International Journal of Creative and Open Research in Engineering and Management
ISSN: 3108-1754 (Online)
Volume 02 Issue 11 November-2026 | Impact Factor: 3.5



AI-Driven Personal and Collaborative Task Management: A Survey of Intelligent Scheduling, LLM Agents, and Multi-Agent Systems

Dnyandeep Bendewar
Department of Computer Science and Engineering
Prof. Ram Meghe Institute of Technology and Research, Badnera-Amravati
Amravati, Maharashtra, India
bendewardnyandeep@gmail.com

Bhaumik Dhore
Department of Computer Science and Engineering
Prof. Ram Meghe Institute of Technology and Research, Badnera-Amravati
Amravati, Maharashtra, India
bhamuik.dhore@gmail.com

Bhumika Baitule
Department of ComputerScience and Engineering
Prof. Ram Meghe Institute of Technology and Research, Banera-Amravati
Amravati, Maharashtra, India
bhumikabaitule@gmail.com

Radha Kokate
Department of ComputerScience and Engineering
Prof. Ram Meghe Institute of Technology and Research, Banera-Amravati
Amravati, Maharashtra, India
radhakokate302005@gmail.com

Dr. S. P. Ingale
Department of ComputerScience and Engineering
Prof. Ram Meghe Institute of Technology and Research, Banera-Amravati
Amravati, Maharashtra, India
spingale@mitra.ac.in

How to Cite this Article:
Bendewar, D., Dhore, B., Baitule, B. & Kokate, R. (2026). AI-Driven Personal and Collaborative Task Management: A Survey of Intelligent Scheduling, LLM Agents, and Multi-Agent Systems. International Journal of Creative and Open Research in Engineering and Management, 02(11), 1-9. https://doi.org/10.55041/ijcope.v2i11.103

License:
This article is published under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.
© The Author(s). Published by International Journal of Creative and Open Research in Engineering and Management.







https://doi.org/10.55041/ijcope.v2i11.103

***Abstract*—Artificial intelligence (AI) is increasingly transforming traditional productivity and task management systems by enabling intelligent assistance for task planning, prioritization, scheduling, collaboration, and decision-making. This paper presents a survey of AI-driven personal and collaborative productivity systems, focusing on intelligent task management, scheduling, meeting management, large language models (LLMs), AI agents, multi-agent systems, and retrieval-augmented generation (RAG). Existing research is reviewed and classified according to the AI techniques employed, application areas, system capabilities, and limitations. The survey examines how these technologies support individual productivity as well as collaborative workflows involving tasks, calendars, meetings, communication, and shared knowledge. A comparative analysis is presented to identify the strengths and limitations of existing approaches. The study further highlights key research gaps, including fragmented productivity tools, limited contextual awareness and personalization, challenges in autonomous task execution and multi-agent coordination, and concerns related to privacy, security, reliability, and human oversight. Finally, emerging trends and future research directions are discussed toward developing more integrated, adaptive, trustworthy, and context-aware AI-driven productivity systems. This survey provides a structured understanding of the current research landscape and identifies opportunities for advancing intelligent productivity and task management systems.**

Keywords*— *Artificial Intelligence, Productivity Management, Task Management, Intelligent Scheduling, Large Language Models, AI Agents, Multi-Agent Systems, Human-AI Collaboration, Retrieval-Augmented Generation.

I. Introduction

A. Background and Motivation

The rapid advancement of artificial intelligence (AI) has transformed traditional productivity and task management systems from basic organizational tools into intelligent systems capable of supporting planning, decision-making, and task execution. Earlier productivity applications primarily focused on storing tasks, managing schedules, tracking deadlines, and organizing information. The development of machine learning, natural language processing, and large language models (LLMs) has expanded these capabilities by enabling systems to understand natural-language instructions and provide more adaptive assistance [1], [2].

The emergence of LLM-based systems has further expanded the role of AI in productivity applications. Modern language models can support reasoning, task decomposition, and interaction with external tools, while AI-agent research has explored capabilities such as planning, memory, tool use, and autonomous decision-making [4]-[6]. These developments create opportunities for productivity systems that can assist users not only in managing information but also in planning and carrying out tasks.

Productivity management also extends beyond individual activities. Modern work environments involve multiple tasks, deadlines, calendars, meetings, documents, and interactions between individuals or teams. Consequently, AI has been investigated in project management and scheduling for activities such as planning, resource allocation, project monitoring, and workflow automation [12]-[16]. Multi-agent approaches have additionally been explored for coordinating activities such as meeting scheduling and collaborative decision-making [7], [17].



Another important development is the integration of external knowledge with generative AI. Retrieval-Augmented Generation (RAG) enables language models to retrieve relevant information from external knowledge sources before generating responses, making AI systems more suitable for knowledge-intensive applications [10], [11]. Meeting-intelligence systems similarly demonstrate how AI can process discussions and generate useful summaries and actionable information [18].

Despite these developments, research on AI-driven productivity remains distributed across several areas, including task management, scheduling, project management, knowledge retrieval, meeting intelligence, and autonomous agents. Integrating these capabilities presents additional challenges related to reliability, personalization, user control, privacy, security, and trust [19], [20]. These challenges highlight the need for a broader examination of how different AI techniques can be combined to support both individual productivity and collaborative workflows.

Therefore, this survey examines the current research landscape of AI-driven personal and collaborative productivity and task management systems, with emphasis on intelligent scheduling, LLM-based agents, multi-agent collaboration, knowledge management, and human-AI interaction. The survey compares existing approaches, identifies their capabilities and limitations, and highlights research gaps that can guide the development of more integrated, adaptive, reliable, and user-centered productivity systems.

B. *Scope and Contributions*

This survey focuses on research related to AI-driven productivity and task management systems that support both individual users and collaborative work environments. The scope covers AI techniques for task and project management, intelligent scheduling, meeting management, LLM-based agents, multi-agent systems, retrieval-augmented knowledge management, and human-AI interaction.

The main contributions of this survey are as follows:

1. Classification of existing approaches: AI-driven productivity systems are organized according to their primary technologies and application areas.
2. Analysis of AI techniques: The survey examines the roles of machine learning, NLP, LLMs, optimization methods, RAG, AI agents, and multi-agent systems in productivity management.
3. Comparison of existing systems: Existing approaches are compared based on capabilities such as task management, scheduling, meeting support, knowledge access, automation, and collaboration.
4. Identification of research gaps: Technical, evaluation, collaboration, governance, privacy, and human-AI interaction challenges are analysed to identify limitations in current approaches.
5. Future research directions: Emerging opportunities for developing more adaptive, reliable, integrated, and human-centered AI productivity systems are discussed.

C. *Research Questions*

To guide the literature analysis, this survey addresses the following research questions:

RQ1: What AI techniques are currently used in intelligent productivity and task management systems?

RQ2: How are AI techniques applied to task prioritization, assignment, scheduling, and project management?

RQ3: How do existing AI systems support personal productivity and collaborative activities involving tasks, meetings, calendars, and knowledge?

RQ4: How are LLM-based agents and multi-agent systems being used for productivity, scheduling, and task automation?

RQ5: What are the major technical, collaborative, security, privacy, and human-AI challenges in existing productivity systems?

RQ6: What future research directions can improve the adaptability, reliability, personalization, and collaboration capabilities of AI-driven productivity systems?

II. Survey and Literature Review Methodology

A. *Search Strategy*

A structured literature search was conducted to identify research relevant to AI-driven personal and collaborative productivity and task management systems. The search focused on major research areas including artificial intelligence, task management, intelligent scheduling, project management, large language models (LLMs), AI agents, multi-agent systems, retrieval-augmented generation (RAG), meeting intelligence, and human-AI collaboration.

Relevant research was identified using academic sources such as IEEE Xplore, ACM Digital Library, SpringerLink, ScienceDirect, Google Scholar, and arXiv. Search terms were formulated using combinations of keywords related to the scope of the survey, including *AI productivity systems*, *AI task management*, *intelligent scheduling*, *AI project management*, *large language model agents*, *multi-agent systems*, *RAG*, *meeting summarization*, and *human-AI interaction*.

The search was primarily focused on recent developments in AI-driven productivity systems while also including selected foundational studies that provide important technical background. The final literature set consists of 20 research papers covering the major themes examined in this survey.

B. *Selection and Screening*

The identified literature was evaluated according to its relevance to the research questions and overall scope of the survey. Papers were selected based on their contribution to one or more of the major research areas, including AI-based productivity, task and project management, scheduling, LLMs, AI agents, multi-agent collaboration, knowledge management, meeting intelligence, and human-AI interaction.

The screening process considered the relevance, technical contribution, research quality, and applicability of each study to the objectives of the survey. Duplicate, unrelated, and insufficiently relevant studies were excluded from further analysis. Preference was given to studies providing sufficient technical information to support meaningful comparison and synthesis. The overall selection and screening process is illustrated in Fig. 1.



Fig. 1. Conceptual architecture of a multi-agent productivity system, synthesized from approaches discussed in [7]-[9], [17].

Following the screening process, 20 papers were selected for detailed analysis based on their relevance to the research questions and the scope of this survey. The selected studies were subsequently categorized and analyzed according to their AI techniques, application areas, system capabilities, and limitations.

C. *Data Extraction and Synthesis*

For each selected paper, relevant information was extracted and organized according to the research objectives. The extracted information included the research problem, AI technique, application domain, methodology, major capabilities, key findings, and reported limitations.

The selected studies were then grouped into thematic categories, including AI and LLM foundations, intelligent task and project management, scheduling and meeting management, AI agents, multi-agent systems, RAG and knowledge management, and human-AI collaboration. This classification enables comparison of different approaches across common dimensions rather than presenting the selected studies as independent summaries.

The synthesis focuses on identifying common approaches, technological developments, strengths, limitations, and unresolved challenges across the literature. The findings are subsequently used to develop a comparative analysis, taxonomy of existing systems, research gaps, and future research directions.

III. Evolution of AI-Based Productivity Systems

A. *Early Digital Productivity Tools*

Early productivity systems were primarily designed to digitize activities that were previously managed manually. Task lists, digital calendars, spreadsheets, project-management software, and scheduling applications provided users with mechanisms for recording tasks, organizing deadlines, tracking progress, and coordinating activities. These systems improved accessibility and organization but generally depended on users to create tasks, determine priorities, update schedules, and make decisions about how work should be organized.

As digital work environments became more complex, productivity systems gradually expanded from individual task tracking toward project and workflow management. Features such as shared workspaces, task dependencies, resource allocation, progress monitoring, and collaborative scheduling enabled multiple users to coordinate their activities within a common environment. However, these systems largely remained user-driven, with limited ability to understand the context of tasks or independently recommend and execute actions.

The increasing complexity of project planning and scheduling also motivated research into computational approaches for handling constraints, resources, dependencies, and deadlines. Resource-constrained project scheduling has been extensively studied as a structured optimization problem in which activities must be assigned within available resources while satisfying precedence and other constraints [15], [16]. These approaches established an important foundation for later intelligent scheduling systems.

B. *Machine Learning and Context-Aware Systems*

The application of machine learning introduced the possibility of moving beyond manually configured productivity workflows toward systems capable of learning from data. Instead of relying exclusively on predefined rules, learning-based approaches can identify patterns in historical project information and use them to support activities such as schedule creation, prediction, and project decision-making.

Recent research has examined the application of machine learning to project schedule creation, demonstrating the growing interest in using data-driven techniques to support planning and scheduling processes [14]. Similarly, systematic studies of AI in project management show that AI techniques are being explored across multiple project-management activities, including planning, monitoring, decision support, and automation [12], [13].

Despite these developments, conventional machine-learning approaches generally depend on structured data and predefined objectives. They can be effective for specific prediction or optimization tasks but may have difficulty interpreting unstructured user requests, understanding complex natural-language context, or flexibly interacting with users. These limitations contributed to increasing interest in natural language processing and, subsequently, large language models.

C. *LLM and Agentic Systems*

The introduction of Transformer-based architectures significantly influenced the development of modern language models. The Transformer architecture enabled efficient processing of relationships between elements in a sequence through attention mechanisms and became a foundation for subsequent large-scale language models [1]. The development of increasingly capable language models demonstrated that a single model could perform a wide range of language-based tasks with limited task-specific training [2].

Subsequent research showed that LLMs could also perform more complex reasoning tasks when provided with appropriate prompting strategies. Chain-of-thought prompting demonstrated the potential of language models to generate intermediate reasoning steps for solving complex problems [3]. This created new possibilities for productivity systems that require natural-language understanding, task decomposition, and planning.

The evolution from language models to agentic systems represents a further shift in productivity-system design. Rather than only generating responses, LLM-based agents can combine reasoning with tool interaction, planning, memory, and actions. ReAct, for example, demonstrated an approach that combines reasoning and acting so that language models can interact with external tools while solving tasks [4]. Research on autonomous language agents and cognitive architectures has subsequently explored more structured combinations of planning, memory, reasoning, and tool use [5], [6].



More recently, multi-agent approaches have extended this concept by allowing multiple specialized agents to communicate and coordinate their activities. Frameworks such as AutoGen, MetaGPT, and CAMEL demonstrate different approaches to communication, role specialization, and collaboration among LLM-based agents [7]-[9]. This evolution creates the possibility of productivity systems in which different AI components can specialize in tasks such as planning, scheduling, knowledge retrieval, or workflow execution.

Overall, the evolution of productivity systems can be viewed as a transition from user-controlled digital tools, to data-driven intelligent systems, and finally toward language-driven and agentic systems capable of reasoning, coordination, and action. This progression provides the foundation for examining the specific AI techniques and productivity applications discussed in the following sections.

IV. AI Techniques for Productivity and Task Management

A. Machine Learning and Predictive Modeling

Machine learning enables productivity systems to identify patterns from historical data and use them for prediction, classification, and decision support. In task and project management, these techniques can support activities such as estimating project duration, identifying scheduling risks, predicting delays, and assisting with workload planning. Their usefulness is particularly strong when structured historical data is available and the prediction objective is clearly defined.

Recent work on machine-learning-based project scheduling shows growing interest in using data-driven approaches to support schedule generation and planning activities [14]. Broader studies of AI in project management also indicate that learning-based techniques can contribute to planning, monitoring, forecasting, and decision support [12], [13].

However, machine-learning approaches generally depend on the quality and availability of training data. Their performance may also be limited when the system must interpret ambiguous natural-language requests or adapt to highly dynamic situations without sufficient historical examples.

B. Natural Language Processing and Large Language Models

Natural language processing allows productivity systems to interpret and generate human language, making interaction with digital tools more intuitive. Earlier NLP approaches typically relied on task-specific models for activities such as classification, information extraction, and text summarization. The emergence of large language models has substantially expanded these capabilities by enabling a single model to handle a wider range of language-based tasks.

Transformer-based architectures provide the technical foundation for many modern language models [1], while large-scale language models have demonstrated strong few-shot and general-purpose language capabilities [2]. In productivity applications, these models can potentially interpret user requests, summarize information, generate task descriptions, assist with planning, and support conversational interaction.

Reasoning-oriented prompting techniques further improve the usefulness of LLMs for tasks requiring multiple intermediate steps. Chain-of-thought prompting, for example, demonstrates that language models can improve performance on complex reasoning tasks by generating intermediate reasoning processes [3]. These capabilities are relevant to productivity systems where complex objectives may need to be decomposed into smaller actions.

Despite their flexibility, LLMs may generate inaccurate or inconsistent outputs and do not inherently guarantee reliable execution of structured tasks. As a result, they are often combined with external tools, retrieval systems, or agent architectures when used in practical productivity applications.

C. Optimization and Scheduling Methods

Optimization techniques play an important role in productivity systems where tasks must be scheduled under constraints such as limited resources, dependencies, deadlines, and availability. Unlike language models, optimization methods are designed to explicitly represent and satisfy such constraints, making them suitable for structured scheduling and resource-allocation problems.

Resource-Constrained Project Scheduling Problems (RCPSP) represent a well-established class of scheduling problems in which activities must be arranged while satisfying precedence and resource limitations [15], [16]. These methods are particularly valuable in project environments where resource conflicts and task dependencies must be handled systematically.

Optimization can also complement AI-based systems. For example, an LLM may interpret user requirements or generate candidate plans, while an optimization component can verify constraints and determine a feasible schedule. Such hybrid approaches can combine the flexibility of natural-language reasoning with the reliability of structured optimization.

D. Comparison of Core Techniques

The major AI techniques used in productivity and task management differ in their capabilities, requirements, and suitability for specific applications. Machine learning approaches are mainly used for identifying patterns from historical data and supporting prediction-based decisions, while NLP and LLM-based approaches enable systems to understand natural-language instructions, generate task-related content, and assist with reasoning and planning. Optimization methods focus on finding feasible or efficient solutions under constraints such as deadlines, resource availability, task dependencies, and scheduling requirements. Therefore, selecting an appropriate technique depends on the type of productivity problem, the available data, and the level of automation required. Table I summarizes the major characteristics, limitations, and applications of these techniques in intelligent productivity systems.

These techniques address different aspects of intelligent productivity and are therefore complementary rather than interchangeable. Machine learning is primarily suitable for data-driven prediction and estimation, whereas NLP and LLM-based approaches provide flexible interaction, task understanding, and reasoning over natural-language information. Optimization methods are particularly effective when tasks involve explicit constraints such as deadlines, resource availability, dependencies, and scheduling requirements. Combining these approaches can provide more comprehensive productivity support by integrating prediction, language-based reasoning, and constraint-aware decision-making.



TABLE I. COMPARISON OF CORE AI TECHNIQUES FOR PRODUCTIVITY

Technique	Main Strength	Main Limitation	Productivity Application
Machine Learning	Prediction and pattern recognition	Requires quality historical data	Delay prediction, schedule estimation
NLP / LLMs	Natural-language understanding and reasoning	May produce inconsistent outputs	Task interpretation, planning, summarization
Optimization	Reliable constraint handling	Requires structured inputs	Scheduling, resource allocation
Hybrid AI	Combines complementary strengths	Greater system complexity	End-to-end intelligent productivity support

V. INGELLIGENT TASK AND PROJECT MANAGEMENT

A. Task Prioritization and Deadline Prediction

AI-based task management systems aim to improve how tasks are prioritized, estimated, and completed by using information such as deadlines, dependencies, project progress, and historical data. Traditional task management generally requires users to manually assign priorities and estimate completion times. AI-based approaches can support these activities by analyzing available project information and identifying patterns that may indicate delays, schedule risks, or changing task requirements. Research on AI-enabled project management shows that AI can support planning, monitoring, decision-making, and prediction across different stages of project execution [12], [13].

Machine learning techniques are particularly useful for estimating project schedules and identifying factors that influence task duration and completion. Studies on machine learning-based project schedule creation demonstrate the potential of data-driven methods to improve schedule generation and decision support [14]. However, effective prediction depends on the availability and quality of historical project data. In addition, task priority is often influenced by contextual factors that may not be directly represented in structured datasets, such as changing user requirements, task importance, or dependencies. Therefore, intelligent task prioritization requires a combination of predictive capabilities and contextual understanding.

B. Task Allocation and Workflow Management

Task allocation is an important component of collaborative project management because tasks must be assigned according to requirements, resources, dependencies, and project constraints. AI-enabled project management approaches can assist in allocating resources, coordinating activities, and managing project workflows [12], [13]. Optimization-based scheduling methods provide a structured approach for handling resource constraints and dependencies, particularly in complex project environments [15], [16].

Recent LLM-based and multi-agent approaches provide additional possibilities for workflow management. Agent-based systems can interpret instructions, reason about tasks, and interact with external tools, while multi-agent frameworks allow different agents to perform specialized roles and coordinate their activities [4], [7]-[9]. Such approaches can support more flexible task distribution and workflow automation. However, autonomous task allocation also introduces challenges related to coordination, conflicting decisions, reliability, and human oversight.

C. Project Monitoring and Workflow Automation

Project monitoring involves tracking task progress, identifying potential delays, updating project information, and supporting timely decision-making. Traditional project management systems primarily depend on users to update task status and monitor progress. AI-based approaches can reduce this manual effort by using predictive models, intelligent analysis, and automated workflows. Research in AI-enabled project management indicates that AI can contribute to project planning, monitoring, risk management, and decision support [12], [13].

LLM-based agents further extend workflow automation by combining reasoning with actions and tool interaction [4]-[6]. Instead of only providing recommendations, an agent-based system can potentially interpret a user's request, break a complex objective into smaller tasks, retrieve relevant information, and interact with connected productivity tools. However, increasing the level of automation requires mechanisms for validation, error handling, human intervention, and reliable execution. Consequently, intelligent project management should balance automation with appropriate human control rather than relying entirely on autonomous decision-making.

The major applications of AI in task and project management, along with their corresponding approaches, capabilities, and challenges, are summarized in Table II.

TABLE II. AI APPLICATIONS IN TASK AND PROJECT MANAGEMENT

Application	AI Approach	Main Capability	Key Challenge
Task Prioritization	ML / LLMs	Priority estimation and contextual understanding	Changing requirements
Deadline Prediction	Machine Learning	Duration and delay prediction	Historical data dependency
Task Allocation	Optimization / Multi-Agent Systems	Resource-aware task assignment	Coordination complexity
Workflow Automation	LLM Agents	Task decomposition and tool interaction	Reliability and human oversight
Project Monitoring	ML / AI Systems	Progress and risk analysis	Data quality and context

VI. INTELLIGENT SCHEDULING AND MEETING MANAGEMENT

A. Calendar Intelligence and Scheduling

Intelligent scheduling focuses on automatically organizing tasks, events, meetings, and other activities while considering constraints such as deadlines, availability, dependencies, and resource requirements. Traditional scheduling systems generally require users to manually select



suitable time slots and update schedules when conditions change. AI-based scheduling approaches can reduce this manual effort by analyzing constraints and identifying feasible or efficient schedules. Research on project scheduling demonstrates the importance of optimization techniques for handling resource limitations, task dependencies, and competing scheduling requirements [15], [16].

Machine learning can further support scheduling by learning from historical project data and improving schedule estimation. Research on machine learning in project schedule creation highlights the potential of data-driven methods for generating and improving project schedules [14]. In collaborative environments, multi-agent scheduling approaches can allow independent agents to negotiate and coordinate available time slots according to their respective requirements [17]. These approaches demonstrate how intelligent scheduling can move from manually configured calendars toward more adaptive and automated scheduling systems.

B. Meeting Preparation and Facilitation

Meetings are an important component of collaborative productivity, but their effectiveness depends on appropriate preparation, participant coordination, and access to relevant information. Intelligent systems can assist with meeting preparation by organizing agendas, identifying relevant tasks or documents, and providing contextual information to participants. Multi-agent approaches are particularly relevant to meeting coordination because different agents can represent participants or organizational requirements and communicate to identify mutually acceptable schedules [17].

LLM-based systems provide additional opportunities for meeting facilitation through natural-language interaction. An AI assistant can interpret meeting objectives, organize information, and assist participants during different stages of a meeting. However, effective meeting assistance requires accurate understanding of conversational context and participant intentions. Therefore, meeting-support systems need to balance automation with human control, particularly when AI-generated information influences collaborative decisions.

C. Meeting Summaries and Action Items

Meeting discussions often contain decisions, task assignments, deadlines, and follow-up activities that need to be converted into structured information. Meeting summarization systems aim to extract the most important information from lengthy conversations and present it in a concise form. Research on meeting summarization identifies summarization as an important component of meeting intelligence, with approaches designed to capture key discussion points and useful meeting information [18].

LLMs can extend this capability by generating natural-language summaries and identifying potential action items from meeting transcripts. Such systems can connect meeting outcomes with task-management workflows by transforming identified actions into structured tasks, assigning responsible participants, or associating deadlines where sufficient information is available. However, automatically generated summaries and action items may contain omissions or incorrect interpretations. Human verification therefore remains important when meeting outputs are used for consequential project decisions.

D. Scheduling System Comparison

Intelligent scheduling approaches differ in their techniques and level of automation. Optimization methods handle explicit constraints, machine learning supports prediction, multi-agent systems enable coordination, and LLMs provide natural-language scheduling assistance. Combining these approaches can provide more flexible and comprehensive scheduling, but introduces additional challenges related to reliability, validation, and human oversight.

The major approaches used in intelligent scheduling and meeting management are compared in Table III based on their primary functions, strengths, and limitations.

TABLE III. COMPARISON OF INTELLIGENT SCHEDULING AND MEETING APPROACHES

Approach	**Primary Function**	**Main Strength**	**Key Limitation**
Optimization-Based Scheduling	Constraint-aware scheduling	Handles dependencies and resource constraints	Requires structured constraints
Machine Learning	Schedule prediction and estimation	Learns from historical data	Data dependency
Multi-Agent Scheduling	Participant coordination	Supports distributed decision-making	Coordination complexity
Meeting Summarization	Meeting information extraction	Reduces information overload	Possible information loss
LLM-Based Scheduling	Natural-language scheduling assistance	Flexible interaction and reasoning	Reliability of generated outputs
Hybrid Approaches	Integrated scheduling and assistance	Combines complementary capabilities	Higher system complexity

VII. LLM-BASED AI AGENTS

A. Agent Architectures and Tool Use

Large language models have evolved from systems that primarily generate or interpret text into more capable architectures that can perform multi-step reasoning and interact with external tools. The development of Transformer-based architectures provided an important foundation for modern language models [1], while subsequent research demonstrated their ability to perform a wide range of language tasks with limited task-specific examples [2]. Reasoning-oriented techniques further improved the ability of language models to handle complex, multi-step problems [3].

LLM-based agents extend these capabilities by combining language models with components such as planning, memory, reasoning, and external tools. ReAct



introduced an approach that combines reasoning with actions, allowing a language model to alternate between internal reasoning and interactions with an external environment [4]. Surveys and architectural studies of language agents further identify planning, memory, tool use, and interaction as important components of autonomous agent systems [5], [6]. These capabilities make LLM agents relevant to productivity applications where users expect systems to understand goals and perform sequences of related actions.

B. Productivity Assistant Applications

LLM agents can support productivity by providing natural-language interfaces for tasks that traditionally require interaction with multiple software tools. Instead of manually navigating different interfaces, users can express an objective in natural language, after which an agent can interpret the request, decompose it into subtasks, retrieve relevant information, and interact with available tools. This approach can support activities such as task creation, information retrieval, scheduling assistance, document processing, and workflow coordination.

The ability of LLM agents to combine reasoning and tool interaction is particularly relevant to productivity environments because many activities require multiple sequential operations rather than a single response [4]-[6]. For example, a complex request may require understanding user intent, checking relevant information, determining an appropriate action, and then executing that action through an external service. However, the effectiveness of such systems depends on the reliability of the underlying model, the quality of available tools, and the accuracy of the context provided to the agent.

C. Autonomy and Evaluation

Increasing the autonomy of LLM-based agents introduces both opportunities and challenges. An agent with greater autonomy can independently plan and execute multiple steps, potentially reducing the amount of manual effort required from users. However, autonomous execution also increases the risk of incorrect reasoning, inappropriate tool use, or unintended actions. Research on autonomous language agents emphasizes the importance of architectural components such as planning, memory, tool use, and interaction for building more capable agents [5], [6].

Evaluation of LLM agents therefore requires more than measuring the quality of generated text. Productivity-oriented agents should also be evaluated based on task completion, planning accuracy, tool-use reliability, consistency, response time, and the ability to recover from errors. Human oversight remains important, particularly for actions that modify schedules, assign tasks, communicate with other users, or affect shared project information. Human-AI interaction guidelines emphasize the need to make AI behavior understandable and provide users with appropriate control over system actions [19].

The major capabilities of LLM-based agents and their relevance to productivity applications are summarized in Table IV.

TABLE IV. COMPARISON OF CORE AI TECHNIQUES FOR PRODUCTIVITY

Agent Capability	**Description**	**Productivity Application**	**Main Challenge**
Reasoning	Handles multi-step problems	Task planning and decision support	Reasoning reliability
Planning	Breaks goals into subtasks	Workflow and project planning	Incorrect plans
Memory	Maintains relevant context	Personalized assistance	Context management
Tool Use	Interacts with external systems	Calendar and task automation	Tool-use errors
Autonomy	Performs actions with limited intervention	End-to-end workflow automation	Human control and safety
Multi-Agent Interaction	Coordinates specialized agents	Collaborative workflows	Coordination complexity

VIII. MULTI-AGENT COLLABORATIVE SYSTEMS

A. Agent Roles and Coordination

Multi-agent systems extend the capabilities of individual AI agents by allowing multiple agents to interact, communicate, and perform specialized roles within a common workflow. Instead of relying on a single agent to complete every task, responsibilities can be distributed among agents according to their capabilities or assigned objectives. This approach is particularly relevant to collaborative productivity environments, where different activities such as task planning, scheduling, information retrieval, and monitoring may need to be performed simultaneously.

Frameworks such as AutoGen demonstrate how multiple LLM-based agents can communicate and cooperate to accomplish complex tasks [7]. MetaGPT further explores role-based multi-agent collaboration by assigning specialized roles to agents within a structured workflow [8]. CAMEL investigates communication between role-playing agents and demonstrates how agents can cooperate through structured interactions [9]. These approaches indicate that role specialization can improve the organization of complex AI workflows, although effective coordination remains an important challenge.

B. Collaboration Workflows

Multi-agent systems can support productivity workflows by assigning different responsibilities to specialized agents, such as task planning, scheduling, information retrieval, and progress monitoring. This specialization allows complex workflows to be divided into smaller tasks and handled in parallel. Agents may also communicate and coordinate decisions, as demonstrated in multi-agent meeting scheduling [17] and LLM-based frameworks [7]-[9]. However, increased agent interaction can lead to communication



overhead, conflicting decisions, and difficulties in maintaining a shared understanding of the workflow.

C. Multi-Agent System Architecture

A multi-agent productivity architecture generally consists of several specialized agents operating around a shared coordination mechanism. A user-facing agent can interpret the user's goals and distribute work to specialized agents responsible for tasks such as scheduling, project management, knowledge retrieval, or meeting management. These agents can communicate through a coordination layer and access shared information or external tools when required.

Fig. 2 presents a conceptual multi-agent architecture where a coordinator distributes tasks among specialized agents for task management, scheduling, meetings, knowledge retrieval, and project monitoring. The architecture is synthesized from [7]-[9] and [17].

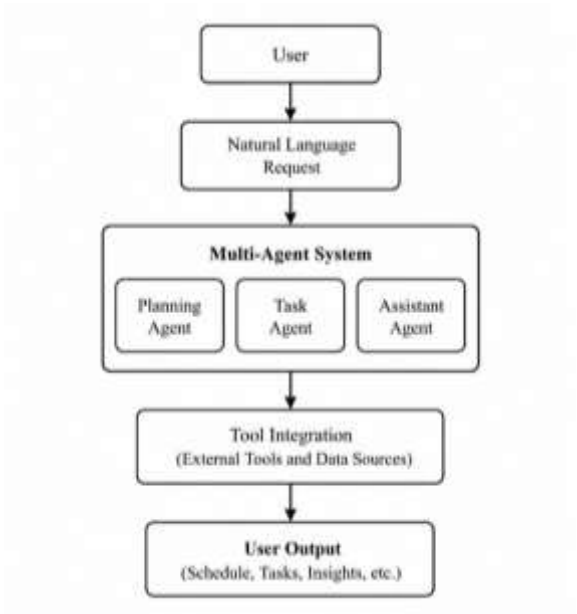


Fig. 2. Conceptual architecture of a multi-agent productivity system, synthesized from approaches discussed in [7]-[9], [17].

Such an architecture allows complex productivity workflows to be decomposed into specialized responsibilities while maintaining coordination through a central agent. The separation of agent roles can improve modularity and allow different capabilities to be combined within a common workflow. However, communication overhead, coordination conflicts, inconsistent decisions, and shared-context management remain important challenges in multi-agent systems.

The surveyed multi-agent approaches differ in their coordination mechanisms, role specialization, and potential applications to productivity workflows, as summarized in Table V.

TABLE V. COMPARISON OF MULTI-AGENT COLLABORATION APPROACHES

Approach	Main Idea	Collaboration Mechanism	Productivity Relevance
AutoGen [7]	LLM agents cooperating through conversations	Agent-to-agent communication	Complex task workflows
MetaGPT [8]	Role-based specialized agents	Structured role collaboration	Project and workflow management
CAMEL [9]	Role-playing communicating agents	Communicative interaction	Collaborative problem solving
Multi-Agent Scheduling [17]	Agents representing scheduling participants	Negotiation and coordination	Meeting and calendar scheduling

IX. RAG And Knowledge Management

A. Knowledge Sources and Retrieval

Modern productivity systems often need access to information that is not contained within the language model itself. Such information may include project documents, meeting records, task descriptions, organizational knowledge, and other user-provided resources. Retrieval-Augmented Generation (RAG) addresses this limitation by retrieving relevant information from an external knowledge source and providing it to the language model as contextual information before generating a response [10].

The original RAG approach combines a language generator with a retrieval component to improve responses for knowledge-intensive tasks [10]. Subsequent research has expanded RAG into a broader framework involving different retrieval, indexing, and generation strategies [11]. In productivity applications, these mechanisms can allow AI systems to retrieve relevant project information before assisting with tasks, meetings, scheduling, or decision-making.

B. Grounded Assistance and Knowledge Sharing

RAG can improve the grounding of AI-generated responses by connecting language models with information sources that can be updated independently of the model. This is particularly useful in productivity environments where information changes frequently. For example, an AI assistant may retrieve relevant documents, previous meeting information, task details, or project records before generating a response.

Knowledge retrieval can also support collaboration by providing multiple users or agents with access to a common information source. In multi-agent environments, a shared knowledge layer can provide contextual information to specialized agents, allowing them to make decisions based on common project information. This creates a connection between RAG systems and multi-agent architectures, where retrieval can serve as a shared contextual foundation for agent coordination [7], [10], [11].



However, retrieval quality directly affects the quality of generated responses. If the relevant information is not retrieved, or if irrelevant information is provided as context, the resulting response may be incomplete or inaccurate. Therefore, effective productivity-oriented RAG systems require appropriate retrieval strategies, document organization, and mechanisms for maintaining contextual relevance.

C. Knowledge Governance Challenges

The use of external knowledge sources introduces important governance challenges. Productivity systems may process sensitive documents, project information, meeting records, personal data, and organizational knowledge. Consequently, access control, data privacy, information security, and appropriate management of stored knowledge become important considerations.

Another challenge is maintaining consistency between retrieved information and the current state of a project. Productivity environments often contain frequently changing tasks, schedules, documents, and meeting records, so outdated or conflicting information can lead to incorrect AI-generated recommendations. RAG systems must therefore consider the relevance, accuracy, and freshness of retrieved information. In collaborative environments, organizations may also require access-control mechanisms to determine which information can be accessed by specific users or agents. Human-centered AI research emphasizes the importance of designing AI systems that remain reliable, understandable, and controllable by their users [20].

For productivity systems, RAG should therefore be viewed as part of a broader knowledge-management architecture rather than only a retrieval-and-generation technique. Combining retrieval with access control, source verification, and human oversight can improve the reliability and trustworthiness of knowledge-grounded AI assistance. Such mechanisms are particularly important when productivity systems handle shared documents, project information, and other sensitive organizational knowledge.

The major components and applications of RAG in productivity-oriented systems are summarized in Table VI.

TABLE VI. COMPARISON OF MULTI-AGENT COLLABORATION APPROACHES

RAG Component	**Function**	**Productivity Application**	**Key Challenge**
Knowledge Sources	Stores relevant information	Documents, tasks, meeting records	Data organization
Retrieval	Finds relevant information	Context retrieval for user queries	Retrieval accuracy
Context Integration	Provides information to LLMs	Grounded task and project assistance	Context relevance
Generation	Produces natural-language responses	Summaries, answers, recommendations	Hallucination
Knowledge Governance	Controls information access	Secure collaborative knowledge sharing	Privacy and access control

X. Human-AI Collaboration, Trust, and Security

A. Human Oversight and Control

As AI systems become increasingly capable of performing planning, reasoning, and task execution, maintaining appropriate human involvement becomes an important design consideration. Productivity systems may assist users with decisions, generate schedules, summarize meetings, or execute actions through external tools. However, fully autonomous operation can introduce risks when the system misunderstands user intentions or performs an inappropriate action. Human-AI interaction research emphasizes the importance of designing systems that provide users with appropriate levels of control and allow them to understand and manage AI behavior [19].

Human oversight is particularly important in collaborative productivity environments because AI-generated actions may affect multiple users, shared schedules, project information, or organizational resources. A practical approach is to allow users to review important AI-generated decisions before execution while permitting lower-risk actions to be automated. This creates a balance between automation and user control and can help reduce the consequences of incorrect AI decisions.

B. Trust and Explainability

Trust is a critical factor in adopting AI-based productivity systems. Users are more likely to rely on AI assistance when they understand its capabilities, limitations, and reasons behind its recommendations. Human-AI interaction guidelines emphasize clear communication of AI actions, appropriate feedback, and user understanding [19].

In productivity applications, explainability is useful when AI prioritizes tasks, recommends schedules, assigns responsibilities, or identifies project risks. Instead of providing only a final recommendation, systems can highlight relevant factors such as deadlines, dependencies, availability, or historical patterns. Human-centered AI research further emphasizes reliability, safety, and human control as key characteristics of trustworthy AI systems [20].

Explainability remains challenging for LLM-based systems because their outputs can result from complex model behavior. Productivity systems should therefore provide clear, contextual explanations without making unsupported claims about the model's internal reasoning.

C. Privacy and Security

AI-driven productivity systems can process sensitive information such as tasks, calendars, meeting conversations, documents, project data, and communication records. The use of LLMs, external tools, and shared knowledge sources therefore introduces important privacy and security concerns.

Access control is essential in collaborative environments where users have different permissions. AI assistants should only access information authorized for the requesting user, while multi-agent and RAG-based systems must ensure that shared tools and retrieved data respect these permissions.

Privacy and security should be integrated into the system architecture. Human-centered AI emphasizes reliable, safe,



and trustworthy design [20], while human-AI interaction guidelines highlight appropriate user control [19]. Combining access control, secure data handling, and human oversight can improve trust in AI-driven productivity systems.

The major human-centered, trust, privacy, and security considerations for AI-driven productivity systems are summarized in Table VII.

TABLE VII. HUMAN-AI CONSIDERATIONS IN PRODUCTIVITY SYSTEMS

Dimension	Key Requirement	Productivity Example	Main Challenge
Human Oversight	User control over AI actions	Approving task assignments	Balancing automation and control
Trust	Reliable and predictable behavior	AI scheduling recommendations	Incorrect outputs
Explainability	Understandable AI decisions	Explaining task priorities	Complex model behavior
Privacy	Protection of personal information	Calendar and task data	Sensitive data exposure
Security	Controlled access to resources	Shared project information	Unauthorized access
Governance	Responsible AI operation	Monitoring autonomous agents	Accountability

XI. Comparative Analysis

A. Comparison Dimensions

AI-driven productivity systems can be compared across several dimensions, including the type of AI technique employed, degree of automation, support for personal and collaborative activities, scheduling capabilities, knowledge access, and interaction with external tools. These dimensions provide a common basis for evaluating systems ranging from traditional machine-learning approaches to LLM-based agents and multi-agent architectures.

Another important dimension is the level of autonomy provided by the system. Conventional AI-based productivity tools generally provide predictions or recommendations, whereas LLM-based agents can support multi-step reasoning and tool interaction [4]-[6]. Multi-agent systems further extend this capability by distributing responsibilities among specialized agents [7]-[9]. At the same time, higher autonomy introduces additional requirements for reliability, human oversight, privacy, and security [19], [20].

B. Cross-System Comparison

Existing research demonstrates a gradual shift from individual, task-specific AI capabilities toward more integrated and agentic productivity systems. Machine-learning approaches are effective for prediction and schedule estimation but generally depend on structured historical data [12]-[14]. Optimization-based methods provide strong support for constraint-based scheduling and resource allocation [15], [16]. However, these approaches may require structured inputs and may have limited ability to interpret natural-language user requirements.

LLM-based systems provide greater flexibility in understanding natural-language instructions and supporting reasoning, planning, and tool interaction [2]-[6]. Multi-agent frameworks extend these capabilities through role specialization and communication among agents [7]-[9]. RAG approaches provide access to external knowledge and can improve context-aware assistance [10], [11]. Nevertheless, these advanced approaches introduce challenges related to hallucination, coordination, retrieval accuracy, tool-use reliability, and autonomous decision-making.

From a productivity perspective, no single approach addresses all requirements effectively. A comprehensive system may require a combination of predictive models, optimization methods, LLMs, retrieval mechanisms, and agents. Such hybrid architectures can support personal productivity while also handling collaborative tasks, scheduling, meetings, shared knowledge, and workflow automation. However, integrating multiple techniques increases system complexity and makes evaluation, security, and governance more challenging.

Table VIII provides a comparative view of the major AI approaches discussed in this survey across key productivity-management capabilities.

TABLE VIII. COMPARATIVE ANALYSIS OF AI APPROACHES FOR PRODUCTIVITY MANAGEMENT

Approach	Task Management	Scheduling	Collaboration	Knowledge Access	Automation	Main Limitation
Machine Learning	Basic support	Basic support	Limited	Limited	Medium	Data dependency
Optimization	Limited	Strong support	Basic support	No	Medium	Structured constraints
NLP / LLMs	Strong support	Basic support	Basic support	Limited	High	Output reliability
LLM Agents	Strong support	Strong support	Basic support	Basic support	High	Tool-use and autonomy risks
Multi-Agent Systems	Strong support	Strong support	Strong support	Basic support	Very High	Coordination complexity



RAG-based Systems	Basic support	Basic support	Basic support	Strong support	High	Retrieval quality
Hybrid AI Systems	Strong support	Strong support	Strong support	Strong support	Very High	System complexity

XII. Taxonomy Of Existing AI-Driven Productivity Systems

A. Taxonomy Dimensions

AI-driven productivity systems can be classified according to several dimensions, including their primary application, underlying AI technique, level of autonomy, collaboration capability, and knowledge integration. From an application perspective, systems may focus on task and project management, scheduling and meeting management, or broader productivity assistance. From a technical perspective, they may employ machine learning, optimization, LLMs, retrieval mechanisms, or combinations of these techniques.

A second dimension is the degree of system autonomy. Traditional AI-based systems primarily provide predictions or recommendations, whereas LLM-based agents can perform reasoning, planning, and tool interaction [4]-[6]. Multi-agent systems further extend autonomy by distributing responsibilities among multiple specialized agents [7]-[9]. Knowledge integration provides another important dimension, distinguishing systems that operate primarily on predefined inputs from those that retrieve and use external information through RAG [10], [11].

Finally, productivity systems can be differentiated by their collaboration scope. Some systems primarily support individual users, while others support interactions among multiple users, agents, or organizational resources. This distinction is particularly important because collaborative productivity introduces additional requirements for coordination, shared knowledge, access control, and human oversight [17], [19], [20].

B. Taxonomy of AI-Driven Productivity Systems

Based on the surveyed literature, AI-driven productivity systems can be organized into five broad categories:

1. Predictive and Optimization-Based Systems: Use machine learning and optimization for schedule estimation, resource allocation, project planning, and constraint handling [12]-[16].
2. LLM-Based Productivity Assistants: Use large language models for natural-language interaction, task interpretation, reasoning, planning, and productivity assistance [1]-[6].
3. Agentic Productivity Systems: Extend LLM capabilities with planning, memory, tool use, and autonomous task execution [4]-[6].
4. Multi-Agent Collaborative Systems: Use multiple specialized agents that communicate and coordinate to perform complex workflows, including collaborative task management and scheduling [7]-[9], [17].
5. Knowledge-Grounded Productivity Systems: Integrate external knowledge through retrieval mechanisms to provide context-aware and grounded assistance [10], [11].

These categories are not mutually exclusive. For example, a modern productivity system can combine LLM agents with RAG and optimization methods, while a multi-agent system can use knowledge retrieval and scheduling components simultaneously. Therefore, the taxonomy should be viewed as a classification of dominant system characteristics, rather than rigid categories.

Based on the characteristics identified across the surveyed literature, AI-driven productivity systems can be classified according to their underlying AI techniques, primary capabilities, level of autonomy, collaboration scope, and knowledge integration. This classification helps organize the diverse approaches discussed in the previous sections and highlights the progression from conventional predictive and optimization-based systems toward LLM-based, agentic, multi-agent, and knowledge-grounded productivity systems. The resulting taxonomy is presented in Fig. 3.

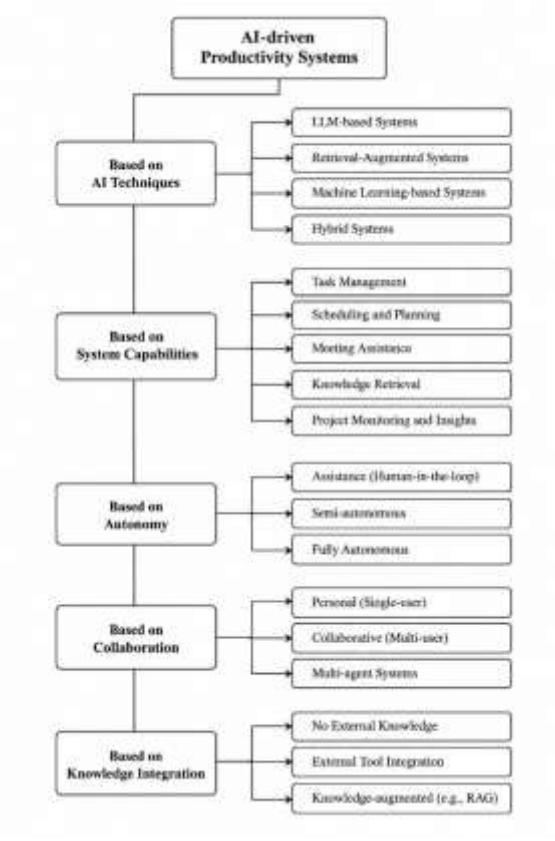


Fig. 3. Taxonomy of AI-driven productivity systems based on AI techniques, system capabilities, autonomy, collaboration, and knowledge integration.



XIII. Research Gaps And Critical Analysis

A. Technical and Evaluation Gaps

Although AI techniques have significantly improved productivity and task-management systems, several technical limitations remain. Machine-learning approaches can provide useful predictions for project schedules and delays, but their effectiveness depends heavily on the availability and quality of historical data [12]-[14]. Optimization-based approaches can handle explicit constraints effectively, but they generally require structured inputs and may have limited ability to interpret ambiguous natural-language requirements [15], [16]. This creates a gap between structured optimization methods and the flexible, language-driven interactions expected from modern productivity systems.

LLM-based agents address some of these limitations by supporting natural-language interaction, reasoning, planning, and tool use [4]-[6]. However, their outputs can be inconsistent and may contain incorrect or unsupported information. RAG can improve access to external knowledge, but retrieval quality, outdated information, and contextual relevance remain important challenges [10], [11]. Multi-agent systems introduce further issues, including communication overhead, conflicting agent decisions, coordination failures, and difficulties in evaluating the behavior of multiple interacting agents [7]-[9].

Another major gap is the lack of standardized evaluation for AI-driven productivity systems. Existing studies often evaluate individual capabilities such as scheduling, summarization, reasoning, or project management rather than evaluating an integrated productivity workflow. A comprehensive evaluation should consider task completion, scheduling accuracy, response quality, tool-use reliability, time savings, user satisfaction, and system robustness. Such multidimensional evaluation is necessary to determine whether increased AI capability actually produces measurable improvements in productivity.

B. Collaboration and Governance Gaps

Existing research also shows a gap between individual AI assistance and reliable collaborative productivity. Multi-agent systems demonstrate the potential for distributed task execution and coordination [7]-[9], while multi-agent scheduling provides evidence for automated coordination among participants [17]. However, collaborative productivity involves additional factors such as shared task ownership, changing responsibilities, conflicting priorities, permissions, and communication between users and AI systems. These factors are not consistently addressed within existing approaches.

Human oversight, trust, privacy, and security represent another significant research gap. As AI systems become more autonomous, users need mechanisms to understand, review, modify, and override AI-generated decisions [19], [20]. This is particularly important when AI agents can modify calendars, assign tasks, access organizational knowledge, or execute actions through external tools. Existing research provides important principles for human-centered and trustworthy AI, but integrating these principles directly into autonomous productivity workflows remains an open challenge.

Privacy and access control become even more important when RAG and multi-agent systems operate over shared organizational information. A system may need to determine not only what information is relevant, but also whether a particular user or agent is authorized to access it. Therefore, future productivity systems require stronger integration between AI reasoning, knowledge governance, authorization mechanisms, and human oversight.

Overall, the literature indicates a clear transition from task-specific intelligent functions toward integrated, autonomous, and collaborative AI systems. However, increased capability does not automatically result in a reliable productivity solution. Existing approaches frequently optimize individual components while leaving integration challenges unresolved. For example, an LLM may interpret a task effectively but generate an infeasible schedule, while an optimization algorithm may produce a valid schedule but lack sufficient contextual understanding of the user's intent.

The major research opportunity therefore lies in hybrid and context-aware architectures that combine the complementary strengths of machine learning, optimization, LLMs, agents, multi-agent coordination, and RAG. Such systems should also incorporate human-centered controls, secure knowledge management, and standardized evaluation. This direction can move productivity systems from isolated AI features toward reliable end-to-end assistance for both personal and collaborative workflows.

XIV. Future Research Directions

A. Adaptive and Reliable Productivity Agents

Future productivity systems should move beyond static task assistance toward adaptive agents that can continuously understand user goals, priorities, deadlines, workload, and changing contextual conditions. LLM-based agents provide capabilities for reasoning, planning, memory, and tool interaction, but their practical deployment requires improvements in reliability, consistency, and controllability [4]-[6]. Future research should investigate agents that can dynamically adjust task priorities and schedules based on changes in deadlines, user preferences, workload, and project status. Hybrid approaches combining LLM reasoning with machine learning and constraint-based optimization may further improve the reliability of planning and scheduling decisions [12]-[16]. Standardized evaluation frameworks are also needed to assess task completion, planning accuracy, tool-use reliability, time efficiency, and user satisfaction.

B. Collaborative and Multi-Agent Research

Future research should further explore multi-agent systems for collaborative productivity environments involving multiple users, tasks, projects, and organizational resources. Specialized agents can coordinate activities such as task management, scheduling, meeting assistance, knowledge retrieval, and project monitoring [7]-[9], [17]. However, communication overhead, conflicting objectives, coordination failures, and inconsistent decisions remain important challenges. Research should therefore focus on improved agent coordination mechanisms, role-based collaboration, conflict resolution, shared memory, and negotiation strategies. Integration of multi-agent systems with scheduling algorithms and knowledge-grounded approaches could enable more effective management of complex collaborative workflows.

C. Human-Centered and Secure Design

Future AI-driven productivity systems should place greater emphasis on human oversight, transparency, privacy,



and security. As systems become increasingly autonomous, users should retain appropriate control over important decisions and externally executed actions [19], [20]. Research should investigate explainable recommendations, user approval mechanisms, configurable levels of autonomy, and mechanisms for correcting or overriding AI decisions. Privacy-preserving approaches are particularly important because productivity systems may process personal schedules, task information, meeting records, documents, and organizational knowledge. Secure access control, reliable data governance, and responsible management of retrieved knowledge will therefore be essential for trustworthy deployment of AI-driven productivity systems.

XV. Discussion

A. Answers to the Research Questions

The survey indicates that AI-driven productivity systems employ a combination of machine learning, natural language processing, large language models, optimization techniques, retrieval-based methods, and agentic approaches. Machine learning is primarily used for prediction and estimation, while LLMs provide natural-language understanding, reasoning, planning, and content generation. Optimization techniques remain important for constraint-based scheduling and resource allocation, whereas RAG enables productivity systems to incorporate external and organizational knowledge. Thus, no single AI technique is sufficient for addressing the diverse requirements of modern productivity management.

For RQ1, the major AI techniques identified include machine learning, NLP and LLMs, optimization methods, LLM-based agents, multi-agent systems, and RAG. These techniques address different aspects of productivity and can be combined to provide broader capabilities.

For RQ2, AI techniques support task prioritization, deadline prediction, task allocation, scheduling, resource management, and project monitoring. Machine learning is particularly useful for prediction, while optimization methods are effective for handling deadlines, dependencies, and resource constraints. LLM-based approaches provide additional flexibility by interpreting natural-language requirements and supporting task planning.

For RQ3, existing systems support both personal and collaborative productivity. Personal applications commonly focus on task planning, scheduling, reminders, and personalized assistance, whereas collaborative systems address shared tasks, meetings, project workflows, and knowledge exchange. However, collaborative environments introduce additional challenges involving coordination, permissions, conflicting priorities, and shared information.

For RQ4, LLM agents extend productivity systems by combining reasoning, planning, memory, and tool interaction. Multi-agent systems further distribute responsibilities among specialized agents and support coordination for complex workflows. These approaches have significant potential for automating multi-step productivity activities, although reliability, coordination, and human oversight remain important concerns.

For RQ5, the major challenges identified across the literature include data dependency, scheduling complexity, unreliable LLM outputs, tool-use errors, retrieval quality, multi-agent coordination, privacy, security, and limited explainability. Increasing system autonomy also creates the need for appropriate human oversight and mechanisms for controlling or correcting AI decisions.

For RQ6, future research should focus on adaptive and reliable agents, hybrid AI architectures, improved multi-agent coordination, knowledge-grounded assistance, standardized evaluation, personalization, and human-centered security and governance. These directions can support productivity systems that are more reliable, context-aware, and capable of operating across both personal and collaborative workflows.

B. Implications for Practice

The findings suggest that practical AI-driven productivity systems should not rely exclusively on a single AI technique. Instead, combining complementary approaches can provide more effective support. For example, LLMs can be used for natural-language interaction and reasoning, machine learning can support prediction, optimization can verify scheduling constraints, and RAG can provide access to relevant organizational knowledge.

For collaborative environments, multi-agent architectures can provide specialized capabilities while allowing different tasks to be coordinated through shared information and controlled workflows. However, increasing automation should be accompanied by appropriate human oversight, permission management, transparency, and security mechanisms. Therefore, practical productivity systems should balance automation with user control, while maintaining reliability and protecting personal and organizational information.

XVI. Conclusion

Artificial intelligence is increasingly transforming productivity and task management systems from conventional organizational tools into intelligent systems capable of supporting planning, scheduling, collaboration, knowledge access, and workflow automation. This survey reviewed the evolution of AI-driven productivity systems and examined major approaches including machine learning, optimization, large language models, LLM-based agents, multi-agent systems, and retrieval-augmented generation.

The analysis shows that each approach provides distinct capabilities. Machine learning supports prediction and estimation, optimization methods address constraint-based scheduling and resource allocation, while LLMs enable natural-language interaction, reasoning, and planning. Agentic and multi-agent approaches extend these capabilities toward autonomous and collaborative workflow execution, while RAG provides access to relevant external and organizational knowledge. However, challenges related to reliability, contextual understanding, coordination, data quality, privacy, security, explainability, and human oversight remain significant.

The survey also highlights a shift from isolated productivity functions toward integrated, adaptive, and context-aware productivity systems capable of supporting both individual and collaborative workflows. Future systems are likely to combine multiple AI techniques rather than relying on a single approach. Hybrid architectures integrating LLMs, agents, optimization, machine learning, RAG, and human oversight provide a promising direction for achieving more reliable and comprehensive productivity assistance.



Overall, the research landscape indicates substantial potential for AI to improve personal productivity and collaborative work. However, achieving trustworthy and practical AI-driven productivity requires a balance between automation, intelligence, reliability, personalization, collaboration, and human control. Continued research in these areas can contribute to the development of more adaptive, secure, and human-centered productivity systems.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to Dr. S. P. Ingale, Department of Computer Science and Engineering, Prof. Ram Meghe Institute of Technology and Research, Badnera-Amravati, for his valuable guidance, continuous support, and constructive suggestions throughout this work.

Finally, the authors would like to thank their institution, colleagues, and everyone who provided support and encouragement during the completion of this survey.

REFERENCES

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[2] T. B. Brown *et al*., "Language Models are Few-Shot Learners," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[3] J. Wei *et al*., "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[4] S. Yao *et al*., "ReAct: Synergizing Reasoning and Acting in Language Models," in *NeurIPS 2022 Workshop on Foundation Models for Decision Making*, 2022.

[5] L. Wang *et al*., "A Survey on Large Language Model based Autonomous Agents," 2023.

[6] J. Sumers *et al*., "Cognitive Architectures for Language Agents," 2023.

[7] Q. Wu *et al*., "AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation," 2023.

[8] S. Hong *et al*., "MetaGPT: Meta Programming for Multi-Agent Collaborative Framework," 2023.

[9] G. Li *et al*., "CAMEL: Communicative Agents for Mind Exploration of Large Scale Language Model Society," 2023.

[10] P. Lewis *et al*., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[11] Y. Gao *et al*., "Retrieval-Augmented Generation for Large Language Models: A Survey," 2023.

[12] E. Taboada *et al*., "Artificial Intelligence Enabled Project Management: A Systematic Literature Review," *Applied Sciences*, vol. 13, no. 8, 2023.

[13] M. Nenni *et al*., "How Artificial Intelligence Will Transform Project Management in the Age of Digitization: A Systematic Literature Review," 2024.

[14] P. Koszykowski and L. Orzeszko, "Machine Learning in Project Schedule Creation: A Systematic Literature Review," *Journal of Scheduling*, 2025.

[15] R. Hartmann and A. Briskorn, "An Updated Survey of Variants and Extensions of the Resource-Constrained Project Scheduling Problem," *European Journal of Operational Research*, 2021.

[16] M. Ghasemi *et al*., "Review of Recent Advances on the Resource-Constrained Project Scheduling Problem," 2026.

[17] A. Garrido and K. Sycara, "Multi-Agent Meeting Scheduling: Preliminary Experimental Results," 1996.

[18] K. Kumar and M. Kabiri, "Meeting Summarization: A Survey of the State of the Art," 2022.

[19] M. Amershi *et al*., "Guidelines for Human-AI Interaction," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019.

[20] B. Shneiderman, "Human-Centered Artificial Intelligence: Reliable, Safe & Trustworthy," *International Journal of Human–Computer Interaction*, 2020.