Blueprint for an AI-Powered Program Management Internship Recommendation Engine: Ten Novel Concepts for a Hackathon

1. The Foundation: Deconstructing the Program Management Intern Role

1.1. Defining the PM Intern: Beyond the Title

The development of an effective AI-based internship recommendation engine necessitates a precise and expert-level understanding of the target role. A common misconception conflates the duties of a Program Management (PM) Intern with a senior Program Manager or other management roles, such as Project and Product Management. An expert system must first delineate these distinctions to avoid fundamental errors in its core logic.

The responsibilities of a PM intern are notably tactical and foundational, serving as a blend of coordination and strategic support. A PM intern is tasked with practical, day-to-day duties, including the development and leadership of team schedules, delivering project-related communications, and tracking progress. They are also expected to provide direct program management support to various project owners to ensure that projects are properly resourced, on track, and within budget. This contrasts sharply with the classical role of a Program Manager, which is often described as a more senior position focused on overseeing a large "book of projects" and ensuring they align with broader strategic outcomes and key performance indicators. The senior Program Manager is considered a "CEO's front line worker" who focuses on the "big picture" and is less involved in the hands-on, day-to-day operations.

This tactical-strategic dichotomy presents a critical challenge for a naive recommender system. If the system is trained on generic "Program Manager" job descriptions, which are laden with senior-level responsibilities, it will inevitably misclassify the requirements of an internship. A system grounded in the nuanced reality of the PM intern role must be trained on a specialized corpus of data from PM internships. The essential skills for an intern are a tactical subset of the senior PM's strategic skillset, aimed at demonstrating a "high potential for exceptional achievement" rather than a track record of executive oversight. This is the foundational problem that any truly unique and helpful hackathon project must solve.

1.2. Core Competencies and Skills: What to Look For

To build a robust recommendation engine, a clear understanding of the core competencies for a PM intern is essential. The qualifications can be segmented into three categories: hard skills, soft skills, and academic background.

Hard skills for this role often include the ability to conduct qualitative and quantitative analysis from research and data sets, visualize complex processes using mappings, and provide program management support for projects. Academic preferences typically lean toward a bachelor's or master's degree in fields such as Engineering, Business, Computer Science, or Finance. A fundamental understanding of project management concepts and a keen eye for detail are also considered prerequisites.

However, the research strongly and repeatedly emphasizes the paramount importance of soft skills. These include leadership, communication (both verbal and written), problem-solving, organizational abilities, and emotional intelligence. Such traits are crucial for success in a role that involves constant cross-functional collaboration, stakeholder engagement, and the need to communicate effectively across diverse audiences.

This emphasis on soft skills introduces a significant challenge, often referred to as the "soft skill paradox." The qualities that are most critical for success in a PM intern role—such as leadership potential, creativity, and adaptability—are the most difficult for a machine learning algorithm to quantify, as they do not "fit neatly into an algorithm". Traditional AI hiring tools, which excel at analyzing structured data like keywords and technical skills, tend to struggle with these human qualities. This contradiction implies that a groundbreaking AI engine cannot simply ignore soft skills; it must devise novel methodologies to infer and assess them from unstructured data.

1.3. The Inadequacies of Traditional Keyword Matching

The current landscape of recruitment technology, dominated by Applicant Tracking Systems (ATS), often relies on a rudimentary form of keyword matching. While this approach offers a degree of efficiency, it is fraught with limitations that result in a significant loss of qualified talent. A key point of contention in the research is whether ATS systems rank candidates by keyword relevance or simply by the order in which they apply. Regardless of the specific mechanism, a reliance on simple keyword filtering creates "significant blind spots" and can reject millions of qualified candidates who do not use the exact terms the system is programmed to detect.

Traditional systems also fail to understand semantic nuances and context-dependent meanings, a critical shortcoming. For instance, they might not recognize that "software developer" is a synonym for "programmer" or grasp the difference in meaning between "lead engineer" and "engineering lead". This lack of contextual understanding leads to suboptimal candidate-job alignment. A more intelligent system is therefore not just a minor improvement; it is a necessary evolution to overcome these systemic failures. The following table consolidates the essential requirements for a PM intern, providing a clear target for the engine's advanced matching algorithms.

Competency Category	Specific Skills and Qualities	Examples from Research
Hard Skills	- Qualitative & Quantitative Analysis	Develop qualitative and quantitative analysis from key research and data sets
	- Project Management Methodologies	Help manage a cross functional team to meet program requirements and milestone goals
	- Process Mapping & Documentation	Visualize complex processes using process mappings
	- Communication & Status Updates	Deliver project-related communications and status updates

Competency Category	Specific Skills and Qualities	Examples from Research
	- Cross-functional Coordination	Work with operations and supply chain to coordinate installation
	- Data Analysis	Generate customer quotes and develop system to track approval rates
Soft Skills	- Communication (Verbal & Written)	Exceptional interpersonal skills, verbal and written communication
	- Organizational & Planning Abilities	Organizational and planning abilities
	- Problem-Solving & Leadership	Strong leadership, communication, and problem-solving skills are essential for growth
	- Emotional Intelligence	Ability to recognize, understand, and manage one's own emotions as well as those of others
	- Self-driven & Creative	Must be a self-driven, creative, self-starter
Academic Background	- Relevant Degree	Bachelor's and Master's Degree in Engineering, Business, Management, Computer Science, MBA
	- Human Resources / Administration	Undergrad student currently pursuing a Bachelor's degree in Human Resources, Business Administration

2. The Technical Core: A Hybrid AI Architectural Framework

2.1. From Simple to Sophisticated: A Look at Recommendation Systems

To architect a powerful recommendation engine, it is necessary to move beyond a single-paradigm approach. The three dominant types of recommendation systems—collaborative filtering, content-based filtering, and hybrid models—each have distinct advantages and disadvantages. Understanding these trade-offs informs the decision to build a hybrid system, as it is the only viable path for a complex problem like job matching.
Collaborative filtering operates on the principle that users who have shown interest in similar items will likely share preferences for new items. This approach is effective for generating personalized recommendations without detailed product metadata. However, it is plagued by the "cold start problem," where new users or internships lack sufficient interaction data, making it

difficult to provide initial recommendations.
Content-based filtering, in contrast, recommends items by analyzing their characteristics and matching them to user preferences. This method relies on item metadata such as descriptions and attributes, and it can handle new items effectively. Its primary limitation is that it tends to reinforce existing preferences, leading to a lack of diversity or "serendipity" in recommendations.
The most advanced and robust systems, as confirmed by the research, employ **hybrid models** to combine the strengths of both approaches while mitigating their limitations. A specific hybrid model architecture integrates traditional techniques, such as Term Frequency-Inverse Document Frequency (TF-IDF), with advanced Large Language Models (LLMs). This multi-modal approach leverages the ability of traditional methods to capture quantifiable, domain-specific aspects while using LLMs to provide deep contextual and semantic understanding. This fusion represents a synergy of old and new, and it is the ideal architectural blueprint for an AI-powered PM internship engine. The final system would not be "LLMs only," but rather a sophisticated, dual-pipeline architecture that demonstrates a nuanced understanding of the technology's strengths and weaknesses.
The following table provides a clear comparison of the three paradigms.

Paradigm	How it Works	Strengths	Limitations
Collaborative Filtering	Based on user interactions. Recommends items liked by similar users.	- Highly personalized. - No item metadata needed.	- Suffers from the "cold start problem" for new users or internships. - Cannot recommend items that no similar user has interacted with.
Content-Based Filtering	Based on item characteristics and user preferences. Recommends items similar to those previously liked.	- No "cold start problem" for new items. - Easy to explain recommendations.	- Tends to reinforce existing preferences, limiting exposure to new content. - Requires detailed item metadata.
Hybrid Models	Combines multiple techniques, often blending collaborative and content-based methods.	- Mitigates the "cold start problem." - Enhances accuracy and personalization. - Provides more diverse recommendations.	- More complex to build and maintain. - Requires more data and computational resources.

2.2. Acknowledging AI's Imperfections: Bias, Transparency, and Soft Skills

An expert-level report on AI in recruitment must address the significant ethical and legal challenges inherent in the technology. Building a responsible system requires an ethical-by-design philosophy that acknowledges AI's potential for algorithmic bias, its "black box" nature, and its limitations in evaluating human qualities.
Algorithmic bias is one of the most pressing concerns. AI systems learn from historical data, and if that data reflects existing human biases, the algorithms can inadvertently replicate and

even amplify them, leading to discriminatory outcomes. The well-known example of Amazon's hiring tool, which favored male candidates because it was trained on historical data that skewed heavily male, serves as a powerful cautionary tale. This not only creates an unfair hiring process but also introduces serious legal risks, potentially violating anti-discrimination laws like Title VII of the Civil Rights Act.

A **lack of transparency** further exacerbates these issues. Many AI tools function as "black boxes," making it difficult for an employer to explain or justify their hiring practices, which can be problematic in the event of legal investigation or litigation. This opacity also creates a poor candidate experience; when applicants are rejected without explanation, they may feel confused, frustrated, and unfairly excluded, which erodes trust in the system. A growing demand for accountability and transparency in technology makes this a critical area to address.

Finally, as noted previously, AI systems struggle to evaluate the human qualities that drive long-term success. They excel at analyzing structured data like keywords but fail to grasp the nuances of soft skills such as emotional intelligence, creativity, and adaptability. Over-reliance on technical skills and keywords can lead to "missed talent" and poor hiring decisions.

Therefore, a truly innovative and responsible hackathon project will not just be functional; it will be auditable and transparent, building a system that turns these liabilities into core features. The most mature and viable solution will address these concerns head-on, differentiating itself from competitors and building trust with both candidates and recruiters.

3. The Concepts: Ten Unique Ideas for the PM Internship Engine

The following ten concepts represent innovative and unique ideas for a hackathon, each addressing a specific problem identified in the analysis.

Concept Title	Primary Purpose	Key Technologies
1. The Project-Based Proficiency Analyst	Uncovering functional experience beyond formal job titles.	Text Mining, NLP, Named Entity Recognition (NER)
2. The Semantic Similarity and Transferable Skills Mapper	Finding highly-qualified candidates from non-traditional backgrounds.	Graph Neural Networks (GNNs), LLMs, Word/Document Embeddings
3. The PM Intern Skill Gap Navigator	Empowering candidates by providing a personalized learning path.	NLP, TF-IDF, Recommendation Systems
4. The Soft Skills Behavioral and Contextual Assessor	Quantifying soft skills by analyzing unstructured text.	NLP, Sentiment Analysis, STAR Method Framework
5. The Ethical AI Transparency Dashboard	Building trust and reducing legal risk through explainable AI.	Explainable AI (XAI)
6. The Dynamic Context-Aware Ranker	Improving the utility of recommendations with real-time data.	Learning to Rank (LTR), Contextual Ranking Algorithms
7. The Community-Driven Collaborative Recommender	Providing peer-validated career path recommendations.	User-based Collaborative Filtering
8. The Gamified PM	Assessing real-world	Rule-based Systems, NLP,

Concept Title	Primary Purpose	Key Technologies
Simulation Interviewer	problem-solving skills in a standardized way.	STAR Method
9. The Resume Optimization Coach	Giving candidates actionable, real-time feedback to improve their resumes.	NLP, Keyword Extraction, Semantic Analysis
10. The End-to-End Predictive Performance Model	Predicting a candidate's likelihood of success in the role.	Supervised Machine Learning (e.g., Classifier/Regressor)

3.1. The Project-Based Proficiency Analyst

This idea moves beyond simple keyword-based resume screening by analyzing a candidate's resume for detailed project descriptions. The engine would leverage text mining, NLP, and machine learning to extract quantifiable experience and skills from the narrative of past projects and match them to the project types mentioned in PM internship job descriptions. For example, instead of just checking for the keyword "Agile," the system would use Named Entity Recognition (NER) to parse a project description and identify a candidate who "managed a cross functional team to meet program requirements" or "tracked hardware deliveries to testing labs," which are core PM intern responsibilities mentioned in the research. This approach addresses a fundamental limitation of traditional systems: their focus on rigid job titles. The analysis indicates that a person can hold all three roles—Project, Program, and Product Manager—or that a Product Manager may take on Project Manager duties. A tool that focuses on the substance of a candidate's project-based experience rather than their job title can uncover hidden talent with transferable skills. This represents a paradigm shift from asking, "What was your job title?" to "What did you *do*?".

3.2. The Semantic Similarity and Transferable Skills Mapper

To find highly qualified candidates from non-traditional backgrounds, this concept proposes a state-of-the-art technical approach using Graph Neural Networks (GNNs) and Large Language Models (LLMs). Traditional models struggle with the semantic nuances of language, such as synonymy and polysemy. This system would represent a resume and a job description as a graph, where nodes are key concepts (e.g., skills, experience, education) and edges capture the relationships between them, such as proficiency levels. An LLM would enrich this graph with contextual and semantic information, enabling the GNN to learn complex patterns and relationships that a keyword-based system would miss. By modeling these semantic and structural relationships, the system can find matches that are semantically close to the requirements, even if they are "extremely distant" in terms of keywords. This enables the engine to serve as an intelligent talent scout, finding exceptional candidates who might be completely overlooked by traditional ATS systems.

3.3. The PM Intern Skill Gap Navigator

This concept flips the script from a passive screening tool to an active development coach. After analyzing a candidate's resume and a target internship's job description, the system would identify specific skill deficiencies and recommend a personalized learning path. Using NLP to extract skills and a comparison method like TF-IDF or vectorization, the engine could pinpoint critical missing skills and link them to relevant courses and certifications from platforms like Coursera or LinkedIn Learning. This idea provides tangible value to candidates even if they are not a perfect match, addressing a growing trend toward continuous learning and development in the workforce. By providing actionable feedback and a clear learning path, the system not only improves a candidate's chances for future internships but also builds trust and loyalty to the platform, inverting the traditional one-sided recruitment paradigm.

3.4. The Soft Skills Behavioral and Contextual Assessor

Directly addressing the "soft skill paradox," this concept proposes a module that uses advanced NLP and sentiment analysis to infer critical human qualities from unstructured text on a resume. The system would analyze a candidate's project descriptions or personal statements, assessing their tone, word choice, and contextual understanding. By applying a framework like the STAR method (Situation, Task, Action, Result) to these written descriptions, the engine could provide a "soft skills" score based on an assessment of a candidate's written communication patterns. For instance, a project summary that uses collaborative language and acknowledges team efforts could be scored higher for teamwork and emotional intelligence. This method provides a non-intrusive way to screen for critical soft skills and mitigates the need for separate, potentially biased, video interview platforms. The way a candidate describes their past work can be a strong predictor of their soft skills.

3.5. The Ethical AI Transparency Dashboard

To combat the "black box" nature of AI hiring tools and build trust, this idea introduces a user-facing dashboard that explains the rationale behind a recommendation. The dashboard would provide a breakdown of the match score, highlighting which hard skills, soft skills, and experiences contributed to the recommendation. It would also inform candidates that AI is being used in the process, a practice that is becoming a legal and ethical requirement in some jurisdictions. By making the AI's logic transparent, this system can differentiate itself from competitors, improve the candidate experience, and reduce legal risk. It transforms a liability into a core feature, positioning the project as a responsible and trustworthy system rather than a simple, opaque tool.

3.6. The Dynamic Context-Aware Ranker

A relevant match is not always a useful match. This concept proposes a system that goes beyond a simple relevance score by incorporating real-time and contextual data to improve the utility of recommendations. Using a Learning to Rank (LTR) algorithm, the engine would factor in data points such as job posting recency, user engagement (e.g., time spent viewing a recommendation), and proximity to the user's preferred location. A perfect match for a job that was posted three months ago is less valuable than a 90% match for a job posted yesterday. This approach demonstrates an understanding of the difference between a proof-of-concept and a production-ready system. It shows a forward-thinking approach to user experience and practical application by balancing precision, diversity, and contextual ranking to provide a complete and useful experience.

3.7. The Community-Driven Collaborative Recommender

This idea leverages the power of social and behavioral data to provide a dynamic, peer-validated recommendation model. The system would recommend internships based on the career paths and educational backgrounds of other successful PM interns, as well as jobs that other professionals with similar traits have previously applied for. This approach is built on the assumption that users with similar tastes will have similar preferences in the future. By grouping new users into "archetypes" based on their profile information, the system can provide relevant and aspirational recommendations that are not just based on a static resume but on a dynamic, community-driven career path. This adds a powerful layer to the engine, making it feel more like a personalized career mentor than a simple matching algorithm.

3.8. The Gamified PM Simulation Interviewer

This concept goes beyond static resume screening by introducing an interactive module that assesses a candidate's problem-solving and decision-making skills in a quantifiable way. The module would present candidates with PM-related scenarios, such as resource allocation or stakeholder communication, in a structured, role-play simulation. Using a pre-defined set of scenarios and evaluating responses based on the STAR method or competency frameworks, the system can objectively assess critical PM skills that are difficult to capture on a resume, such as adaptability, critical thinking, and negotiation. This approach addresses the bias and inconsistency found in traditional human-led interviews and provides a standardized, scalable way to evaluate a candidate's behavioral tendencies and future potential.

3.9. The Resume Optimization Coach

A highly pragmatic and marketable feature, this idea proposes a tool that provides real-time, actionable feedback to candidates on their resume. It would analyze a user's resume against a specific PM job description, telling them which skills are missing and providing suggestions for how to "naturally weave in the keyword". The system would leverage NLP and keyword extraction to provide feedback on ATS compatibility, word choice, and brevity. This concept solves a clear and present user need—the frustration of trying to get past an ATS and land an interview. By providing immediate, tangible value to the user, this feature would drive engagement and implicitly train the model by gathering data on how candidates optimize their resumes.

3.10. The End-to-End Predictive Performance Model

This is the ultimate long-term vision for the project, moving beyond simple matching to genuine predictive analytics. The concept proposes a comprehensive system that uses both candidate data and historical hiring outcomes (e.g., intern success, retention, performance reviews) to predict a candidate's likelihood of success in a PM role. The model would be a supervised machine learning model (e.g., a classifier or regressor) trained on a large dataset of past intern applications and their performance metrics. The evidence suggests that a system that learns from a feedback loop of actual outcomes will continuously improve its accuracy and provide a powerful, end-to-end solution for recruiters and candidates alike.

4. Implementation and Evaluation: Building a Robust

System

4.1. Data Collection, Preprocessing, and Transformation

The practical implementation of the proposed engine begins with meticulous data collection and preparation. The two primary data sources are unstructured text, such as resumes and job descriptions, and structured data, including user profiles and job metadata. Unstructured text requires a rigorous preprocessing pipeline to be usable for machine learning. This involves several key steps: tokenization, which splits the text into individual words; stop-word removal, which eliminates common words that do not contribute to analysis; and stemming or lemmatization, which reduces words to their root form. After preprocessing, the text must be transformed into a structured, numerical format using techniques like Named Entity Recognition (NER), which extracts and classifies key entities like skills, and vectorization, which converts text into numerical vectors that represent word frequency or presence.

4.2. Selecting and Training the Optimal Model

For the core recommendation engine, a hybrid architecture is the most effective approach, blending traditional methods with modern deep learning models. The system can be built using established machine learning frameworks like TensorFlow and PyTorch. A supervised learning model, such as a classifier, could be trained to predict the likelihood of a successful match. The training process would involve optimizing the model's parameters to maximize the probability of a positive match while minimizing the probability of a negative match, using a Bayesian Personalized Ranking criterion.

4.3. The Critical Metrics for Success: Precision, Diversity, and Novelty

To prove the effectiveness of the engine, evaluation must go beyond simple accuracy. A sophisticated framework is necessary to measure the system's performance across multiple dimensions.
Predictive Metrics are crucial for measuring the engine's ability to recommend relevant items. **Precision@K** measures how many of the top K recommendations are relevant, while **Recall@K** measures how many of the relevant items were successfully captured within the top K results.
Ranking Metrics assess the quality of the ranked list, a critical component of the dynamic context-aware ranker. **Normalized Discounted Cumulative Gain (NDCG)** measures the gain of an item based on its position in the ranked list, rewarding a more relevant item placed higher up. **Mean Reciprocal Rank (MRR)** measures the position of the first relevant item in the ranked list.
Behavioral Metrics provide a holistic view of the user experience. **Diversity** measures the variety of recommendations to ensure the system does not simply reinforce existing preferences, while **Novelty** measures how new or unexpected a recommendation is to the user. The final evaluation framework should balance precision, diversity, and contextual ranking to provide a truly complete and valuable user experience. The following table provides a clear guide to these metrics.

Metric Type	Metric Name	Purpose
Predictive	**Precision@K**	Measures the proportion of

Metric Type	Metric Name	Purpose
		relevant recommendations in the top K results.
Predictive	**Recall@K**	Measures the proportion of all relevant items that were successfully included in the top K recommendations.
Ranking	**Normalized Discounted Cumulative Gain (NDCG)**	Measures the quality of the ranked list by accounting for the position of relevant items.
Ranking	**Mean Reciprocal Rank (MRR)**	Measures the inverse of the rank of the first relevant item in a ranked list of recommendations.
Behavioral	**Diversity**	Measures the dissimilarity between recommended items to ensure a variety of suggestions.
Behavioral	**Novelty**	Measures how new or unexpected a recommendation is to the user, ensuring serendipity.

5. Conclusion: Beyond the Hackathon

The design and development of an AI-based PM internship recommendation engine for a hackathon is a problem that requires a multifaceted and nuanced approach. A simple keyword-matching system is insufficient, as it fails to account for the unique tactical nature of the PM intern role, the intangible qualities of soft skills, and the inherent biases of AI. The most successful solutions will be those that are not only technically advanced but also ethically responsible and user-centric.

The proposed ten concepts provide a robust blueprint for such a system, moving beyond the simple "matchmaker" paradigm to an intelligent, end-to-end platform. Concepts like the **Project-Based Proficiency Analyst** and the **Semantic Similarity Mapper** address the fundamental challenge of finding qualified candidates from non-traditional backgrounds by focusing on the substance of their experience rather than rigid job titles. The **Soft Skills Behavioral Assessor** and the **Gamified Simulation Interviewer** tackle the critical problem of quantifying elusive human qualities.

Furthermore, concepts such as the **Ethical AI Transparency Dashboard** and the **Resume Optimization Coach** directly address user pain points, building trust and providing tangible value to both the candidate and the recruiter. This approach transforms the project from a simple tool into a trusted partner in a candidate's career journey. By focusing on these innovative concepts, a hackathon team can build a compelling and highly impactful product that not only wins the competition but also lays the groundwork for a commercially viable and ethically sound solution for the evolving landscape of Program Management.

Works cited

1. What is the difference between product, program, project management : r/ProductManagement - Reddit, https://www.reddit.com/r/ProductManagement/comments/zpn2d0/what_is_the_difference_between_product_program/ 2. Product Management vs Project/Program Management Perspectives : r/PMCareers - Reddit, https://www.reddit.com/r/PMCareers/comments/18x9r2c/product_management_vs_projectprogram_management/ 3. Program Management Intern Job Description | Velvet Jobs, https://www.velvetjobs.com/job-descriptions/program-management-intern 4. Internship Program Manager Responsibilities | Career Guide - JobzMall, https://www.jobzmall.com/careers/internship-program-manager/responsibilities 5. Career Path for Program Manager - Job Scope and Salaries - Sprintzeal.com, https://www.sprintzeal.com/blog/career-path-for-program-manager 6. www.sprintzeal.com, https://www.sprintzeal.com/blog/career-path-for-program-manager#:~:text=It%20typically%20begins%20with%20experience,for%20growth%20in%20this%20role. 7. AI Recruitment Mistakes: Top Pitfalls and How to Avoid Them - GoCo, https://www.goco.io/blog/common-ai-recruitment-pitfalls-to-avoid 8. How NLP Beats Traditional Resume Screening: A Technical Analysis, https://talentbusinesspartners.com/articles/how-nlp-beats-traditional-resume-screening-a-technical-analysis/ 9. Full Guide to Optimizing Resume Keywords to Pass ATS Screening : r/jobsearchhacks, https://www.reddit.com/r/jobsearchhacks/comments/1j530wc/full_guide_to_optimizing_resume_keywords_to_pass/ 10. Comparison Of Models For Resume-JD Matching: BERT, Gemini, And Llama 3.1 - IOSR Journal, https://www.iosrjournals.org/iosr-jce/papers/Vol27-issue2/Ser-5/A2702050110.pdf 11. 7 machine learning algorithms for recommendation engines - Lumenalta, https://lumenalta.com/insights/7-machine-learning-algorithms-for-recommendation-engines 12. Recommendation systems for recruitment: an overview - HR Cloud, https://www.hrcloud.com/blog/recommendation-systems-for-recruitment-an-overview 13. How to Build a Hybrid Recommender System - Motivus, https://motivus.com/insights/how-to-build-a-hybrid-recommender-system/ 14. 10 metrics to evaluate recommender and ranking systems - Evidently AI, https://www.evidentlyai.com/ranking-metrics/evaluating-recommender-systems 15. A Hybrid Approach for Job Recommendation Systems - ResearchGate, https://www.researchgate.net/publication/385535884_A_Hybrid_Approach_for_Job_Recommendation_Systems 16. AI Tools in Human Resources: Benefits, Risks and How to Stay Compliant, https://www.fredlaw.com/alert-ai-tools-in-human-resources-benefits-risks-and-how-to-stay-compliant 17. How To Measure Soft Skills In Recruitment - CPS HR Consulting, https://www.cpshr.us/blog-article/how-to-effectively-assess-soft-skills-during-recruitment/ 18. Text Mining in Organizational Research - PMC - PubMed Central, https://pmc.ncbi.nlm.nih.gov/articles/PMC5975701/ 19. deepakpadhi986/AI-Resume-Analyzer - GitHub, https://github.com/deepakpadhi986/AI-Resume-Analyzer 20. Text Mining: Analyzing Data & Best Practices | InMoment, https://inmoment.com/blog/text-mining/ 21. AI Resume Screening Revolution: How Smart Algorithms Identify Top Talent 10X Faster, https://www.mokahr.io/myblog/ai-resume-screening-revolution-identify-top-talent-10x-faster/ 22. Job-Resume Compatibility Scoring Using Graph Neural Networks and Large Language Models, https://uwindsor.scholaris.ca/server/api/core/bitstreams/8497dc1f-392b-4d0b-8079-dad82e4c7b5f/content 23. Graph Neural Networks for Candidate-Job Matching: An Inductive Learning Approach,

https://www.semanticscholar.org/paper/Graph-Neural-Networks-for-Candidate-Job-Matching%3A-Frazzetto-Haq/876e90b04d1bfeecad7397e665b9bc41e0de1b89 24. Semantic Analysis and Search - Inda, https://inda.ai/en/semantic-analysis-and-search/ 25. www.ksolves.com, https://www.ksolves.com/blog/artificial-intelligence/6-ways-to-automate-the-resume-shortlisting-process-with-nlp-text-analysis#:~:text=By%20using%20NLP%20to%20extract,skills%2C%20qualifications%2C%20and%20experience. 26. Using AI to Assess Soft Skills: Can Machines Evaluate Emotional Intelligence?, https://www.ezintervuez.com/blog/ai-assess-soft-skills-job-interviews/ 27. Essential Natural Language Processing Skills - Coursera, https://www.coursera.org/articles/natural-language-processing-skills 28. NLP Communication Model- Communicate & Influence Effectively - NLP Limited, https://nlplimited.com/nlp-communication-model-communicate-influence-effectively/ 29. How do you assess leadership ability when hiring? - Streamline Hiring with AI Candidate Screening - Serand, https://serand.co.uk/guide/how-do-you-assess-leadership-ability-when-hiring/ 30. Building a Job Recommender engine | by RAI00007 - Medium, https://medium.com/@manish.rai21bit/building-a-job-recommender-engine-2cb327d0bd40 31. Leadership Skills Assessments: The Secret To Identifying Great Leaders | Vervoe Blog, https://vervoe.com/leadership-skills-assessments-the-secret-to-identifying-great-leaders/ 32. SkillSyncer: Free ATS Resume Scanner, https://skillsyncer.com/ 33. ResyMatch.io | Free ATS Resume Scanner & Job Description Match Tool, https://cultivatedculture.com/resume-scanner/ 34. Score My Resume - Get a Free Resume Score - Resume Checker and Grader - Resume Worded, https://resumeworded.com/score 35. What is a Recommendation System? | Data Science | NVIDIA Glossary, https://www.nvidia.com/en-us/glossary/recommendation-system/