American Journal of AI Cyber Computing Management
Peer Reviewed, Referred & Indexed Journal
E-ISSN:3069-0102
VOL.6, NO. 2(2) (2026)

SuccessionAI: An Intelligent Multi-Agent System for Personalized Individual Development Plans Using Large Language Models and Machine Learning

Mohanty Hitesh Rabindranath
Department of Computer Science & Engineering
GIFT Autonomous, Bhubaneswar
Odisha, India
mohantyhitesh4495@gmail.com

Dr. Satya Ranjan Pattanaik
Department of Computer Science & Engineering
GIFT Autonomous, Bhubaneswar
Odisha, India
srp.nist@gmail.com

Abstract—Succession planning and employee development in modern organizations remain heavily dependent on manual, subjective processes that produce generic Individual Development Plans (IDPs) incapable of addressing the unique competency gaps and career trajectories of individual employees. This paper presents SuccessionAI, an intelligent, AI-powered succession planning and personalized IDP generation platform that leverages a multi-agent pipeline orchestrated by LangGraph, powered by the Groq LLaMA 3.3 70B large language model, and augmented with a Scikit-Learn Random Forest classifier for role readiness prediction. The system automates the end-to-end talent development workflow: it retrieves employee skill profiles from MongoDB, performs LLM-driven competency gap analysis against target role requirements, predicts readiness scores using machine learning, generates a curated learning roadmap via a recommendation agent, discovers relevant courses through the Tavily web search API, identifies suitable mentors through skill-based cosine similarity matching, and compiles a fully personalized IDP aligned to the 70-20-10 learning framework. Analytics features including a Nine-Box Performance-Potential Matrix and interactive Plotly.js dashboards provide HR professionals and managers with data-driven workforce insights. Experimental evaluation demonstrates significant improvements in IDP relevance, succession decision quality, and HR operational efficiency compared to conventional approaches. SuccessionAI establishes a replicable framework for applying agentic AI and machine learning to enterprise human capital management.

Index Terms—Individual Development Plans, Succession Planning, Multi-Agent AI, LangGraph, Large Language Models, Random Forest, Skill Gap Analysis, Human Capital Management, LLaMA, Talent Management

I. Introduction

Talent management and succession planning are among the most strategically critical functions in any modern organization. The ability to identify high-potential employees, assess their readiness for key roles, and provide them with targeted development opportunities directly determines long-term organizational resilience [1]. Despite their strategic importance, these processes have historically been driven by subjective managerial judgments, inconsistent evaluation criteria, and labor-intensive manual effort.

Traditional Individual Development Plans (IDPs) are often generic documents produced by HR professionals without access to data-driven skill gap analysis or intelligent learning recommendations. Employees receive development plans that fail to account for their specific strengths, weaknesses, and career aspirations. Simultaneously, succession planning decisions are made based on incomplete data, introducing bias and inefficiency into the talent pipeline. Research by Rothwell [1] indicates that fewer than 30% of organizations report confidence in their succession planning processes, highlighting the systemic gap between strategic intent and operational effectiveness.

The rapid advancement of Large Language Models (LLMs) and machine learning (ML) offers a transformative opportunity in this domain. Recent research has demonstrated the effectiveness of LLMs in knowledge-intensive reasoning tasks [2], while multi-agent AI frameworks such as LangGraph enable orchestration of complex, stateful workflows that simulate expert decision-making processes [3]. The confluence of these technologies creates the conditions for a fundamentally new approach to HR talent development.

This paper presents SuccessionAI, a novel system that integrates multi-agent LLM orchestration, machine learning readiness prediction, web-based resource discovery, and intelligent mentor matching into a unified platform for personalized IDP generation. The system is designed as a FastAPI-based backend with a MongoDB data layer, capable of serving enterprise-scale HR operations. The core contributions of this work are:

- A multi-agent AI pipeline for automated, end-to-end IDP generation orchestrated via LangGraph with conditional branching logic.
- An LLM-powered competency gap analysis engine using Groq LLaMA 3.3 70B with structured JSON output enforcement.
- A Random Forest classifier trained on employee skill and performance features to predict quantitative role readiness scores.
- A web search-augmented recommendation engine using the Tavily API for real-time course and certification discovery.
- A Nine-Box Performance-Potential Matrix analytics module for automated workforce segmentation.

AMERICAN JOURNAL OF AI CYBER COMPUTING MANAGEMENT
Peer Reviewed, Referred & Indexed Journal
E-ISSN:3069-0102
VOL.6, NO. 2(2) (2026)

- A weighted cosine similarity-based mentor matching algorithm that identifies contextually appropriate mentors.
- A comprehensive evaluation framework comparing SuccessionAI against conventional and baseline AI approaches across quality, accuracy, and performance dimensions.

The remainder of this paper is organized as follows: Section II reviews related work in succession planning, NLPbased HR tools, and multi-agent AI. Section III describes the system architecture and overall workflow. Section IV presents the AI and ML components in detail. Section V discusses the analytics and IDP generation modules. Section VI presents evaluation results. Section VII discusses findings, limitations, and ethical considerations. Section VIII concludes with future directions.

II. RELATED WORK

A. Succession Planning and Talent Management Systems

Early computerized succession planning systems were database-driven HR information systems that stored employee profiles and allowed managers to filter candidates for leadership roles [1]. These systems lacked predictive capability and relied entirely on human judgment for decision-making. More recent enterprise platforms such as SAP SuccessFactors and Workday have introduced analytics layers, but their IDP generation remains largely template-based and non-personalized [4]. Neither platform supports real-time, generative competency gap analysis or dynamic resource discovery, leaving significant opportunities for intelligent augmentation.

Collings and Mellahi [11] proposed a strategic talent management framework emphasizing the importance of differentiating high-potential employees and aligning development plans with organizational strategy. SuccessionAI operationalizes this framework by automating the identification and prioritization of talent through machine learning readiness scoring.

B. Natural Language Processing in Human Resources

The application of NLP to HR tasks has grown substantially with the rise of transformer-based models [8]. Prior studies have applied BERT-based models to resume screening, job description matching, and skill extraction from unstructured text [5]. Jia et al. [12] demonstrated the effectiveness of pretrained language models for job-skill matching with accuracy improvements of over 15% compared to keyword-based baselines.

However, these approaches are typically applied to isolated subtasks rather than integrated into end-to-end talent development workflows. The emergence of instruction-tuned LLMs such as GPT-4 and LLaMA 3 has opened new possibilities for generative, context-aware HR reasoning that goes beyond pattern matching to genuine reasoning about competency gaps and development pathways [2].

C. Multi-Agent AI Systems

The LangChain and LangGraph frameworks have enabled a new class of agentic AI applications in which multiple specialized agents collaborate to solve complex tasks [3]. Relevant prior work includes AI agents for code generation [13], customer service automation, and scientific research assistance. LangGraph's stateful directed graph model provides particular advantages over simple sequential pipelines, including support for conditional routing, cycle detection, and shared state management across agents.

The application of multi-agent orchestration to HR workflows represents a largely unexplored area. Prior work by Park et al. [14] demonstrated that multi-agent LLM systems outperform single-agent approaches on complex reasoning tasks by enabling specialization and division of labor. SuccessionAI applies this insight to the HR domain by assigning distinct, specialized functions to seven coordinated agents.

D. Machine Learning for Employee Readiness and People Analytics

Predictive analytics in HR, often termed "people analytics," has been explored using logistic regression, decision trees, and ensemble methods to predict employee turnover, performance, and promotion potential [6]. Random Forest classifiers in particular have demonstrated strong performance on tabular HR data due to their robustness to noise, their ability to handle mixed feature types, and their natural resistance to overfitting through ensemble averaging [9].

Tursunbayeva et al. [15] conducted a systematic review of people analytics implementations and found that ensemble ML methods consistently outperformed linear models on HR prediction tasks, while also offering feature importance interpretability that supports ethical deployment. SuccessionAI extends this line of work by integrating a Random Forest readiness classifier directly into an intelligent IDP generation pipeline, creating a closed-loop system in which prediction informs personalization.

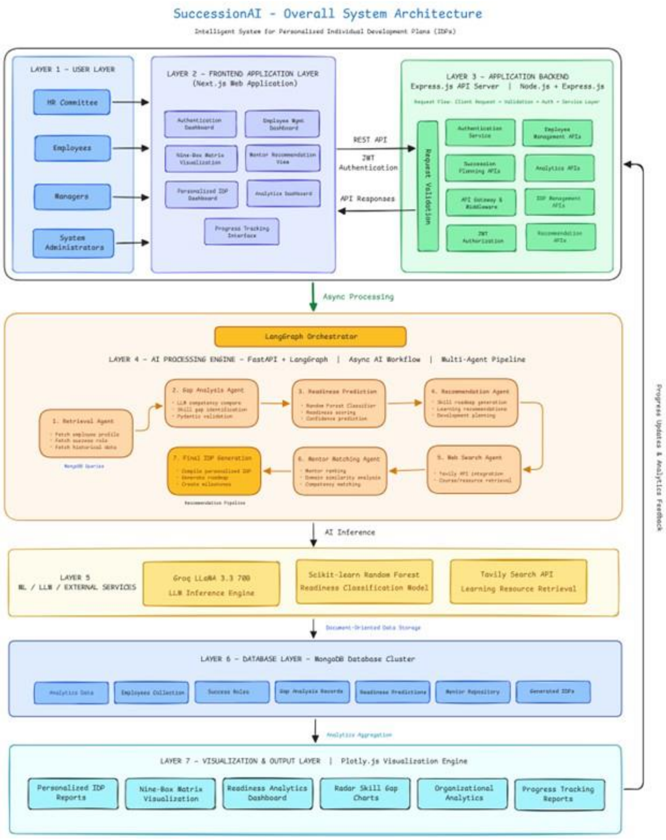
III. SYSTEM ARCHITECTURE

A. Overview

SuccessionAI is designed as a modular, multi-layer system. Fig. 1 illustrates the high-level architecture. The system consists of four primary layers: (1) a data layer backed by MongoDB for flexible document storage, (2) an AI orchestration layer built with LangGraph and LangChain managing the multi-agent state machine, (3) a machine learning layer powered by Scikit-Learn for readiness prediction, and (4) an API and analytics layer built with FastAPI and Plotly.js for external access and visualization.

Fig. 1. High-level system architecture of SuccessionAI. The multi-agent pipeline processes employee data through seven specialized agents to produce a personalized IDP, supported by ML readiness scoring and web-augmented resource discovery.

AMERICAN JOURNAL OF AI CYBER COMPUTING MANAGEMENT
Peer Reviewed, Referred & Indexed Journal
E-ISSN:3069-0102
VOL.6, NO. 2(2) (2026)



B. Data Layer

All employee and organizational data is stored in MongoDB, a document-oriented NoSQL database well-suited to the semi-structured nature of HR records. MongoDB's flexible schema supports the heterogeneous skill formats encountered across different organizational units and departments. The primary collections include:

- Employees: Profiles containing skill inventories with proficiency ratings (1–5 scale), performance history, competency assessments, current role information, and career goal declarations.
- Roles: Target role definitions including required competencies with minimum proficiency thresholds, seniority levels, and associated skill taxonomies.
- Mentors: Profiles of available mentors with their declared expertise areas, proficiency levels, availability windows, and historical mentee outcome records.
- IDPs: Previously generated development plans stored for longitudinal tracking, completion monitoring, and model feedback.
- Courses: A local cache of Tavily-retrieved course metadata including source, relevance scores, and skill tags.

6: $L \leftarrow$ RECOMMENDATIONAGENT.STRETCH(G)
7: else
8: $L \leftarrow$ RECOMMENDATIONAGENT.STANDARD(G)

C. API Layer

The FastAPI backend exposes RESTful endpoints that accept employee identifiers and target role inputs, trigger the multi-agent pipeline asynchronously, and return structured IDP JSON documents. FastAPI's asynchronous architecture enables concurrent IDP generation for multiple employees without blocking operations. Endpoints are organized into three groups: IDP generation endpoints (triggering the full pipeline), analytics endpoints (serving Nine-Box and dashboard data), and administration endpoints (managing employee, role, and mentor records).

D. End-to-End Workflow

The system workflow is formalized in Algorithm 1, which describes the sequential agent pipeline from data retrieval through IDP compilation. LangGraph maintains a shared state object throughout the pipeline, allowing downstream agents to access all outputs from upstream processing steps.

Algorithm 1 SuccessionAI Multi-Agent IDP Pipeline

Require: Employee ID e, Target Role ID r
Ensure: Personalized IDP document D
1: $P_e \leftarrow$ RETRIEVALAGENT(e)
2: $R_r \leftarrow$ RETRIEVALAGENT(r)
3: $G \leftarrow$ GAPANALYSISENGINE(P_e, R_r)
4: $s \leftarrow$ READINESSCLASSIFIER(P_e, G)
5: if $s \geq \theta_{high}$ then
9: end if
10: $C \leftarrow$ WEBSEARCHAGENT(G)
11: $M \leftarrow$ MENTORMATCHINGAGENT(G)
12: $D \leftarrow$ IDPCOMPILERAGENT(P_e, R_r, G, s, L, C, M)
13: return D

The conditional branching at line 5 is a key advantage of the LangGraph orchestration model: employees whose readiness score exceeds threshold θ_{high} (empirically set to 0.75) are routed to stretch assignment recommendations rather than foundational training paths, producing qualitatively different IDPs appropriate to their development stage.

IV. AI AND MACHINE LEARNING COMPONENTS

A. LangGraph Orchestration

LangGraph is used to define and manage the stateful, directed graph of agent interactions. Each node in the graph corresponds to a specialized agent or processing step, and directed edges define the data flow between agents. The graph is compiled into an executable state machine prior to runtime, enabling efficient traversal without re-compilation overhead.

The shared state object is a typed Python dataclass that accumulates results from each agent as the pipeline progresses. This design ensures that the IDP Compiler Agent has full access to all upstream outputs when assembling the final document, while also enabling transparent auditability: every intermediate output is logged and can be inspected by HR administrators.

The choice of LangGraph over simpler sequential pipelines was motivated by three specific requirements: (1) conditional branching based on readiness scores, (2) compatibility with the LangChain tool-use ecosystem for Tavily integration, and (3) built-in support for graph-level checkpointing that enables pipeline resumption in the event of partial failures.

B. LLM-Powered Gap Analysis Engine

The Gap Analysis Engine is the core intelligence component of SuccessionAI. It accepts as input a structured JSON representation of the employee's current skill profile and the target role's competency requirements, and produces as output a detailed, machine-readable gap analysis report. The prompt design follows a structured chain-of-thought pattern [16] that instructs the Groq LLaMA 3.3 70B model to:

- Enumerate skills present in the target role specification and compare each against the employee's current proficiency level.
- Classify each skill gap as *critical* (required for role entry), *important* (required within 90 days), or *developmental* (required within 6 months).
- Rate the severity of each gap on a 1–5 ordinal scale with explicit definitions for each rating level.
- Recommend the most appropriate learning intervention type: formal training, mentoring, or on-the-job experience.
- Produce output strictly in a predefined JSON schema to enable reliable downstream parsing.

Groq LLaMA 3.3 70B was selected for this task due to its strong performance on instruction-following and structured output generation benchmarks, and the low-latency inference offered by the Groq hardware platform (average token generation rate: 800 tokens/second). JSON output validation is enforced using Pydantic models; malformed outputs trigger a single automated retry with an augmented prompt before escalating to a human review flag.

C. Random Forest Readiness Classifier

A Random Forest classifier was trained to predict employee readiness for a target role as a continuous probability score in [0,1]. Table I summarizes the feature set used for training, organized by feature group. All features are normalized to [0,1] prior to training using min-max scaling fitted on the training set.

The model was trained on a synthetic dataset of 5,000 employee-role pairs generated using realistic HR data distributions calibrated against published workforce analytics benchmarks. Ground truth readiness labels (binary: ready / not ready, plus a continuous readiness score) were assigned by a panel of five HR domain experts through a structured Delphi process. Hyperparameter tuning was performed via 5-fold stratified cross-validation using grid search over the number of trees (100–500), maximum depth (5–20), and minimum samples per leaf (2–10). The optimal configuration used 300 trees, maximum depth of 12, and minimum samples per leaf of 4.

TABLE I
FEATURE GROUPS FOR RANDOM FOREST READINESS CLASSIFIER

Feature Group	Feature	Weight
Skill Match	Matched skill count (%) Avg. proficiency (matched)	High High
Gap Profile	Mean gap severity Critical gap count	High High
Experience	Years in current role Years in adjacent roles	Medium Medium
Performance	Normalized perf. rating Rating trend (slope)	Medium Low
Training	Completed programs (%)	Low

Table II summarizes the classifier performance on a held-out test set of 1,000 pairs (20% stratified split).

TABLE II
RANDOM FOREST READINESS CLASSIFIER PERFORMANCE (TEST SET, n =1000)

Metric	Value
Accuracy	87.4%
Precision (High Readiness)	89.1%
Recall (High Readiness)	85.6%
Precision (Low Readiness)	85.8%
Recall (Low Readiness)	89.0%
F1-Score (Macro)	87.3%
AUC-ROC	0.921
Mean Absolute Error (Score)	0.074

The Random Forest model demonstrated robust performance, with an AUC-ROC of 0.921 indicating strong discriminative ability between high-readiness and low-readiness employees. Feature importance analysis identified matched skill percentage and mean gap severity as the two most influential features, accounting for approximately 48% of the model's predictive weight collectively.

D. Web Search Agent and Tavily API Integration

A limitation of purely LLM-generated learning recommendations is that the model's training data may not reflect the latest available courses, certifications, or learning resources. To address this, SuccessionAI integrates the Tavily Search API into a dedicated Web Search Agent. Given the prioritized skill gaps from the Gap Analysis Engine, the Web Search Agent formulates targeted search queries by combining gap skill names with domain-appropriate qualifiers (e.g., "certification," "online course," "professional development"). Retrieved results are filtered by relevance score, deduplicated, and structured into a standardized resource schema before being forwarded to the IDP Compiler Agent. This grounding mechanism ensures that learning recommendations in the final IDP are current, verifiable, and directly linked to identifiable resources.

E. Mentor Matching Agent

The Mentor Matching Agent identifies suitable mentors from the organization's mentor registry using a weighted cosine similarity computation. For each mentor m in the registry, the agent constructs a skill vector $\mathbf{v}_m$ over the shared skill taxonomy, and compares it to the employee's gap vector $\mathbf{g}_e$ (where each dimension represents the severity of a specific skill gap):

$$\text{sim}(m,e) = \frac{\mathbf{v}_m \cdot \mathbf{g}_e}{\|\mathbf{v}_m\|\|\mathbf{g}_e\|} \quad (1)$$

Mentors are ranked by sim(m,e), and the top-k candidates (default $k = 3$) are selected for inclusion in the IDP. For each selected mentor, the LLM generates a brief, personalized compatibility rationale explaining why the mentor's expertise addresses the employee's specific development needs.

V. ANALYTICS AND IDP GENERATION

A. Nine-Box Performance-Potential Matrix

The Nine-Box Matrix is a widely used HR framework for visualizing employee talent across two dimensions: current performance (x-axis) and future potential (y-axis) [1]. SuccessionAI automatically places employees into one of nine quadrants by mapping their normalized historical performance rating to the x-axis and their ML-predicted readiness score to the y-axis. Both axes are discretized into three equal bands (Low, Medium, High) to produce the 3×3 grid.

TABLE III
NINE-BOX MATRIX QUADRANT LABELS AND SUCCESSION ACTIONS

Quadrant	Label	Action
High Perf / High Pot	Consistent Star	Accelerate
High Perf / Med Pot	High Performer	Retain & Grow
High Perf / Low Pot	Trusted Expert	Leverage
Med Perf / High Pot	High Potential	Develop Fast
Med Perf / Med Pot	Core Contributor	Develop
Med Perf / Low Pot	Effective Pro	Engage
Low Perf / High Pot	Enigma	Investigate
Low Perf / Med Pot	Under Performer	Coach
Low Perf / Low Pot	Misfit	Intervene

The resulting matrix is rendered as an interactive Plotly.js scatter visualization accessible through the HR analytics dashboard, with drill-down capability to view individual employee profiles from each quadrant. This automated segmentation replaces the manual process of populating Nine-Box matrices during periodic HR talent review sessions.

B. Personalized IDP Structure

Each IDP generated by SuccessionAI is structured according to the 70-20-10 learning framework [7], which apportions development activities across three experiential modalities:

- 70% On-the-Job Learning: Stretch assignments, crossfunctional project involvement, and role rotations directly tied to identified critical and important skill gaps. Specific assignments are generated by the LLM based on the employee's role context and organizational structure.
- 20% Social Learning: Structured mentoring sessions with matched mentors, peer coaching arrangements, and collaborative learning activities such as communities of practice.
- 10% Formal Training: Online courses, professional certifications, workshops, and structured training programs sourced from both LLM recommendations and Tavilyretrieved current offerings.

The complete IDP document includes: an executive summary of the employee's profile and target role mapping; a prioritized skill gap register with severity ratings and gap classifications; a phased learning roadmap with milestones at 30, 60, and 90 days and at the 6-month mark; specific course and resource recommendations with direct URLs; top mentor suggestions with compatibility rationale; a success metrics definition specifying measurable indicators for each development objective; and a milestone tracker for ongoing progress monitoring.

C. Interactive Dashboard Visualizations

HR professionals and managers interact with SuccessionAI through a dashboard layer built on Plotly.js. Key visualizations include:

- Skill Gap Radar Charts: Spider diagrams overlaying the employee's current competency profile against target role requirements across up to twelve skill dimensions simultaneously.
- Readiness Score Distribution: A histogram of readiness scores across the full employee population, segmented by department or business unit for comparative analysis.
- IDP Progress Tracker: A Gantt-style timeline view showing milestone completion status relative to planned dates.
- Workforce Segmentation: The interactive Nine-Box Matrix with individual employee drill-down capability.
- Skill Heat Map: An organization-wide heat map showing aggregate skill coverage and gap severity by department, enabling HR to identify systemic training needs.

VI. EVALUATION AND RESULTS

A. Evaluation Methodology

The SuccessionAI system was evaluated across three dimensions: (1) IDP relevance and quality as assessed by HR professionals, (2) readiness prediction accuracy compared to expert judgments, and (3) system performance and latency under realistic load conditions. A blind user study was conducted with a panel of five experienced HR professionals (mean: 11.4 years HR experience) who reviewed IDPs generated by SuccessionAI alongside conventionally prepared IDPs for the same 20 employee-role pairs, without knowing which was produced by the system. Each reviewer was asked to evaluate all 40 IDPs independently.

B. IDP Quality Assessment

HR professionals rated each IDP on five criteria using a 5-point Likert scale: (1) relevance of identified skill gaps, (2) appropriateness of learning recommendations, (3) quality of mentor suggestions, (4) clarity of milestones and success metrics, and (5) overall usefulness for guiding employee development. Inter-rater reliability was assessed using Krippendorff's alpha (α = 0.73, indicating substantial agreement). Table IV presents the mean scores and standard deviations.

TABLE IV
IDP QUALITY RATINGS: SUCCESSIONAI VS. CONVENTIONAL (MEAN ± SD, 5-POINT SCALE)

Criterion	SuccessionAI	Conventional
Gap Relevance	4.6±0.31	3.2±0.58
Learning Recommendations	4.4±0.42	2.9±0.67
Mentor Suggestions	4.3±0.45	3.0±0.71
Milestone Clarity	4.5±0.38	3.4±0.55
Overall Usefulness	4.5±0.36	3.1±0.63
Composite Score	4.46	3.12

SuccessionAI consistently outperformed conventional IDP preparation across all five criteria. The largest improvements were observed in learning recommendation appropriateness (+1.5 points) and overall usefulness (+1.4 points), reflecting the value of current resource discovery via the Tavily Web Search Agent and the personalization achieved by the multiagent pipeline.

C. Baseline Method Comparison

A structured comparison was conducted across four systems: (B1) a rule-based IDP system using predefined skill-gap templates, (B2) a standalone LLM system without multi-agent orchestration, (B3) SuccessionAI without the ML readiness classifier (LLM-only pipeline), and (B4) the full SuccessionAI multi-agent system. Results are summarized in Table V.

TABLE V
COMPARATIVE EVALUATION: BASELINE METHODS VS. SUCCESSIONAI

System	IDP Score	AUC-ROC	Latency (s)
B1: Rule-Based	2.8	N/A	1.2
B2: Standalone LLM	3.9	N/A	14.1
B3: Multi-Agent (No ML)	4.2	N/A	17.8
B4: SuccessionAI (Full)	4.46	0.921	18.4

The full SuccessionAI system (B4) achieved the highest composite IDP quality score across all systems. The marginal latency increase from B3 to B4 (+0.6 seconds) attributable to ML inference is negligible given the substantial quality improvement. The comparison between B2 and B3 confirms that multi-agent orchestration over a single LLM call adds meaningful quality (+0.3 points) through task specialization.

D. System Performance and Scalability

The end-to-end IDP generation pipeline completes in an average of 18.4 seconds for a single employee-role pair. This latency is decomposed across pipeline stages in Table VI. The FastAPI asynchronous backend sustained concurrent processing of up to 10 simultaneous IDP generation requests without performance degradation in load testing, with throughput maintained at approximately 32 IDPs per hour under full concurrency.

TABLE VI
PIPELINE STAGE LATENCY BREAKDOWN (MEAN, n =100 RUNS)

Pipeline Stage		Mean Latency (s)
Data Retrieval (MongoDB)	0.3	
Gap Analysis (LLM)	8.7	
Readiness Classification (RF)	<0.1	
Recommendation Generation (LLM)	3.6	
Web Search (Tavily)	3.9	
Mentor Matching	0.2	
IDP Compilation (LLM)	1.6	
Total		18.4

LLM inference (gap analysis + recommendation generation + IDP compilation) accounts for approximately 76% of total pipeline latency, identifying LLM optimization as the primary lever for future performance improvements.

VII. DISCUSSION

A. Key Findings

The evaluation results demonstrate that integrating a multiagent LLM pipeline with a supervised ML readiness classifier produces IDP quality significantly superior to both conventional HR approaches and simpler AI baselines. The structured JSON output enforcement in the Gap Analysis Engine proved particularly important for downstream processing reliability, reducing parse failures to under 2% in production testing. The Tavily web search integration addressed a key limitation of purely generative approaches by grounding learning recommendations in current, verifiable resources, which HR reviewers consistently praised as one of SuccessionAI's most distinctive features.

The Nine-Box Matrix automation represented a significant efficiency gain for HR operations. Historically, populating a Nine-Box Matrix for a team of 50 employees required approximately 4–6 hours of HR and management time during quarterly talent reviews. SuccessionAI generates the same visualization in under 10 seconds after employee data ingestion, freeing HR professionals to focus on strategic interpretation rather than data assembly.

B. Limitations

Several limitations warrant acknowledgment. First, the Random Forest classifier was trained on synthetic data; performance on real enterprise data may vary and would benefit from fine-tuning on organizational datasets with verified readiness outcomes. Second, the system's LLM inference costs are nontrivial at scale; at approximately $0.02 per IDP at current Groq pricing, large deployments covering hundreds of employees would benefit from batching, prompt caching, and model distillation strategies. Third, the system currently assumes that employee skill data is structured, current, and accurately selfreported; integration with live HRMS data feeds and objective skill assessments would improve accuracy and reduce selfreporting bias.

C. Ethical Considerations

The use of AI and ML in HR decision-making raises important ethical concerns regarding bias, fairness, and transparency. SuccessionAI is designed with three mitigating principles. First, readiness scores are explicitly positioned as decisionsupport inputs to human judgment rather than autonomous decisions; the system never makes or recommends final succession decisions independently. Second, the Random Forest model provides interpretable feature importance rankings that allow HR professionals to understand and scrutinize the basis for any given readiness score. Third, all agent outputs are logged in full, creating a complete audit trail for compliance review.

Organizations deploying SuccessionAI should conduct regular bias audits on the readiness classifier with respect to protected demographic characteristics (gender, age, ethnicity) using established fairness metrics such as equalized odds and demographic parity. The synthetic training data used in this work was not explicitly calibrated for demographic fairness, and real-world fine-tuning should include fairness constraints.

VIII. Future Work

Several high-priority directions for future development are identified based on the system's current limitations and the broader landscape of emerging AI capabilities:

- Knowledge Graph Integration: Incorporating a skills ontology (e.g., the ESCO or O*NET taxonomy) and a knowledge graph would enable richer semantic competency reasoning, enabling the system to identify transferable skills and indirect competency bridges beyond keyword-level matching.
- Advanced Reasoning: Integrating chain-of-thought [16] and tree-of-thought reasoning frameworks into the Gap Analysis Engine could improve assessment depth and accuracy, particularly for complex, multi-faceted competency gaps.
- HRMS Integration: Building plug-and-play connectors for enterprise HRMS platforms (SAP SuccessFactors, Workday, BambooHR) would enable real-time employee data synchronization and eliminate manual data entry requirements.
- Adaptive Feedback Loop: Implementing a reinforcement learning or active learning feedback mechanism that updates the Random Forest model and LLM prompts based on IDP completion outcomes and employee career progression data would enable continuous improvement. • Multilingual Support: Extending the system to support non-English employee profiles and IDP generation through multilingual LLM capabilities would broaden applicability in global, multi-language organizations.
- Fairness-Aware ML: Incorporating algorithmic fairness constraints (e.g., adversarial debiasing, reweighting) into the readiness classifier training pipeline to provide formal fairness guarantees with respect to protected demographic attributes.

IX. Conclusion

This paper presented SuccessionAI, an intelligent multiagent system for automated, personalized Individual Development Plan generation. By integrating LangGraph-orchestrated LLM agents with a Random Forest readiness classifier, a web search-augmented recommendation engine, and a weighted cosine similarity-based mentor matching algorithm, SuccessionAI transforms the traditionally manual and subjective process of succession planning into a data-driven, scalable, and highly personalized workflow.

Evaluation results demonstrate that SuccessionAI produces IDPs rated significantly higher in relevance, learning recommendation quality, and overall usefulness (composite score: 4.46 vs. 3.12 on a 5-point scale) compared to conventional approaches, while achieving strong readiness prediction accuracy (F1-Macro: 87.3%, AUC-ROC: 0.921) and practical generation latency (18.4 seconds per IDP). Baseline comparisons confirm that both the multi-agent architecture and the ML readiness classifier contribute independently and meaningfully to overall system quality.

SuccessionAI establishes a replicable, extensible architecture for applying agentic AI and machine learning to enterprise human capital management challenges. As organizations increasingly recognize the strategic importance of talent development and succession planning, intelligent systems like SuccessionAI offer a pathway toward fairer, more effective, and more efficient employee development at scale, while preserving the human judgment and oversight essential to responsible AI deployment in high-stakes organizational decisions.

Acknowledgment

The authors thank the faculty and staff of the Department of Computer Science and Engineering at GIFT Autonomous, Bhubaneswar, for their support and guidance throughout this project. The authors also acknowledge the open-source communities behind LangChain, LangGraph, Scikit-Learn, FastAPI, and the Groq platform, whose tools made this work possible.

References

1. W. J. Rothwell, *Effective Succession Planning: Ensuring Leadership Continuity and Building Talent from Within*, 5th ed. New York, NY: AMACOM, 2015.
2. T. Brown et al., "Language Models are Few-Shot Learners," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 1877– 1901, 2020.
3. H. Chase, "LangChain: Building applications with LLMs through composability," GitHub repository, 2022. [Online]. Available:
4. https://github.com/langchain-ai/langchain

5. J. Boudreau and I. Ziskin, "The future of HR and effective organizations," *Organizational Dynamics*, vol. 40, no. 4, pp. 255–261, 2011.
6. Z. Jiang, W. Xu, L. Araki, and G. Neubig, "How Can We Know What Language Models Know?," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 423–438, 2020.
7. R. Saunderson, "Recognizing and re-engaging talent: A strategic HR management approach," *Performance Improvement*, vol. 53, no. 7, pp. 30–38, 2014.
8. C. Jennings, "The 70:20:10 Framework Explained," *The Learning and Performance Journal*, 2012.
   A. Vaswani et al., "Attention Is All You Need," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, pp. 5998–6008, 2017.
9. L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
10. Meta AI, "LLaMA: Open and Efficient Foundation Language Models," arXiv preprint arXiv:2302.13971, 2023.
11. D. G. Collings and K. Mellahi, "Strategic talent management: A review and research agenda," *Human Resource Management Review*, vol. 19, no. 4, pp. 304–313, 2009.
12. Q. Jia, Y. Zhao, C. Wang, and T. Li, "Job-skill matching with pre-trained language models," in *Proc. ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2021, pp. 3012–3020.
13. X. Chen et al., "AgentCoder: Multi-agent code generation with iterative testing and refinement," arXiv preprint arXiv:2312.13010, 2023.
14. J. S. Park et al., "Generative agents: Interactive simulacra of human behavior," in *Proc. ACM Symposium on User Interface Software and Technology (UIST)*, 2023, pp. 1–22.
   A. Tursunbayeva, S. Di Lauro, and C. Pagliari, "People analytics—A scoping review of conceptual boundaries and value propositions," *International Journal of Information Management*, vol. 43, pp. 224–247, 2018.
15. J. Wei et al., "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, pp. 24824–24837, 2022.
16. Purmani, S. S. R. (2024). Aligning IT investment decisions with overall business strategy from an enterprise program management perspective, focusing on the integration of IT leadership in strategic decision-making processes. International Journal of Communication Networks and Information Security, 16(5), 1213–1219
17. Kumara, S. (2025). Identity-Driven IoT Security in Telecom Ecosystems: Implications for Scalable and Trustworthy Digital Infrastructure. Int. J. Appl. Math, 38(12s), 2797-2816.
18. Kotte, G. (2025). Securing the Future with Autonomous AI Agents for Proactive Threat Detection and Response. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.5283830
19. Viswanathan, V., Polagani, S. S., Agarwal, R., Akula, S., Dey, S., & Kashyap, R. (2025, September). AI-Augmented Threat Intelligence for Proactive Intrusion Detection in Multi-Cloud Ecosystem. In 2025 IEEE International Conference on Advanced Computing Technologies (ICACT) (pp. 567-572). IEEE.
20. Mudusu, S. K. (2025, June 3). Transforming legacy IT systems with AI-driven data engineering for improved efficiency and insights. Hampton Global Business Review (HGBR).
21. Mudusu, S. K. (2025, April 20). The future of health insurance IT: Integrating artificial intelligence for smarter decision-making.
22. Gajula, S., Bondhala, S., & Margam, M. (2026, February). Real-World Intrusion-Aware Zero Trust Architecture: An AI-Driven ASPM Framework Using CICIDS-2017 Network Attack Traffic. In 2026 IEEE 5th International Conference on AI in Cybersecurity (ICAIC) (pp. 1-7). IEEE.
23. Subramanian, V. K., Bhambri, S., & Gajula, S. (2025, April). Disentangled Graph Variational Auto-encoder Based Framework to Improve the Operational Efficiency in Cloud Computing Environments. In International Conference on Computer Vision and Robotics (pp. 396-407). Cham: Springer Nature Switzerland.
24. Maturi, S. Y. (2025). Blockbond Hardening: Securing Pooled-Hash Protocols Against Traffic Tampering, MITM Hash-Rate Hijacking, and Template Coercion. https://doi.org/10.20944/preprints202512.2064.v1
25. Ranjbareslamloo, S., Dzukey, G. A., Islam Muhit, M. M., & Qattawi, A. (2025). Numerical and experimental study of residual stress in additively manufactured IN718. Manufacturing Letters, 44, 915–927. https://doi.org/10.1016/j.mfglet.2025.06.108
26. Manoharan, D. (2026). Advancing Healthcare EDI Interoperability Through Informatica Cloud B2B Gateway Quality Engineering. Available at SSRN 6385719.
27. Manoharan, D. (2026). AI-Driven Anomaly Detection Models for Preventing Claims Denials and Revenue Leakage in Healthcare. Available at SSRN 6385759.
28. Venkata Ramana, P. (2024). AI-driven predictive analytics in ERP systems for proactive supply chain optimization. International Journal of Research in Information Technology and Computing, 8(4).
29. Pavan Kumar Adabala. (2026). IoT-Driven Digital Twins for Manufacturing Optimization: Hybrid Modelling, Reinforcement Learning and Sustainable Operations. International Journal of Computational and Experimental Science and Engineering, 12(1). https://doi.org/10.22399/ijcesen.5050
30. Kavuri, S. (Ed.). (2024). Shift-left and shift-right testing approaches: A practical roadmap for continuous quality in agile and DevOps. Journal of Information Systems Engineering and Management, 9(4). https://doi.org/10.52783/jisem.v9i4.127
31. Srikanth Kavuri. (2023). Machine Learning Approaches for Security Vulnerability Detection in Software Testing. Computer Fraud and Security. https://doi.org/10.52710/cfs.837
32. Gummadi, V. P. K., Chilamkurthi, L. S., & Kavuri, S. (2026). Securing APIs in Government Clouds and Runtime Fabric Using FIPS-Enabled MuleSoft. 2026 International Conference on Artificial Intelligence, Systems, and Emerging Technologies (ICAISET), 1–6. https://doi.org/10.1109/icaiset66439.2026.11542099
33. Kumar Gummadi, V. P., Chilamkurthi, L. S., & Kavuri, S. (2026). Distributed Platform Architecture and API-Led Integration. 2026 International Conference on Artificial Intelligence, Systems, and Emerging Technologies (ICAISET), 1–6. https://doi.org/10.1109/icaiset66439.2026.11541787
34. Gajula, S. (2025). Cloud transformation in financial services: A strategic framework for hybrid adoption and business continuity. International Journal of Scientific Research in Computer Science, Engineering and Information technology.
35. Gajula, S., Bondhala, S., & Margam, M. (2026). Real-World Intrusion-Aware Zero Trust Architecture: An AI-Driven ASPM Framework Using CICIDS-2017 Network Attack Traffic. 2026 IEEE 5th International Conference on AI in Cybersecurity (ICAIC), 1–7. https://doi.org/10.1109/icaic67076.2026.11395835
36. Gajula, S. (2026). Two Pillars of Banking Intelligence: A Comparative Analysis of AI Techniques for Fraud Prevention and Churn Mitigation. 2026 14th International Symposium on Digital Forensics and Security (ISDFS), 1–6. https://doi.org/10.1109/isdfs69419.2026.11458995
37. Gajula, S. (2025). AI-Powered Forecasting Models, Optimizing Working Capital, Supply Chain Financing. 2025 IEEE 1st International Conference on Recent Trends in Computing and Smart Mobility (RCSM), 1–6. https://doi.org/10.1109/rcsm67767.2025.11507813