0

NAMIBIA UNIVERSITY
OF SCIENCE AND TECHNOLOGY

Individual Development Plan Recommender Model Using Deep Learning

Master's Thesis

Student: Hubert Patrick Mouton

Programme: Master's in Data Science (RIT912S)

Supervisor: Prof. Muyingi, Hippolyte N'sung-Nza

Date: Sunday 11th January, 2026

Digital Signature of Supervisor: ______________________

Digital Signature of Programme Coordinator: ______________________

Digital Signature of FCI HDC Representative: ______________________

DECLARATION

I, Hubert Patrick Mouton, hereby declare that the work contained in this thesis, for the degree **Master's of Data Science**, entitled:

INDIVIDUAL DEVELOPMENT PLAN RECOMMENDER MODEL USING DEEP LEARNING

is my own original work and that I have not previously, in its entirety or in part, submitted it at any university or other higher education institution for the award of a degree. I further declare that I will fully acknowledge any sources of information I use for the research in accordance with the Institution's rules.

____________________ ____________________

Signature Date

SIGNATURE OF THE SUPERVISOR

____________________ ____________________

Supervisor's Signature Date

Preface

<preface>

Contents

Nomenclature

List of Abbreviations

A/B Testing A/B Testing for Model Evaluation

AE Autoencoder

ANN Artificial Neural Network

AUC-ROC Area Under the Receiver Operating Characteristic Curve

Bias Bias in AI models

CF Collaborative Filtering

CNN Convolutional Neural Network

Cross Validation k-Fold Cross Validation

DL Deep Learning

DRL Deep Reinforcement Learning

EDA Exploratory Data Analysis

F1 F1-Score

GD Gradient Descent

GNN Graph Neural Network

ICF Item-based Collaborative Filtering

IDF Inverse Document Frequency

IDP Intelligent Development Plan

LIME Local Interpretable Model-agnostic Explanations

LSTM Long Short-Term Memory

MAE Mean Absolute Error

MF Matrix Factorization

ML Machine Learning

NLP Natural Language Processing

PCA Principal Component Analysis

Precision Precision

R2 R-squared

RBCF Recurrent-Based Collaborative Filtering

Recall Recall

RL Reinforcement Learning

RMSE Root Mean Square Error

RNN Recurrent Neural Network

SGD Stochastic Gradient Descent

SHAP SHapley Additive exPlanations

SVD Singular Value Decomposition

UCF User-based Collaborative Filtering

List of Symbols

α Learning Rate

β Bias Parameter

δ TD-Error (Temporal Difference Error)

ϵ Exploration Rate (in Reinforcement Learning)

η Entropy Temperature (in Model Explainability)

γ Discount Factor (for RL)

$\hat{r}_{ui}$ Predicted Rating for User u and Item i

κ Huber-Loss Parameter

λ Regularization Parameter (L2)

i Item Vector (in Collaborative Filtering)

R Ratings Matrix (in Recommender Systems)

u User Vector (in Collaborative Filtering)

W Weights Matrix (in Neural Networks or Collaborative Filtering)

X Input Features (in Neural Networks or Recommender Models)

y Model Output (in supervised learning or regression tasks)

z Latent Factor Vector (in Matrix Factorization)

Ψ Distortion Risk-Measure (Fairness Metric)

τ Temperature (in Model Interpretability)

ξ Risk-Distortion Parameter (Fairness Measure)

t Time-Step (in Reinforcement Learning or Sequential Models)

List of Figures

List of Tables

1. Introduction

1.1. Background to the Study

This study explores the development of Individual Development Plans (IDPs) using a deep learning-based recommender system. IDPs offer a structured framework for self-assessment, goal setting, and action planning, and are important for aligning personal growth with organizational objectives in complex career contexts (Vanderford , Evans , Weiss , Bira , & Beltran-Gastelum , 2018).

1.2. Choice of Methods

The study adopts a mixed-methods approach, integrating qualitative insights and quantitative validation (et al. , 2021; Guetterman , 2016). Data collection will involve semi-structured interviews and structured questionnaires. Analysis will include thematic coding and statistical modeling, supporting deep learning model training and evaluation.

1.3. The Research Time Horizon

A longitudinal design will be employed to monitor participants' development plans over time. Regular assessments will capture evolving employee competencies and measure the recommender system's long-term effectiveness (Jung , Kim , Lee , & An , 2023; Kelley & Rausch , 2011). Careful sampling and structured engagement strategies will maintain data integrity and participant involvement.lores the development of Individual Development Plans (IDPs) using a deep learning-based recommender system. IDPs, introduced by the Federation of American Societies for Experimental Biology, offer a structured framework for self-assessment, goal setting, and action planning, particularly for doctoral trainees Vanderford et al., 2018. They are crucial for aligning personal growth with organizational goals in today's complex career landscapes.

Recommender systems, widely used in e-commerce, education, and healthcare, provide personalized suggestions. However, traditional systems often struggle with dynamic user behaviors and lack interpretability Sahoo , Pradhan , Barik , and Dubey , 2019. Deep learning enhances these systems by uncovering intricate user-item patterns, enabling stronger personalization and adaptability Li et al., 2024a; Mu , 2018.

Advanced deep learning techniques like attention mechanisms, knowledge graphs, and Graph Convolutional Networks (GCNs) are highly effective for modeling relationships among employee skills, career goals, and resources Chen and Zhong , 2024; Li et al., 2024a. These methods form the basis for building an adaptive IDP recommender system.

1.4. Problem Statement

Telecom Namibia currently conducts competency assessments manually using Excel templates and email exchanges. This process is inefficient, prone to errors, and lacks real-time insights, thereby hindering timely decision-making. As the organization grows, scalability issues become apparent, necessitating an automated, centralized platform for more efficient and accurate IDP creation.

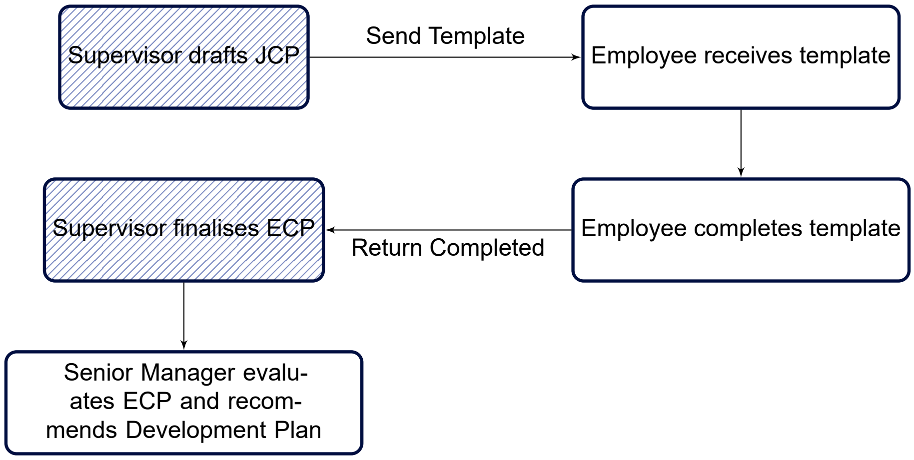

Figure 1.1: Current IDP Creation Process at Telecom Namibia (Interaction View).

1.5. Research Aim

The aim is to enable access to real-time data,through a platform where management, staff and supervisors may interact through training and skill audit assessments. This would automate workflows associated with individual development plan formulation and implementation.

1.6. Research Questions

In what ways may deep learning techniques be applied to create a personalized and efficient recommender model that appropriately evaluates employee competencies and suggests chances for customized growth for individual development plans?

Research Question 1

Which employee data types, e.g., skills, evaluations, performance reviews, etc, are most crucial to the construction of a well performing deep learning model that generates recommendations for individual development plans?

Research Question 2

How can employee characteristics and past performance data be used to refine deep learning algorithms to produce precise and customized recommendations for development plans?

Research Question 3

In what ways can the use of real-time performance data and feedback enhance the individual growth plan recommendations' adaptability and relevance?

Research Question 4

What is the difference between the accuracy, employee skill development, and organizational productivity of the deep learning-based recommender system for individual development plans and the conventional approaches?

1.7. Research Objectives

Research Objective

The primary objective of this research is to design an individual development plan recommender model using deep learning.

- **RO1:** To compile comprehensive datasets containing employee skills, evaluations, and development history.
- **RO2:** To design a deep learning model that analyzes skill gaps and generates customized growth plans.
- **RO3:** To integrate real-time feedback loops to dynamically update employee development recommendations.
- **RO4:** To evaluate the effectiveness of the proposed system compared to traditional IDP development methods.

1.8. Rationale and Motivation

The rapid evolution of employee competencies and organizational objectives makes manual development planning increasingly unsustainable. Automating IDP generation through AI enables scalability, accuracy, and responsiveness. Personally, this project reflects my passion for applying machine learning to real-world human resource development challenges, specifically optimizing organizational talent management systems.

1.9. Research Approach Overview

The research follows an interpretivist philosophical stance, utilizing an abductive mixed-methods strategy. Data will be collected from surveys, interviews, and organizational records. Analyses will include both thematic qualitative coding and quantitative statistical techniques, supporting the development of a deep learning recommender system.

1.9.1. Interpretivism as a Research Philosophy

Interpretivism emphasizes the individuality of human experiences (Irshaidat , 2019; Saunders , Lewis , & Thornhill , 2012). Each employee's development needs are influenced by personal, social, and cultural factors (Myers , 2008). Interpretivism opposes generalization and instead values subjective interpretations, aligning well with the need for personalized development plans.

1.9.2. The Research Approach

Following Hurley , Dietrich , and Rundle-Thiele (2021) and Mantere and Ketokivi (2013), an abductive approach combines inductive and deductive reasoning. Thompson (2022) provides an 8-step abductive framework, guiding data collection, feature extraction, theme development, theorizing, model implementation, and evaluation through continuous reflection.

1.9.3. Research Strategy: Action Research

Action Research (AR), particularly Canonical Action Research (CAR) as outlined by Avison , Lau , Myers , and Nielsen (1999) and Davison , Martinsons , and Malaurent (2021), is adopted. AR fosters collaboration between researcher and participants in an iterative cycle of problem identification, intervention, observation, and reflection—making it highly suitable for information system development like the proposed IDP recommender.

1.10. Expected Deliverables and Contributions

- A fully functional deep learning-based IDP recommender model.

- A prototype centralized platform for employee development planning.
- Contributions to the field of AI-driven HR systems and personalized learning pathways.

1.11. Structure of the Thesis

- **Chapter 1: Introduction** — Context, problem statement, objectives, and thesis outline.
- **Chapter 2: Literature Review** — Review of IDPs, recommender systems, and deep learning techniques.
- **Chapter 3: Methodology** — Research design, model development strategy, and data collection methods.
- **Chapter 4: Results and Analysis** — Experimental results and system evaluation.
- **Chapter 5: Discussion and Conclusion** — Findings interpretation, research limitations, and future work.

2. Literature Review

2.1. Introduction

The increasing complexity of the global market necessitates innovative approaches to workforce development, with artificial intelligence emerging as a pivotal technology for upskilling and reskilling employees Ramachandran et al., 2024. This is particularly crucial for companies aiming to personalize learning experiences and enhance productivity by leveraging advanced AI algorithms to create tailored training programs Ramachandran et al., 2024. Such personalized training recommendations are vital for improving individual and organizational performance, allowing companies to maintain a competitive edge in fast-paced business environments Wang et al., 2020. These AI-powered solutions leverage data analytics and machine learning to adapt training modules based on individual needs, skill gaps, and performance metrics, thereby enhancing engagement and knowledge retention within organizations Oladele, Anector, and Foluwa, 2025. Importantly, these systems consider both employees' current competencies and their future career development preferences to offer relevant and explainable recommendations Wang et al., 2020. Such models can also forecast necessary competencies for software development teams, aligning training with future organizational demands and even contributing to comprehensive workforce optimization Nosratabadi, Zahed, Ponkratov, and Kostyrin, 2022, 3. This strategic alignment ensures that learning and development initiatives are not merely reactive but proactively address skill shortages and contribute to organizational agility Nurjaman, 2025. However, challenges such as data sparsity and cold-start problems, along with the need for accurate skill profile reflection, remain pertinent considerations in developing such systems Wang et al., 2020.

2.1.1. Individual Development Plans

Individual Development Plans are structured documents outlining personal and professional goals, competencies needed for sustained career growth, and actionable strategies to address skill gaps, typically co-developed by employees and managers Azizi, Vajargha, Arefi, and Abolghasemi, 2020, 4; da Silva, Borges, Sarsur, Nunes, and de Amorim, 2019, 4. Their primary purpose is to facilitate targeted training, self-assessment, career exploration, and setting of SMART objectives, aligning individual aspirations with organizational needs Chang, Hui, Justus-Smith, and Wang, 2024; Flood, Skrabalak, and Yu, 2021, 23. In organizational contexts, IDPs boost employee engagement, retention, leadership skills, and overall alignment between personal development and business goals Sumartik and Ambarwati, 2023; van der Merwe, Nel, and Hoole, 2024. In educational settings, particularly STEM graduate programs, they enhance self-awareness, self-efficacy, goal clarity, and continuous improvement through iterative planning and mentoring Coopersmith, 2021, 1; Rubio et al., 2023, 1.

IDPs offer significant benefits, including increased motivation, productivity, talent retention, and organizational agility by fostering continuous learning and employability Sumartik and Ambarwati, 2023; van der Merwe et al., 2024. Organizations gain from effective IDP implementation through strengthened leadership pipelines, reduced absenteeism, higher performance, and strategic workforce planning Geerts, 2024, 10; Zafar, 2025, 2.

However, traditional IDPs often fall short in dynamically adapting to rapidly changing demands, relying on static frameworks that fail to capture evolving skill requirements or individual learning paces Kastrati, Imran, and Kurti, 2019; Nurjaman, 2025. Additional challenges include lack of interactivity, limited mentor access, poor compliance, and perceived ineffectiveness for career success Rubio et al., 2023, 1. Common barriers encompass resource constraints, manager time limitations, employee resistance, goal misalignment, inconsistent evaluation, and organizational culture issues Zafar, 2025, 2.

Over recent years, IDPs have evolved from paper-based, static tools to digital platforms like myIDP and AI-integrated systems offering real-time feedback, personalized paths, and mentor-AI hybrids, enhancing adaptability amid technological disruption Chang et al., 2024; Eason, Bruno, and Böttjer-Wilson, 2020, 7. This shift incorporates AI analytics for dynamic L&D, addressing prior static limitations Nurjaman, 2025; Oladele et al., 2025.

2.1.2. Recommender Systems

Recommender systems are advanced algorithms designed to filter and suggest relevant items from large datasets to users, mitigating information overload and enhancing personalization across various domains Raza et al., 2024; Roy and Dutta, 2022, 1. These systems integrate artificial intelligence to analyze vast datasets, including user profiles, behaviors, preferences, and item characteristics, thereby facilitating highly personalized recommendations such as adaptive development plans in employee training Oladele et al., 2025.

Overview and Types of Recommender Systems

RS are broadly categorized into three primary types: collaborative filtering, content-based filtering, and hybrid systems Alshbanat, Benhidour, and Kerrache, 2024; Raza et al., 2024; Roy and Dutta, 2022, 1. CF leverages collective user behavior, assuming similar users share preferences Hsieh et al., 2024. CBF matches items to user profiles based on feature similarities Roy and Dutta, 2022, 1. Hybrid systems combine both to address individual limitations, improving accuracy and diversity Alshbanat et al., 2024; Tey, Wu, Lin, and Chen, 2020.

Algorithms and Primary Approaches: Differences

Traditional algorithms include:

Recommendation Technique	Advantages	Disadvantages
Collaborative Filtering Alshbanat, Benhidour, and Kerrache, 2024; Hsieh et al., 2024	No content analysis needed; discovers serendipitous items	Cold start problem; sparsity; scalability issues Raza et al., 2024
Content-Based Filtering Roy and Dutta, 2022, 1; Tey, Wu, Lin, and Chen, 2020	Handles new users; explainable	Limited to similar items; new item problem
Hybrid Alshbanat, Benhidour, and Kerrache, 2024; Jomsri, Prangchumpol, Poonsilp, and Panityakul, 2023	Balances strengths; higher accuracy	Increased complexity

CF is divided into user-based, item-based, memory-based, and model-based Alshbanat et al., 2024; Tey et al., 2020. CBF uses techniques like TF-IDF and cosine similarity on item features Raza et al., 2024. Hybrids weight or switch methods for optimal results Hsieh et al., 2024.

Application Domains Beyond IDPs

Beyond individual development plans, RS power e-commerce, entertainment Gomez-Uribe and Hunt, 2015, 4; Raza et al., 2024, healthcare Raza et al., 2024, news, and social media Akkaya, 2025; Taghavi, Bentahar, Bakhtiyari, and Hanachi, 2017, 3. They drive sales, engagement, and retention in these areas Raza et al., 2024; Zou et al., 2024.

Suitability for Educational and Professional Development Contexts

Hybrid RS are most suitable for educational and professional development, combining CF for peer/learning similarities with CBF for skill gaps and career goals da Silva, Slodkowski, da Silva, and Cazella, 2022, 3; Hemmler, Rasch, and Ifenthaler, 2022, 1; Raj and Renumol, 2021, 1. In education, they personalize learning paths, courses, and content da Silva et al., 2022, 3; Multiple Authors, 2020; in professional contexts, they recommend training and IDPs based on performance and aspirations Jangda and ur Rahman Raazi, 2024; Lahoud, Moussa, Obeid, Khoury, and Champin, 2022, 7; Srivastava, Palshikar, Chaurasia, and Dixit, 2018, 3.

Impact on User Experiences

RS significantly boost engagement, satisfaction, retention, and outcomes: 60

2.1.3. Deep Learning

Fundamentals of Deep Learning

Deep learning represents a powerful subset of machine learning, utilizing multi-layered neural networks to learn intricate patterns from vast datasets, which makes it particularly adept at overcoming the aforementioned limitations of traditional recommender systems, such as scalability and sparsity challenges Hsieh et al., 2024; Sami, Adrousy, Sarhan, and Elmougy, 2024, 1. Core concepts include neurons, layers, activation functions, backpropagation for training via gradient descent, and loss functions to measure prediction errors. DL excels with unstructured data like images, text, and sequences due to its representation learning capability.

Advantages:

- Automatic feature extraction, reducing manual engineering.
- Superior performance on large-scale, high-dimensional data.
- Handles non-linearity and complex patterns effectively, improving accuracy in tasks like recommendation Sami et al., 2024, 1.

Limitations:

- Requires massive datasets and computational resources.
- Prone to overfitting without regularization.
- Interpretability issues.
- Vulnerability to adversarial attacks and data biases.

Key Architectures

DL architectures vary by data type and task. Key ones include:

Deep Learning vs. Traditional Machine Learning Techniques

Traditional ML relies on hand-crafted features and shallower models, while DL automates feature learning through deep hierarchies. Traditional methods scale poorly with data volume/complexity; DL thrives on ”big data” via end-to-end learning Hsieh et al., 2024.

Differences in Feature Extraction and Performance

DL differs fundamentally in feature extraction: Traditional ML requires explicit feature design (e.g., TF-IDF for text Roy and Dutta, 2022, 1), whereas DL learns features progressively—low-level to high-level—via convolutions, embeddings, or attention. This end-to-end approach minimizes information loss.

In performance, DL outperforms on scalability, sparsity (embeddings capture latent factors Sami et al., 2024, 1), and accuracy (e.g., hybrids beat traditional CF Alshbanat et al., 2024; Jomsri, Prangchumpol, Poonsilp, and Panityakul, 2023). However, it demands more compute/time for training.

Relevant Architectures for Recommender Systems

For RS, key DL architectures address CF/CBF limitations:

- Autoencoders: Model user-item interactions.

Architecture	Description	Key Components	Common Applications
Feedforward Neural Networks	Basic deep networks processing data in one direction	Fully connected layers, backpropagation	Classification, regression
Convolutional Neural Networks	Designed for grid-like data	Convolutional layers, pooling, filters	Image recognition, content-based recommendation
Recurrent Neural Networks/LSTMs/GRUs	Handle sequential data with memory	Recurrent connections, gates	Time-series, sequential recommendations
Autoencoders	Unsupervised networks for data compression/reconstruction	Encoder-decoder structure, latent space	Dimensionality reduction, denoising, collaborative filtering
Transformers	Attention-based models for sequences	Self-attention, positional encoding, multi-head attention	NLP, sequential recommendation
Graph Neural Networks	Operate on graph-structured data	Message passing, node embeddings	Social networks, knowledge graphs in RS Hsieh et al., 2024

Aspect	Traditional ML	Deep Learning
Feature Engineering	Manual	Automatic Addagarla, 2019, 1
Data Requirements	Smaller datasets suffice	Large datasets essential
Model Complexity	Simpler, interpretable	Complex, high capacity
Performance	Good for tabular/small data	State-of-the-art for unstructured/large-scale data Sami, Adrousy, Sarhan, and Elmougy, 2024, 1

- CNNs: Process textual/image content in CBF.
- RNNs/Transformers: Capture sequential user behavior.
- GNNs: Leverage graph data for hybrid RS Sami et al., 2024, 1.
- Hybrids combine these for superior personalization in IDPs Sami et al., 2024, 1.

These enable dynamic, adaptive recommendations aligning with IDP evolution toward AI-integrated systems Chang et al., 2024.

2.1.4. Deep Learning in Recommender Systems

Deep learning has emerged as a transformative paradigm in recommender systems, fundamentally altering how user preferences and item characteristics are modeled by leveraging multi-layered neural networks to discern intricate patterns in vast, complex datasets Zhang, Yao, Sun, and Tay, 2019b, 1; Zhang, Quintana, and Miller, 2024.

Integration of Deep Learning Techniques into Recommender Systems

Deep learning integrates into RS through specialized architectures that address traditional limitations like sparsity, cold starts, and scalability. Neural Collaborative Filtering replaces matrix factorization's inner product with multi-layer perceptrons to capture non-linear user-item interactions Addagarla, 2019, 1; Grida, Fayed, and Hassan, 2020, 12. Autoencoders enable unsupervised learning of latent representations for collaborative filtering, while CNNs process multimodal content in content-based filtering

Zhang et al., 2024. RNNs, LSTMs, GRUs, and Transformers model sequential user behaviors for next-item prediction Bathla, Rani, and Aggarwal, 2018, 10; Stergiopoulos, Vassilakopoulos, Tousidou, and Corral, 2024, 8. Graph Neural Networks exploit relational data in social or knowledge graphs Zhang et al., 2024. Hybrid deep models combine these with traditional methods for enhanced accuracy Grida et al., 2020, 12.

Comparative Analysis with Traditional Recommender Methods

Deep learning outperforms traditional methods on large-scale, unstructured data by automating feature extraction and handling non-linearity, but requires more data and compute Akkaya, 2025; Gheewala, Xu, and Yeom, 2025.

Aspect	Traditional Methods	Deep Learning Methods
Feature Engineering	Manual	Automatic Addagarla, 2019, 1
Data Requirements	Smaller, structured	Large-scale, unstructured Bathla, Rani, and Aggarwal, 2018, 10
Handling Sparsity/Cold Start	Poor	Strong Zhang, Quintana, and Miller, 2024
Scalability	Limited	High Grida, Fayed, and Hassan, 2020, 12
Performance on Complex Patterns	Linear assumptions	Superior accuracy Gheewala, Xu, and Yeom, 2025; Zhang, Yao, Sun, and Tay, 2019b, 1

Traditional methods like memory-based CF excel on small datasets but falter on sparsity; DL hybrids mitigate this Dong, Li, and Schnabel, 2023; Verma, 2020, 1.

Successful Examples and Case Studies

Prominent real-world implementations demonstrate DL's impact:

These cases highlight DL's deployment in production, with A/B tests showing 5-30% lifts in metrics like precision/recall Gheewala et al., 2025; Tewari, Rautela, Sharma, and Garg, 2024.

Performance Improvements in Real-World Applications

DL enhances RS by 4-47% in precision/recall/NDCG over baselines, boosting engagement, personalization, and handling multimodal/sequential data Gheewala et al., 2025; Taha, Yoo, and Taha, 2024. It mitigates biases via better representations, increases sales/stickiness, and supports real-time adaptation Rane, Ömer Kaya, Mallick, and Rane, 2024; Zhang et al., 2024.

2.1.5. Data Collection Methods

Effective data collection is paramount for training robust deep learning models in recommender systems, involving the systematic acquisition and preprocessing of diverse data types such as explicit user ratings, implicit behavioral signals, and rich contextual information.

2.1.6. Data Preprocessing

This foundational step ensures data quality, consistency, and suitability for complex deep learning architectures in recommender systems, directly impacting model accuracy, convergence, and the reliability of individual development plan recommendations Mumuni and Mumuni, 2024, 2.

Data preprocessing is critical for deep learning models because neural networks are highly sensitive to input scales, noise, and sparsity—common in RS datasets with implicit feedback, cold starts, and

Platform	DL Techniques	Key Outcomes Gangadharan et al., 2024; Steck et al., 2021, 3; Zhang, Yao, Sun, and Tay, 2019b, 1
Netflix	Various DNNs	80% of views from recommendations; outperforms baselines across tasks Steck et al., 2021, 3
YouTube	DNN candidate generation + ranking	60%+ video clicks from recs Zhang, Yao, Sun, and Tay, 2019a, 1
Google Play	Wide & Deep model	Significant gains over traditional Zhang, Yao, and Sun, 2022
Yahoo News	RNN-based	Improved online performance Zhang, Yao, Sun, and Tay, 2019b, 1
Facebook	DLRM	Handles categorical features at scale Naumov et al., 2019
Pinterest	TransAct	Real-time sequential modeling Xia et al., 2023

heterogeneous features. Poor preprocessing leads to unstable gradients, slow convergence, vanishing/exploding gradients, and suboptimal performance on large-scale, unstructured data Lin et al., 2024; Zhu, Jiang, and Alonso, 2024; Đuričić, Kowald, Lacić, and Lex, 2023. Proper techniques mitigate these by standardizing inputs, enabling effective feature learning, and addressing RS challenges like data sparsity Grida et al., 2020, 12.

Key Preprocessing Techniques

- **Cleaning:** Removing duplicates, outliers, noise – Ensures data integrity, prevents model bias from anomalies Zhu et al., 2024.
- **Normalization/Scaling:** Min-max or z-score for dense features – Facilitates stable gradient descent, faster convergence in neural networks Zhu et al., 2024.
- **Missing Data Imputation:** Mean/median fill, advanced methods like KNN or model-based – Handles sparsity/cold starts in user-item matrices Zhu et al., 2024.
- **Categorical Encoding:** One-hot, label encoding, or low-dim embeddings for IDs, text, and other nominal features Zhu et al., 2024.

2.1.7. Model Evaluation Methods

Effective evaluation of recommender systems is crucial for ensuring their utility and performance, particularly in specialized domains like Individual Development Plans. Various metrics are employed to assess different aspects of a recommender system's effectiveness, from prediction accuracy to the relevance and ranking quality of recommendations Jadon and Patil, 2023, 2024.

Common Evaluation Metrics for Recommender Systems

Commonly used evaluation metrics for recommender systems can be broadly categorized into those measuring prediction accuracy and those assessing ranking quality or relevance:

- **Mean Absolute Error**: Quantifies the average absolute difference between predicted ratings and actual user ratings. Lower MAE values indicate better prediction accuracy Fakhfakh, □□, and Ben, 2017, 12; Payandenick, Wang, Othman, and Payandenick, 2025; Raghavendra, K.C, and R, 2018, 10.

- **Root Mean Squared Error**: Similar to MAE, RMSE measures the square root of the mean of the squared differences between predicted and actual ratings. RMSE places a greater emphasis on larger errors, and lower values signify better prediction accuracy Gheewala et al., 2025; Nawi, Noah, and Zakaria, 2020, 2; Osman, Noah, Darwich, and Mohd, 2021, 3; Papaleonidas, Pimenidis, and Iliadis, 2019; Payandenick et al., 2025.
- **Precision**: Indicates the proportion of recommended items that are actually relevant to the user Fakhfakh et al., 2017, 12; Osman et al., 2021, 3; Raghavendra et al., 2018, 10. Precision@k evaluates the proportion of relevant items among the top-k recommendations, especially when the cost of false positives is high Jadon and Patil, 2023.
- **Recall**: Measures the proportion of relevant items that were successfully recommended by the system out of all relevant items Fakhfakh et al., 2017, 12; Osman et al., 2021, 3; Raghavendra et al., 2018, 10. Recall@k prioritizes the model's ability to capture all relevant items within the top-k suggestions, particularly when missing a relevant recommendation is costly Jadon and Patil, 2023.
- **F1-score**: The harmonic mean of precision and recall, providing a balanced measure of a system's accuracy Fakhfakh et al., 2017, 12; Qin et al., 2025.
- **Normalized Discounted Cumulative Gain (nDCG)**: Used when the ranking relevance matters, evaluating the quality of recommendations in a list by considering the position of highly relevant items Jadon and Patil, 2023.
- **Mean Reciprocal Rank**: Evaluates the effectiveness of a system in returning the first relevant item in a ranked list, emphasizing the importance of the top-most recommendation Jadon and Patil, 2023.
- **Area Under the Receiver Operating Characteristic Curve**: A metric for binary classification problems, often used in recommender systems to evaluate the ability to distinguish between relevant and irrelevant items Qin et al., 2025.
- **Hit@N**: Indicates whether any of the top-N recommended items are found in the test set for a user Cortes, 2021; Qin et al., 2025.

Comparative Analysis of Evaluation Metrics

The choice of an appropriate evaluation metric is critical, as different metrics may favor different algorithms and capture distinct aspects of a recommender system's performance Gunawardana and Shani, 2009, 100; Jadon and Patil, 2023, 2024. While MAE and RMSE are useful for assessing the accuracy of rating predictions, they may not truly reflect the user's experience, as users typically receive ranked lists rather than individual rating predictions Nawi et al., 2020, 2; Osman et al., 2021, 3; Papaleonidas et al., 2019. Precision and Recall, on the other hand, are better indicators of the relevance and quality of the recommended items within these ranked lists Nawi et al., 2020, 2; Osman et al., 2021, 3; Papaleonidas et al., 2019; Payandenick et al., 2025. Metrics such as nDCG, MRR, and ARHR@k are specifically designed to evaluate the effectiveness of the order in which recommendations are presented Jadon and Patil, 2023. Beyond technical performance, business-oriented metrics like Click-Through Rate, conversion rate, and user retention are vital for aligning the recommender system's performance with organizational objectives Askarbekuly and Luković, 2024; Silveira, Zhang, Xiao, Liu, and Ma, 2017, 5.

Issues in Model Evaluation and Validation

Offline evaluation of recommender systems presents significant challenges, making it difficult to accurately assess true progress Schnabel, 2022. A major issue stems from the fact that user behavior data is observational rather than experimental, leading to various biases Chen et al., 2020. These include:

- **Selection Bias**: Occurs when data is influenced by how users self-select or how the existing recommendation system operates Chen et al., 2020; Schnabel, Swaminathan, Singh, Chandak, and Joachims, 2016.

- **Exposure Bias**: Arises because users only interact with items they are exposed to, which might not represent their full preferences Chen et al., 2020; Pereira, Said, and Santos, 2025.
- **Popularity Bias**: Leads to algorithms and metrics disproportionately favoring popular items, potentially hindering the discovery of niche content and affecting diversity Chen et al., 2022a, 3; Mena-Maldonado, Cañamares, Castells, Ren, and Sanderson, 2020; Sun, 2023.

These biases can distort empirical measurements, making the interpretation and comparison of results across experiments challenging and potentially leading to discrepancies between offline evaluation results and actual online metrics Bellogín, Castells, and Cantador, 2017, 6; Castells and Moffat, 2022, 2; Chen et al., 2020. Furthermore, the reliability of sampling strategies used in offline evaluation can also be a concern Pereira et al., 2025.

Most Appropriate Evaluation Metrics for IDP Recommender Systems

For IDP recommender systems, which fall under educational contexts, the evaluation should extend beyond standard technical metrics to include outcome-based measures that focus on pedagogical effectiveness Askarbekuly and Luković, 2024. In addition to traditional metrics like AUC, Recall@N, Precision@N, F1@N, Hit@N, NDCG@N, and MAP@N used for course recommendation Qin et al., 2025, IDP recommenders should prioritize metrics that assess:

- **Learning Gain or Improvement**: Measures the increase in knowledge or skills after engaging with recommended educational resources, often assessed through pre-tests and post-tests Askarbekuly and Luković, 2024.
- **Concept Mastery and Skill Acquisition**: Evaluates how well learners have mastered specific concepts or acquired particular skills recommended by the system, typically through assessments Askarbekuly and Luković, 2024.

These learning-centered metrics are crucial because they directly align with the core purpose of IDPs: facilitating targeted training and addressing skill gaps for sustained career growth. While standard metrics like precision and recall assess the relevance of recommendations, outcome-based metrics directly measure the impact on employee development and organizational goals. Business metrics such as engagement and retention are also relevant, indicating the system's success in maintaining user interest and facilitating ongoing usage Askarbekuly and Luković, 2024.

Explainability of Recommender Models

The increasing complexity of recommender systems, particularly those employing deep learning, has brought the concept of explainability to the forefront. Moving beyond mere accuracy, explainable AI in recommender systems aims to provide transparency and build user trust, leading to better user experience and system adoption Pasrija and Pasrija, 2024, 2; Paz-Ruza, Alonso�Betanzos, Guijarro�Berdiñas, Cancela, and Eiras�Franco, 2023.

Importance and Need for Model Explainability

The need for explainability in recommender systems is driven by several critical factors. Modern recommender systems often function as "black boxes," making it challenging for users to understand why specific recommendations are generated Paz-Ruza et al., 2023. This lack of transparency can lead to diminished user experience, reduced trust, and potentially harmful outcomes Paz-Ruza et al., 2023. By providing explanations, these systems become more understandable, which is vital for fostering user trust and increasing the likelihood of engagement with suggested items Abdollahi, 2017; Guesmi et al., 2023a, 22. Explainable recommendations help users to comprehend the rationale behind suggestions, thereby improving their decision-making process Cramer et al., 2008, 5.

Furthermore, explainability is crucial for system designers. Explanations facilitate better system debugging by highlighting the features or data points that most influence a recommendation, allowing developers to identify and address biases or errors within the model Zhang and Chen, 2020, 1. In sensitive applications, such as those with ethical implications, explainable recommender systems are essential for ensuring fairness and accountability Guttmann and Ge, 2024.

Introduction to SHAP and LIME Frameworks

SHAP and LIME are widely used model-agnostic frameworks that provide insights into the predictions of complex machine learning models, including those used in recommender systems Said, 2024; Salih et al., 2023

- **SHAP**: This method is grounded in cooperative game theory, attributing the contribution of each feature to a model's prediction Zhong and Negre, 2022. SHAP values offer a unified measure of feature importance, indicating how much each feature impacts a prediction relative to a baseline Ragodos, Wang, Feng, Yu, and Hu, 2024; Zhong and Negre, 2022 SHAP ensures properties like local accuracy and consistency, meaning that if a feature's importance increases, its SHAP value will not decrease Ragodos et al., 2024. It can provide both local (individual prediction) and global (overall model behavior) explanations Zhong and Negre, 2022.
- **LIME**: LIME generates explanations for individual predictions by constructing a locally faithful, interpretable model around the specific instance being explained e Oliveira et al., 2024, 3; Said, 2024 It works by perturbing the input data, observing the model's output on these perturbations, and then training a simpler, interpretable model (e.g., a linear regressor) on this locally weighted synthetic dataset Brughmans, Melis, and Martens, 2024, 3; e Oliveira et al., 2024, 3 The coefficients of this simple model then indicate the importance of each feature for that particular prediction Ragodos et al., 2024.

Implementation and Effectiveness of Explainability Methods in Recommender Systems

The implementation of explainability methods like SHAP and LIME in recommender systems involves integrating these frameworks to produce explanations alongside recommendations M and Basheer, 2025, 6. Their effectiveness is evaluated based on their ability to generate meaningful, actionable, and user-understandable explanations that can enhance user experience and trust Chen, Zhang, and Wen, 2022b; Wardatzky, Inel, Rossetto, and Bernstein, 2025, 4

For example, by using collaborative signals and large language models, frameworks like XRec can analyze and explain model behavior after recommendations have been generated, focusing on specific decisions Ma, Ren, and Huang, 2024; Mishra, Bibo, van Engelen, and Schaapman, 2025 This post-hoc approach aligns with efforts to provide local explanations, similar to LIME Mishra et al., 2025. While SHAP is often chosen for its theoretical guarantees, LIME was one of the earliest model-agnostic post-hoc explainers Ragodos et al., 2024; Zhong and Negre, 2022 Research suggests that further advancements are needed to make deep learning models more explainable for recommendations, especially in understanding "what makes something recommended versus other options" Saeed and Omlin, 2022. Extracting explanations from latent factor models by training association rules on the outcomes of a matrix factorization model can balance interpretability and accuracy Areeb et al., 2023, 6.

How Do Explainability Methods Enhance User Trust and Model Transparency?

Explainability methods significantly enhance user trust and model transparency by converting opaque algorithmic processes into understandable insights Guesmi et al., 2023b; Zhang and Chen, 2020, 1 When users receive explanations for recommendations, they are better equipped to comprehend why certain items are suggested, which builds confidence in the system's capabilities Abdollahi, 2017; Guesmi et al., 2023a, 22 Transparency, achieved through clear explanations, removes the "black box" perception of recommender systems Gedikli, Jannach, and Ge, 2014, 4. For instance, if an explanation highlights that a recommendation is based on a user's past purchases or ratings of similar items, the user is more likely to accept and trust the suggestion Cramer et al., 2008, 5.

By understanding the logic behind recommendations, users can form more accurate mental models of how the system operates Guesmi et al., 2023b. This allows users to identify potential "filter bubbles" and understand whether a recommendation is personalized or random Areeb et al., 2023, 6. Research consistently shows that providing explanations increases user trust and satisfaction Nunes and Jannach, 2017; Pu and Chen, 2007, 6

What Are the Specific Benefits and Limitations of LIME and SHAP Methods?

oprule	**Benefits**	**Limitations**
extbfLIME	**Model-agnostic**: Can be applied to any black-box model e Oliveira et al., 2024, 3; Said, 2024. **Local explanations**: Provides insights for individual predictions, which is crucial for user understanding e Oliveira et al., 2024, 3. **Speed**: Often faster than computing exact Shapley values e Oliveira et al., 2024, 3. **Versatility**: Works with various data types (tables, images, text) and can use different features than the model's training features e Oliveira et al., 2024, 3.	**Stability issues**: Explanations can be unstable and vary depending on the samples used for perturbation, making them manipulable Brughmans, Melis, and Martens, 2024, 3; e Oliveira et al., 2024, 3. **Parameter dependence**: Poor choices in parameters can lead to missing important features e Oliveira et al., 2024, 3. **Neighborhood definition**: Defining the local neighborhood and sampling process remains an open problem e Oliveira et al., 2024, 3. **Lack of local accuracy**: Unlike SHAP, LIME does not guarantee that feature contributions sum to the model prediction difference Ragodos, Wang, Feng, Yu, and Hu, 2024.
extbfSHAP	**Model-agnostic**: Applicable to a wide range of models Ragodos, Wang, Feng, Yu, and Hu, 2024; Said, 2024. **Theoretical soundness**: Based on game theory, ensuring fair calculation of each feature's contribution e Oliveira et al., 2024, 3; Henninger and Strobl, 2024, 1; Zhong and Negre, 2022. **Consistency and local accuracy**: Guarantees that if a model changes to increase a feature's importance, its SHAP value will not decrease; contributions sum to the prediction difference Ragodos, Wang, Feng, Yu, and Hu, 2024. **Local and global explanations**: Can provide both detailed individual and overall explanations Said, 2024.	**High computational cost**: Exact Shapley values are often intractable and require approximations e Oliveira et al., 2024, 3; Ragodos, Wang, Feng, Yu, and Hu, 2024. **Requires representative data**: Often needs training data or representative samples for accuracy Zhong and Negre, 2022. **Interpretation complexity**: Outputs can be complex for non-experts e Oliveira et al., 2024, 3.

How Have Explainability Techniques Improved User Engagement and Acceptance in Recommender Systems?

Explainability techniques have a significant impact on user engagement and acceptance by making recommender systems more transparent and trustworthy (Pasrija & Pasrija, 2024, 2; Zhang & Chen, 2020, 1) When explanations accompany recommendations, users are more likely to interact with the suggested items, leading to improved user activity and reduced decline in engagement due to poor recommendations Pasrija and Pasrija, 2024, 2. Studies indicate that providing appropriate explanations can increase user perception of recommendation quality by a notable margin Pasrija and Pasrija, 2024, 2.

Moreover, explainable recommender systems contribute to increased user satisfaction and user retention Pu and Chen, 2007, 6. Users who understand the "why" behind recommendations feel more in control and less manipulated by the system Abdollahi, 2017. This enhanced understanding empowers users to make better decisions and even provide more targeted feedback, which can further improve the system's performance Wardatzky et al., 2025, 4. Ultimately, by fostering trust and comprehension, explainability techniques transform passive users into engaged participants, leading to greater acceptance and a more positive overall experience with the recommender system Cramer et al., 2008, 5.

Future Research Directions
The field of Individual Development Plan recommender systems, particularly those leveraging deep learning, offers numerous avenues for future research and improvement. Addressing existing gaps and integrating emerging technologies can significantly enhance their effectiveness, personalization, and ethical considerations.

Potential Improvements and Research Gaps

Current research highlights several areas for improvement and critical gaps in IDP recommender systems:

- **Multidimensional Evaluation Frameworks**: A significant gap exists in the evaluation of educational recommender systems, which primarily focuses on accuracy rather than pedagogical effectiveness. Future research should develop multidimensional evaluation frameworks that assess the actual impact of recommendations on teaching and learning processes, going beyond traditional accuracy metrics da Silva et al., 2022, 3.
- **Reliable Data Utilization**: Most studies rely on learners' explicit data and evaluations. Future work should investigate more reliable data types, such as implicit actions performed by learners, to generate recommendations that align better with individual learning paces and motivations Souabi, Retbi, Idrissi, and Bennani, 2021, 5.
- **Social Learning Integration**: There is a noticeable lack of studies on recommender systems adapted for social learning networks. Research into incorporating community detection within recommender systems could lead to more tailored recommendations for groups of learners with shared interests or characteristics Souabi et al., 2021, 5.
- **Organizational Perspective in Workplace Learning**: Studies often overlook the organizational perspective in workplace learning. Future research should focus on how IDP recommender systems can align with organizational goals by examining input from supervisors and managers, and evaluating systems based on categorized workplace learning goals Hemmler et al., 2022, 1.
- **Dynamic Job Market Alignment**: Existing course recommender systems frequently neglect the rapidly changing demands of the job market. Research is needed on unsupervised skill extraction from job listings, course descriptions, and resumes, and on developing metrics to align recommendations with evolving job market requirements and user career goals (Frej, Dai, Montariol, Bosselut, & Käser, 2024a, 2024b)
- **Privacy and Ethical Considerations**: The balance between personalization and data protection presents a fundamental tension. The centralized collection of sensitive educational data raises ethical concerns and compliance challenges. Future research must focus on privacy-preserving techniques and ethical frameworks Tertulino, 2025.
- **User Personalization in Federated Settings**: While federated learning offers privacy benefits, personalized recommendation models that capture heterogeneous user preferences in decentralized and non-IID (non-independently and identically distributed) data settings remain underexplored Zhang et al., 2025.

Emerging Trends and Technologies

Several emerging trends and technologies are poised to significantly impact the development and efficacy of IDP recommender systems:

- **Generative AI**:
  - There is growing interest in using generative language models for crafting personalized learning paths Bayly-Castaneda, Ramirez-Montoya, and Morita-Alexander, 2024.
  - LLMs can be utilized for advanced skill extraction from diverse textual sources like job listings and resumes, facilitating better matching and recommendations Frej et al., 2024b.

- Generative job recommendation systems powered by LLMs can create suitable job descriptions and offer personalized job-seeking experiences Qin et al., 2025. These systems can also leverage graph information to understand behavioral semantics for personalized job recommendations Qin et al., 2025.
- LLMs are also proving transformative in personalized career guidance systems within vocational education Duan and Wu, 2024, 2.

Federated Learning:

- - FL is emerging as a critical solution for user privacy in recommender systems by enabling model training on local devices without centralizing raw data Harasic, Keese, Mattern, and Paschke, 2023, 4; Kaur, Gujar, and Jain, 2024 This addresses the privacy paradox inherent in personalized education Tertulino, 2025.
  - FL can be applied to generate personalized, context-aware, and sequential recommendations while preserving data privacy Kaur et al., 2024.
- **Ethical AI**:
  - Future research and practice should prioritize the development of ethical frameworks, guidelines, and policies to ensure recommender systems operate responsibly, respect learner privacy, and promote equitable access to educational opportunities Oussouaddi, Hassouny, and Ismaili, 2023.
  - Ethically aligned personalization, which integrates fairness checks, cultural diversity, and governance structures into the system architecture, is gaining importance Damera, 2025, 4.
  - Fair federated recommendation learning is essential to characterize and mitigate the impact of system and data heterogeneity on fairness Maeng et al., 2022.

Combination of Symbolic and Machine Learning Techniques: Advancements in systems that combine symbolic reasoning with machine learning, particularly semantic-based recommender systems, are expected to enhance explainability and robustness Broisin, 2020.

- **Context-aware and Deep Learning-based Recommendations**: These advanced techniques are considered more efficient than traditional methods for future e-learning applications Salau et al., 2022, 23.
- **Multimodal Recommendation**: Integrating diverse data types, such as text, images, and video, into recommendations is a promising direction for federated recommender systems Li et al., 2024b.

How Emerging Trends Could Further Enhance IDP Recommender Systems

These emerging trends offer significant potential to enhance IDP recommender systems:

- **Hyper-personalization with Generative AI**: LLMs can move beyond static recommendations to dynamically generate highly personalized learning content, job descriptions, and career advice that adapts in real-time to an individual's evolving skills, goals, and the fluctuating job market. This dynamic capability can make IDPs truly adaptive (Duan & Wu, 2024, 2; Qin et al., 2025)
- **Privacy-Preserving Personalization with Federated Learning**: By implementing FL, IDP recommender systems can offer deep personalization without compromising the sensitive personal and performance data of employees. This can foster greater trust and adoption within organizations, especially given stringent data protection regulations (Kaur et al., 2024; Tertulino, 2025)
- **Fair and Equitable Development with Ethical AI**: Integrating ethical AI principles will ensure that IDP recommendations are free from biases related to gender, race, or other protected characteristics. This promotes equitable opportunities for all employees and builds organizational trust, ensuring that the system is a tool for growth, not discrimination (Damera, 2025, 4; Oussouaddi et al., 2023)
- **Holistic Skill Development with Multimodal Integration**: Combining various data types (e.g., performance reviews, project outcomes, online course interactions, certifications) through multimodal recommender systems can create a more comprehensive view of an individual's skill profile, leading to more accurate and impactful IDP recommendations.

Critical Research Gaps and How Future Research Can Address Them

Critical research gaps that currently exist in this field and how future research can address them include:

- **Bridging the Evaluation Gap**: Future research should focus on developing and validating new evaluation metrics and methodologies that specifically measure the *pedagogical effectiveness* and *learning outcomes* of IDP recommendations, rather than just predictive accuracy. This could involve longitudinal studies tracking skill acquisition and career progression da Silva et al., 2022, 3.
- **Robust Data Strategies for Dynamic Environments**: Research needs to explore advanced techniques for incorporating implicit behavioral data and real-time feedback into IDP recommender systems. This would address data sparsity challenges and allow systems to adapt more quickly to dynamic individual and organizational needs (Roy & Dutta, 2022, 1; Souabi et al., 2021, 5)
- **Contextual Understanding in Recommendations**: More research is required to integrate a deeper understanding of social learning contexts and organizational dynamics into recommender algorithms. This includes developing models that account for peer influence, team-based learning, and specific corporate culture or policy objectives Hemmler et al., 2022, 1.
- **Proactive Skill Development for Future Job Markets**: A key gap is the ability of IDP recommenders to proactively identify and recommend skills for future job markets. Future research should leverage advanced LLMs and natural language processing to continuously analyze job market trends and skill requirements, translating these into actionable, forward-looking IDP suggestions (Frej et al., 2024a, 2024b)
- **Ethical AI Implementation and Governance**: Research is crucial in developing practical, implementable ethical AI frameworks for IDP systems, focusing on bias detection and mitigation, transparency in decision-making, and user control over data and recommendations. This also extends to developing governance models for AI in HR technologies (Damera, 2025, 4; Oussouaddi et al., 2023)
- **Personalization in Decentralized Data Settings**: Further investigation into personalized recommendation models within federated learning environments is needed, especially concerning how to effectively handle heterogeneous user preferences and non-IID data distribution while maintaining privacy Zhang et al., 2025.
- **Explainability of Complex Deep Learning Models**: While progress has been made with methods like SHAP and LIME, research needs to develop more intuitive and user-friendly explainability techniques for the increasingly complex deep learning architectures used in IDP recommenders. The goal is to provide explanations that are both accurate and easily understood by employees and managers, fostering trust and enabling informed decision-making.

Table 2.1: Summary of Related Works for the IDP Recommender System Study

oprule extbfAuthor(s) & Year	**Title**	**Methodology**	**Key Findings and Relevance**
Vanderford et al. (2018)	Use of IDPs for doctoral trainees	Survey study	IDPs improve self-assessment and career planning; foundational to this study.
Sahoo et al. (2019)	Deep learning in health recommender systems	Deep collaborative filtering (RBMs)	Deep learning improves recommendation accuracy; supports choice of deep learning.
Mu (2018)	Survey of deep learning recommender systems	Literature review	Validates deep learning as superior for complex recommendation tasks.
Li et al. (2024)	Knowledge graph-enhanced recommender systems	Attention and residual networks	Knowledge graphs improve personalization; important for IDP mappings.
Chen & Zhong (2024)	GCN-based course recommendation system	Graph Convolutional Networks (GCNs)	GCNs model complex relationships; informs modelling employee competencies.
Ertürkman et al. (2019)	Personalized health management platforms	Collaborative platform development	Personalized plans outperform generic; supports personalizing IDPs.
Gulzar et al. (2018)	Personalized course recommender systems	Hybrid recom- mendation methods	Combining methods improves engagement; supports multi-input IDP systems.
Dabak et al. (2022)	Career development for Gen Y and Z	Developmental cycle framework	Adaptive planning needed for evolving careers; supports dynamic IDP updates.
Ghaffar et al. (2022)	Impact of personality traits on planning	Empirical study in finance	Personality traits influence decision-making; suggests incorporating traits into IDPs.
Wang et al. (2020)	Employee training course recommendations	Bayesian variational network modeling	Career goals improve recommendations; aligns with dynamic IDP needs.
Bui et al. (2016)	Text classification from PDFs	Multi-pass sieve technique	Text extraction improves preprocessing; useful for automating employee documents.
Viani et al. (2019)	Clinical event extraction with RNNs	Supervised learning	RNNs extract structured info from text; supports skill extraction automation.
Lin et al. (2018)	Sparse linear method for recommendation	L0 regularization technique	Improved recommendation precision; useful for refining IDP suggestions.

3. Theoretical Framework

4. Research Design

4.1. Overview of the Research Design

The research design for this study focuses on the development, training, and evaluation of a deep learning-based recommender system to generate personalized Individual Development Plans (IDPs) for employees.

The design follows a modular, data-driven approach, where raw employee data undergoes preprocessing, feature engineering, and is subsequently used to train and evaluate several deep learning models. These models are trained independently to predict suitable development plans based on employee skills, evaluations, learning goals, and performance histories.

Four primary deep learning models are considered: Neural Collaborative Filtering (NCF), Recurrent Neural Networks (RNN), Graph Neural Networks (GNN), and Transformer-based architectures. Each model is trained, validated, and evaluated independently using consistent data splits and evaluation metrics. The goal is not to ensemble or aggregate model outputs, but rather to identify the model that offers the best predictive performance according to established evaluation criteria such as Precision, Recall, F1-Score, and AUC-ROC.

The system architecture is organized into five major phases:

Table 4.1: Models and Techniques Applied Across Research Phases

Model/Technique	**Purpose**	**Research Phase**
Data Collection and Preprocessing	Collect, clean, normalize employee data (skills, feedback, evaluations)	**Data Collection**
Principal Component Analysis (PCA)	Reduce feature dimensionality and highlight important patterns	**Feature Engineering**
Autoencoder	Denoise and compress employee data for feature extraction	**Feature Engineering**
Clustering (e.g., K-Means)	Group employees based on similar goals and skill gaps	**Feature Engineering**
Neural Collaborative Filtering (NCF)	Predict personalized skill development plans from interaction data	**Model Training**
RNN / LSTM	Model sequential learning behaviors and engagement history	**Model Training**
Graph Neural Network (GNN)	Model relationships between skills and career objectives via Knowledge Graphs	**Model Training**
Transformer Model (BERT)	Understand textual employee career goals and aspirations	**Model Training**
Precision, Recall, F1, AUC-ROC	Evaluate model effectiveness and recommendation quality	**Model Evaluation**
k-Fold Cross-Validation	Validate model robustness and prevent overfitting	**Model Evaluation**
Explainability (SHAP, LIME)	Interpret model predictions and detect potential biases	**Model Evaluation**
Feedback Integration	Use best model to generate IDPs; integrate feedback for retraining	**Deployment**

This research design ensures both methodological rigour and flexibility, allowing for fair comparison between models and supporting the goal of building an adaptable, scalable, and accurate IDP recommender system.

4.2. System Architecture

The system architecture for the Individual Development Plan (IDP) Recommender is designed to modularly process employee data, engineer useful features, independently train multiple deep learning models, evaluate their performance, and deploy the best model to generate personalized development plans.

Figure 4.1 illustrates the overall architecture of the system. The flow begins with raw employee data and proceeds through several critical stages: preprocessing, feature engineering, model training, evaluation, and selection. A feedback loop is incorporated to ensure that the model continues to improve over time based on new data and organizational feedback.

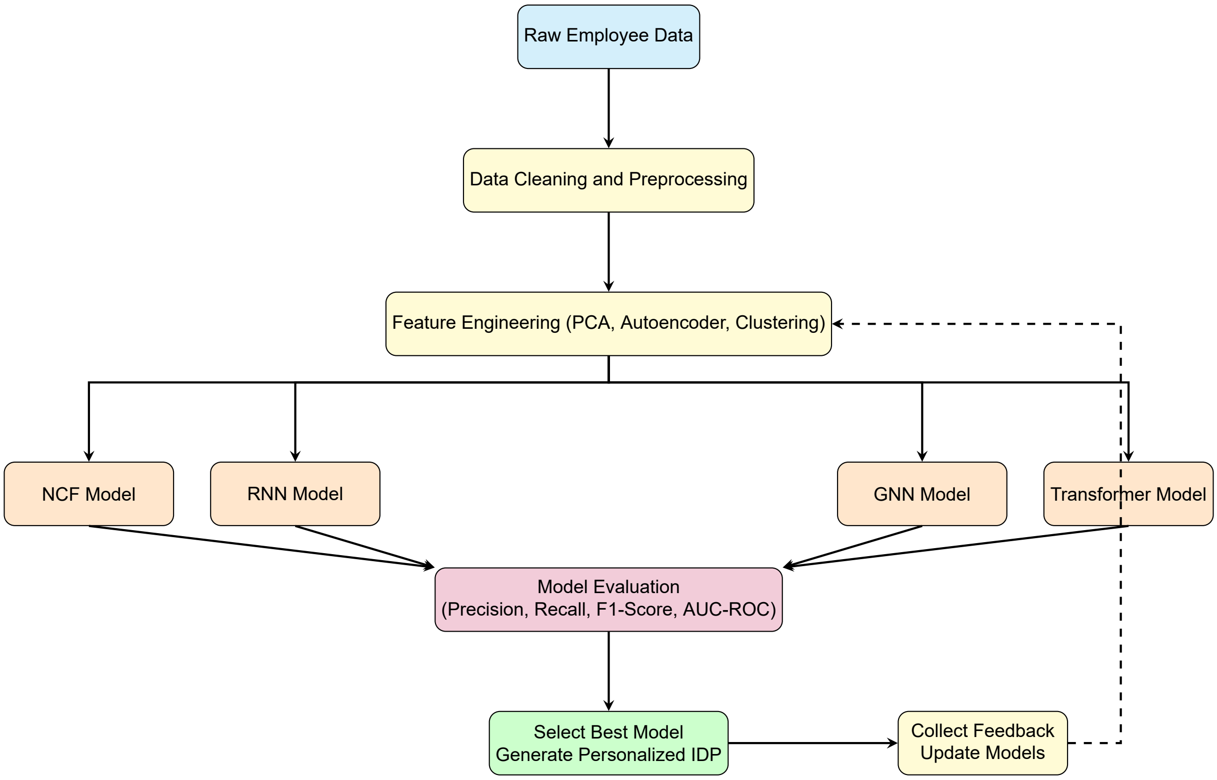

Figure 4.1: System Architecture for the Deep Learning-based IDP Recommender. The models (NCF, RNN, GNN, Transformer) are independently trained and evaluated to identify the best-performing model for generating personalized IDPs.

The system is composed of the following key modules:

4.2.1. Data Collection and Preprocessing

Employee data—including skills assessments, performance reviews, career goals, and feedback—is collected from organizational databases and manual sources. This raw data often contains inconsistencies, missing values, and unstructured formats. Therefore, a preprocessing module cleans the data by handling missing entries, standardizing formats, and preparing the inputs for feature engineering.

4.2.2. Feature Engineering

Feature engineering transforms raw data into structured formats suitable for deep learning models. Dimensionality reduction is performed using Principal Component Analysis (PCA) and Autoencoders to extract the most informative features while minimizing redundancy. Clustering techniques are also applied to group employees by similar career paths and competencies, providing additional categorical features.

4.2.3. Model Training

Four deep learning models—Neural Collaborative Filtering (NCF), Recurrent Neural Networks (RNN), Graph Neural Networks (GNN), and Transformer-based models—are independently trained using the engineered feature sets. Each model is designed to predict the most appropriate Individual Development Plan recommendations for a given employee profile.

4.2.4. Model Evaluation

The models are evaluated separately using consistent data splits (training, validation, and testing). Evaluation metrics include Precision, Recall, F1-Score, and AUC-ROC to provide a comprehensive view of each model's predictive performance. The best-performing model based on these metrics is selected for deployment.

4.2.5. Deployment and Feedback Integration

Once the best model is selected, it is used to generate personalized IDPs for employees. A feedback mechanism is implemented to collect post-recommendation evaluations from employees and supervisors. This feedback is used to update the training dataset and periodically retrain models, ensuring that the system adapts to evolving career development needs within the organization.

4.3. Module Descriptions

The system is divided into several specialized modules, each responsible for a key part of the IDP recommendation pipeline. This modular design ensures scalability, maintainability, and clear responsibilities across the system components.

4.3.1. Data Preprocessing Module

The Data Preprocessing Module is responsible for cleaning and standardizing the raw employee datasets. Key operations performed include:

- Handling missing data through imputation or removal strategies.
- Standardizing categorical variables (e.g., job titles, departments).
- Normalizing numerical features (e.g., skill ratings, evaluation scores).
- Parsing and cleaning unstructured textual fields, such as career goal descriptions.

This preprocessing ensures that the data fed into the Feature Engineering Module is consistent and machine-readable.

4.3.2. Feature Engineering Module

The Feature Engineering Module focuses on enhancing the dataset by extracting relevant features:

- **Principal Component Analysis (PCA):** Reduces the dimensionality of numerical data, capturing the most informative features.
- **Autoencoders:** Learn compact, noise-resistant representations of the original employee feature space.
- **Clustering (e.g., K-Means):** Groups employees with similar learning styles, skill gaps, or career aspirations, generating new categorical features.

The output from this module serves as the input for the model training phase.

4.3.3. Neural Collaborative Filtering (NCF) Model

The NCF Model extends traditional collaborative filtering by using a multi-layer perceptron (MLP) to learn nonlinear user-item interaction patterns. In this context, employees are treated as users and development plans as items. The model predicts the most suitable development plans based on historical training, assessment results, and career progressions.

Algorithm 1: Neural Collaborative Filtering (NCF)

Input: User-item interaction matrix **R**
Output: Predicted user preferences $\hat{\mathbf{R}}$
begin
  Initialize embeddings for users and items
  Define neural network $f(\cdot)$ to model interaction between embeddings
  for *each epoch* **do**
    for *each user-item pair* (u, i) **do**
      Predict score: $\hat{r}_{ui} = f(\text{embedding}(u), \text{embedding}(i))$
      Compute loss: $L = \text{MSE}(r_{ui}, \hat{r}_{ui})$
      Update model parameters via backpropagation
    end
  end
end

4.3.4. Recurrent Neural Network (RNN) Model

The RNN Model is designed to capture sequential patterns in employee development histories. Employees' past training sessions, promotions, and evaluations form a temporal sequence, which the RNN learns to model. Long Short-Term Memory (LSTM) cells are used to mitigate the vanishing gradient problem and better capture long-term dependencies in employee progression paths.

Algorithm 2: Training and Prediction using Long Short-Term Memory (LSTM) Networks

Input: Sequential employee development data $\mathcal{D} = \{(x_t, y_t)\}_{t=1}^{T}$
Output: Trained LSTM model capable of predicting next development steps
begin
  Initialize LSTM parameters (weights, biases) randomly
  Set learning rate η
  for *each training epoch* **do**
    for *each employee sequence* $(x_1, \ldots, x_T)$ *in* $\mathcal{D}$ **do**
      Initialize hidden state $h_0 = 0$ and cell state $c_0 = 0$
      for *each time step* $t = 1$ *to* T **do**
        Compute input gate:
        $i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i)$
        Compute forget gate:
        $f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f)$
        Compute output gate:
        $o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o)$
        Compute candidate cell state:
        $\tilde{c}_t = \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c)$
        Update cell state:
        $c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$
        Update hidden state:
        $h_t = o_t \odot \tanh(c_t)$
        Compute output prediction:
        $\hat{y}_t = \sigma(W_{hy}h_t + b_y)$
      end
      Compute loss L between predicted outputs $\hat{y}_t$ and ground truth y_t
      Backpropagate error through time (BPTT) to compute gradients
      Update LSTM parameters using gradient descent:
      $\theta \leftarrow \theta - \eta \nabla_\theta L$
    end
  end
end

4.3.5. Graph Neural Network (GNN) Model

The GNN Model treats employee competencies, career goals, and development resources as nodes in a graph. Skills are connected based on prerequisite relationships or co-occurrence patterns. The GNN aggregates information from an employee's local skill neighborhood to predict personalized development plans, effectively leveraging relational data structures.

Algorithm 3: Graph Neural Network (GNN) for Skill Recommendation

```
Input: Graph G = (V, E) with node features H^0
Output: Updated node embeddings H^L
begin
    for layer l = 0 to L − 1 do
        for each node v ∈ V do
            Aggregate neighbor features:
            h^(l)_N(v) = AGGREGATE^(l)({h^(l)_u : u ∈ N(v)})
            Update node representation:
            h^(l+1)_v = σ(W^(l) · CONCAT(h^(l)_v, h^(l)_N(v)))
        end
    end
end
```

4.3.6. Transformer Model

The Transformer Model is applied particularly to unstructured textual data, such as employee career goals or self-assessments. Utilizing self-attention mechanisms, the Transformer captures long-range dependencies and contextual relationships in the text, allowing the model to generate more contextually relevant development plan recommendations.

Algorithm 4: Transformer Encoder for Text Embeddings

```
Input: Tokenized text input X = (x_1, ..., x_n)
Output: Contextual embeddings H = (h_1, ..., h_n)
begin
    Embed tokens: E = Embedding(X)
    Add positional encodings to E
    for each layer do
        Compute multi-head self-attention:
        Z = MultiHead(Q = E, K = E, V = E)
        Apply feedforward network:
        H = FFN(Z)
    end
end
```

4.3.7. Model Evaluation Metrics

Each model's predictions are evaluated using standard classification metrics:

- **Precision:** Measures the proportion of recommended development plans that were relevant.
- **Recall:** Measures the proportion of relevant plans that were correctly recommended.
- **F1-Score:** The harmonic mean of Precision and Recall, balancing both concerns.
- **AUC-ROC:** Measures the trade-off between true positive and false positive rates across thresholds.

These metrics are used to select the best-performing model for deployment.

4.4. Experimental Design

The experimental design phase establishes the procedures for training, validating, and evaluating the deep learning models developed for the IDP recommender system. Each model—Neural Collaborative

Filtering (NCF), Recurrent Neural Network (RNN), Graph Neural Network (GNN), and Transformer—is independently trained and assessed to ensure fair performance comparisons.

4.4.1. Dataset Preparation

Employee data collected from organizational systems is preprocessed as described in Section 4.3. After cleaning and feature engineering, the dataset is split into three subsets:

- **Training Set (70%):** Used to train the models by minimizing the loss function.
- **Validation Set (15%):** Used for hyperparameter tuning and early stopping to prevent overfitting.
- **Testing Set (15%):** Used solely for final performance evaluation.

All data splits are stratified where applicable, ensuring balanced distributions of key employee attributes across the subsets.

4.4.2. Cross-Validation Strategy

In addition to a fixed train-validation-test split, 5-fold cross-validation is applied during hyperparameter tuning. This involves partitioning the training set into five folds and performing training and validation iteratively to assess the generalization capability of each model.

4.4.3. Model Training Details

Each deep learning model is trained independently using the prepared training data. The following general settings are applied:

- **Optimizer: Adam Optimizer** with an initial learning rate of 0.001.

  The Adam optimizer updates model parameters based on estimates of first (m_t) and second (v_t) moments of the gradients:

  $$\theta_{t+1} = \theta_t - \eta \times \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

  where:

  - θ_t are the model parameters at time step t,
  - η is the learning rate,
  - $\hat{m}_t$ is the bias-corrected first moment estimate (mean of gradients),
  - $\hat{v}_t$ is the bias-corrected second moment estimate (uncentered variance of gradients),
  - ϵ is a small constant for numerical stability.

- **Loss Function: Binary Cross-Entropy (BCE)** for recommendation tasks.

  The Binary Cross-Entropy loss function is defined as:

  $$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)]$$

  where:

  - N is the number of samples,
  - y_i is the true label (0 or 1),
  - $\hat{y}_i$ is the predicted probability for the positive class.

- **Batch Size:** 64 samples per batch.

  During training, the model updates its weights based on the gradient computed from mini-batches of 64 samples, rather than using the entire training dataset at once (stochastic gradient descent).

- **Epochs:** 100, with early stopping after 10 consecutive epochs without validation loss improvement.

  An epoch is defined as one full pass through the entire training dataset. Early stopping is a regularization technique where training is halted if the model performance on the validation set does not improve for 10 consecutive epochs.

Hyperparameters such as learning rate, dropout rate, number of hidden layers, and embedding sizes are fine-tuned based on validation performance during cross-validation.

4.4.4. Evaluation Criteria

- **Precision (P):** Measures the proportion of correctly recommended development plans among all recommendations made.

$$P = \frac{TP}{TP + FP}$$

  where TP = True Positives and FP = False Positives.

- **Recall (R):** Measures the proportion of relevant development plans successfully recommended.

$$R = \frac{TP}{TP + FN}$$

  where FN = False Negatives.

- **F1-Score (F_1):** Harmonic mean of Precision and Recall, providing a balance between the two.

$$F_1 = 2 \times \frac{P \times R}{P + R}$$

- **AUC-ROC:** Area Under the Receiver Operating Characteristic Curve, measuring the ability of the model to distinguish between classes.

  The AUC-ROC is calculated by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings:

$$\text{TPR} = \frac{TP}{TP + FN} \quad \text{and} \quad \text{FPR} = \frac{FP}{FP + TN}$$

  where TN = True Negatives.

Each model's performance is reported on the testing set, and the model achieving the best F1-Score and AUC-ROC is selected for deployment.

4.4.5. Software and Hardware Environment

The experiments are conducted using the following environments:

- Programming Language: Julia
- Deep Learning Libraries: Flux, DataFrames, Plots, Statistics
- Hardware: CPU-only training due to Hardware Constraints.

4.5. Technical Decisions Justification

This section provides the rationale behind the selection of models, methods, and techniques used in designing the deep learning-based IDP recommender system.

4.5.1. Why Deep Learning Approaches?

Traditional machine learning algorithms such as decision trees or support vector machines often struggle with high-dimensional, unstructured, and sequential data commonly found in employee development records. Deep learning models offer superior capabilities for:

- Automatically extracting complex features from large datasets.
- Capturing nonlinear relationships between employee attributes and development paths.
- Handling sequential data (e.g., training histories, performance progressions) through models like RNNs and LSTMs.
- Processing unstructured textual data such as career aspirations using architectures like Transformers.

Given these advantages, deep learning was chosen to maximize the predictive accuracy and flexibility of the IDP recommender system.

4.5.2. Why Neural Collaborative Filtering (NCF)?

NCF extends traditional collaborative filtering by replacing dot-product operations with neural networks, enabling the learning of complex user-item interaction patterns. In the context of IDPs:

- Employees are treated as users and recommended career development options as items.
- NCF can model intricate relationships between employee history and future development needs.
- It improves over matrix factorization methods by introducing non-linearity through hidden layers.

4.5.3. Why Recurrent Neural Networks (RNNs)?

Career development is inherently sequential—previous training, promotions, and evaluations influence future development plans. RNNs are selected because:

- They capture temporal dependencies in employee development trajectories.
- Using Long Short-Term Memory (LSTM) cells addresses the vanishing gradient problem, allowing modeling of long-term dependencies.
- Sequential modeling is essential for recommending development paths that logically follow an employee's career history.

4.5.4. Why Graph Neural Networks (GNNs)?

Skills, roles, and career goals form natural graph structures with prerequisite relationships and interdependencies. GNNs are appropriate because:

- They capture relational information between skills and learning objectives.
- They allow knowledge propagation across the graph, enhancing recommendations based on employee competencies and career objectives.
- They model interconnected skill frameworks more effectively than flat vector representations.

4.5.5. Why Transformer Models?

Employee career goals and self-assessments often involve natural language, which is unstructured and complex. Transformers, such as BERT, are ideal because:

- They leverage self-attention mechanisms to capture contextual relationships within text.
- They handle long-range dependencies without sequential bias, unlike RNNs.
- They have proven superior performance in many natural language processing (NLP) tasks relevant to understanding career aspirations.

4.5.6. Why Principal Component Analysis (PCA) and Autoencoders?

To reduce redundancy and enhance model training efficiency, dimensionality reduction techniques are applied:

- **PCA** projects features into a lower-dimensional space while preserving variance, useful for structured numerical data.
- **Autoencoders** learn compact, non-linear representations, beneficial when relationships among features are complex.

Both methods ensure that the input features are more manageable and that models can focus on the most informative aspects of employee data.

4.5.7. Why Standard Evaluation Metrics?

Metrics like Precision, Recall, F1-Score, and AUC-ROC are chosen because:

- They provide a comprehensive understanding of model performance.
- Precision and Recall address the correctness and completeness of recommendations.
- F1-Score balances Precision and Recall into a single figure of merit.
- AUC-ROC provides insight into model discrimination capability across decision thresholds.

4.6. Summary

This chapter presented the complete research design for developing and evaluating a deep learning-based recommender system for Individual Development Plans (IDPs).

The design is modular and data-driven, beginning with the collection and preprocessing of employee data, followed by feature engineering using dimensionality reduction and clustering techniques. Four deep learning models—Neural Collaborative Filtering (NCF), Recurrent Neural Network (RNN), Graph Neural Network (GNN), and Transformer—are independently trained and evaluated based on standard metrics such as Precision, Recall, F1-Score, and AUC-ROC.

Each model's architecture, training process, and evaluation strategy were explained in detail, including technical justifications for selecting deep learning approaches over traditional machine learning techniques. The experimental design ensures fair comparison across models through consistent data splitting, cross-validation, and hyperparameter tuning.

The best-performing model will ultimately be deployed within the IDP generation system, supported by a feedback loop for continuous system improvement.

The next chapter presents the results of the experiments and provides a detailed analysis of the models' performances.

5. Implementation

stuff

6. Results and Evaluation

7. Conclusion

7.1. Closing Remarks

7.2. Research Questions

The research questions posed in Chapter 1 are repeated below for convenience.

Research Question 1

What state-of-the-art methods are most suitable for XXX?

Reflect on research questions.

8. Recommendations

This chapter provides a brief overview of the primary recommendations for the future continuation of this research project.

Rec 1
Rec 2

References

Abdollahi, B. . (2017, January). *Accurate and justifiable : New algorithms for explainable recommendations.* doi:10.18297/etd/2744

Addagarla, S. K. . (2019). A survey on comprehensive trends in recommendation systems & applications. *International Journal of Electronic Commerce Studies*, *10*, 65–88. doi:10.7903/ijecs.1705

Akkaya, B. . (2025). Current trends in recommender systems: A survey of approaches and future directions. *DergiPark (Istanbul University)*. doi:10.53070/bbd.1652022

Alshbanat, H. I., Benhidour, H., & Kerrache, S. . (2024). A survey of latent factor models in recommender systems. *Information Fusion*, 102905–102905. doi:10.1016/j.inffus.2024.102905

Areeb, Q. M., Nadeem, M., Sohail, S. S., Imam, R., Doctor, F., Himeur, Y., … Amira, A. . (2023). Filter bubbles in recommender systems: Fact or fallacy—a systematic review. *Wiley Interdisciplinary Reviews Data Mining and Knowledge Discovery*, *13*. doi:10.1002/widm.1512

Askarbekuly, N., & Luković, I. . (2024, June). Learning outcomes, assessment, and evaluation in educational recommender systems: A systematic review. doi:10.48550/arxiv.2407.09500

Avison, D. E., Lau, F., Myers, M., & Nielsen, P. A. . (1999). Action research. *Communications of the ACM*, *42*(1), 94–97.

Azizi, A., Vajargha, K. F., Arefi, M, & Abolghasemi, M. . (2020). Explaining the human resources training and improvement paradigm: Focusing on the individual development model (idp). *Research in Medical Education*, *12*, 59–69. doi:10.52547/rme.12.4.59

Bathla, G., Rani, R., & Aggarwal, H. . (2018). Improving recommendation techniques by deep learning and large scale graph partitioning. *International Journal of Advanced Computer Science and Applications*, *9*. doi:10.14569/ijacsa.2018.091049

Bayly-Castaneda, K., Ramirez-Montoya, M.-S., & Morita-Alexander, A. . (2024). Crafting personalized learning paths with ai for lifelong learning: A systematic literature review. *Frontiers in Education*, *9*. doi:10.3389/feduc.2024.1424386

Bellogín, A., Castells, P., & Cantador, I. . (2017). Statistical biases in information retrieval metrics for recommender systems. *Information Retrieval*, *20*, 606–634. doi:10.1007/s10791-017-9312-z

Broisin, J. . (2020). Towards intelligent technology-enhanced learning solutions for transforming higher education: Contributions and future directions. *HAL (Le Centre pour la Communication Scientifique Directe)*. Retrieved from https://hal.science/tel-03684452

Brughmans, D., Melis, L., & Martens, D. . (2024). Disagreement amongst counterfactual explanations: How transparency can be misleading. *Top*, *32*, 429–462. doi:10.1007/s11750-024-00670-2

Castells, P., & Moffat, A. . (2022). Offline recommender system evaluation: Challenges and new directions. *AI Magazine*, *43*, 225–238. doi:10.1002/aaai.12051

Chang, C., Hui, J. C., Justus-Smith, C., & Wang, T. . (2024). Navigating stem careers with ai mentors: A new idp journey. *Frontiers in Artificial Intelligence*, *7*. doi:10.3389/frai.2024.1461137

Chen, J., Dong, H., Wang, X., Feng, F., Wang, M., & He, X. . (2020). Bias and debias in recommender system: A survey and future directions. *arXiv (Cornell University)*. doi:10.48550/arxiv.2010.03240

Chen, J., Dong, H., Wang, X., Feng, F., Wang, M., & He, X. (2022a). Bias and debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems*, *41*, 1–39. doi:10.1145/3564284

Chen, L., & Zhong, J. (2024). Intelligent recommendation system for college english courses based on graph convolutional networks. *Heliyon*, *10*(8), e29052–e29052. doi:10.1016/j.heliyon.2024.e29052

Chen, X., Zhang, Y., & Wen, J.-R. (2022b). Measuring ”why” in recommender systems: A comprehensive survey on the evaluation of explainable recommendation. *arXiv (Cornell University)*. doi:10.48550/arxiv.2202.06466

Coopersmith, K. (2021). Personal development planning and vertical leadership development in a vuca world. *Journal of Values-Based Leadership*, *15*. doi:10.22543/0733.151.1380

Cortes, D. (2021, July). Recometrics: Evaluation metrics for implicit-feedback recommender systems. doi:10.32614/cran.package.recometrics

Cramer, H., Evers, V., Ramlal, S., van Someren, M., Rutledge, L., Stash, N., … Wielinga, B. (2008). The effects of transparency on trust in and acceptance of a content-based art recommender. *User Modeling and User-Adapted Interaction*, *18*, 455–496. doi:10.1007/s11257-008-9051-3

da Silva, F. L., Slodkowski, B. K., da Silva, K. K. A., & Cazella, S. C. (2022). A systematic literature review on educational recommender systems for teaching and learning: Research trends, limitations and opportunities. *Education and Information Technologies*, *28*, 3289–3328. doi:10.1007/s10639-022-11341-9

da Silva, J. F., Borges, J. F. F., Sarsur, A. M., Nunes, S. C., & de Amorim, W. A. C. (2019). Careers: Workers’ perceptions of organizations’ support for their professional growth. *Tourism & Management Studies*, *15*, 35–43. doi:10.18089/tms.2019.150404

Damera, T. (2025). Ethically aligned personalization: Reimagining large-scale ai through distributed systems for an inclusive digital society. *International Journal of Computational and Experimental Science and Engineering*, *11*. doi:10.22399/ijcesen.4185

Davison, R. M., Martinsons, M. G., & Malaurent, J. (2021). Research perspectives: Improving action research by integrating methods. *Journal of the Association for Information Systems*, *22*(3). doi:10.17705/1jais.00682

Dong, Y., Li, J., & Schnabel, T. (2023). When newer is not better: Does deep learning really benefit recommendation from implicit feedback? *arXiv (Cornell University)*. doi:10.48550/arxiv.2305.01801

Duan, J., & Wu, S. (2024). Beyond traditional pathways: Leveraging generative ai for dynamic career planning in vocational education. *International Journal of New Developments in Education*, *6*. doi:10.25236/ijnde.2024.060205

e Oliveira, E., Rodrigues, M., Pereira, J. P. G., Lopes, A. M., Mestric, I. I., & Bjelogrlic, S. (2024). Unlabeled learning algorithms and operations: Overview and future trends in defense sector. *Artificial Intelligence Review*, *57*. doi:10.1007/s10462-023-10692-0

Eason, D. E., Bruno, B. C., & Böttjer-Wilson, D. (2020). Individual development plans (idps): An underutilized advising tool in the geosciences. *GSA Today*, *30*, 34–35. doi:10.1130/gsatg3gw.1

et al., O. (2021). Mixed methods appraisal tool: Strengthening the methodological rigor of mixed methods research studies in nursing. *Texto & Contexto - Enfermagem*. doi:10.1590/1980-265x-tce-2020-0603

Fakhfakh, R., □□, □., & Ben, C. (2017). Deep learning-based recommendation: Current issues and challenges. *International Journal of Advanced Computer Science and Applications*, *8*. doi:10.14569/ijacsa.2017.081209

Flood, A. H., Skrabalak, S. E., & Yu, Y. (2021). Individual development plans — experiences made in graduate student training. *Analytical and Bioanalytical Chemistry*, *413*, 5681–5684. doi:10.1007/s00216-021-03540-z

Frej, J., Dai, A., Montariol, S., Bosselut, A., & Käser, T. (2024a). Course recommender systems need to consider the job market. *arXiv*. doi:10.48550/ARXIV.2404.10876

Frej, J., Dai, A., Montariol, S., Bosselut, A., & Käser, T. (2024b). Course recommender systems need to consider the job market. *arXiv (Cornell University)*. doi:10.1145/3626772.3657847

Gangadharan, K., Kanagasabai, M., Purandaran, A., Subramanian, B., Jeyaraj, R., & Jung, S. K. (2024). From data to decisions: The transformational power of machine learning in business recommendations. *arXiv (Cornell University)*. doi:10.1109/access.2025.3532697

Gedikli, F., Jannach, D., & Ge, M. (2014). How should i explain? a comparison of different explanation types for recommender systems. *International Journal of Human-Computer Studies*, *72*, 367–382. doi:10.1016/j.ijhcs.2013.12.007

Geerts, J. M. (2024, October). Maximizing the impact and roi of leadership development: A theory- and evidence-informed framework. doi:10.3390/bs14100955

Gheewala, S., Xu, S., & Yeom, S. (2025). In-depth survey: Deep learning in recommender systems—exploring prediction and ranking models, datasets, feature analysis, and emerging trends. *Neural Computing and Applications*. doi:10.1007/s00521-024-10866-z

Gomez-Uribe, C. A., & Hunt, N. T. (2015). The netflix recommender system. *ACM Transactions on Management Information Systems*, *6*, 1–19. doi:10.1145/2843948

Grida, M., Fayed, L., & Hassan, M. (2020). Convsvd++: A hybrid deep cf recommender model using convolutional neural network. *Journal of Computer Science*, *16*, 1697–1708. doi:10.3844/jcssp.2020.1697.1708

Guesmi, M., Chatti, M. A., Joarder, S., Ain, Q. U., Alatrash, R., Siepmann, C., & Vahidi, T. (2023a). Interactive explanation with varying level of details in an explainable scientific literature recommender system. *International Journal of Human-Computer Interaction*, *40*, 7248–7269. doi:10.1080/10447318.2023.2262797

Guesmi, M., Chatti, M. A., Joarder, S., Ain, Q. U., Siepmann, C., Ghanbarzadeh, H., & Alatrash, R. (2023b). Justification vs. transparency: Why and how visual explanations in a scientific literature recommender system. *arXiv (Cornell University)*. doi:10.48550/arxiv.2305.17034

Guetterman. (2016). What distinguishes a novice from an expert mixed methods researcher? *Quality & quantity*. doi:10.1007/s11135-016-0310-9

Gunawardana, A., & Shani, G. (2009). A survey of accuracy evaluation metrics of recommendation tasks. *10*, 2935–2962. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.332.5041

Guttmann, M., & Ge, M. (2024). Research agenda of ethical recommender systems based on explainable ai. *Procedia Computer Science*, *238*, 328–335. doi:10.1016/j.procs.2024.06.032

Harasic, M., Keese, F.-S., Mattern, D., & Paschke, A. (2023). Recent advances and future challenges in federated recommender systems. *International Journal of Data Science and Analytics*, *17*, 337–357. doi:10.1007/s41060-023-00442-4

Hemmler, Y. M., Rasch, J., & Ifenthaler, D. (2022, September). A categorization of workplace learning goals for multi-stakeholder recommender systems: A systematic review. doi:10.1007/s11528-022-00777-y

Henninger, M., & Strobl, C. (2024). Interpreting machine learning predictions with lime and shapley values: Theoretical insights, challenges, and meaningful interpretations. *Behaviormetrika*, *52*, 45–75. doi:10.1007/s41237-024-00253-2

Hsieh, W. C., Bi, Z., Chen, K., Peng, B., Zhang, S., Xu, J., … Liu, M. (2024). Deep learning, machine learning, advancing big data analytics and management. *arXiv (Cornell University)*. doi:10.48550/arxiv.2412.02187

Hurley, E., Dietrich, T., & Rundle-Thiele, S. (2021). Integrating theory in co-design: An abductive approach. *Australasian Marketing Journal*, *29*(1), 66–77.

Irshaidat, R. (2019). Interpretivism vs. positivism in political marketing research. *Journal of Political Marketing*, *21*(2), 1–35. doi:10.1080/15377857.2019.1624286

Jadon, A., & Patil, A. (2023). A comprehensive survey of evaluation techniques for recommendation systems. *arXiv (Cornell University)*. doi:10.48550/arxiv.2312.16015

Jadon, A., & Patil, A. (2024, September). A comprehensive survey of evaluation techniques for recommendation systems. (pp. 281–304). doi:10.1007/978-3-031-71484-9_25

Jangda, S. I., & ur Rahman Raazi, S. M. K. (2024). Recommender system f or educational and professional growth, 1–8. doi:10.1109/iconics64289.2024.10824485

Jomsri, P., Prangchumpol, D., Poonsilp, K., & Panityakul, T. (2023). Hybrid recommender system model for digital library from multiple online publishers. *F1000Research*, *12*, 1140–1140. doi:10.12688/f1000research.133013.1

Jung, H., Kim, H., Lee, I., & An, S. (2023). Designing a longitudinal database for cohort construction in medical education. *Korean Medical Education Review*, *25*(2), 84–101. doi:10.17496/kmer.23.012

Kastrati, Z., Imran, A. S., & Kurti, A. (2019). Integrating word embeddings and document topics with deep learning in a video classification framework. *Pattern Recognition Letters*, *128*, 85–92. doi:10.1016/j.patrec.2019.08.019

Kaur, K., Gujar, S., & Jain, S. (2024). Towards fairness in provably communication-efficient federated recommender systems. *arXiv (Cornell University)*. doi:10.48550/arxiv.2405.15788

Kelley, K., & Rausch, J. (2011). Sample size planning for longitudinal models: Accuracy in parameter estimation for polynomial change parameters. *Psychological Methods*, *16*(4), 391–405. doi:10.1037/a0023352

Lahoud, C., Moussa, S., Obeid, C., Khoury, H. E., & Champin, P.-A. (2022). A comparative analysis of different recommender systems for university major and career domain guidance. *Education and Information Technologies*, *28*, 8733–8759. doi:10.1007/s10639-022-11541-3

Li, W., Zhong, H., Zhou, J., Chang, C., Lin, R., & Tang, Y. (2024a). An attention mechanism and residual network-based knowledge graph-enhanced recommender system. *Knowledge-Based Systems*, *299*, 112042–112042. doi:10.1016/j.knosys.2024.112042

Li, Z., Long, G., Zhang, C., Zhang, H., Jiang, J., & Zhang, C. (2024b). Navigating the future of federated recommendation systems with foundation models. *arXiv (Cornell University)*. doi:10.48550/arxiv.2406.00004

Lin, J., Liu, J., Zhu, J., Xi, Y., Liu, C., Zhang, Y., … Zhang, W. (2024). A survey on diffusion models for recommender systems. *arXiv (Cornell University)*. doi:10.48550/arxiv.2409.05033

M, N. K., & Basheer, M. (2025). A hybrid explainability framework for recommender systems using shap with natural language interpretations. *International Journal of Innovative Research and Scientific Studies*, *8*, 687–703. doi:10.53894/ijirss.v8i6.9669

Ma, Q., Ren, X., & Huang, C. (2024). Xrec: Large language models for explainable recommendation. *arXiv (Cornell University)*. doi:10.48550/arxiv.2406.02377

Maeng, K., Lu, H., Melis, L., Nguyen, J., Rabbat, M., & Wu, C.-J. (2022). Towards fair federated recommendation learning: Characterizing the inter-dependence of system and data heterogeneity, 156–167. doi:10.1145/3523227.3546759

Mantere, S., & Ketokivi, M. (2013). Reasoning in organization science. *Academy of Management Review*, *38*, 70–89.

Mena-Maldonado, E., Cañamares, R., Castells, P., Ren, Y., & Sanderson, M. (2020). Agreement and disagreement between true and false-positive metrics in recommender systems evaluation, 841–850. doi:10.1145/3397271.3401096

Mishra, R., Bibo, J. I., van Engelen, Q., & Schaapman, H. (2025). Reproducibility study of "xrec: Large language models for explainable recommendation". *arXiv (Cornell University)*. doi:10.48550/arxiv.2510.06275

Mu, R. (2018). A survey of recommender systems based on deep learning. *IEEE Access*, *6*, 69009–69022. doi:10.1109/access.2018.2880197

Multiple Authors. (2020). Various studies on recommendation systems and educational personalization. Illustrative placeholder entry added by automated tooling.

Mumuni, A., & Mumuni, F. (2024). Automated data processing and feature engineering for deep learning and big data applications: A survey. *Journal of Information and Intelligence*, *3*, 113–153. doi:10.1016/j.jiixd.2024.01.002

Myers, M. (2008). *Qualitative research in business & management*. Sage Publications.

Naumov, M., Mudigere, D., Shi, H.-J. M., Huang, J., Sundaraman, N., Park, J., ... Smelyanskiy, M. (2019). Deep learning recommendation model for personalization and recommendation systems. *arXiv (Cornell University)*. doi:10.48550/arxiv.1906.00091

Nawi, R. M., Noah, S. A. M., & Zakaria, L. Q. (2020). Evaluation of group modelling strategy in model-based collaborative filtering recommendation. *International Journal of Machine Learning and Computing*, *10*, 330–338. doi:10.18178/ijmlc.2020.10.2.939

Nosratabadi, S., Zahed, R. K., Ponkratov, V. V., & Kostyrin, E. V. (2022). Artificial intelligence models and employee lifecycle management: A systematic literature review. *Organizacija*, *55*, 181–198. doi:10.2478/orga-2022-0012

Nunes, I., & Jannach, D. (2017, October). A systematic review and taxonomy of explanations in decision support and recommender systems. doi:10.1007/s11257-017-9195-0

Nurjaman, K. (2025, August). Technological disruption in human resource management: A review of machine learning algorithms for strategic decision-making. doi:10.18421/tem143-86

Oladele, S., Anector, M., & Foluwa, E. (2025). Ai-powered personalization in employee training and development programs authors. doi:10.2139/ssrn.5226745

Osman, N. A., Noah, S. A. M., Darwich, M., & Mohd, M. (2021). Integrating contextual sentiment analysis in collaborative recommender systems. *PLoS ONE*, *16*. doi:10.1371/journal.pone.0248695

Oussouaddi, R., Hassouny, A. E., & Ismaili, Z. E. A. A. (2023). Advance in online education recommender systems during and after covid-19 a survey. *Research Square (Research Square)*. doi:10.21203/rs.3.rs-2982259/v1

Papaleonidas, A., Pimenidis, E., & Iliadis, L. S. (2019, January). Hybrid data set optimization in recommender systems using fuzzy t-norms. (pp. 647–659). doi:10.1007/978-3-030-19823-7_54

Pasrija, V., & Pasrija, S. (2024). Demystifying recommendations: Transparency and explainability in recommendation systems. *International Journal for Research in Applied Science and Engineering Technology*, *12*, 1376–1383. doi:10.22214/ijraset.2024.58541

Payandenick, M., Wang, Y. D., Othman, M. K., & Payandenick, M. (2025). Resnetmf: Improving recommendation accuracy and speed with matrix factorization enhanced by residual networks. *Acta Informatica Pragensia*. doi:10.18267/j.aip.280

Paz-Ruza, J., Alonso□Betanzos, A., Guijarro□Berdiñas, B., Cancela, B., & Eiras□Franco, C. (2023). Sustainable transparency in recommender systems: Bayesian ranking of images for explainability. *arXiv (Cornell University)*. doi:10.48550/arxiv.2308.01196

Pereira, B. L., Said, A., & Santos, R. L. T. (2025). On the reliability of sampling strategies in offline recommender evaluation, 360–369. doi:10.1145/3705328.3748086

Pu, P., & Chen, L. (2007). Trust-inspiring explanation interfaces for recommender systems. *Knowledge-Based Systems*, *20*, 542–556. doi:10.1016/j.knosys.2007.04.004

Qin, C., Zhang, L., Cheng, Y., Zha, R., Shen, D. Z., Zhang, Q., … Xiong, H. (2025). A comprehensive survey of artificial intelligence techniques for talent analytics. *Proceedings of the IEEE*, 1–47. doi:10.1109/jproc.2025.3572744

Raghavendra, C. K., K.C, S., & R, V. K. (2018). Personalized recommendation systems (pres): A comprehensive study and research issues. *International Journal of Modern Education and Computer Science*, *10*, 11–21. doi:10.5815/ijmecs.2018.10.02

Ragodos, R., Wang, T., Feng, L., Yu, & Hu. (2024). From model explanation to data misinterpretation: Uncovering the pitfalls of post hoc explainers in business research. *arXiv (Cornell University)*. doi:10.48550/arxiv.2408.16987

Raj, N. S., & Renumol, V. G. (2021). A systematic literature review on adaptive content recommenders in personalized learning environments from 2015 to 2020. *Journal of Computers in Education*, *9*, 113–148. doi:10.1007/s40692-021-00199-4

Ramachandran, K. K., Srivastava, A., Panjwani, V., Kumar, D., Cheepurupalli, N. R., & Mohan, C. R. (2024). Developing ai-powered training programs for employee upskilling and reskilling. *Journal of Informatics Education and Research*. doi:10.52783/jier.v4i2.903

Rane, J., Ömer Kaya, Mallick, S. K., & Rane, N. L. (2024, October). Enhancing customer satisfaction and loyalty in service quality through artificial intelligence, machine learning, internet of things, blockchain, big data, and chatgpt. doi:10.70593/978-81-981271-7-4_3

Raza, S., Rahman, M., Kamawal, S., Toroghi, A., Raval, A., Navah, F., & Kazemeini, A. (2024, July). A comprehensive review of recommender systems: Transitioning from theory to practice. doi:10.48550/arxiv.2407.13699

Roy, D., & Dutta, M. (2022, May). A systematic review and research perspective on recommender systems. doi:10.1186/s40537-022-00592-5

Rubio, D. M., Mayowski, C. A., Meagher, E. A., Patino, C. M., Thakar, M. S., Welch, J. L., & White, G. E. (2023). Customized career development platform (ccdp) for clinical and translational researchers: A pragmatic cluster-randomized controlled trial. *Journal of Clinical and Translational Science*, *7*. doi:10.1017/cts.2023.687

Saeed, W., & Omlin, C. W. (2022). Explainable ai (xai): A systematic meta-survey of current challenges and future opportunities. *arXiv (Cornell University)*. doi:10.48550/arxiv.2111.06420

Sahoo, A. K., Pradhan, C., Barik, R. K., & Dubey, H. (2019). Deepreco: Deep learning based health recommender system using collaborative filtering. *Computation*, *7*(2), 25. doi:10.3390/computation7020025

Said, A. (2024, November). A review of llm-based explanations in recommender systems. doi:10.48550/arxiv.2411.19576

Salau, L., Hamada, M., Prasad, R., Hassan, M., Mahendran, A., & Watanobe, Y. (2022). State-of-the-art survey on deep learning-based recommender systems for e-learning. *Applied Sciences*, *12*, 11996–11996. doi:10.3390/app122311996

Salih, A., Raisi□Estabragh, Z., Galazzo, I. B., Radeva, P., Petersen, S. E., Menegaz, G., & Lekadir, K. (2023). A perspective on explainable artificial intelligence methods: Shap and lime. *arXiv (Cornell University)*. doi:10.48550/arxiv.2305.02012

Sami, A., Adrousy, W. E., Sarhan, S., & Elmougy, S. (2024). A deep learning based hybrid recommendation model for internet users. *Scientific Reports*, *14*. doi:10.1038/s41598-024-79011-z

Saunders, M., Lewis, P., & Thornhill, A. (2012). *Research methods for business students* (6th). Pearson Education Limited.

Schnabel, T. (2022). Where do we go from here? guidelines for offline recommender evaluation. *arXiv (Cornell University)*. doi:10.48550/arxiv.2211.01261

Schnabel, T., Swaminathan, A., Singh, A., Chandak, N., & Joachims, T. (2016). Recommendations as treatments: Debiasing learning and evaluation. *arXiv (Cornell University)*, *48*, 1670–1679. doi:10.48550/arxiv.1602.05352

Silveira, T., Zhang, M., Xiao, L., Liu, Y., & Ma, S. (2017). How good your recommender system is? a survey on evaluations in recommendation. *International Journal of Machine Learning and Cybernetics*, *10*, 813–831. doi:10.1007/s13042-017-0762-9

Souabi, S., Retbi, A., Idrissi, M. K., & Bennani, S. (2021, November). Recommendation systems on e-learning and social learning: A systematic review. doi:10.34190/ejel.19.5.2482

Srivastava, R., Palshikar, G. K., Chaurasia, S., & Dixit, A. M. (2018). What's next? a recommendation system for industrial training. *Data Science and Engineering*, *3*, 232–247. doi:10.1007/s41019-018-0076-2

Steck, H., Baltrunas, L., Elahi, E., Liang, D., Raimond, Y., & Basilico, J. (2021). Deep learning for recommender systems: A netflix case study. *AI Magazine*, *42*, 7–18. doi:10.1609/aimag.v42i3.18140

Stergiopoulos, V., Vassilakopoulos, M., Tousidou, E., & Corral, A. (2024). An academic recommender system on large citation data based on clustering, graph modeling and deep learning. *Knowledge and Information Systems*, *66*, 4463–4496. doi:10.1007/s10115-024-02094-7

Sumartik, S., & Ambarwati, R. (2023, September). *Manajemen talenta dan implementasinya di industri*. doi:10.21070/2023/978-623-464-074-8

Sun, A. (2023, July). Take a fresh look at recommender systems from an evaluation standpoint. doi:10.1145/3539618.3591931

Taghavi, M., Bentahar, J., Bakhtiyari, K., & Hanachi, C. (2017). New insights towards developing recommender systems. *The Computer Journal*, *61*, 319–348. doi:10.1093/comjnl/bxx056

Taha, K., Yoo, P. D., & Taha, A. (2024). Empirical and experimental perspectives on big data in recommendation systems: A comprehensive survey. *arXiv (Cornell University)*. doi:10.48550/arxiv.2402.03368

Tertulino, R. (2025). Privacy-preserving personalization in education: A federated recommender system for student performance prediction. *arXiv (Cornell University)*. doi:10.48550/arxiv.2509.10516

Tewari, B., Rautela, M., Sharma, S. K., & Garg, N. (2024, November). Revolutionizing recommendations: Exploring recent trends in deep learning for modern systems. (pp. 443–447). doi:10.1201/9781003559092-223

Tey, F. J., Wu, T., Lin, C.-L., & Chen, J.-L. (2020). Accuracy improvements for cold-start recommendation problem using indirect relations in social networks. *Research Square (Research Square)*. doi:10.21203/rs.3.rs-137823/v1

Thompson, J. (2022). A guide to abductive thematic analysis. *The Qualitative Report*, *27*(5), 1410–1421. doi:10.46743/2160-3715/2022.5340

van der Merwe, M., Nel, P., & Hoole, C. (2024). How talent management execution impacts career experiences: Exploring the organization-individual intersect. *Frontiers in Psychology*, *15*. doi:10.3389/fpsyg.2024.1272645

Vanderford, N., Evans, T., Weiss, L., Bira, L., & Beltran-Gastelum, J. (2018). A cross-sectional study of the use and effectiveness of the individual development plan among doctoral students. *F1000research*, *7*, 722. doi:10.12688/f1000research.15154.2

Verma, A. (2020). A comparative study of ai-based recommender systems. *Türk bilgisayar ve matematik eğitimi dergisi*, *11*, 827–830. doi:10.17762/turcomat.v11i1.13564

Wang, C., Zhu, H., Zhu, C., Zhang, X., Chen, E., & Xiong, H. (2020). Personalized employee training course recommendation with career development awareness, 1648–1659. doi:10.1145/3366423.3380236

Wardatzky, K., Inel, O., Rossetto, L., & Bernstein, A. (2025). Whom do explanations serve? a systematic literature survey of user characteristics in explainable recommender systems evaluation. *ACM Transactions on Recommender Systems*, *3*, 1–35. doi:10.1145/3716394

Xia, X., Eksombatchai, P., Pancha, N., Badani, D., Wang, P.-W., Gu, N., … Zhai, A. (2023). Transact: Transformer-based realtime user action model for recommendation at pinterest. *arXiv (Cornell University)*. doi:10.48550/arxiv.2306.00248

Zafar, M. (2025). The impact of leadership development programs on organizational resilience and change management. *International Journal of Scientific Research and Management (IJSRM)*, *13*, 8327–8343. doi:10.18535/ijsrm/v13i02.em02

Zhang, C., Long, G., Zhang, Z., Li, Z., Zhang, H., Yang, Q., & Yang, B. (2025). Personalized recommendation models in federated settings: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 1–20. doi:10.1109/tkde.2025.3606643

Zhang, S., Yao, L., & Sun, A. (2022). Deep learning based recommender system: A survey and new perspectives. *arXiv (Cornell University)*. doi:10.48550/arxiv.1707.07435

Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019a, February). Deep learning based recommender system. doi:10.1145/3285029

Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019b, February). Deep learning based recommender system: A survey and new perspectives. Cornell University. Retrieved from http://arxiv.org/abs/1707.07435

Zhang, W., Quintana, M., & Miller, C. (2024). Recommender systems and reinforcement learning for building control and occupant interaction: A text-mining driven review of scientific literature. *arXiv (Cornell University)*. doi:10.48550/arxiv.2411.08734

Zhang, Y., & Chen, X. (2020). Explainable recommendation: A survey and new perspectives. *Foundations and Trends® in Information Retrieval*, *14*, 1–101. doi:10.1561/1500000066

Zhong, J., & Negre, E. (2022). Amélioration des explications contrefactuelles pour les recommandations à l'aide de shap. *HAL (Le Centre pour la Communication Scientifique Directe)*. Retrieved from https://hal.science/hal-03835986

Zhu, Y., Jiang, W., & Alonso, G. (2024). Efficient tabular data preprocessing of ml pipelines. *arXiv (Cornell University)*. doi:10.48550/arxiv.2409.14912

Zou, K., Sun, A., Jiang, X. W., Ji, Y., Zhang, H., Wang, J., & Guo, R. (2024). Hesitation and tolerance in recommender systems. *arXiv (Cornell University)*. doi:10.48550/arxiv.2412.09950

Đuričić, T., Kowald, D., Lacić, E., & Lex, E. (2023, December). Beyond-accuracy: A review on diversity, serendipity, and fairness in recommender systems based on graph neural networks. doi:10.3389/fdata.2023.1251072

A. Algorithms

This appendix contains an example algorithm.

Algorithm 5: Principal Component Analysis (PCA)

Input: Dataset $\mathbf{X} \in \mathbb{R}^{n \times d}$ (n samples, d features)
Output: Reduced representation $\mathbf{X}_{\text{reduced}}$
begin
  Center the data: $\mathbf{X} \leftarrow \mathbf{X} - \text{mean}(\mathbf{X})$
  Compute covariance matrix: $\mathbf{C} = \frac{1}{n-1}\mathbf{X}^T\mathbf{X}$
  Compute eigenvectors and eigenvalues of $\mathbf{C}$
  Sort eigenvectors by descending eigenvalues
  Select top k eigenvectors to form matrix $\mathbf{W}$
  Project data: $\mathbf{X}_{\text{reduced}} = \mathbf{XW}$
end

Algorithm 6: Autoencoder Training for Feature Extraction

Input: Training data $\mathbf{X}$
Output: Encoded representation $\mathbf{Z}$
begin
  Define encoder network f_θ and decoder network g_ϕ
  Initialize parameters θ, ϕ
  for *each epoch* **do**
    for *each mini-batch* **do**
      Encode: $\mathbf{Z} = f_\theta(\mathbf{X})$
      Decode: $\hat{\mathbf{X}} = g_\phi(\mathbf{Z})$
      Compute reconstruction loss: $L = ||\mathbf{X} - \hat{\mathbf{X}}||^2$
      Update θ, ϕ using gradient descent to minimize L
    end
  end
end

Algorithm 7: Neural Collaborative Filtering (NCF)

Input: User-item interaction matrix $\mathbf{R}$
Output: Predicted user preferences $\hat{\mathbf{R}}$
begin
  Initialize embeddings for users and items
  Define neural network $f(\cdot)$ to model interaction between embeddings
  for *each epoch* **do**
    for *each user-item pair* (u, i) **do**
      Predict score: $\hat{r}_{ui} = f(\text{embedding}(u), \text{embedding}(i))$
      Compute loss: $L = \text{MSE}(r_{ui}, \hat{r}_{ui})$
      Update model parameters via backpropagation
    end
  end
end

Algorithm 8: Graph Neural Network (GNN) for Skill Recommendation

---

Input: Graph $G = (V, E)$ with node features $\mathbf{H}^0$
Output: Updated node embeddings $\mathbf{H}^L$
begin
  for *layer* $l = 0$ **to** $L - 1$ **do**
    for *each node* $v \in V$ **do**
      Aggregate neighbor features:
      $\mathbf{h}^{(l)}_{\mathcal{N}(v)} = \text{AGGREGATE}^{(l)}(\{\mathbf{h}^{(l)}_u : u \in \mathcal{N}(v)\})$
      Update node representation:
      $\mathbf{h}^{(l+1)}_v = \sigma\left(\mathbf{W}^{(l)} \cdot \text{CONCAT}(\mathbf{h}^{(l)}_v, \mathbf{h}^{(l)}_{\mathcal{N}(v)})\right)$
    end
  end
end

Algorithm 9: Transformer Encoder for Text Embeddings

Input: Tokenized text input $X = (x_1, \ldots, x_n)$
Output: Contextual embeddings $H = (h_1, \ldots, h_n)$
begin
    Embed tokens: **E** $=$ Embedding(X)
    Add positional encodings to **E**
    for *each layer* **do**
        Compute multi-head self-attention:
        $Z =$ MultiHead$(Q = E, K = E, V = E)$
        Apply feedforward network:
        $H =$ FFN(Z)
    end
end

Algorithm 10: Deep Learning-based IDP Recommender System Pipeline

Input: Raw employee data: skills assessments, job descriptions, performance reviews, career goals
Output: Personalized Individual Development Plans (IDPs)
begin
    extbfStep 1: Data Collection and Preprocessing
        Collect employee datasets (skills, evaluations, feedback, goals)
        Clean and normalize data (handling missing values, text cleaning)
        Extract features from structured data and unstructured text using NLP
    extbfStep 2: Feature Engineering
        Apply PCA for dimensionality reduction
        Train Autoencoder to learn compact feature representations
        Perform Clustering to identify similar employee profiles
    extbfStep 3: Model Training
        extbfSub-step 3.1: NCF-based Interaction Modeling
            Train Neural Collaborative Filtering (NCF) to predict personalized recommendations based on historical data
        extbfSub-step 3.2: Graph Modeling
            Construct Knowledge Graph of skills, learning paths, career objectives
            Train Graph Neural Network (GNN) to capture relational dependencies
        extbfSub-step 3.3: Textual Understanding
            Train Transformer model (e.g., BERT) on employee goal statements
            Generate embeddings for career goal descriptions
    extbfStep 4: Inference and Recommendation
        For a given employee:
        - Embed structured and unstructured data
        - Aggregate predictions from NCF, GNN, and Transformer models
        - Rank and select top development plan recommendations
        Generate customized IDP output for the employee
    extbfStep 5: Feedback Integration
        Collect employee and supervisor feedback post-recommendation
        Update training datasets and retrain models periodically for dynamic adaptation
end