Journal of the Korean Institute of Industrial Engineers
Vol. 52, No. 4, pp. 303-319, August 2026.
ISSN 1225-0988 | EISSN 2234-6457
https://doi.org/10.7232/JKIIE.2026.52.4.303

AI 기반 자동화 블루프린트: SaaS 네트워크 기반 에이전트 툴킷 및 워크플로우 시나리오 설계

김경아[1] · 이학연[2†]

[1]서울과학기술대학교 데이터사이언스학과 / (주)베가스 AA그룹
[2]서울과학기술대학교 산업공학과

Blueprints for AI-Driven Automation: Mining a SaaS Dependency Network for Agent Toolkits and Workflow Scenarios

Kyeonga Kim[1] · Hakyeon Lee[2]

[1]Department of Data Science, Seoul National University of Science and Technology / Adaptive AI Group, Begas
[2]Department of Inudstrial Engineering, Seoul National University of Science and Technology

While artificial intelligence (AI)-driven automation is rapidly advancing, design processes remain largely ad-hoc due to a lack of empirical analysis on real-world software-as-a-service (SaaS) workflows. To address this gap, this study proposes a data-driven framework that transforms SaaS usage patterns into actionable design resources. We analyzed 5,602 automation templates consisting of several SaaS applications from Zapier to construct a SaaS dependency network capturing cross-app connectivity. Based on this topology, the framework yields two core solutions for system configuration. First, for AI agents, the Leiden algorithm identifies compatible tool clusters, defined as “Toolkits,” to optimize tool selection boundaries. Second, for intelligent workflows, a subgraph pattern miner (SPMiner) extracts sequential motifs, which are subsequently translated into executable natural language scenarios using a large language model (LLM). The results demonstrate that empirical usage patterns can be systematically converted into ready-to-use automation templates with minimal engineering effort. This research contributes to the field by shifting the focus of business automation from adoption determinants to proactive engineering design. Practically, the proposed framework provides a scalable methodology for configuring AI agents and workflows, offering strategic insights for organizations seeking to navigate and automate within complex SaaS environments.

Keywords: AI-driven Automation, Business Process Automation, AI Agents, Intelligent Workflows, SaaS Dependency Network

1. Introduction

In the contemporary digital landscape, the proliferating adoption of Software-as-a-Service (SaaS) solutions has become a cornerstone for enhancing operational efficiency and organizational productivity (Gunawan *et al.*, 2025; Hyrynsalmi, 2022). SaaS has emerged as the predominant paradigm for business software delivery, offering unprecedented scalability and economic efficiency. By transitioning

This article was excerpted and adapted from the master's thesis of the first author, Kyeonga Kim. This work was supported by the National Research Foundation of Korea (NRF) grant funded by Korean Government (MSIT) (RS-2025-23324126).
† Corresponding Author : Professor, Hakyeon Lee, Department of Industrial Engineering, Seoul National University of Science and Technology, 232 Gongneung-ro, Nowon-gu, Seoul 01811, Republic of Korea, Tel: +82-2-970-6469, Fax: +82-2-974-284, hylee@seoultech.ac.kr
Received April 14, 2026; Accepted May 29, 2026.

from monolithic on-premises systems to specialized, best-of-breed SaaS applications, organizations can optimize discrete business functions. However, the strategic value of these digital assets is often constrained by functional silos, which impede the seamless flow of data and tasks across the enterprise.

To mitigate these limitations, workflow automation platforms, such as Zapier, n8n, and Microsoft Power Automate, have emerged as critical technological infrastructure, enabling the integration of disparate applications into cohesive end-to-end processes (Chen *et al.*, 2025). Recent advancements in artificial intelligence (AI) have further catalyzed this evolution, shifting the focus from deterministic automation to autonomous AI agents. Unlike traditional automation that relies on rigidly defined, "if-then" procedural constraints, AI agents possess the capacity to autonomously orchestrate complex workflows based on high-level goals and available toolsets (Marchena Sekli and De La Vega, 2025).

Despite this technological leap, a critical knowledge gap persists in both academia and practice. The performance of autonomous agents is inherently contingent upon the quality and relevance of the tools (SaaS) provided to them. Yet, there is a notable scarcity of standardized design protocols or empirical guidelines for selecting optimal toolsets for specific organizational objectives. While extant literature has extensively explored the determinants of SaaS adoption and user acceptance (Oliveira *et al.*, 2019; Rahman and Subriadi, 2022), it offers limited insight into the operational interdependencies and sequential connectivity of these applications within automation ecosystems. Consequently, the design of AI-driven workflows remains largely heuristic and ad-hoc, disconnected from systematic methodologies supported by empirical evidence.

This study seeks to bridge this gap by proposing a data-driven framework for designing AI automation and agent systems. Utilizing real-world integration data from Zapier, we construct a SaaS dependency network to capture and analyze empirical usage patterns. Specifically, this research employs community detection (Leiden algorithm) to derive structured Toolkits for AI agents and motif analysis (subgraph pattern miner, SPMiner) to identify frequent connection patterns for automated workflow recommendations. By transitioning from intuitive configuration to a design paradigm rooted in evidence, this study provides actionable insights for managers and system architects to enhance technological agility in the era of AI-driven business transformation.

2. Theoretical Background

2.1 Automation Within the SaaS Ecosystem

SaaS has redefined the organizational technological landscape by providing scalable, cloud-based applications delivered via subscription models. In the realm of workflow automation, platforms such as Zapier, IFTTT and Make orchestrate cross-application processes through trigger-action primitives and public APIs. These platforms generate high-resolution operational logs, which serve as a critical resource for understanding digital transformation at the process level. Current literature on automation powered by SaaS generally converges into three research streams.

First, a significant body of work focuses on user behavior and business analytics. Oliveira *et al.* (2019) utilized SaaS automation logs to characterize usage patterns and identify drivers of productivity, while others have leveraged such data to predict customer churn (Ge *et al.*, 2017) or analyze decision-making dynamics (Jiao *et al.*, 2020). Second, research has concentrated on the technical evaluation of automation platforms, examining the functional limitations of services like Zapier (Coronado *et al.*, 2015) and establishing typologies for comparative analysis (Abdou *et al.*, 2021; Rahmati *et al.*, 2017). Third, studies have explored the practical implications of automation on operational efficiency, particularly through process mining in IoT (Kim *et al.*, 2022) and robotic process automation (RPA) (Aguirre and Rodriguez, 2017).

Despite these advancements, a notable theoretical gap remains. Existing research largely overlooks the latent structural interdependencies within end-to-end processes. Most studies treat SaaS applications as isolated entities rather than integrated nodes within a complex network, necessitating a more rigorous approach to recover the underlying execution structures hidden within automation logs.

2.2 Transitioning to AI-driven Automation: Agents and Intelligent Workflows

The automation landscape is undergoing a paradigm shift from static, rule-based mechanisms to intelligent, autonomous systems. The integration of large language models (LLMs) is transforming workflows from rigid script execution to dynamic, cognitive decision making (Hosseini and Seilani, 2025; Schwartz *et al.*, 2023). This transition allows organizations to manage non-deterministic tasks that transcend simple trigger-action logic (Birr *et al.*, 2024).

AI-driven automation can be bifurcated into two complementary paradigms: AI agents as autonomous actors and AI automation as intelligent workflows (Li, 2025). The agent-centric paradigm emphasizes autonomous actors capable of orchestrating tasks using a Toolkit, a collection of external APIs that extend the functional reach of the agent (Paranjape *et al.*, 2023). However, most current frameworks rely on manually defined or retrieval-based tool selection (Lumer *et al.*, 2024), failing to reflect the empirical co-usage ecosystems of SaaS applications (Mishra *et al.*, 2024).

Conversely, the workflow-centric paradigm focuses on systems that learn from event logs to recommend or optimize process sequences (Maisenbacher and Weidlich, 2017). While advanced methods like graph neural networks (GNNs) and sequential models have been employed (Xiong *et al.*, 2025), there is a persistent disconnect between granular data analytics and actual system design. Current research rarely utilizes the higher-order structures of SaaS dependency networks to inform the construction of toolkits derived from communities or reusable workflow components (Legallois and Koch, 2020).

This gap indicates the need to move beyond adoption-oriented or performance-oriented analyses of SaaS systems toward a design-oriented view of AI automation. In such a view, workflow data are not only retrospective records of system use but also empirical design resources that reveal which applications tend to work together and in what order they are connected. This perspective is particularly relevant for AI agents and intelligent workflows because their performance depends not only on model capability but also on the configuration of external tools, task sequences, and operational contexts. Therefore, a network-based analysis of SaaS dependencies can provide a structural foundation for translating observed usage patterns into agent Toolkits and reusable workflow components.

2.3 Motif Analysis as a Structural Framework

To bridge the gap between empirical data and AI system design, this study adopts motif analysis. A network motif is defined as a subgraph that appears in an empirical network significantly more frequently than in randomized networks (Milo *et al.*, 2002). Motifs function as the fundamental building blocks of complex systems, much like logic gates in electronic circuits (Alon, 2007).

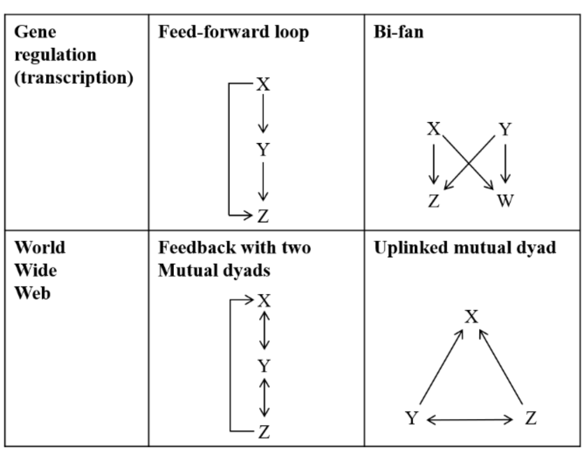


Figure 1. Example of Network Motifs (Milo *et al.*, 2002)

<Figure 1> illustrates fundamental motif structures identified across different complex systems, ranging from biological gene regulation to the World Wide Web. Specific topologies, such as the 'Feed-forward loop', act as functional units for signal processing or noise filtering (Mangan *et al.*, 2003), while 'Bi-fan' and 'Feedback' structures represent distinct coordination patterns. While motif analysis originated in the natural sciences for disease pathway prediction (Agrawal *et al.*, 2018) and molecular substructure identification (Zhang *et al.*, 2021), it has increasingly been applied to social and online networks to reveal organizational fingerprints and community structures (Li *et al.*, 2017; Topirceanu *et al.*, 2016).

Methodologically, the field has evolved from exact counting algorithms like FANMOD (Wernicke, 2006) to deep learning-based approaches such as DotMotif (Matelsky *et al.*, 2021). This study specifically utilized SPMiner (Ying *et al.*, 2024), a GNN-based framework that autonomously discovers frequent motifs without prior knowledge. By identifying these recurring higher-order patterns, we can derive structurally grounded toolkits and workflow recommendations that faithfully mirror real-world SaaS automation ecosystems, providing a meso-scale representation that facilitates evidence-based AI design (Jazayeri and Yang, 2020). In this study, motif analysis is therefore used not as an end in itself, but as a structural bridge between empirical SaaS dependencies and actionable automation design.

3. Network Construction

The entire process of constructing the SaaS dependency network is described in this section, including data collection from Zap data and how that network supports downstream analysis for motif mining and community detection. <Figure 2> depicts the overall framework. The framework is organized into two independent analysis tracks that share a common input network. In the track of AI automation, SPMiner is utilized as a motif mining model in a SaaS dependency network to extract recurrent sequential patterns. These motifs are then translated into workflow scenarios for AI automation templates. Meanwhile, in the track of AI agents, the Leiden algorithm is applied to the input network, and the resulting communities are interpreted as "Toolkits" for an AI agent. Within each track, the output of the earlier component of the framework directly feeds the subsequent component.

3.1 Data Collection and Domain Selection

To implement the proposed framework, this study collected

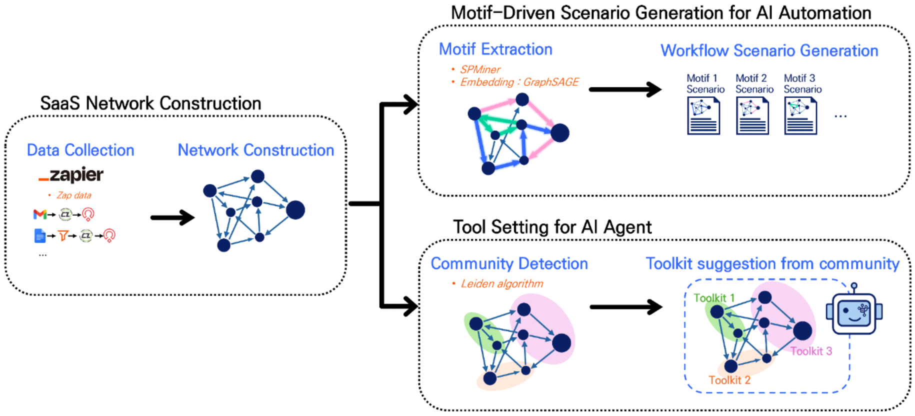


Figure 2. Proposed Framework

empirical workflow data from Zapier, an integration platform that facilitates automation consisting of SaaS applications. The data collection, conducted in April 2025, targeted the HR domain. HR was selected as the empirical domain for three reasons. First, HR processes are highly repetitive and process-oriented, including recruitment, onboarding, training, assessment, scheduling, and employee communication. These characteristics make HR suitable for observing sequential automation patterns. Second, HR work frequently requires cross-functional interactions across multiple SaaS applications, such as applicant tracking systems, learning management platforms, calendars, email services, document tools, and communication platforms. This makes the domain appropriate for analyzing SaaS-to-SaaS dependencies. Third, as organizations undergo digital transformation, HR has increasingly adopted automation to reduce manual coordination and improve responsiveness, making it a relevant setting for studying AI-driven workflow configuration (Sestino and De Mauro, 2022). Therefore, the HR domain provides both practical relevance and sufficient structural complexity for examining how empirical SaaS workflow data can be transformed into AI automation design resources.

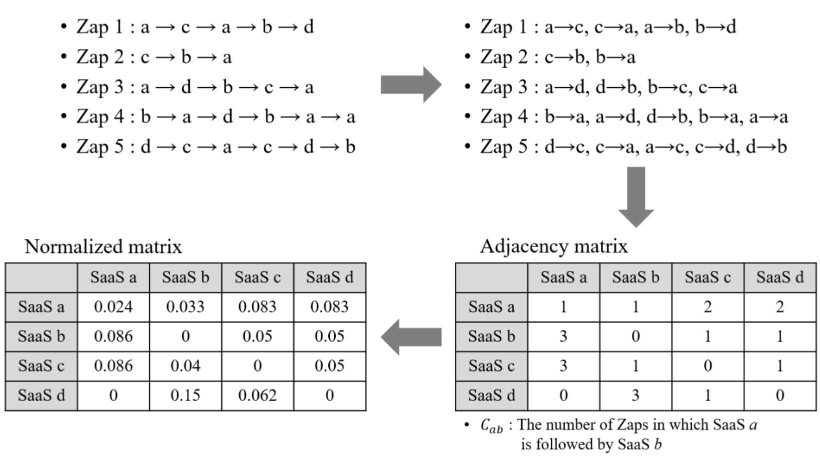
A custom web crawler developed in Python was employed for the collection process. We collected data on all SaaS applications categorized under HR and their associated Zap templates. In this context, a Zap represents a sequential workflow where a Trigger from one application initiates Action in others. A total of 249 SaaS applications were collected. However, applications with no associated Zaps were excluded, resulting in a subset of 190 target SaaS applications. The final dataset comprised 5,602 valid Zap templates. The distribution of these templates by length is summarized in <Table 1>.

Table 1. Distribution of Human Resources Zap lengths in Zapier

Length	2	3	4	5	6	8	10
Frequency	5,073	173	380	7	3	1	1

3.2 SaaS Dependency Network Construction

The network construction process involves projecting the bipartite structure of Zap-SaaS incidence onto a homogeneous di-



Normalized matrix

	SaaS a	SaaS b	SaaS c	SaaS d
SaaS a	0.024	0.033	0.083	0.083
SaaS b	0.086	0	0.05	0.05
SaaS c	0.086	0.04	0	0.05
SaaS d	0	0.15	0.062	0

Adjacency matrix

	SaaS a	SaaS b	SaaS c	SaaS d
SaaS a	1	1	2	2
SaaS b	3	0	1	1
SaaS c	3	1	0	1
SaaS d	0	3	1	0

Figure 3. Network Construction Process

rected graph, $G=(V,E)$, where V denotes the set of SaaS applications and E represents the functional dependencies between them. Unlike traditional process mining that focuses on individual execution instances, this study aggregated immediate adjacencies across thousands of templates to capture pattern-level sequential dependencies. The overall process of SaaS dependency network construction from Zap data is explained in <Figure 3>.

An ordered pair (a,b) is assigned to a directed edge if SaaS a (the source) is immediately followed by SaaS b (the target) within a Zap template. A Zap template with a sequence $[a\to b\to c\to d\to e]$ therefore contributes one count to each of the ordered pairs (a,b), (b,c), (c,d) and (d,e). Let C_{ab} denote the number of Zaps in which a is followed by b (immediate adjacency). To ensure the network reflects substantively significant ties rather than idiosyncratic user preferences, we applied a directed variant of association strength (Eck and Waltman, 2009) for normalization:

$$S_{ab}=\frac{C_{ab}}{W_a^{out}\times W_b^{in}}$$

Here, W_a^{out} is the total number of immediate sequential connections that originate from a to any SaaS, and W_b^{in} is the total number of immediate sequential connections that terminated at b from any SaaS. S_{ab} represents the normalized strength of a directed connection from a to b. This normalization accounts for the differing popularity of applications, allowing us to identify latent design opportunities that are structurally significant within the SaaS ecosystem.

To enhance interpretability and focus on the backbone of the SaaS ecosystem, the normalized matrix was dichotomized using a threshold τ. Following a sensitivity analysis across values 0.001, 0.005 and 0.01 (Borgatti *et al.*, 2024), we selected $\tau=0.005$. This threshold was found to effectively filter out weak, idiosyncratic ties while preserving the essential connectivity required for downstream motif analysis and community detection.

The resulting SaaS dependency network consists of 678 nodes and 938 edges, with a density of 0.002. This sparse yet structured graph, visualized in <Figure 4>, represents the empirical foundation upon which our AI agent Toolkits and workflow recommendations are constructed. By transforming ad-hoc automation logs into this structured representation, we provide the evidence-based methodology necessary for next generation AI-driven system design.

4. Motif-Driven Scenario Generation for AI Automation

4.1 Motif Extraction

SPMiner (Ying *et al.*, 2024) consists of an encoder-decoder pipeline. The encoder learns graph-level and subgraph sensitive representations that preserve relations among substructures. The decoder then infers highly frequent subgraphs over a collection of

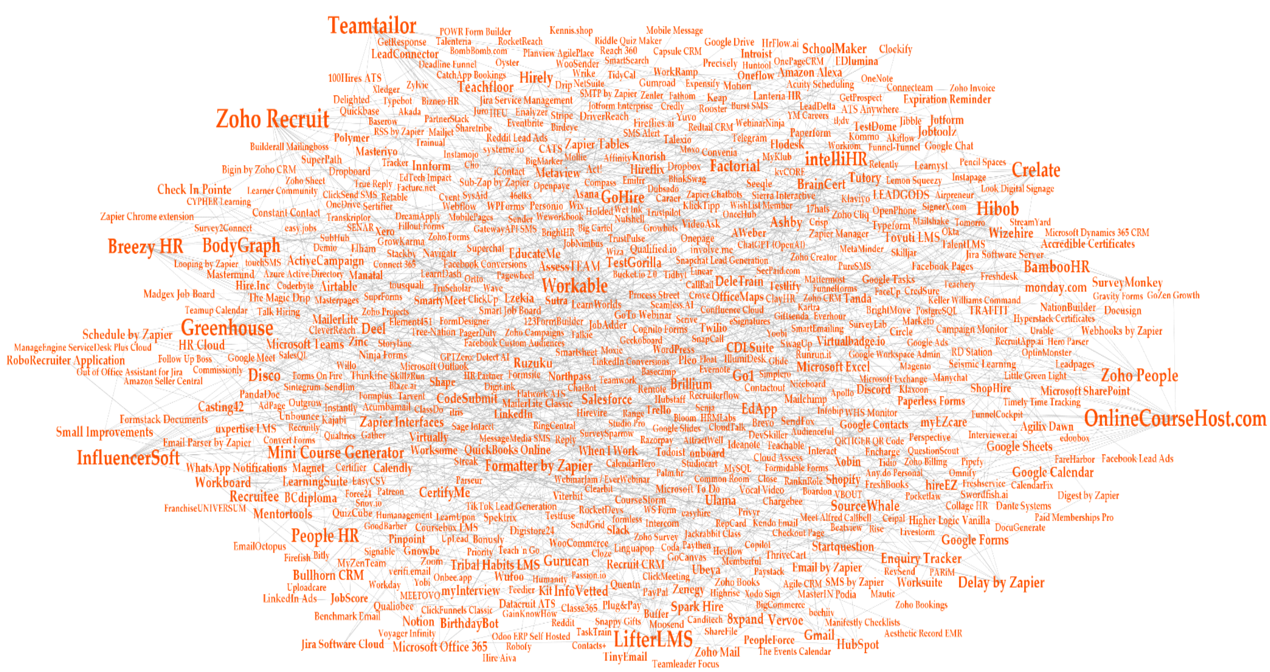


Figure 4. SaaS Dependency Network

graphs, which can be interpreted as the discovery of motifs suitable for downstream analysis. In its original formulation, SPMiner assumes an undirected graph setting. Therefore, additional care is required when the application depends on the order of interaction, as is the case in workflow automations where execution proceeds from an upstream component to a downstream component. In addition, motif extraction in this study operates on an aggregated SaaS dependency network, not on individual Zap instances. As a result, a mined linear motif should not be interpreted as a verbatim reconstruction of a single end-to-end template. Instead, motifs represent reusable sequential design patterns that emerge from repeatedly observed adjacent dependencies and can suggest compositional workflow candidates beyond explicitly authored sequences.

For subgraph matching, the encoder of SPMiner is instantiated with GraphSAGE and trained while explicitly preserving the direction of edges in the Zap workflow graph. Training is conducted on the complete HR SaaS network so that the learned embeddings reflect global structural regularities as well as local ordering. And then, for subgraph mining, the decoder of SPMiner performs motif discovery under a directed assumption using the post cut-off HR SaaS network as input to remove extremely weak ties prior to search. Following the experimental protocol of the original study, the search procedure is implemented with Monte Carlo Tree Search, which effectively explores the subgraph space by balancing the expansion of promising candidates with continued exploration of alternatives.

Formally, SPMiner maps graphs into an order embedding space where the subgraph relationship is preserved geometrically. Let $\phi: G \rightarrow \mathbb{R}^d$ be the embedding function. The order embedding constraint asserts that if a graph A is a subgraph of (denoted as $A \subseteq B$), then their embeddings must satisfy the inequality $\phi(A) \leq \phi(B)$ element-wise. To enforce this property during the training of the encoder, the model minimizes a margin-based ranking loss involving a penalty term $E(A,B)$, which is defined as:

$$E(A,B) = \|\max(0, \phi(A) - \phi(B))\|^2$$

Here, a penalty of zero ($E(A,B) = 0$) indicates that the subgraph order constraint is satisfied, implying the model is confident that A is a subgraph of B.

In the motif discovery phase, the decoder aims to identify a subgraph pattern G of size k that appears frequently in the target graph G_T. Instead of computationally expensive exact counting, SPMiner utilizes a soft frequency objective $m(G)$. This objective aggregates the order embedding violations against the set of all node-anchored neighborhoods N extracted from the target graph:

$$m(G) = \sum_{N \in N} \|\max(0, \phi(G) - \phi(N))\|^2$$

The estimated frequent motif G_{freq} is obtained by finding the subgraph that minimizes this total penalty $m(G)$. This formulation allows the search procedure to effectively navigate the embedding space via a monotonic walk. As established in the methodology, iteratively adding nodes to a candidate subgraph results in a monotonic increase in its embedding coordinates.

To align motif discovery with the objective of recommending executable workflows, the end-to-end pipeline is modified to impose strict structural constraints on the search procedure. The restriction to directed linear paths is an intentional design boundary rather than an assumption that all business processes are linear. The objective of this study is not to reconstruct a complete process model such as a business process model and notation (BPMN) diagram, but to extract sequential workflow fragments that can be directly transformed into AI automation templates. Unlike the original formulation where the walk allows for expansion into any arbitrary connected subgraph, this study introduces a sequential constraint that restricts the valid transition steps in the embedding space. Specifically, the expansion mechanism is constrained to only permit the addition of nodes that extend the current subgraph into a linear sequence, that is, a path graph, thereby explicitly prohibiting branching operations and cycles. This constraint effectively prunes the search space during the walk, ensuring that the optimization of $m(G)$ is performed solely over the set of directed linear paths. Consequently, the pipeline operates on fully directed structures, preserving edge orientation throughout both representation learning in the encoder and the constrained enumeration in the decoder. This design supports interpretability, preserves execution order, and helps keep downstream scenario generation aligned with sequential workflow fragments rather than unsupported branches, conditions, or loops. Even if a complete end-to-end Zap sequence is not present in any individual Zap, a directed path-like motif remains a potential candidate for authoring so long as each adjacent dependency along the path is strongly supported in the aggregated network.

Under this sequential constraint, motifs are defined as path-like subgraphs. No global upper bound is imposed on their length. This approach allows the procedure to capture the short, medium, and long routines encountered in practice. In addition, size-specific searches are conducted on fixed path lengths. These searches serve two purposes. First, they enable comparative analyses across different levels of procedural complexity. And second, they

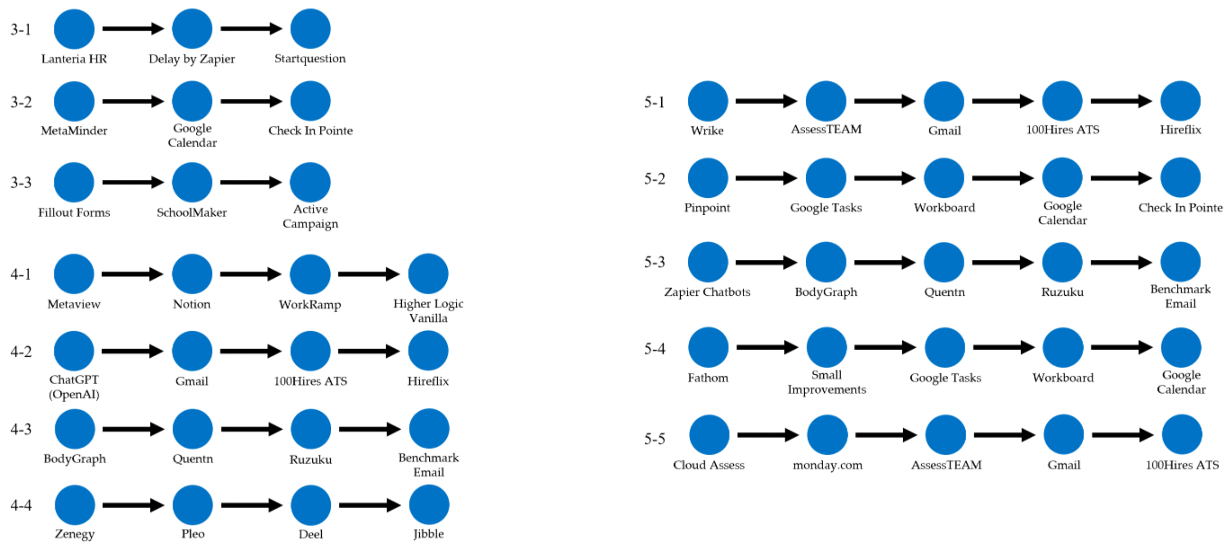


Figure 5. Extracted motifs (length three to five)

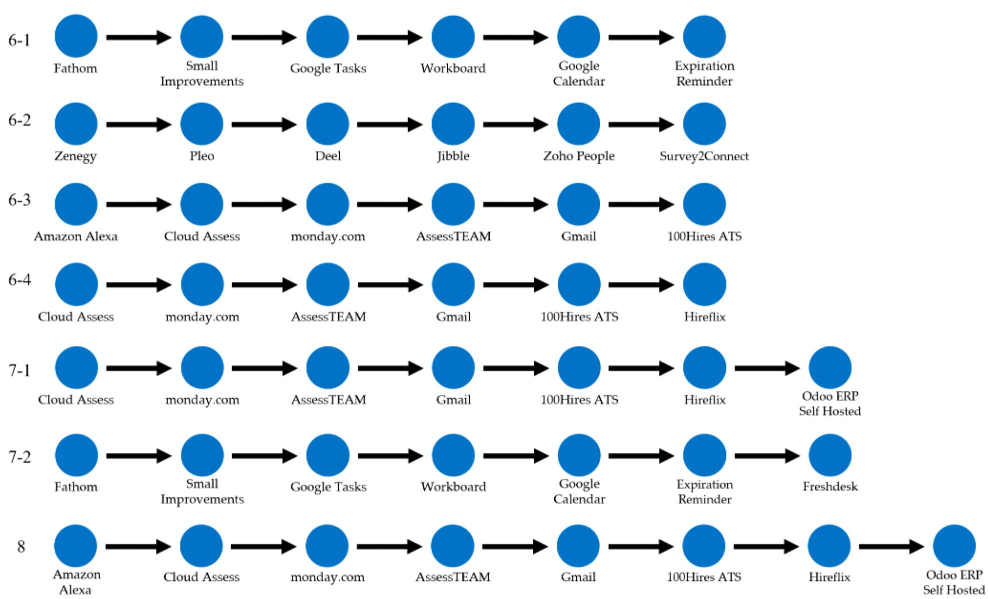


Figure 6. Extracted motifs (length six to eight)

facilitate template selection for recommendations. This combination of directed learning, sequential filtering, and flexible yet size-aware extraction yields motifs that directly mirror how users compose workflow automations. As a result, this method improves the fidelity of subsequent automation recommendations. As illustrated in <Figure 5> and <Figure 6>, the comprehensive set of motifs has been extracted from the SaaS dependency network.

4.2 Workflow Scenario Generation

To translate the motifs extracted from the network into actionable workflows, this research utilizes a scenario generation procedure based on an LLM configured for few-shot learning. Although motifs encode validated SaaS sequences as directed path-like structures, they remain structural artifacts that specify connectivity and execution order but do not fully articulate the operational semantics required for implementation, such as trigger-action intent, data artifacts exchanged across steps, and the business objective that the sequence fulfills in an HR process. This gap is especially consequential for AI automation, which is pre-defined and non-autonomous. Unlike an AI agent that can plan and disambiguate underspecified steps at runtime, a pre-defined workflow must be explicitly specified before execution begins. Accordingly, the LLM is used as a prompt-based generator that instantiates the semantics of each motif while preserving its topology, producing human-readable scenarios that can serve as initial candidates for AI automation template design.

This research uses the GPT-5 model to translate the extracted SaaS motifs into actionable workflow scenarios. The role of the model is limited to bounded structure-to-text generation rather than unrestricted workflow ideation. In other words, GPT-5 is not used to freely invent new automation processes, but to convert network-derived motif structures into natural language scenarios while preserving the sequence, node composition, and domain context of the original motifs.

For each motif, GPT-5 is provided with structured input consisting of the ordered list of SaaS applications, brief functional explanations of each application, and the community information to which the applications belong. The prompt was designed around six constraints. First, the model must preserve the execution order encoded by the directed motif. Second, it must include all SaaS applications in the motif without skipping any node. Third, it must not add SaaS applications or external tools that are not included in the motif. Fourth, it must describe how data or events flow from one SaaS application to the next. Fifth, it must identify a clear trigger and final outcome for the automation. Sixth, it must interpret the motif within the HR workflow automation domain. These constraints were introduced to reduce unsupported generation and to ensure that the generated scenarios remain grounded in the motif structure. The concrete instructions used in the prompt are explained in Appendix A.

The generated scenarios were subsequently reviewed for structural consistency with the motifs from which they were generated, rather than treated as independently validated business process models. This step is not a quantitative conformance evaluation in the sense of process mining, where a process model is compared against an external event log. Instead, it is a prompt-grounded review that verifies whether the constraints specified in the prompt are reflected in the LLM output. The review drew on the general logic of conformance and process quality in process mining and process modeling research, which emphasizes preserving relevant activity order, avoiding unsupported behavior, and maintaining a coherent process description (Rozinat and van der Aalst, 2008; Buijs *et al.*, 2014; Mendling *et al.*, 2010). In this study, the check focused on whether each scenario preserved the directed order of the motif, included all SaaS applications in the motif, avoided adding unsupported SaaS applications or external tools, described a plausible data or event flow, and remained consistent with the HR workflow context. Therefore, the purpose of this step was not to claim field-level validation of the generated workflow, but to ensure that the LLM output remained a structurally grounded interpretation of the network-derived motif.

<Figure 7> presents one of the example motifs (5-5) and its corresponding workflow scenario. The goal of this automation is connecting assessment results to improvement tasks, performance tracking, and recruitment opportunities in a single HR workflow. The scenario is as follows:

When an evaluation is completed in "Cloud Assess" (online assessment and training solution for corporates, training providers, and government entities), scores and competency recommendations for each employee are generated and passed into the automation. These results flow to "monday.com" (work management platform for projects and task coordination), where concrete improvement tasks are automatically created with clear owners, deadlines, and priorities aligned to the identified skill gaps. As employees complete these tasks, "AssessTEAM" (cloud-based performance and productivity analysis software) logs outcomes, updates performance profiles, and, when appropriate, triggers new review cycles. "Gmail" (email service) then delivers targeted follow-up messages to employees and managers, such as sharing assessment results, notifying them of new development plans, or recognizing progress. For individuals identified as high-potential talent, "100Hires ATS" (recruitment software with applicant tracking) creates or updates candidate records in a talent pool, so they can be considered for internal promotions or future openings. This scenario ties assessment, development actions, performance monitoring, and talent pipeline building into one coherent, sequential workflow.

Reviewing the scenario in <Figure 7> against the criteria above indicates that the directed order of the motif is preserved, all five SaaS applications are included, no external tool is introduced, the data flow from assessment completion (trigger) to talent-pool registration (outcome) is articulated, and the description remains consistent with the HR assessment-to-recruitment context. This example illustrates how a network-derived motif can be translated

Figure 7. Extracted motif 5-5

into a human-readable workflow recommendation while preserving the sequence of SaaS applications. Rather than validating the workflow as an operationally deployed process, the scenario demonstrates how motif-based logic can provide an empirically grounded starting point for further specification, evaluation, and implementation.

5. Tool Setting for AI Agent

5.1 Community Detection

Community detection provides a principled way to reduce the complexity of large co-usage networks, while preserving their meso-scale organization (Lancichinetti and Fortunato, 2009). In contrast to centroid-based methods such as k-means, commonly used network approaches do not require predefining the number of clusters and can better respect inherent topology of the graph (Blondel *et al.*, 2008). Moreover, community methods can expose multi-level structure when combined with hierarchical or aggregation strategies, thereby supporting both overview and drill-down analysis (Bonald *et al.*, 2018; Lancichinetti and Fortunato, 2009). These properties improve interpretability, scalability, and robustness for empirical studies of complex networks.

This study employs the Leiden algorithm (Traag *et al.*, 2019) to identify communities in the HR SaaS network. This method addresses well known limitations of the conventional Louvain algorithm (Blondel *et al.*, 2008) and is suitable for large-scale networks. Leiden iteratively relocates nodes to neighboring communities to improve the chosen quality function and modularity. It refines each community into connected subparts to guarantee the internal connectivity of all communities and then aggregates the updated communities into super-nodes. By repeating this refinement aggregation cycle, Leiden typically yields more suitable partitions and better connected communities.

The Leiden algorithm is particularly appropriate for the objective of this study because the resulting communities are used not merely as descriptive clusters but as candidate Toolkits for AI agents. A Toolkit should consist of applications that are not only similar in function but also structurally connected within actual workflow configurations. The connectivity guarantee of Leiden therefore supports the interpretation of each community as a coherent set of tools that can be jointly considered by an agent. In addition, Leiden does not require the number of communities to

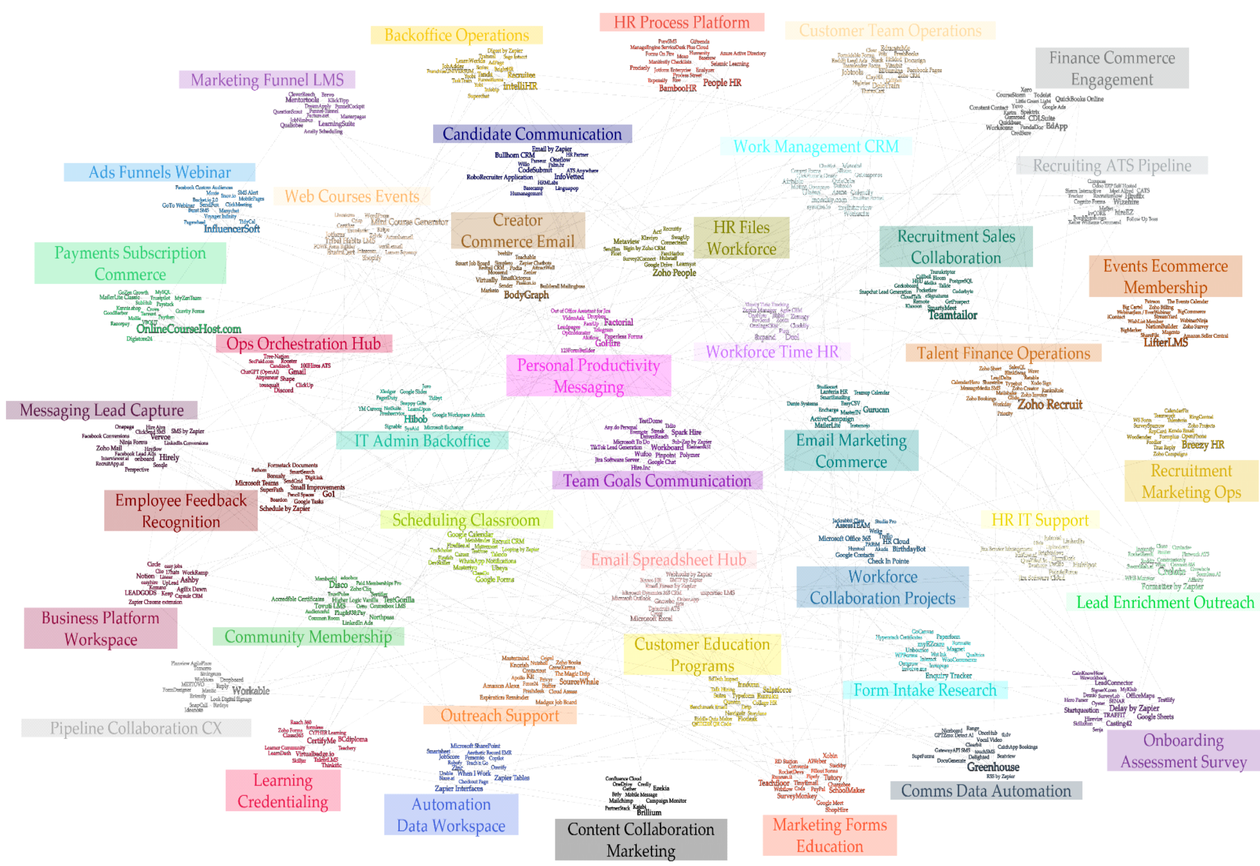


Figure 8. Human Resources SaaS Communities

be specified in advance, which is useful when the natural grouping of SaaS applications is unknown. The resolution parameter also enables the analysis to control the granularity of communities so that the resulting Toolkits are neither too broad for practical agent planning nor too narrow to support meaningful workflow composition.

The community detection procedure in this study proceeds as follows. First, partition the graph into very small initial communities. The algorithm then merges communities with high similarity, where similarity is defined as the inter-community connection strength, measured as the sum of edge weights between two communities. After obtaining an initial partition, search over the resolution parameter and the random seed to satisfy community size constraints, targeting communities with a minimum of 10 nodes and a maximum of 20 nodes. The resulting medium sized communities capture groups of SaaS applications that organizations frequently use together when implementing HR workflow automations. Each community can be interpreted as a candidate Toolkit for an AI agent, that is, a coherent set of tools that can support related HR tasks.

Community names, descriptions and all the SaaS applications of each community are listed in Appendix B. The names and the descriptions of each community were written in consideration of the information of the SaaS applications belonging to that community and their purpose in automating HR tasks. <Figure 8> visualizes their placement in the SaaS dependency network. These labeled communities constitute the Toolkit from which the AI agent can later select appropriate tools when constructing HR automation workflows.

5.2 Toolkit Suggestion from Community

The communities derived from the SaaS network constitute an agent-facing Toolkit for composing business automation in the HR domain. In this study, Toolkit is defined as a context specific repository of SaaS applications that share a common usage purpose and operational context within HR workflows. For an AI agent to construct valid HR automations, it must select tools that are not only functionally capable but also contextually appropriate. Without a predefined structure, an agent faces a global search problem across the entire SaaS catalog, leading to combinatorial complexity and the risk of contextual hallucination, such as combining incompatible tools simply because they share a generic API feature. By conceptualizing the detected communities as Toolkits, this study furnishes the AI agent with a detailed repository of resources. When a specific HR task is requested, the agent first identifies the relevant Toolkit, constraining the subsequent planning process to a safe, empirically validated set of tools known to work together for that specific domain.

The following scenario illustrates how an AI agent can leverage specific communities as Toolkits, thereby demonstrating the practical utility of this approach. First, consider a recruitment scenario where a user requests a workflow to send updated emails to candidates who passed an interview and automatically schedule a follow-up call. The agent analyzes the semantic intent and maps the request to Community 19 (Candidate Communication). Instead of searching the entire catalog, the agent activates this specific Toolkit and constructs a pipeline using only its member applications. For instance, it configures a trigger based on candidate status changes in “ATS Anywhere”, chains this to a personal-

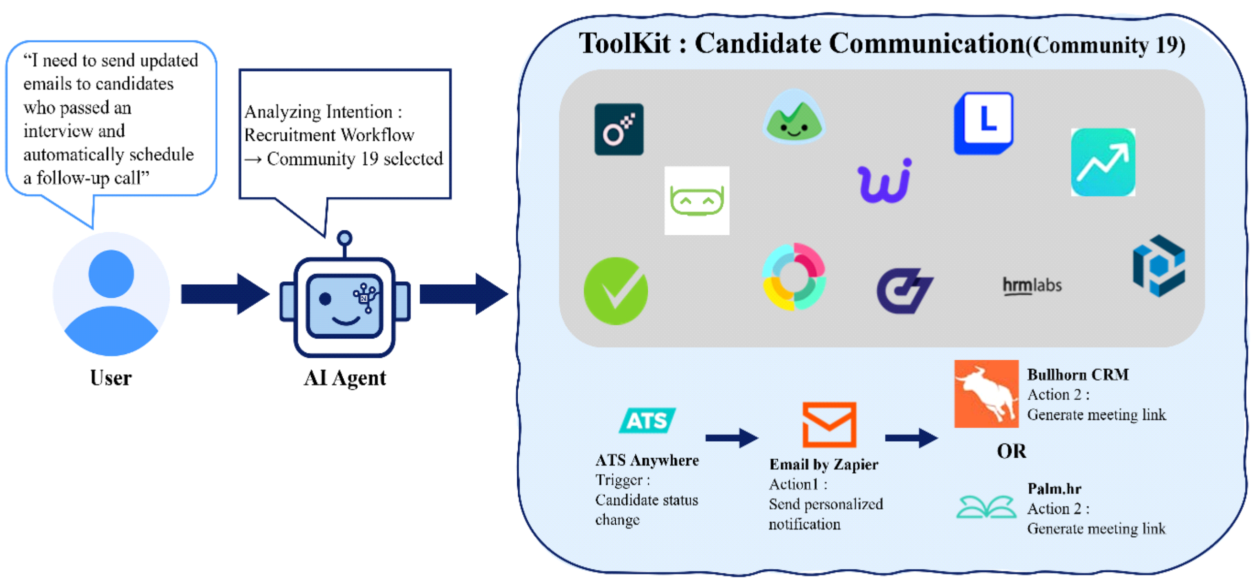


Figure 9. Overview of how AI Agent uses the Candidate Communication (19) as a Toolkit

ized notification via “Email by Zapier” and generates a meeting link using the scheduling features within “Bullhorn CRM” or “Palm.hr”. This constraint ensures the agent builds a coherent recruiting workflow without accidentally invoking internal messaging tools or unrelated marketing platforms. A detailed explanation of this case is provided in <Figure 9>.

6. Conclusions

This study turns SaaS workflow data into design resources for AI-driven automation by deriving community-based Toolkits for AI agents and motif-based workflow candidates for AI automation from a directed SaaS dependency network. Theoretically, it shifts the focus of SaaS research from adoption and usage toward design-oriented automation intelligence by conceptualizing SaaS applications as networked design components rather than independent systems. In this view, SaaS workflow data are not only retrospective records of system use but also empirical resources that can inform how AI agents and workflows should be configured.

Practically, the framework can help managers and system architects design AI-driven automation in complex SaaS environments. Community-based Toolkits reduce the tool selection space for AI agents by grouping structurally coherent applications, while motif-based workflow candidates provide initial automation designs grounded in observed SaaS-to-SaaS dependencies. Although this study focuses on HR workflows, the analytical procedure can be applied to other domains where SaaS-based workflow templates or automation logs are available, such as marketing, DevOps, and customer support. However, the resulting Toolkits and motifs should be interpreted in relation to each domain’s specific workflow objects, events, and operational goals. Therefore, the generalizability of this study lies primarily in the framework and analytical logic, while the substantive interpretation of the results remains domain-dependent.

This study has two main limitations. First, the motif mining procedure intentionally restricts the search space to directed linear paths. This choice supports the extraction of sequential workflow fragments that can be transformed into AI automation templates, but it does not imply that all business processes are linear. The objective of this study is not to reconstruct a complete process model such as a BPMN diagram, but to identify reusable straight-through automation candidates grounded in empirical SaaS-to-SaaS dependencies. Accordingly, the extracted motifs should be interpreted as sequential workflow fragments rather than complete representations of business processes. Real-world processes may include conditional branches, parallel tasks, exception handling, approval gates, human-in-the-loop decisions, and iterative loops, which are outside the current scope. Future research can extend the framework through branch-aware motif mining, loop-aware motif mining, directed acyclic subgraph discovery, and BPMN-compatible process representations.

Second, the SaaS dependency network was constructed from Zapier HR templates collected at a specific point in time, so it captures a static snapshot rather than the temporal evolution of the SaaS automation ecosystem. This is a meaningful limitation because cloud-based service ecosystems continuously change as new applications enter the market, existing applications modify their application programming interfaces (APIs), platform policies evolve, and organizations revise their automation practices. Such changes may alter both the community structures used as AI agent Toolkits and the sequential motifs used as workflow candidates. Future research can address this limitation through time-sliced SaaS dependency networks, dynamic community detection, temporal motif mining, link prediction, and concept drift detection. These extensions would make it possible to examine how Toolkits and workflow motifs emerge, persist, merge, split, or disappear over time.

References

Abdou, M., Ezz, A. M., and Farag, I. (2021), Digital automation platforms comparative study, *2021 4th International Conference on Information and Computer Technologies (ICICT)*, 279-286. https://ieeexplore.ieee.org/abstract/document/9476914/

Agrawal, M., Zitnik, M., and Leskovec, J. (2018), Large-scale analysis of disease pathways in the human interactome, *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, **23**, 111. https://pmc.ncbi.nlm.nih.gov/articles/PMC5731453/

Aguirre, S. and Rodriguez, A. (2017), Automation of a Business Process Using Robotic Process Automation (RPA): A Case Study. In J. C. Figueroa-García, E. R. López-Santana, J. L. Villa-Ramírez, & R. Ferro-Escobar (Eds.), *Applied Computer Sciences in Engineering* (Vol. 742, pp. 65-71). Springer International Publishing. https://doi.org/10.1007/978-3-319-66963-2_7

Alon, U. (2007), Network motifs: Theory and experimental approaches, *Nature Reviews Genetics*, **8**(6), 450-461.

Birr, T., Pohl, C., Younes, A., and Asfour, T. (2024, July 15), AutoGPT+P: Affordance-based Task Planning with Large Language Models, *Robotics: Science and Systems XX*. https://doi.org/10.15607/RSS.2024.XX.112

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008), Fast unfolding of communities in large networks, *Journal of Statistical Mechanics: Theory and Experiment*, **2008**(10), P10008. https://doi.org/10.1088/1742-5468/2008/10/P10008

Bonald, T., Charpentier, B., Galland, A., and Hollocou, A. (2018), *Hierarchical Graph Clustering using Node Pair Sampling* (arXiv:1806.01664). arXiv. https://doi.org/10.48550/arXiv.1806.01664

Borgatti, S. P., Agneessens, F., Johnson, J. C., and Everett, M. G.

(2024), *Analyzing social networks*, SAGE Publications Ltd. https://doi.org/10.4135/9781036236021

Buijs, J. C. A. M., Van Dongen, B. F., and Van Der Aalst, W. M. P. (2014), Quality Dimensions in Process Discovery: The Importance of Fitness, Precision, Generalization and Simplicity, *International Journal of Cooperative Information Systems*, **23**(01), 1440001. https://doi.org/10.1142/S0218843014400012

Chen, M., Fang, S., Wang, G., and Yin, Y. (2025), The impact of digital transformation on employment and workforce composition: Evidence from China's evolving digital economy, *Technology Analysis & Strategic Management*, 1-16. https://doi.org/10.1080/09537325.2025.2547816

Coronado, M., Iglesias, C. A., and Serrano, E. (2015), Modelling rules for automating the Evented WEb by semantic technologies, *Expert Systems with Applications*, **42**(21), 7979-7990. https://doi.org/10.1016/j.eswa.2015.06.031

Eck, N. J. V. and Waltman, L. (2009), How to normalize cooccurrence data? An analysis of some well-known similarity measures, *Journal of the American Society for Information Science and Technology*, **60**(8), 1635-1651. https://doi.org/10.1002/asi.21075

Ge, Y., He, S., Xiong, J., and Brown, D. E. (2017), Customer churn analysis for a software-as-a-service company, *2017 Systems and Information Engineering Design Symposium (SIEDS)*, 106-111. https://doi.org/10.1109/SIEDS.2017.7937698

Gunawan, A., Wijaya, M. I., Andika, N., and Lius, A. (2025), Empirical Study of Robotic Process Automation: Implementation and Evaluation, *Human Behavior and Emerging Technologies*, **2025**(1), 2876164. https://doi.org/10.1155/hbe2/2876164

Hosseini, S. and Seilani, H. (2025), The role of agentic ai in shaping a smart future: A systematic review, *Array*, **26**, 100399. https://doi.org/10.1016/j.array.2025.100399

Hyrynsalmi, S. M. (2022), The state-of-the-art of the integration platforms as a service research, *Proceedings of the 5th International Workshop on Software-Intensive Business: Towards Sustainable Software Business*, 17-22. https://doi.org/10.1145/3524614.3528634

Jazayeri, A. and Yang, C. C. (2020), Motif discovery algorithms in static and temporal networks: A survey, *Journal of Complex Networks*, **8**(4), cnaa031.

Jiao, W., Chen, H., and Yuan, Y. (2020), Understanding users' dynamic behavior in a free trial of IT services: A three-stage model, *Information & Management*, **57**(6), 103238. https://doi.org/10.1016/j.im.2019.103238

Kim, S., Suh, Y., and Lee, H. (2022), What IoT devices and applications should be connected? Predicting user behaviors of IoT services with node2vec embedding, *Information Processing & Management*, **59**(2), 102869. https://doi.org/10.1016/j.ipm.2022.102869

Lancichinetti, A. and Fortunato, S. (2009), Community detection algorithms: A comparative analysis, *Physical Review E*, **80**(5), 056117. https://doi.org/10.1103/PhysRevE.80.056117

Legallois, D. and Koch, S. (2020), The Notion of Motif Where Disciplines Intersect: Folkloristics, Narrativity, Bioinformatics, Automatic Text Processing and Linguistics. In I. Novakova & D. Siepmann (Eds.), *Phraseology and Style in Subgenres of the Novel* (pp. 17-46). Springer International Publishing. https://doi.org/10.1007/978-3-030-23744-8_2

Li, P., Dau, H., Puleo, G., and Milenkovic, O. (2017), Motif clustering and overlapping clustering for social network analysis, *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*, 1-9. https://doi.org/10.1109/INFOCOM.2017.8056956

Li, X. (2025), A review of prominent paradigms for llm-based agents: Tool use, planning (including rag), and feedback learning, *Proceedings of the 31st International Conference on Computational Linguistics*, 9760-9779. https://aclanthology.org/2025.coling-main.652/

Lumer, E., Subbiah, V. K., Burke, J. A., Basavaraju, P. H., and Huber, A. (2024), *Toolshed: Scale Tool-Equipped Agents with Advanced RAG-Tool Fusion and Tool Knowledge Bases* (arXiv:2410.14594). arXiv. https://doi.org/10.48550/arXiv.2410.14594

Maisenbacher, M. and Weidlich, M. (2017), Handling concept drift in predictive process monitoring, *2017 IEEE International Conference on Services Computing (SCC)*, 1-8. https://doi.org/10.1109/SCC.2017.10

Mangan, S., Zaslaver, A., and Alon, U. (2003), The coherent feedforward loop serves as a sign-sensitive delay element in transcription networks, *Journal of Molecular Biology*, **334**(2), 197-204. https://doi.org/10.1016/j.jmb.2003.09.049

Marchena Sekli, G. and De La Vega, I. (2025), Addressing challenges and constructing a blueprint for effective generative AI integration in business operations, *Technology Analysis & Strategic Management*, 1-24. https://doi.org/10.1080/09537325.2025.2577709

Matelsky, J. K., Reilly, E. P., Johnson, E. C., Stiso, J., Bassett, D. S., Wester, B. A., and Gray-Roncal, W. (2021), DotMotif: An open-source tool for connectome subgraph isomorphism search and graph queries, *Scientific Reports*, **11**(1), 13045. https://doi.org/10.1038/s41598-021-91025-5

Mendling, J., Reijers, H. A., and van der Aalst, W. M. (2010), Seven process modeling guidelines (7PMG), *Information and Software Technology*, **52**(2), 127-136. https://doi.org/10.1016/j.infsof.2009.08.004

Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., and Alon, U. (2002), Network Motifs: Simple Building Blocks of Complex Networks, *Science*, **298**(5594), 824-827. https://doi.org/10.1126/science.298.5594.824

Mishra, S., Shinde, M., Yadav, A., Ayyub, B., and Rao, A. (2024), *An AI-Driven Data Mesh Architecture Enhancing Decision-Making in Infrastructure Construction and Public Procurement* (arXiv:2412.00224). arXiv. https://doi.org/10.48550/arXiv.2412.00224

Oliveira, T., Martins, R., Sarker, S., Thomas, M., and Popovič, A. (2019), Understanding SaaS adoption: The moderating impact of the environment context, *International Journal of Information Management*, **49**, 1-12. https://doi.org/10.1016/j.ijinfomgt.2019.02.009

Paranjape, B., Lundberg, S., Singh, S., Hajishirzi, H., Zettlemoyer, L., and Ribeiro, M. T. (2023), *ART: Automatic multi-step reasoning and tool-use for large language models* (arXiv:2303.09014). arXiv. https://doi.org/10.48550/arXiv.2303.09014

Rahman, A. and Subriadi, A. P. (2022), Software as a service (SaaS) adoption factors: Individual and organizational perspective, *2022 2nd International Conference on Information Technology and Education (ICIT&E)*, 31-36. https://doi.org/10.1109/ICITE54466.2022.9759891

Rahmati, A., Fernandes, E., Jung, J., and Prakash, A. (2017), IFTTT vs. Zapier: A comparative study of trigger-action programming frameworks, *arXiv Preprint arXiv:1709.02788*. https://doi.org/10.48550/arXiv.1709.02788

Rozinat, A. and Van Der Aalst, W. M. P. (2008), Conformance checking of processes based on monitoring real behavior, *Information Systems*, **33**(1), 64-95. https://doi.org/10.1016/j.is.2007.07.001

Schwartz, S., Yaeli, A., and Shlomov, S. (2023), *Enhancing Trust in LLM-Based AI Automation Agents: New Considerations and Future Challenges* (arXiv:2308.05391). arXiv. https://doi.org/10.48550/arXiv.2308.05391

Sestino, A. and De Mauro, A. (2022), Leveraging Artificial Intelligence in Business: Implications, Applications and Methods, *Technology Analysis & Strategic Management*, **34**(1), 16-29. https://doi.org/10.

1080/09537325.2021.1883583

Topirceanu, A., Duma, A., and Udrescu, M. (2016), Uncovering the fingerprint of online social networks using a network motif based approach, *Computer Communications*, **73**, 167-175. https://doi.org/10.1016/j.comcom.2015.07.002

Traag, V. A., Waltman, L., and Van Eck, N. J. (2019), From Louvain to Leiden: Guaranteeing well-connected communities, *Scientific Reports*, **9**(1), 1-12. https://doi.org/10.1038/s41598-019-41695-z

Wernicke, S. (2006), Efficient detection of network motifs, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **3**(4), 347-359. https://doi.org/10.1109/TCBB.2006.51

Xiong, Y., Wang, J., Li, B., Zhu, Y., and Zhao, Y. (2025), *Self-Organizing Agent Network for LLM-based Workflow Automation* (arXiv:2508.13732). arXiv. https://doi.org/10.48550/arXiv.2508.13732

Ying, R., Fu, T., Wang, A., You, J., Wang, Y., and Leskovec, J. (2024), *Representation Learning for Frequent Subgraph Mining* (arXiv:2402.14367). arXiv. https://doi.org/10.48550/arXiv.2402.14367

Zhang, Z., Liu, Q., Wang, H., Lu, C., and Lee, C.-K. (2021), Motif-based graph self-supervised learning for molecular property prediction, *Advances in Neural Information Processing Systems*, **34**, 15870-15882.

Appendix

Appendix A. Prompt for generating workflow scenario from extracted motifs

Prompt: Motif-to-Scenario Generation for HR SaaS workflows

You are given motifs, extracted from workflow automation data that use SaaS applications in the Human Resources (HR) category. Each motif is a directed sequence of SaaS applications that are frequently used together in HR workflow automation. Your task is to convert each motif into a clear business automation scenario.

For each motif, you will receive the ordered list of SaaS applications, a short functional description for each application, and the community information (name and description) to which the application belongs. Using this information, write what kind of HR workflow automation the motif represents. You must preserve the execution order of SaaS applications and describe how data or events flow from one application to the next. Do not add SaaS applications, external systems, or tools that are not included in the given motif. Do not skip any SaaS application in the motif. If a specific operational detail is not provided, infer only a general HR workflow role that is consistent with the SaaS description and community information.

For every motif, produce the following outputs:

1. <u>Goal of the automation</u>: a short description of the main business purpose of the workflow.
2. <u>Scenario paragraph</u>: one coherent paragraph that describes the workflow from the first SaaS application to the last. Each SaaS name must appear with a brief functional summary at its first mention.

Below is the one of examples of workflow scenarios from the Zapier website.

Example. Create new Zoho Recruit candidates from CandidateZip parsed Dropbox resume files

1. Components: Dropbox + Zoho Recruit + CandidateZip Resume/Job Parser
2. Description: Storing all of your received resume files in a shared folder is great for keeping track of candidates. Although, it is likely that you will want to use an applicant management tool to stay up-to-date on the job listing progress. Once activated, Zapier will automatically parse resumes added to a Dropbox folder via CandidateZip and add the information to Zoho Recruit.

Appendix B. Name, description and application list of the communities

No.	Community Name	Primary role	SaaS
1	Learning Credentialing	Training delivery and digital credential issuance	BCdiploma, CertifyMe, Classe365, CYPHER Learning, formless, LearnDash, Learner Community, Reach 360, Skilljar, TalentLMS, Teachery, Thinkific, Virtualbadge.io, Zoho Forms
2	Community Membership	Learning communities and membership management	Accredible Certificates, Audienceful, Common Room, Coursebox LMS, Disco, edoobox, Higher Logic Vanilla, LinkedIn Ads, Memberful, Northpass, Ortto, Paid Memberships Pro, Plug&Pay, Sertifier, TestGorilla, Tovuti LMS, TrustPulse, Zoho Cliq
3	Customer Education Programs	Credential-based learning and career pathway management	Benchmark Email, Collage HR, Drip, EdTech Impact, Flodesk, Innform, Navigatr, QRTIGER QR Code, Quentn, Riddle Quiz Maker, Ruzuku, Salesforce, Storylane, Sutra, Talk Hiring, Typeform
4	Automation Data Workspace	Shared data workspaces for HR automation	Aesthetic Record EMR, Blaze.ai, Checkout Page, Copilot, JobScore, Microsoft SharePoint, Omnify, Personio, Robofy, Smartsheet, Teach 'n Go, Urable, When I Work, Zapier Interfaces, Zapier Tables, Zinc
5	Outreach Support	Candidate sourcing and outbound recruiting support	Amazon Alexa, Apollo, Buffer, Ceipal, Cloud Assess, Contactout, Expiration Reminder, Force24, Freshdesk, GrowKarma, Kit, Knorish, Madgex Job Board, Mastermind, Nutshell, Privyr, SourceWhale, The Magic Drip, Zoho Books
6	Team Goals Communication	Task coordination and HR goal communication	Any.do Personal, DriverReach, Element451, Evernote, Google Chat, Hire.Inc, Jira Software Server, Microsoft To Do, Pinpoint, Polymer, Spark Hire, Streak, Sub-Zap by Zapier, TestDome, Tidio, TikTok Lead Generation, Workboard, Wufoo

No.	Community Name	Primary role	SaaS
7	Form Intake Research	Form and survey-based HR data intake	Enquiry Tracker, Formsite, GoCanvas, Hyperstack Certificates, Instapage, Interact, involve.me, Magnet, myEZcare, Outgrow, Paperform, Qualtrics, Unbounce, Wet Ink, WooCommerce, WPForms
8	Personal Productivity Messaging	Personal task and messaging support for HR work	123FormBuilder, Akiflow, Dropbox, FaceUp, Factorial, GoHire, Leadpages, OptinMonster, Out of Office Assistant for Jira, Paperless Forms, Telegram, VideoAsk
9	Scheduling Classroom	Scheduling, classroom sessions, and assessments	Caraer, ClassDo, DevSkiller, Firefish, Fireflies.ai, Google Calendar, Google Forms, Looping by Zapier, Masteriyo, Mattermost, MetaMinder, Recruit CRM, Talexio, Testfuse, TruScholar, Ubeya, WhatsApp Notifications
10	Email Spreadsheet Hub	Email parsing and spreadsheet-based HR data management	Bizneo HR, Cvent, Datacruit ATS, Email Parser by Zapier, Gnowbe, itris, Microsoft Dynamics 365 CRM, Microsoft Excel, Microsoft Outlook, Onbee.app, SMTP by Zapier, uxpertise LMS, Webhooks by Zapier
11	Email Marketing Commerce	HR-related email campaigns and communication	ActiveCampaign, Dante Systems, EasyCSV, Encharge, Gurucan, Instamojo, Lanteria HR, MailerLite, MasterIN, SmartEmailing, Studiocart, Teamup Calendar
12	Workforce Time HR	Time tracking, payroll, and workforce administration	8xpand, Agile CRM, Clockify, Deel, Jibble, OneNote, OnePageCRM, Pleo, RevSend, Timely Time Tracking, Zapier Manager, Zenegy, Zoom
13	Creator Commerce Email	Learning programs with commerce and email operations	AttractWell, beehiiv, BodyGraph, Builderall Mailingboss, EmailOctopus, Marketo, Moosend, Passion.io, Podia, Redtail CRM, Sender, Simplero, Smart Job Board, Teachable, Virtually, Zapier Chatbots, Zenler
14	HR IT Support	HR-triggered IT support and account provisioning	BrightMove, Everhour, HrFlow.ai, HubSpot, IllumiDesk, Introist, Jira Service Management, Jira Software Cloud, LinkedIn, Okta, PeopleForce, Qualified.io, Twilio, Uploadcare
15	Employee Feedback Recognition	Employee feedback, recognition, and engagement	Boardon, Bonusly, Digit.ink, Fathom, Formstack Documents, Go1, Google Tasks, Microsoft Teams, Pencil Spaces, Schedule by Zapier, SendGrid, Small Improvements, SmartSearch, SuperPath
16	Lead Enrichment Outreach	Candidate enrichment and outbound outreach	Affinity, Cloze, Commissionly, Connect 365, Contacts+, Crelate, Emitrr, Flatwork ATS, Formatter by Zapier, Growbots, Instantly, Reddit, RocketReach, Seamless.AI, Swordfish.ai, WHS Monitor, Wiza
17	HR Files Workforce	Employee files, scheduling, and workforce resources	Act!, Bigin by Zoho CRM, Connecteam, FareHarbor, Float, Google Drive, Hubstaff, Klaviyo, Learnyst, Metaview, Recruitly, SendJim, Survey2Connect, SwagUp, Zoho People
18	Web Courses Events	HR-related courses, events, and webinars	Acumbamail, BrainCert, Certifier, Crisp, Eventbrite, Intercom, Jotform, Lemon Squeezy, Livestorm, Mini Course Generator, POWR Form Builder, Shopify, Stripe, Tribal Habits LMS, verifi.email, WordPress, Zylvie
19	Candidate Communication	End-to-end candidate communication	ATS Anywhere, Basecamp, Bullhorn CRM, CodeSubmit, Email by Zapier, HR Partner, HRMLabs, Humanagement, InfoVetted, Linguapop, Oneflow, Palm.hr, Parseur, RoboRecruiter Application, Willo
20	Pipeline Collaboration CX	Collaborative recruitment pipeline management	Birdeye, Dropboard, FormDesigner, Ideanote, Look Digital Signage, Mautic, MEETOVO, Planview AgilePlace, Reply, Retently, Sintegrum, SnapCall, Tomorro, Workable, Workiom
21	Work Management CRM	Work management, CRM, and scheduling	Airtable, Asana, Calendly, ChatBot, ClickFunnels Classic, Convert Forms, Deadline Funnel, Dubsado, Elham, GetResponse, Manatal, monday.com, Motion, myInterview, Openpaye, QuizCube, systeme.io, Ulama, Worksuite
22	Content Collaboration Marketing	HR content creation and engagement tracking	Bitly, Brillium, Campaign Monitor, Confluence Cloud, Credly, Ezekia, Gather, Kajabi, Mailchimp, Mobile Message, OneDrive, PartnerStack
23	Marketing Forms Education	Recruitment marketing and form-based experiences	AWeber, Chargebee, Coda, Convenia, Fillout Forms, Google Meet, PayPal, Pipefy, RD Station, RocketDevs, Runrun.it, SchoolMaker, ShopHire, Stackby, SurveyMonkey, Teachfloor, TinyEmail, Tutory, Webflow, Xobin

No.	Community Name	Primary role	SaaS
24	Ops Orchestration Hub	Multi-step HR operations orchestration	100Hires ATS, Airpreneur, Canditech, ChatGPT (OpenAI), ClickUp, Discord, Gmail, Rooster, SecPaid.com, Shape, tousquali, Tree-Nation
25	Business Platform Workspace	Integrated workspace for HR and talent operations	17hats, Agilix Dawn, Ashby, Capsule CRM, Circle, Clio, easy.jobs, easyhire, Keap, Kommo, LEADGODS, Linear, Notion, UpLead, WorkRamp, Zapier Chrome extension
26	Messaging Lead Capture	Multi-channel message and lead capture	ClickSend SMS, Facebook Conversions, Facebook Lead Ads, Heyflow, Hire Aiva, Hirely, Interviewer.ai, LinkedIn Conversions, Ninja Forms, onboard, Onepage, Perspective, RecruitApp.ai, Seeqle, SMS by Zapier, Vervoe, Zoho Mail
27	IT Admin Backoffice	HR-related IT administration and backoffice tasks	Freshservice, Google Slides, Google Workspace Admin, Hibob, Juro, LearnUpon, Microsoft Exchange, NetSuite, PagerDuty, Signable, Snappy Gifts, SysAid, Tidbyt, Xledger, YM Careers
28	Payments Subscription Commerce	Payments and subscriptions for training programs	Crove, Digistore24, GoodBarber, GoZen Growth, Gravity Forms, Kennis.shop, MailerLite Classic, Mollie, MySQL, MyZenTeam, OnlineCourseHost.com, Paystack, Paythen, Razorpay, SubHub, Tarvent, Trustpilot, VBOUT
29	Ads Funnels Webinar	Recruitment funnels and webinar-based campaigns	Bucket.io 2.0, Burst SMS, ClickMeeting, Facebook Custom Audiences, GoTo Webinar, InfluencerSoft, Manychat, MobilePages, Moxie, Pagewheel, SendFox, SMS Alert, Snov.io, TidyCal, Voyager Infinity
30	Marketing Funnel LMS	LMS and marketing-funnel integration	Acuity Scheduling, Brevo, CleverReach, DreamApply, Facture.net, FunnelCockpit, Funnel-Tunnel, JobNimbus, KlickTipp, LearningSuite, Masterpages, Mentortools, Qualiobee, QuestionScout
31	Backoffice Operations	Recruitment, contract, and financial backoffice operations	AdPage, BrightHR, Digest by Zapier, FranchiseUNIVERSUM, Funnelforms, Infobip, intelliHR, JobAdder, LearnWorlds, Recruitee, Sage Intacct, Scrive, Superchat, Tanda, TaskTrain, Trainual, Yobi, Yoobi
32	HR Process Platform	Repeatable HR process and checklist automation	Azure Active Directory, BambooHR, Baserow, Enalyzer, Expensify, Forms On Fire, Giftsenda, Humanity, Jotform Enterprise, ManageEngine ServiceDesk Plus Cloud, Manifestly Checklists, Moxo, People HR, Precisely, Process Street, PureSMS, Rise, Seismic Learning
33	Comms Data Automation	Recruitment communication and HR analytics	Beatview, CatchApp Bookings, Clearbit, Delighted, DocuGenerate, GatewayAPI SMS, GPTZero: Detect AI, Greenhouse, Niceboard, OnceHub, Range, RSS by Zapier, SuprForms, tl;dv, touchSMS, Vocal Video
34	Onboarding Assessment Survey	Onboarding, assessments, and survey-based decisions	Casting42, Delay by Zapier, Demio, GainKnowHow, Google Sheets, Hero Parser, Hirevire, LeadConnector, MyKlub, OfficeMaps, Oyster, SENAR, Senja, SignerX.com, SkillzRun, Startquestion, SurveyLab, Testlify, TRAFFIT, Weworkbook
35	Workforce Collaboration Projects	Workforce collaboration and people operations	Akada, AssessTEAM, BirthdayBot, Check In Pointe, Google Contacts, HR Cloud, Huntool, Jackrabbit Class, Microsoft Office 365, PARiM, Studio Pro, Trello, Wrike
36	Recruitment Sales Collaboration	Collaborative recruiting and candidate evaluation	46elks, Bloom, Callbell, CloudTalk, Coderbyte, eSignatures, Geckoboard, GetProspect, HEU, Klaxoon, Pocketlaw, PostgreSQL, Remote, SmartyMeet, Snapchat Lead Generation, Talkie, Teamtailor, Transkriptor
37	Recruitment Marketing Ops	Recruitment marketing operations	Breezy HR, CalendarFix, Feedier, Formplus, Kendo Email, OpenPhone, RepCard, RingCentral, SurveySparrow, Talenteria, Teamwork, True Reply, WooSender, WS Form, Zoho Campaigns, Zoho Projects
38	Talent Finance Operations	Talent management with finance and operations	BlinkSwag, CalendarHero, Glide, LeadDelta, Mailshake, MessageMedia SMS, Priority, RanknRole, Retable, SalesQL, Sharetribe, Typebot, Wave, Workday, Xodo Sign, Zoho Bookings, Zoho Creator, Zoho Invoice, Zoho Recruit, Zoho Sheet

No.	Community Name	Primary role	SaaS
39	Events Ecommerce Membership	HR events, memberships, and training commerce	Amazon Seller Central, Big Cartel, BigCommerce, BigMarker, iContact, LifterLMS, Magento, NationBuilder, Patreon, ShareFile, StreamYard, The Events Calendar, WebinarJam / EverWebinar, WebinarNinja, WishList Member, Zoho Billing, Zoho Survey
40	Recruiting ATS Pipeline	ATS-centered recruiting pipeline management	BombBomb.com, CATS, Cognito Forms, Compass, Follow Up Boss, hireEZ, Hireflix, Keller Williams Command, kvCORE, Mailjet, Meet Alfred, Odoo ERP Self Hosted, Recruiterflow, Sierra Interactive, Tracker, Wizehire
41	Finance Commerce Engagement	HR initiatives linked to finance and engagement	CDLSuite, Constant Contact, CourseStorm, CredSure, EdApp, Google Ads, Gumroad, Kartra, Little Green Light, PandaDoc, Quickbase, QuickBooks Online, Spektrix, Todoist, Worksome, Xero, Yuvo
42	Customer Team Operations	HR team operations and external outreach	CallRail, ClayHR, Close, DeleTrain, Docusign, EDlumina, EducateMe, Facebook Pages, Formidable Forms, FreshBooks, Highrise, Holded, Jobtoolz, Reddit Lead Ads, Slack, Teamleader Focus, ThriveCart, Viterbit, Wix, Zoho CRM

저자소개

김경아 : 서울과학기술대학교 데이터사이언스학과에서 석사학위를 취득하였으며, 현재 (주)베가스 AA그룹에 재직 중이다. 연구 분야는 그래프 마이닝, 비즈니스 애널리틱스, AI 자동화 등이다.

이학연: 서울대학교 산업공학과에서 학사학위를 취득하였으며, 동 대학원에서 박사학위를 받았다. 현재 서울과학기술대학교 산업공학과 교수로 재직 중이다. 주요 연구 분야는 기술예측, 이노베이션 애널리틱스, 디지털 혁신 전략 등이다.