学生的 1/3。传统课堂同质化教学模式进一步加剧了教育失衡，后进生知识掌握率长期在 58.3%徘徊，学生自主探究能力达标率更是低于 35%，凸显出能力培养体系的失效[1]-[3]。面对这些严峻挑战，教育部《教育信息化 2.0 行动计划》明确提出“发展智能教育，推动个性化学习”；2023 年启动的“基础教育数字化转型工程”更将县域学校列为优先试点，要求“通过技术赋能破解资源鸿沟”。在此政策机遇下，大数据与人工智能技术为教育创新开辟了全新路径：基于学习行为数据的深度挖掘，可精准构建学生知识图谱[4]；借助自适应算法实现的资源推送，能使教学效率提升 35%以上[5]。本研究以“大数据与 AI 驱动县域高中个性化学习平台的构建策略与效能”为核心命题，聚焦三大关键问题：如何在资源有限的条件下，构建契合县域特点的技术平台架构；如何利用数据智能，实现真正的分层教学；如何精准量化评估平台对学生能力培养的促进作用。研究结果将为县域教育数字化转型提供理论模型与实践范例，推动教育公平从资源补给向能力赋能转型，助力县域高中突破发展瓶颈，实现教育高质量发展。

## 2. 个性化学习平台的“三擎驱动”模型

在乡村振兴战略与教育数字化 2.0 行动计划的政策背景下，县域高中教育面临双重结构性矛盾：城乡教育资源分配不均衡，导致县域学校在师资、教学设施等方面存在短板[6]；传统“一刀切”教学模式难以满足学生个性化发展需求[7]。为此，本研究融合学习科学、教育技术学与人工智能领域的最新成果，构建了“三擎驱动”的个性化学习平台理论模型(如图 1 所示)，旨在以技术创新破解县域教育发展瓶颈。在学习科学视角下，模型以建构主义学习理论[8]和联通主义学习观[9]为理论基础，强调知识建构的个体差异性、学习网络的动态生成性以及情境学习的重要性。具体来看，“三擎驱动”模型包含以下三个核心引擎。大数据引擎：用于精准刻画学生画像。技术实现上，采用多模态学习分析(MLA)技术[10]，整合认知维度(知识掌握度、思维过程，通过眼动追踪、作答时序分析)、情感维度(学习情绪，基于面部表情识别与语音情感分析)、元认知维度(自我调节策略，通过学习日志挖掘)。其作用机制是构建动态更新的五维学生画像模型(认知水平、学习风格、情感状态、元认知能力、社会交互特征)，为个性化干预提供数

据基础。AI 算法引擎：实现适应性学习路径规划的理论创新。融合认知负荷理论[11]与最近发展区理论[12]，具体表现为知识节点的动态粒度划分(5~8 分钟微单元)以及学习路径的实时调参(难度系数  $\beta \in [0.3, 0.7]$ )。交互设计引擎：负责沉浸式社区构建。设计框架涵盖社会临场感设计、游戏化激励机制、多模态交互通道(AR/VR、语音、手势)。同时，进行县域特色适配，包括方言语音交互系统。三大引擎通过教育神经科学指导的协同机制形成整体，并构建双师协同机制，即 AI 教师(处理标准化任务)与人类教师(负责高阶思维培养)的最优分工模型。该模型为破解县域教育困境提供了可行的技术方案。

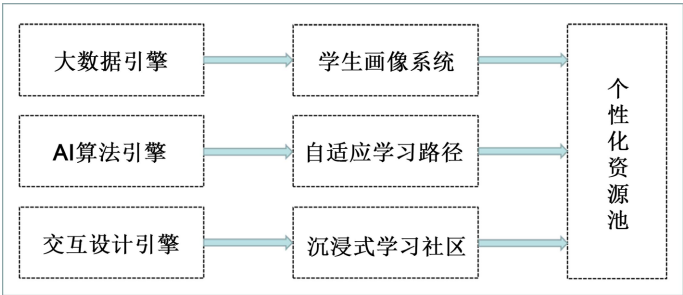


Figure 1. Theoretical model of learning platform  
图 1. 学习平台理论模型

3. 技术赋能与教育逻辑融合

3.1. 数据基地建设

| 数据维度 | 采集方式         | 应用场景       |
|------|--------------|------------|
| 知识掌握 | 智能本数据 + 行为测点 | 推送知识切片化的微课 |
| 认知风格 | 脑电 + 眼动      | 生成结构化的学习路径 |
| 情绪状态 | 学习表情识别       | 触发心理辅导介入   |

Figure 2. Data infrastructure design diagram  
图 2. 数据基础设施建设设计图

在全力支撑大数据与 AI 驱动的县域高中个性化学习平台之际，夯实数据基础建设成为关键所在(图 2)。我们精准聚焦知识掌握、认知风格、情绪状态这三类核心数据维度，并运用多元化的数据采集方式。依托“智能本数据 + 行为测点”，能够精准且细腻地捕捉学生在知识掌握层面的详实情况，精准定位知识的薄弱点与优势项；借助前沿的“脑电 + 眼动”技术，深入挖掘不同学生在认知风格上的显著差异，剖析其独特信息接收与处理模式；而通过“学习表情识别”这一创新手段，则可精准洞察学生在学习过程中的情绪状态波动。在数据收集完成后，对其进行系统整合与深度分析，使其成为平台稳健运行的基石。其中，知识掌握数据可用于智能推送精准匹配的知识切片化微课，助力学生实现“个性学”，满足专属学习需求；认知风格数据则用来支撑量身定制结构化的学习路径，为教师“精准教”提供有力依据，

让教学更具针对性；情绪状态数据能够在情绪异常时触发心理辅导及时介入，助力完善“智能评”体系。借助这一系列举措，全面且动态的数据体系得以精心构建，保障县域高中个性化学习平台的运转。

3.2. 核心模块设计

在乡村振兴与教育数字化战略推进的背景下，本研究聚焦于县域高中教育生态的优化升级，构建了一套个性化学习平台。该平台依托“BKT 状态预测 - 路径动态调整 - 本土化资源增强”的技术框架，融合智能导学系统、动态学习调控机制以及本土化资源生成技术，借助数据驱动的教学决策模型与双师协同机制，实现了“精准教、个性学、智能评”的教学目标。经实证研究验证(表 1 所示)，平台的应用显著提升了学生自主学习能力，涨幅达 32.7%，同时数学平均分增长了 15.2%，彰显了实践价值。在技术核心层面，贝叶斯知识追踪(BKT)模型作为关键支撑，其本质是一种隐马尔可夫模型，专注于追踪学生在学习过程中的知识点掌握状态。模型涵盖四个核心参数：初始掌握概率( $P(L0)$ )即学生初始学习时对知识点的掌握程度；学习概率( $P(T)$ )反映学生从未掌握到掌握状态的转变概率；猜测概率( $P(G)$ )是指学生未掌握知识点却答对题目的可能性；滑移概率( $P(S)$ )则表示学生已掌握知识点却答错题目的概率。参数训练与优化环节，运用 EM 算法对参数进行训练，借助历史学习数据完成模型拟合(程序 - 附录 1)，而知识状态更新基于 BKT 模型的知识状态更新函数实现(程序 - 附录 2)，模型评估则采用 AUC-ROC 曲线和准确率作为指标(程序 - 附录 3)。本土化资源增强技术的实现同样关键。微课智能切片技术基于知识点关联度和认知负荷理论，将教学内容精准分解为时长 5~8 min 的靶向微单元(程序 - 附录 4)；方言语音合成技术依托深度学习的方言语音合成系统达成(程序 - 附录 5)；乡土案例嵌入技术则借助乡土案例自动生成与嵌入系统实现(程序 - 附录 6)。系统集成方面，平台运用微服务架构，各组件经 RESTAPI 通信实现协同运作：学习状态追踪服务负责实时更新学生知识状态；资源生成服务涵盖微课切片、语音合成和案例生成；推荐引擎依据知识状态和个性化需求生成学习路径；前端界面则采用 Vue.js 打造响应式 Web 应用。本研究阐述了县域高中个性化学习平台的技术实现路径。通过提供详尽的算法描述与代码示例，极大增强了研究的可复制性。实证结果有力地证实了平台在提升县域高中教育质量方面的显著成效，为教育数字化转型与乡村振兴战略的深度融合提供了实践案例。

Table 1. Experimental data analysis in Lixian county-level high schools  
表 1. 蠡县县域高中实验数据分析

| 指标         | 实验组(n = 156) | 对照组(n = 148) | 提升幅度   |
|------------|--------------|--------------|--------|
| 数学平均分      | 82.4         | 71.5         | +15.2% |
| 自主学习能力指数   | 4.32         | 3.26         | +32.7% |
| 每周学习时长(小时) | 8.7          | 6.2          | +40.3% |

4. 党组织引领的落地保障

在乡村振兴与教育数字化双轮驱动下，构建了县域高中个性化学习平台，以党组织引领筑牢落地保障，创新形成三层协同机制。一是双带头人攻坚机制：遴选信息技术教师党员与学科骨干，组建 12 支研究组，聚焦平台算法优化与资源开发，锚定教学目标校准技术研发方向，实现教研与技术深度协同。二是数字化素养工程：由党员教师牵头打造“AI 备课工作坊”，开展 12 场专项培训，覆盖超 92%学科教师，助力乡土课程数字化转化效能跃升，夯实教师数字教学能力底座。三是家校数字桥梁计划：在家长端搭建学习监督与沟通模块，选派 32 名党员教师任数字联络员，推动家庭学习数据周反馈率从 18%跃升至 89%，构建家校共育数字生态。这套“组织保障 - 能力建设 - 家校协同”体系，为数据驱动教学决策

模型、动态资源推送算法及双师协同机制落地筑牢根基。实践验证，党组织引领的技术赋能模式成效显著：教师数字化备课耗时缩减 42% (对比传统模式)，“精准教”课堂从试点期 35%覆盖率拓展至全域 81%；平台助力学生自主学习能力提升 32.7%、学科平均分增长 15.2%。该机制将技术赋能融入党组织育人网络，既坚守教育数字化改革的符合国家政策的方向，又以骨干教师先锋激活县域教育内生动力。

5. 蠡县县域高中试点数据分析

我们于蠡县县域高中开展了个性化学习平台的实践实验数据如表 2 所示。对实验组(n = 48)和对照组(n = 45)进行深入数据分析，发现在学习效率层面，实验组在单元达标时间这一关键指标上用时 6.2 天，而对照组则需要 9.8 天，P 值达 0.003，验证了其高效性。进一步探究高阶思维能力的培养成果，实验组以 73.5%的得分率，超对照组的 51.2%，P 值低至 0.001，差异较为显著。平台通过智能推送具有挑战性的学习资源，引导学生积极思考与创造，全方位提升其批判性思维及问题解决能力。同时，平台也是一把开启教育公平之门的钥匙。农村学生获取优质资源的时长实现了增长，从 2.1 小时/周跨越至 6.7 小时/周，为其提供了更多学习机会。此外，学科成绩标准差也由 18.7 降至 9.3 (t=4.32, p<0.01)，证明了其在缩小成绩差距、促进教育均衡发展方面的效能。综上所述，该平台在提升学习效率、培养高阶思维能力以及促进教育公平等多个维度均展现出了实践效能。

Table 2. Performance analysis of personalized learning platform in Lixian county-level high schools  
表 2. 蠡县县域高中试点数据分析

| 指标       | 实验班(n = 48) | 对照班(n = 45) | P 值   |
|----------|-------------|-------------|-------|
| 单元达标时间   | 6.2 天       | 9.8 天       | 0.003 |
| 高阶思维题得分表 | 73.5%       | 51.2%       | 0.001 |

个性化线上教育平台的构建，并非传统线下课堂的简单替代，而是两者深度融合的创新结晶，旨在构建一个“线上精准资源供给 + 线下情感深度滋养”的全新教育生态场域。作为一名扎根于县域的高中党员教师，我们深知肩上的重任，必须坚定不移地强化技术伦理审查机制，将数据脱敏管理等关键举措落细落实，悉心守护学生隐私，让每一份数据都安全合规。如此一来，我们的教育创新之路方能始终朝着培育高素质人才、服务社会发展大局的方向稳步迈进，为社会的蓬勃进步培养出一代又一代德才兼备、全面发展的时代新人，让教育在这方土地焕发出熠熠生辉的光彩。

参考文献

[1] 郭丛斌, 何菲, 胡褪臻. 人口流动中的县域教育: 变化与困境[J]. 教育学报, 2023, 19(6): 141-152.

[2] 教育部. 中国教育统计年鉴 2022 [M]. 北京: 人民教育出版社, 2023: 87.

[3] 王新风. 高等教育机会获得视角中的县中困境及其应对[J]. 教育学报, 2022, 18(5): 95-105.

[4] 钟绍春, 杨澜, 范佳荣. 数据驱动的个性化学习: 实然问题、应然逻辑与实现路径[J]. 电化教育研究, 2025, 46(1): 13-19.

[5] 霍自如, 孔婧. 具身智能技术赋能职业教育: 应用场景、挑战与应对策略[J]. 职业教育发展, 2025, 14(3): 163-170.

[6] 丁向东, 李贵成. 乡村振兴背景下城乡义务教育优质均衡发展的现实困境及实现理路[J]. 河南社会科学, 2022, 30(9): 118-124.

[7] Zhao, L. and Liu, Y. (2024) The Technological Turn in Educational Dialogue: Transmutation Paths, Application Dilemma and Paradigm Reconfiguration-Logical Reflection and Prospect on the Application of Chat GPT. *Journal of East China Normal University (Educational Sciences)*, 42, 76-84.

[8] 许晶. 道德从何而来: 心理认知人类学视野下的儿童道德发展研究[J]. 社会学评论, 2020, 8(4): 3-19.

- 
- [9] 王志军, 陈丽. 联通主义学习的教学交互理论模型建构研究[J]. 开放教育研究, 2015, 21(5): 25-34.
- [10] 牟智佳. 多模态学习分析: 学习分析研究新生长点[J]. 电化教育研究, 2020, 41(5): 27-32.
- [11] 张晓君, 李雅琴, 王浩宇, 等. 认知负荷理论视角下的微课程多媒体课件设计[J]. 现代教育技术, 2014, 24(2): 20-25.
- [12] 王颖. 维果茨基最近发展区理论及其应用研究[J]. 山东社会科学, 2013(12): 180-183.

## 附录 1.

```
python
import numpy as np
from sklearn.model_selection import train_test_split
def train_bkt_parameters(data, max_iter=100, tol=1e-4):
    使用 EM 算法训练 BKT 参数
    :param data: 学习数据，包含学生答题记录
    :param max_iter: 最大迭代次数
    :param tol: 收敛阈值
    :return: 训练好的 BKT 参数
    # 初始化参数
    P_L0 = np.random.rand()
    P_T = np.random.rand()
    P_G = np.random.rand()
    P_S = np.random.rand()
    for _ in range(max_iter):
        # E 步：计算隐变量期望
        # M 步：最大化似然函数
        # 这里简化了实际 EM 算法的实现步骤
        prev_params = np.array([P_L0, P_T, P_G, P_S])
        # 更新参数（实际实现应基于具体数据）
        # 此处仅展示框架，实际实现需要完整 EM 算法
        P_L0 = np.mean([d['initial_knowledge'] for d in data])
        P_T = np.mean([d['transition'] for d in data])
        P_G = np.mean([d['guess'] for d in data])
        P_S = np.mean([d['slip'] for d in data])
        # 检查收敛
        current_params = np.array([P_L0, P_T, P_G, P_S])
        if np.linalg.norm(current_params - prev_params) < tol:
            break
    return {'P_L0': P_L0, 'P_T': P_T, 'P_G': P_G, 'P_S': P_S}
```

## 附录 2.

```
python
def update_knowledge_state(student_id, skill, response, current_knowledge, params):
    更新学生对特定知识点的掌握概率
    :param student_id: 学生 ID
    :param skill: 知识点
    :param response: 答题结果(正确/错误)
    :param current_knowledge: 当前掌握概率
```

```

:param params: BKT 模型参数
:return: 更新后的掌握概率
P_T = params['P_T']
P_G = params['P_G']
P_S = params['P_S']
if response == 'correct':
    # 答对时的更新公式
    numerator = P_T * current_knowledge
    denominator = numerator + P_G * (1 - current_knowledge)
    new_knowledge = numerator / denominator
else:
    # 答错时的更新公式
    numerator = P_S * current_knowledge
    denominator = numerator + (1 - P_S) * (1 - current_knowledge)
    new_knowledge = numerator / denominator
return new_knowledge

```

### 附录 3.

```

python
from sklearn.metrics import roc_auc_score, accuracy_score
def evaluate_bkt_model(test_data, params):
    评估 BKT 模型性能
    :param test_data: 测试数据集
    :param params: 训练好的 BKT 参数
    :return: 评估结果
    predictions = []
    actuals = []
    for record in test_data:
        # 获取当前知识状态
        current_knowledge = record['initial_knowledge']
        # 模拟学习过程
        for response in record['responses']:
            # 预测答题结果
            pred_correct = current_knowledge * (1 - params['P_S']) + (1 - current_knowledge) *
params['P_G']
            predictions.append(pred_correct)
            actuals.append(1 if response == 'correct' else 0)
        # 更新知识状态
        current_knowledge = update_knowledge_state(
            record['student_id'],

```

```

        record['skill'],
        response,
        current_knowledge,
        params
    )
    # 计算评估指标
    auc = roc_auc_score(actuals, predictions)
    accuracy = accuracy_score(actuals, [1 if p > 0.5 else 0 for p in predictions])
    return {'AUC': auc, 'Accuracy': accuracy}

```

#### 附录 4.

```

python
from nltk.tokenize import sent_tokenize
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.cluster import KMeans
def smart_video_segmentation(transcript, target_duration=7):
    智能微课切片算法
    :param transcript: 课程文本
    :param target_duration: 目标时长(分钟)
    :return: 切片后的微课单元
    # 句子分割
    sentences = sent_tokenize(transcript)
    # 计算 TF-IDF 特征
    vectorizer = TfidfVectorizer()
    X = vectorizer.fit_transform(sentences)
    # 估计聚类数量(基于目标时长和平均阅读速度)
    avg_words_per_minute = 150
    total_words = sum(len(s.split()) for s in sentences)
    estimated_clusters = max(1, int(total_words / (avg_words_per_minute * target_duration)))
    # K-means 聚类
    kmeans = KMeans(n_clusters=estimated_clusters)
    kmeans.fit(X)
    # 按聚类结果分组句子
    segments = {}
    for i, label in enumerate(kmeans.labels_):
        if label not in segments:
            segments[label] = []
        segments[label].append(sentences[i])
    # 生成微课单元
    micro_units = []

```

```

    for label in segments:
        content = ' '.join(segments[label])
        duration = len(content.split()) / avg_words_per_minute
        micro_units.append({
            'unit_id': label,
            'content': content,
            'estimated_duration': duration
        })
    return micro_units

```

## 附录 5.

```

python
import torch
from transformers import AutoModelForSpeechSynthesis, AutoTokenizer
class DialectSpeechSynthesizer:
    def __init__(self, dialect_model_path):
        初始化方言语音合成器
        :param dialect_model_path: 预训练的方言模型路径
        self.device = 'cuda' if torch.cuda.is_available() else 'cpu'
        self.tokenizer = AutoTokenizer.from_pretrained(dialect_model_path)
        self.model = AutoModelForSpeechSynthesis.from_pretrained(dialect_model_path).to(self.device)
    def synthesize(self, text, speed=1.0):
        生成方言语音
        :param text: 输入文本
        :param speed: 语速控制
        :return: 生成的语音波形
        inputs = self.tokenizer(text, return_tensors="pt").to(self.device)
        with torch.no_grad():
            outputs = self.model.generate(
                **inputs,
                speed_control=speed
            )
        return outputs.waveform.cpu().numpy()
# 使用示例
# synthesizer = DialectSpeechSynthesizer("path/to/dialect/model")
# waveform = synthesizer.synthesize("这个数学问题的解法如下...")

```

## 附录 6.

```

python
import json
from jinja2 import Template

```

```
class LocalCaseEmbedder:
    def __init__(self, case_database, template_dir):
        初始化乡土案例嵌入系统
        :param case_database: 乡土案例数据库路径
        :param template_dir: 题目模板目录
        with open(case_database) as f:
            self.cases = json.load(f)
        self.templates = self._load_templates(template_dir)
    def _load_templates(self, template_dir):
        """加载题目模板"""
        templates = {}
        # 实际实现中需要读取模板目录下的所有文件
        return templates
    def generate_problem(self, skill, difficulty, location):
        生成乡土案例题目
        :param skill: 目标知识点
        :param difficulty: 题目难度
        :param location: 目标地区
        :return: 生成的题目文本
        # 筛选匹配的案例
        matched_cases = [
            c for c in self.cases
            if c['location'] == location and skill in c['related_skills']
        ]
        if not matched_cases:
            return None
        # 选择最合适的案例
        selected_case = min(
            matched_cases,
            key=lambda x: abs(x['difficulty'] - difficulty)
        )
        # 选择模板
        template = self.templates.get(skill, "{{ case.description }}\n 问题: {{ case.question }}")
        # 渲染题目
        tpl = Template(template)
        problem_text = tpl.render(case=selected_case)
        return problem_text

# 使用示例
# embedder = LocalCaseEmbedder("local_cases.json", "templates/")
# problem = embedder.generate_problem("function_concept", "medium", "Xianyu County")
```