

Article

Not peer-reviewed version

Autonomous Resources-Loaded Project Scheduling System Development

[Mazen A Al-Sinan](#)*, [Abdulaziz Bubshait](#), Zainab A Aljaroudi

Posted Date: 2 February 2024

doi: 10.20944/preprints202402.0139.v1

Keywords: autonomous systems; BIM; ChatGPT; construction; GPT; LLMs; machine learning; project scheduling



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

Autonomous Resources-Loaded Project Scheduling System Development

Mazen A. Al-Sinan ^{1,*}, Abdulaziz A. Bubshait ² and Zainab Aljaroudi ¹

¹ Procurement & Supply Chain Management, Saudi Aramco, Dhahran 31311, Saudi Arabia

² King Fahd University of Petroleum and Minerals, Dhahran 31261, Saudi Arabia

* Correspondence: mazen.alsinan@hotmail.com

Abstract: The fast-paced evolution of artificial intelligence (AI), machine learning (ML), natural language processing (NLP), large language model (LLM) applications, and generative pre-trained transformer (GPT) have set the stage for improved and autonomous project scheduling models. This study demonstrates how project scheduling can be enhanced by employing building information modeling (BIM) to predict reliable schedules for each project element. The research is based on design science research (DSR) methodology and focus group discussion. The proposed system consists of six components: project analyzer, data warehouse, activity identifier and sequencing, activity magnitude and direct cost calculator, activity duration calculator, and schedule analyzer. Unlike other solutions, the proposed solution sequences activities based on an ML model rather than constraint matrices. The proposed solution thus enhances project performance through proactive planning and risk management. Integrating the autonomous project scheduling solution with the other components of an autonomous project management system would surmount schedule challenges and would result in greater cost efficiency and fewer delays.

Keywords: autonomous systems; BIM; ChatGPT; construction; GPT; LLMs; machine learning; project scheduling

1. Introduction

Project scheduling require knowledge and expertise of the activities, resources, and equipment required to complete a project. In addition, a rapid response to delays, malfunctions, and any other problems that might arise is necessary for prompt, fruitful alterations to schedules. Creating a schedule that considers existing plans and the possible need for new plans is a task that artificial intelligence (AI) can be trained to manage extremely well. Through supervised learning, AI can be repeatedly fed reliable information to generate a new collection of terms, or a dictionary, for different activities, jobs, and equipment in the field. These data would be the foundation on which schedules would be built. Furthermore, previous schedules, their success rates, and their problems can be used as inputs to further improve the accuracy of new schedules. The process of learning and prediction should be repeated and tested until a near-perfect algorithm is built.

Although vast amounts of research on the employment of AI in various aspects of project management have been conducted, research on construction practical applications is scarce [1]. This could be due to the nature of the construction industry, which has always resisted integrating the latest technologies and changing the status quo in terms of its practices [2]. Its reluctance toward change is further intensified by other deterrents, including costs, difficulty adopting new technologies, and fragmentation of the industry. As a result, the success of integrating AI in the construction industry in general and in project management in particular depends on the technology providers who should develop user-friendly, affordable solutions. AI has the power to improve every facet of ESG score and transform the conventional construction industry into one that is more socially responsible, ecologically sustainable, and technologically advanced.

This study attempts to explore the potential of machine learning (ML), large language models (LLMs) (e.g., ChatGPT), and AI-powered search tools for developing autonomous solutions, mainly for applications where textual and real-time data extracted from the web are core components of the process. The provided blueprint demonstrates how to convert building information modeling (BIM) elements into a set of construction activities that can be identified, classified, and sequenced. Each activity will initially be allotted a duration originating from 4D BIM models and then refined through further supervised ML.

While the development of autonomous systems in general has been widely studied, this study focuses on autonomous project scheduling and management systems powered by AI. In particular, the authors focus on the development of autonomous systems paired with construction schedules. The ethical and economic implications [3] of autonomous systems are not within the scope of this research. The proposed method is generic while all the previously proposed methods in the literature have been project-specific and difficult to replicate in other projects.

The paper is organized as follows: Section II discusses the methodology. Section III reviews the literature and provides relevant background information, focusing on the concepts of AI, ML, BIM, natural language processing (NLP), LLM, and generative pre-trained transformer (GPT). Section IV describes the proposed blueprint for building an autonomous construction project scheduling system. Finally, Section V concludes the study.

2. Methodology

The research strategy was based on design science research (DSR) methodology, a problem-solving paradigm aimed at developing, evaluating, and employing new artifacts to address practical problems. Typically building on existing solutions [4–6], the DSR adheres to the following guidelines [5]:

- (1) Design as an Artifact: The production of an artifact in the form of a construct, model, method, or instantiation;
- (2) Problem Relevance: The development of technology-based solutions to important and relevant business problems;
- (3) Design Evaluation: The rigorous demonstration of the utility, quality, and efficacy of a design artifact;
- (4) Research Contributions: The clear and verifiable contribution to the areas of the design artifact, design foundations, and/or design methodologies;
- (5) Research Rigor: The scrupulous application of methods in both the construction and evaluation of the design artifact;
- (6) Design as a Search Process: The utilization of available means to reach desired ends while satisfying laws in the problem environment; and
- (7) Communication of Research: The effective presentation of the research, both to technology-and management-oriented audiences.

The product of the first two guidelines of the DSR is the subject of this study, while the paper itself is a meticulous execution of the third and fourth guidelines. Section IV follows the fifth guideline in its examination of the proposed artifact, the artifact's technical shortcomings, and the remedies required for successful implementation of the proposed solution. The proposed artifact employs the latest relevant technology—an existing and accessible resource—that satisfies the sixth guideline's demands. Finally, in line with the seventh guideline, this paper and the proposed artifact are presented in a comprehensible manner for both subject matter experts (SMEs) and non-technical laypeople.

To arrive at the proposed artifact, the researchers conducted a literature review of relevant papers. These papers investigated problems in autonomous construction project scheduling. Concentrating on the latest advancements in AI, this research explores the potential of new

computational capabilities and advanced AI algorithms. The authors aimed for a solution that could be built and commercialized as an initial step toward building an autonomous construction project management solution.

3. AI in the Literature

The following subjects are presented in this section: BIM and 4D BIM, AI, web scraping, autonomous systems, and autonomous construction project scheduling.

3.1. BIM and 4D BIM

BIM is a collaborative process that the architecture, engineering, and construction (AEC) sectors employ to create, plan, design, and construct buildings. Contrary to popular belief, BIM does not refer to a single software program or application. Instead, BIM is a set of tools, technologies, and processes that enable collaboration between AEC personnel within a shared 3D model. The process is divided into four main stages: planning, designing, building, and operating. During the planning stage, the 3D visual model of the building is created using real-world data. The 3D model components store comprehensive information about an object, such as its geometry, spatial relationships, and material properties. When any component is altered in the model, the BIM software updates other components accordingly to maintain a building's consistency. Moreover, collaborators can simultaneously edit the data and track each other's modifications in real time. The second stage tends to be the most complicated, as team members begin conceptual design, analysis, and documentation. BIM facilitates this stage through improved communication and coordination, increased efficiency, time reduction, and quality control. Furthermore, BIM tools can be used to assess and identify problems within a design, a structure, and the sustainability of a building, as well as to recommend solutions and improvements. The common data environment (CDE) bridges the time between the second and third stages, where construction is guided by the model. Finally, the operational stage depends on the model for the future renovations or demolitions of a building.

Another dimension, time, is added to 3D BIM to create 4D BIM. With 4D BIM, the construction sequence becomes part of the information stored in the model. The software helps to create and refine a project's schedule based on individual components and their designs. Consequently, the project seamlessly transitions from the second to the third stage of the process.

All BIM projects, regardless of dimension, adhere to the Industry Foundation Classes (IFC) format. The IFC format ensures the interoperability of data among various BIM software programs [7].

Researchers were quick to begin studying AI in coordination with the 4D BIM process [8]. Kim et al. [30] proposed a framework for automating schedule generation from BIM, where the sequencing rules are predefined. Hong et al. [9] proposed graph-based automated scheduling (GAS) to automatically store the best practices of past projects and reuse them to generate a new project schedule. Faghihi et al. [10] proposed a method for retrieving information from BIM to develop construction sequencing using a genetic algorithm (GA). Faghihi's method uses a rule-based approach in which the dependencies of project elements are predefined in the constructability constraint matrix.

3.2. Artificial Intelligence

In the context of autonomy, AI is defined as "a system's ability to correctly interpret external data, to learn from such data, and to use those learnings to achieve specific goals and tasks through flexible adaptation" [11]. This is an all-encompassing definition that covers all of AI's general abilities, whereas AI's specific applications in the field of autonomic scheduling are discussed below.

Figure 1 shows the subfields that fall under AI.

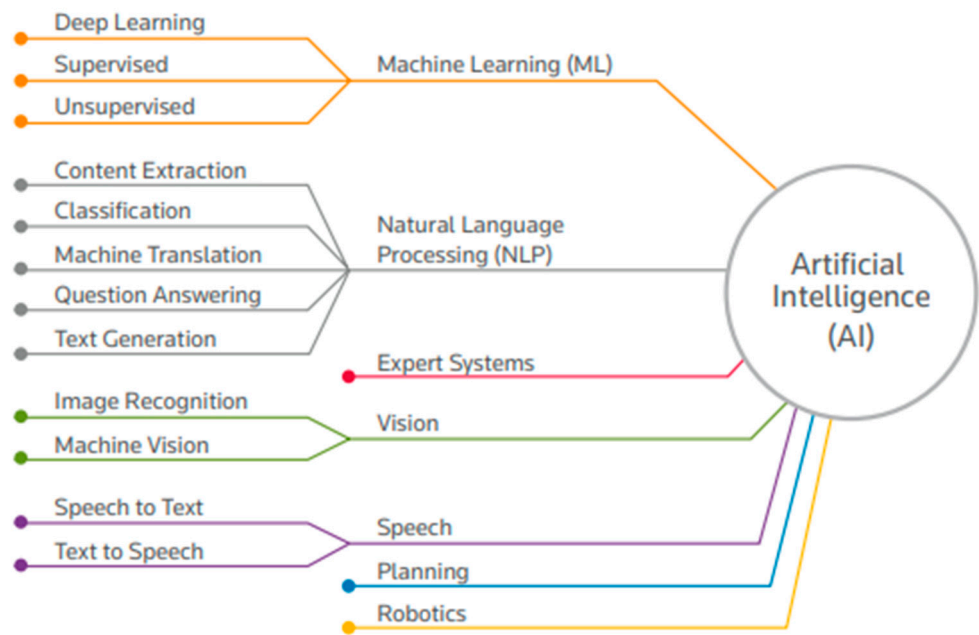


Figure 1. Artificial intelligence (AI) subfields [12].

3.2.1. Machine Learning.

From the 1960s until the emergence of ML, AI was mainly a rule-based expert system (if-then rules) that can process a defined problem in a particular domain [13]. ML, an algorithm that processes large amounts of data to generate patterns (e.g., classification and categorization), has expanded AI’s potential applications. ML algorithms are fed data from which they learn by running statistical and probabilistic analyses. The algorithm then extrapolates possible outcomes based on the results of the analyses.

ML became a hot topic after 2010 for multiple reasons [14], the first of which is the availability of big data, an essential breakthrough that exponentially increased the amount of training data available for learning models. Second, the increase in and affordability of computing power allowed algorithms to run on GPUs instead of CPUs. This contributed to the global spread of ML, as GPUs power increased the number of training processes by 10 to 20-fold. Third, advanced ML algorithms that are more flexible, robust, and capable of handling various types of problems have become available publicly. Finally, the creation of open-source code libraries provided a platform for developers to collaborate and build upon each other’s contributions. Figure 2 shows the fundamental differences between classical programming and ML.

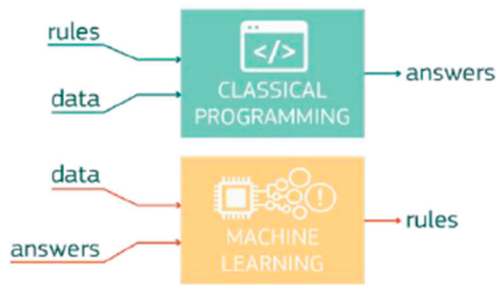


Figure 2. The difference between classical programming and machine learning (ML) [15].

There are three main ML techniques, distinguished by the method of learning [15,16], as described below.

1. **Supervised learning:** In supervised learning, a set of data points, called training data, and corresponding labels are analyzed by the model. The model identifies similarities between data points and patterns. Afterward, the model is fed with unlabeled data to be classified based on what the model learned. This method is best suited for data classification (discrete data) or regression (continuous data). Running the model repeatedly improves its performance, as the previous outcome of classifying unlabeled data is used as training data. Face recognition and spam detection programs employ supervised ML.
2. **Unsupervised learning:** By clustering data without labels, regardless of the number of classes known, the model focuses on understanding data patterns and relationships rather than classification. This method is best suited for clustering (fitting data into discrete groups) and density estimation (numeric estimate to fit).
3. **Reinforcement learning:** In reinforcement learning, a computer program or agent interacts with an environment and learns to act within that environment. The model attempts to improve the outcome through cumulative reward (capable of adjustment). For example, robots use reinforcement learning to enhance accuracy in movement. Autonomous vehicles and video games also use reinforcement learning for development.

Some researchers occasionally refer to a fourth ML technique called deep learning, a semi-supervised method in which a large mix of labeled and unlabeled data is used in training and analysis [14]. In this technique, a special neural network imitates the human learning process by learning by example. One of the most popular deep learning neural networks is the convolutional neural network (CNN). Deep learning models can directly classify images, sounds, or texts with high accuracy, outperforming human beings [16].

3.2.2. Natural Language Processing

NLP is a model's ability to process human speech and text and simultaneously understand the associated meaning [17]. Computational linguistics, rule-based modeling of human language, ML, and deep learning techniques are used in NLP development. The understanding of human language involves many variables for NLP models:

1. **Speech to text:** Speech to text converts spoken data into written data. Analyzing human speech can be quite difficult, as grammar, accents, and tone of voice differ greatly among people.
2. **Grammatical tagging:** Grammatical tagging is the process of identifying the part of speech to which a word belongs. Whether a word is a verb, noun, adjective, adverb, or other part of speech can change the entire meaning of a sentence.
3. **Word sense disambiguation:** Because some words can have multiple meanings, word sense disambiguation is the process of choosing the definition of a word that best fits the context. This adds to the amount of information models must learn.
4. **Sentiment analysis:** Sentiment analysis controls a model's ability to understand the sentiment, intent, emotion, and attitude behind the data. Through sentiment analysis, models can determine the collective meaning of the data after establishing the individual meanings of the words.

These variables require immense and repetitive training for accurate interpretation, which then contributes to the accuracy of the model's generated outcomes. NLP applications include spam detection, text mining, information retrieval, language translation, Q&A systems, and dialogue systems (chatbots) [18].

3.2.3. Large Language Models

LLMs are a type of NLP model that have substantially higher number of parameters than other NLPs. Parameters are the attention weights and biases that a model learns during training. Large volumes of text are used as training data to ensure that models learn the relationships between words. The model identifies words that are statistically more likely to appear after one another and uses this

pattern to generate human-like text. Using deep learning techniques, LLMs are equipped with a transformer to generate more accurate results [19]. Transformers are a type of neural network suitable for processing sequential data, such as text, due to their “self-attention” mechanism. The self-attention mechanism allows the model to divide the input data into smaller segments and dynamically assign them an attention weight based on importance and relevance to each other. During training, the model refers to these attention weights to determine the order of words in the generated text. The process of predicting the next word is known as language modeling. Usually, the model is fine-tuned to perform a specific task, such as answering a question or translating a language [20]. LLMs are classified by their parameters and the amount of training data sampled. As with any ML model, the more training data fed and the more parameters used, the more accurate the response the model will generate.

3.2.4. Generative Pre-Trained Transformers

GPTs are a group of LLMs developed by OpenAI, an AI organization and deployment company. GPT models produce natural language texts based on input prompts, like other LLMs, but they have transformed beyond that. With four versions released since 2018, GPTs have been trained using data from the *entire* open internet, creating their own library of training data and outshining previous LLMs [19]. The latest version of the GPT models is ChatGPT, which can now answer text in a conversational style, solve problems, generate short and long text data upon request (e.g., blogposts and essays), edit or summarize data, translate different languages, and even create or respond to visual art. The model is now used globally by ordinary people and has become an extremely popular tool.

GPT models have revolutionized the world of AI and extended the limits of LLMs in the workforce. With 175 billion parameters and 500 billion weights, GPT-3 was the last model in the GPT group to have its parameters and weights published. Since then, OpenAI has reversed its open code policy and kept ChatGPT's code to itself. Despite the astronomical number of parameters and training data, GPT models have not yet perfected human thinking. Bias from training data, inability to reason like humans, lack of common sense, and problems with deterministic answers are some of ChatGPT's biggest weaknesses, as the algorithm of the model is probabilistic in nature [17,21].

3.3. Web Scraping

Web scraping refers to the process by which a program extracts various data from the web [22]. Previously a manual job, AI has automated the process through ML, NLP, and computer vision. Computer vision is a branch of AI dedicated to understanding the content and meaning of visual data, allowing models to include data from infographics, videos, images, and other multimedia sources. After a web scraper is given a URL, it loads the HTML, JavaScript, and CSS elements of the entire page. Next, the scraper will collect all the data, or the requested data based on its programming and output the data into spreadsheets, CSV, or JSON formats [23]. AI gave web scrapers the ability to scour unstructured data, unlike manual web scrapers, which were limited to structured data. ML enables the perfecting of the scraper's accuracy in targeting the desired information and understanding it, as the scraper repeats the learning and training processes. However, integrating AI with web scraping has created an ethical dilemma in which the scraper might scan data that a website states are confidential. Thus, web scrapers must be manually programmed to respect the boundaries set by websites and companies to ensure ethical scraping [24].

Furthermore, AI-powered web scrapers have become a common tool business use to collect and analyze data, draw conclusions, and make informed decisions. Examples of web scrapers include Google's Search Generative Experience (SGE), a search engine that integrates generative AI-powered results into the Google search engine query response [25]. Whereas Google's traditional search engine depends only on data indexed and ranked, SGE uses AI to supplement conventional search results. As web scrapers become more advanced with AI, their applications in the workforce become immeasurable, limitless agents for ensuring the success of businesses, companies, and researchers.

3.4. Autonomous Systems

Systems that operate to collect data and achieve a certain set of goals in a changing environment without human intervention or control are classified as autonomous systems. The two main roles of autonomous systems are management and operation automation. Bolstered by AI, autonomous systems not only collect data but also make intelligent decisions on their own based on training and repetitive ML. For example, robots and driverless cars depend on visual data that autonomous systems have been trained to perceive and evaluate. Both cases, however, require precise risk assessment and cautious decision-making, holding the automotive and robotic industries back from creating fully autonomous products to minimize dangers and risks. For businesses, tasks usually depend on textual analysis and data retrieval for decision-making, both of which are special abilities that AI has honed and perfected. The automation of these tasks results in a more efficient and instantaneous work environment. It is worth mentioning that there is a significant difference between “automation” and AI. While automation depends on predefined rules and preprogrammed logic, AI attempts to simulate human thinking [26].

3.5. Autonomous Construction Project Scheduling

The benefits of the autonomous systems explained above could extend to construction project scheduling. In addition to BIM, various algorithms and AI tools, such as case-based reasoning (CBR), knowledge-based systems, GAs, expert systems, and neural networks, have been considered for automating schedules [27].

El-Menshawy et al. [28] proposed a model for automated schedule generation from BIM, where the output is exported to Primavera. The user can then modify the generated schedule. In addition, they proposed an optimization model that provides a near-optimum solution resulting from various generated scenarios.

Text-based automation is another approach to automatically generating a project schedule, where the focus is on analyzing the textual data of the schedule. For example, CBR involves a text base in which an old, similar case is recalled to the present case and used to suggest a solution to the new problem [29]. Researchers have recently explored using NLP to analyze descriptions of construction schedule activities [30].

Amer et al. [31] proposed an ML transformer language model to learn the implicit dependency constraints and the flexibility of construction schedule relationships. Alikhan et al. [32] proposed training a deep learning model on historical project schedules to predict sequential activities. The model employed bidirectional long short-term memory (LSTM) recurrent neural networks (RNNs) to learn the activity predecessors in the forward direction and the activity successors in the backward direction. Amer et al. [33] proposed an ML-based method using LSTM-RNNs, where construction sequences extracted from previous project schedules were used as training data. The model learned potential successor activities based on a sequence of previous schedule activities.

A review of the literature revealed a single paper that addressed the use of ChatGPT in scheduling construction projects [34]. The study assessed the capability of ChatGPT to generate resource-loaded project schedules, revealing that ChatGPT is promising but needs to be explicitly trained for this task to generate acceptable results [34]. The authors believed that ChatGPT could be essential to building an autonomous construction project scheduling solution but not the main backbone for the solution. Additionally, ChatGPT could play a more effective role when an autonomous project management system is built based on extracting recommendations and requirements from big data to address ad hoc situations.

4. Autonomous Construction Project Scheduling System Development

Any autonomous project management solution should imitate the role played by the project management team in terms of meeting the project schedule within the budget and per the specifications. Figure 3 shows a schematic diagram of an autonomous project management system. Accordingly, the authors consulted a focus group of experienced project managers and schedules to

identify the imperative tasks that an autonomous project management system should execute. The focus group concluded that any autonomous construction project management system should be capable of performing the following tasks with minimum human intervention, if any:

1. Develop a schedule autonomously from project specifications, drawings, and contracts.
2. Assign the duration and resources required (manpower, equipment, and materials) for each activity.
3. Determine the cost of each activity.
4. Optimize project costs and schedules by analyzing the assigned resources (manpower, equipment, and material).
5. Monitor the progress of the project (schedule and cost).
6. Monitor quality by ensuring that applicable engineering standards, safety, codes, regulations, and other environmental requirements are addressed.
7. Provide a recourse plan in case of delays or overruns in the budget.
8. Trigger purchase orders for materials and procure third-party services (subcontractors) in a timely manner.
9. Trigger requests for inspection.
10. Trigger payments upon completion of a milestone or on due dates.
11. Provide technical recommendations to solve ad hoc technical troubles, such as when an element fails to meet the minimum bearing test.
12. Trigger a request for a change order if there is an error or conflict between specifications and design drawings.
13. Issue an autonomous progress report to stakeholders (e.g., client).

The proposed autonomous construction project scheduling system should provide a solution for the first four tasks listed above. Project scheduling is one of the most significant activities affecting a project's success and requires a high degree of experience and intelligence. Accordingly, conducting project planning autonomously may foster the development of the remaining components of an autonomous project management solution.

The proposed solution is cloud-based, allowing multiple users to access it. Having multiple users is important in the retrieval of real-life data for training data; this ensures the model is regularly updated, thus improving its performance.

Users should be able to integrate the solution into their enterprise resource planning (ERP) system. This might entail reconfiguring some of the existing databases to make them compatible with the solution. If users do not have an ERP, they can manually upload some of their data instead. User data, especially those related to human resources and equipment, are essential for developing a resource-loaded project schedule. Moreover, real-time data on these resources aid the solution in identifying the activities that must be subcontracted.

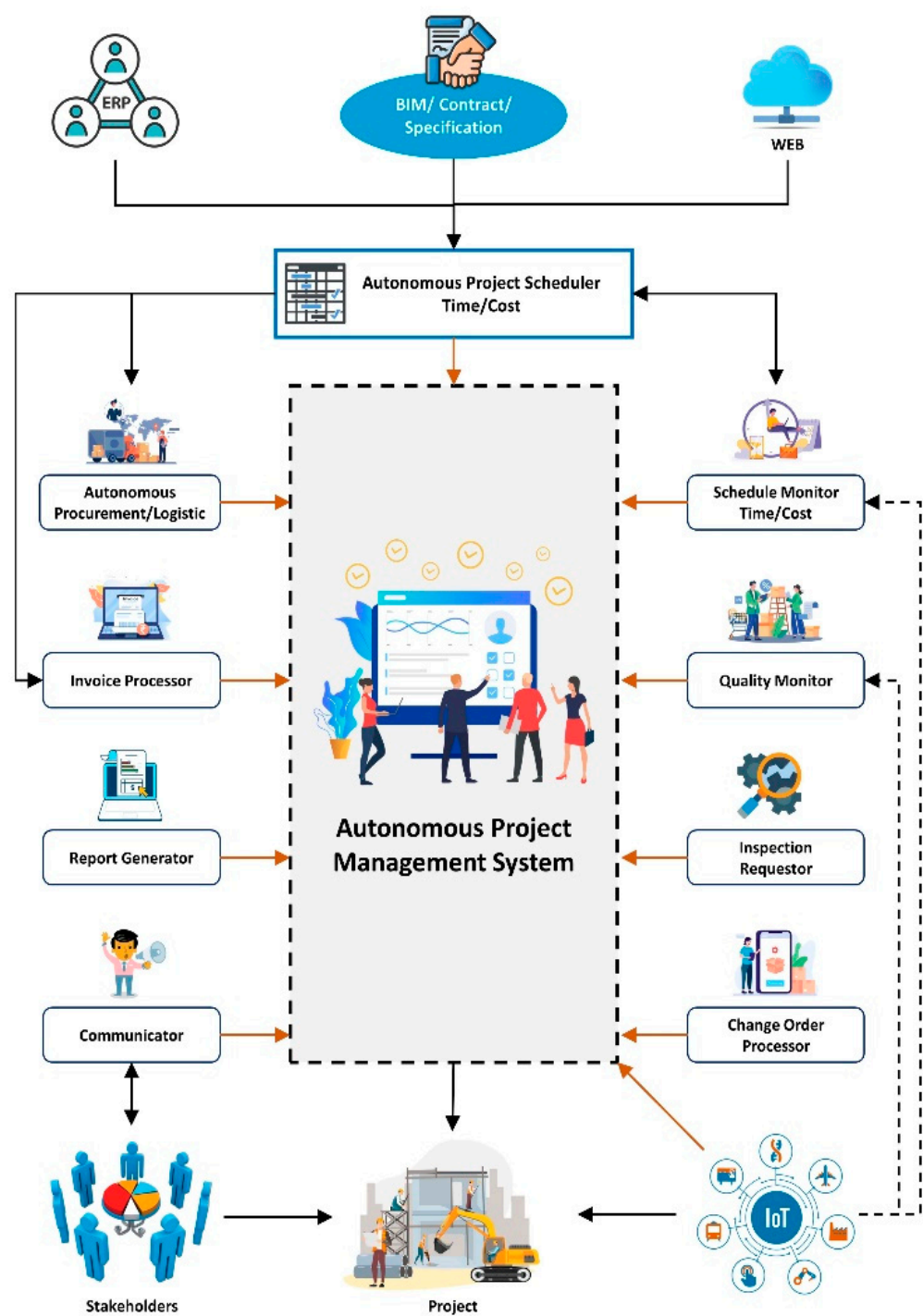


Figure 3. Autonomous Project Management System.

4.1. Autonomous Project Scheduling System Structure

Generating a resource-loaded schedule requires identifying project activities, sequencing them, assigning resources, and determining the size, cost, and duration of each activity according to the assigned resources. The proposed solution would employ BIM data files to determine scheduled activities. Unlike other solutions, the proposed solution would sequence activities based on an ML model rather than constraint matrices. Accordingly, separate iterations of training should be conducted for each type of project (e.g., multi-story buildings or roads) to improve the system's

performance. Ultimately, the solution should be able to generate a resource-loaded schedule for any project type after training. For demonstration, the proposed solution assumes that the project is a multi-story building. The proposed solution would implicitly develop the constraints within the model during the training. This could provide more flexibility to accommodate unique designs. A diagram of the proposed autonomous project scheduling solution is shown in Figure 4.

The proposed solution consists of the following components:

1. Project analyzer
2. Data warehouse
3. Activity identifier and sequencing
4. Activity magnitude and direct cost calculator
5. Activity duration calculator
6. Schedule analyzer (project actual performance)

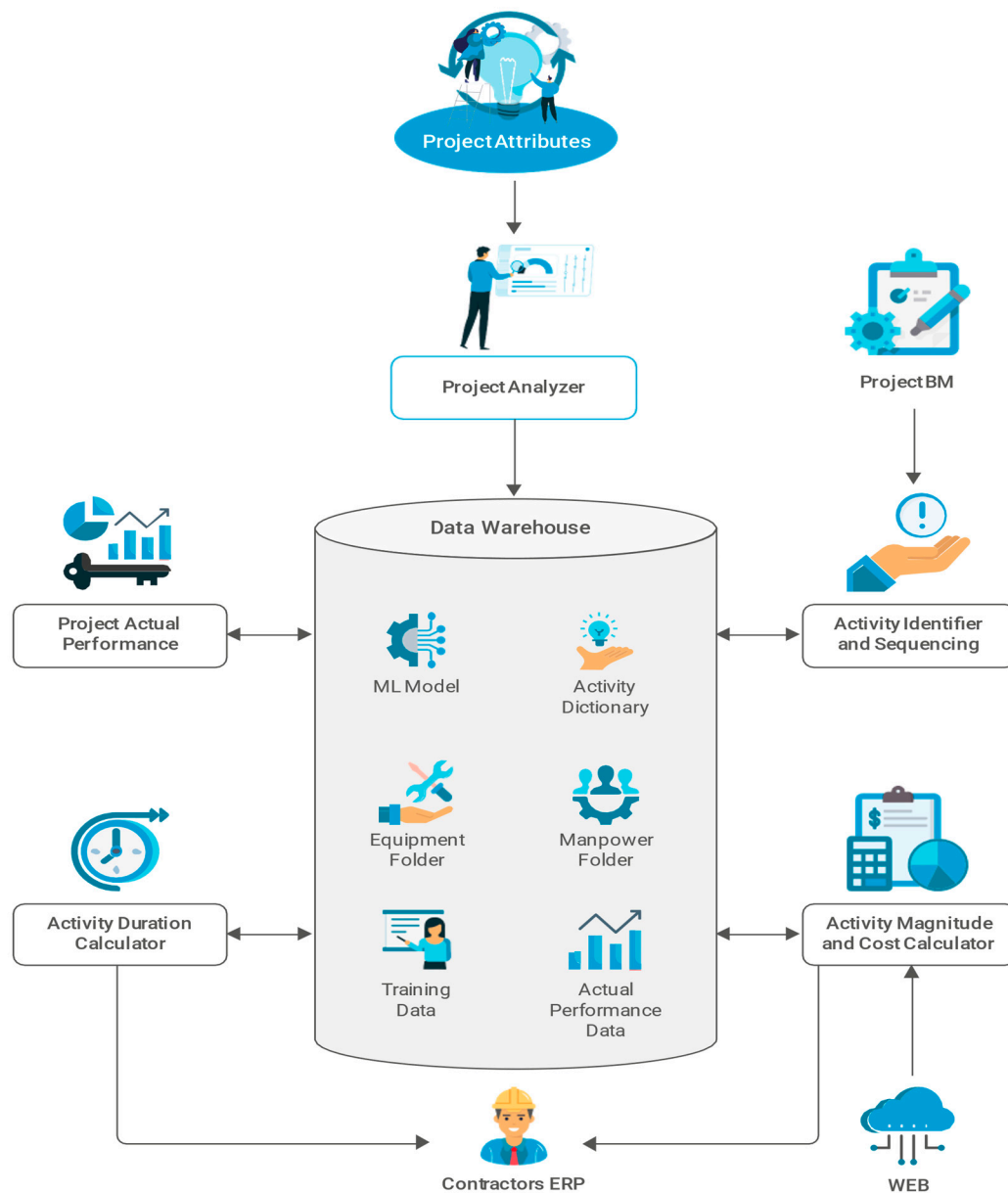


Figure 4. Autonomous Project Scheduling System.

4.1.1. Project Analyzer

The project type (e.g., commercial, industrial, or residential), size, and location (offshore or onshore) affect the entire schedule development. Initially, users should enter the general attributes of the project into the project analyzer. The project attributes could include the project’s budget, duration, applicable standards, and the maximum subcontracting percentage permissible. Using an LLM like GPT, the solution should then process the scope of the work and identify the project type accordingly. The algorithm should be trained using previous contract documents to identify the project type and other project-specific attributes. However, certain project attributes might not be included in contract documents, such as the project budget. Thus, manually structuring and labeling some of the training data might be necessary.

4.1.2. Data Warehouse

- The data warehouse consists of the following folders:
- Activity Master Data Folder (Activity Dictionary)
- Equipment Directory Folder
- Manpower Directory Folder
- Training Data Repository
- ML Model
- Actual Performance Data
- Activity Master Data Folder (Activity Dictionary)

To develop an efficient project schedule using an autonomous solution, it is essential to standardize project activities. Universal project activity master data (UPAMD) is an advanced, expanded construction elements dictionary. Since the most comprehensive source for identifying activities is 3D drawings, these elements would be based on BIM IFC models, with each element regarded as a standalone activity. The project type and contractor’s resources were also considered and addressed during the development of the UPAMD. Additionally, an identifier is assigned to each element based on its location. For example, in a multi-story building, there would be a unique code for each element on the first floor, second floor, and so on.

Each activity (construction element) will have a file with an assigned code that provides a description of the activity and its characteristics. There could be multiple versions of the same activity, which would require multiple files with different codes. The activity file could contain a certain number of fields, as shown in Table 1.

Table 1. Dictionary Activity File Contents.

No.	Field	Description
1	Activity Code	Each activity type assigned a specific code
2	Activity Name	Shows the activity title, such as “Painting”
3	Activity Type Description	Provides a general description to define the activity
4	Required Equipment	Lists the equipment and tools required to perform the activity
5	Required Worker Skills	Lists the work skills required to perform the activity
6	Required Materials	Lists the materials required to perform the activity
7	Measurement Unit	Identifies the measurement unit based on the nature of the activity (e.g., square meter or cubic meter)
8	Hard and soft logic	If hard and soft logic are both required, an activity code is shown here.
9	Workers’ productivity	Indicates the number of manhours of each worker skill required to perform one measurement unit
10	Productivity of Equipment	Displays the number of equipment hours to complete one measurement unit

11	Required Material Quantity Per Measurement Unit	For example, the number of gallons of paint per square meter
12	Safety Requirements	Applicable safety requirements
13	Quality Test	Applicable quality test
14	Special Requirements	Reports on special requirements

Developing an activity master data folder (dictionary) is a prerequisite for conducting system training. Developing an activity dictionary is a demanding task in which GPT can assist. For a cloud-based solution where many users are going to utilize the solution, the time and cost to build an activity dictionary should justify such efforts.

Equipment Directory Folder

For each piece of equipment available on the market, a file with several fields is created where the equipment information can be retained. Table 2 shows an example of a piece of equipment file.

Table 2. Equipment File Contents.

No.	Field	Description
1	Equipment Code	Each piece of equipment assigned a specific code
2	Equipment Name	Name(s) of the piece of equipment
	Equipment Description	Brief description of equipment
	Equipment Manufacturer	The name and address of the manufacturer
3	Equipment Uses	All activities in which the equipment could be used are listed as activity codes. This field should feed the required equipment field in the activity master data dictionary.
4	Equipment Lease Cost	Indicated daily or hourly lease cost; this could be either scraped from the web (in real time) or from a vendor site. The rate in this field should feed project schedule activity equipment costs.
5	Required Operators	The job code and number of required operators
6	Energy Type	Diesel, gasoline, or electrical energy sources indicated
7	Energy Consumption	The average consumption of energy per hour; this field could be used to adjust the activity cost during construction when the cost of energy changes.
8	Equipment Capacity	The standard capacity measurement of the equipment (e.g., 1 horsepower for a compressor)
9	Equipment Model	Specifies the model available; this field could be used for analysis.
10	Equipment Operation Cost	This cost is extracted either from the web or as an input from the manufacturer.
11	Equipment Weight/Dimensions	Equipment weight or dimensions shown

The original equipment manufacturer (OEM) could be granted temporary access to create a file for any piece of equipment. In the case of contractor-owned equipment, there should be an equipment database using a file structure like that used in the equipment directory folder. Additional fields could be added to capture information related to location, amount of available equipment, maintenance costs, preventive maintenance schedule, and other relevant information, as applicable. The proposed solution prioritizes utilizing owned equipment to optimize scheduling over leasing additional equipment.

Manpower Directory Folder

A universal folder with the job titles of all those working in the AEC industry will be created with a file for each job title. Table 3 shows an example of a job file, along with possible fields.

Table 3. Job File Contents.

No.	Field	Description
1	Job Code	Each job assigned a specific code
2	Job Title	Each job assigned a specific title
	Job Description	A brief description of the job
3	Job Requirements	The minimum qualifications and years of experience required for the job
4	Activities to be Assigned	All activities to which this job can be assigned to; this field should feed the manpower requirements field in the activity database dictionary.
5	Hourly Rate	This could be extracted from the web or ERP and fed to the activity cost in the project activity cost.

Contractors should have a digital human resources system with files similar to those found in the manpower directory folder. Each job title could have additional fields to capture further information, such as the available headcount. A subfile could also be created for each worker to capture historic productivity from previous projects, availability schedules, and other qualifications. The contractor might have limited available human resources, with a need to either hire temporary workers or subcontract certain activities.

Training Data Repository, ML Model, and Actual Performance Data Folders

All training data, including contract documents (contracts, specifications, or drawings from BIM) with their developed schedules, should be uploaded to the training data repository. In addition, schedules developed by the system and the actual project’s progress after project execution should be retained in the training data repository and used as training data to improve the future system’s performance. For each completed project, a record of the estimated resources, costs, and duration of each activity compared to the actual duration and cost after completion should be stored under Actual Performance Data for analysis and further model training.

4.1.3. Training Data Preparation

To develop the proposed solution, the training data must include all of the elements to be expected in a generated schedule. The following processes can provide near-perfect training data: data preparation, feature extraction, activity classification, activity and relationship recognition, and sequencing. Because manually developed schedules and BIM models lack the terminology and depth required for the universal standardization of activities, data preparation is a crucial step. During data preparation, activities in previous schedules and BIM models are renamed using the activity dictionary, schedules are refined, and errors are cleared. The next steps will be completed using AI for greater efficiency.

Important features, such as object geometry, dimensions, materials, and relationships, will be extracted from the BIM model. Then, the AI model will use these features to inform object detection and classification. Using deep learning techniques, a neural network that can identify and classify BIM components (e.g., walls, doors, and windows) within the BIM 3D model will be created. Once they are classified, the AI model needs to recognize how these components relate to specific construction activities. NLP will be used to generate descriptions for each BIM component based on their attributes, relationships, and contexts. These descriptions should be formulated in a way that aligns with the language used in the predefined UPAMD to link the components with the required activities. Finally, the model will determine the logical sequence of activities based on the relationships between the BIM components.

4.1.4. Model Training

After identifying construction activity dependencies during data preparation, the model can finally be trained. The refined schedules and BIM files will be used as inputs to generate an ensemble model that is fully capable of autonomous scheduling. Ensemble modeling is a more effective method than using predefined sets of rules or structural stability rules for identifying project activity dependencies [7,35]. The training process entails developing a schedule for each BIM model, in which each structural element is considered a standalone activity. The resulting output will be a comprehensive, thorough schedule that does not require monitoring from project managers. Any changes or delays in the project will automatically trigger the creation of a new schedule. Each type of project would require separate training and a separate model rather than one generic model to create multiple models rather than one generic model.

4.1.5. Activity Identifier and Sequencing

After feeding the solution with project documents (contracts, specifications, and BIM data), the system will choose the applicable model based on the project type determined or entered into the project attributes. Subsequently, the system will generate activities with their dependencies. Each activity in the activity sequence should depend on its preceding activity. For each identified activity, an activity file similar to the activity dictionary file will be created. Table 4 shows the project schedule activity file.

4.1.6. Activity Magnitude and Direct Cost Calculator

Since each structural element is treated as a standalone activity, the dimensions of the elements are retained in the BIM database. Accordingly, the magnitude of the work of each activity can be determined. This should be fed to the activity magnitude field in the project activity file (Field 6 in Table 5). From the element size, the material quantity, manpower, and equipment can be determined. Since worker productivity for each unit of measurement is already one of the variables of the activity file, the required manhours for each activity and equipment hours can be estimated. The total activity cost can then be estimated by multiplying the resources with the updated actual cost of materials, manpower, and equipment. The material costs could be retrieved directly from the web using ChatGPT or an AI-powered search engine, while the costs of equipment and manpower could be sourced from the contractor’s ERP system. The estimated total direct project cost can be determined by aggregating the direct costs of all activities. Initially, for the sake of calculating a project’s direct cost, resource constraints could be ignored. The indirect cost is a function of the project duration, which can be determined once the project duration is set.

Table 5. Project Schedule Activity File.

No.	Field	Description
1	Activity Code	Activity dictionary code number
2	Activity Title	Activity title, such as “Excavation”
3	Activity Sequence	This is generated from the ML model.
4	Activity Predecessors	This field should be used to determine the project’s critical path and the total project duration.
5	Activity Type Description	A general description of the activity
6	Activity Magnitude	To be fed from the activity magnitude calculator (see Section 4.2.4)
7	Required Equipment	Used to calculate the required hours based on the magnitude of the activity (see Field 6) and the standard equipment hour per activity unit of measurement
8	Required Worker Skills	Used to calculate the required manhours by multiplying the magnitude from Field 6 by the

		standard required manhours per activity unit of measurement
9	Required Materials	Used to calculate the required material by multiplying the magnitude of the activity from Field 6 by the required material per unit of measurement
10	Activity Measurement Unit	Identifies the measurement unit (e.g., square meter or cubic meter) based on the nature of the activity
11	Mandatory Pre-Activity	The prior activity or activities identified by the ML model
12	Mandatory Post-Activity (Predecessors)	Predecessor activity or activities identified by the ML model

4.1.7. Activity Duration Calculator

The duration of each activity depends on the magnitude of the work and the allocated resources (manpower and equipment). In addition, there is a relationship between allocated resources, cost, and duration. A resource constraint matrix should be developed to allocate resources for each activity. Accordingly, for each activity and concurrent activity, the maximum allocated resources will not exceed the resources reflected in the resource’s constraint matrix. The resource constraint matrix could be developed based on information from the ERP or contractors’ policies.

4.1.8. Schedule Analyzer

The system can calculate the direct costs of the project. The indirect costs should be determined by the contractor. The indirect cost could be a predefined amount per day or could be a function of the project’s active resources. Determining the indirect cost is important for determining the optimal project cost and duration schedule, as there is a trade-off between project cost and duration [36].

Multi-objective linear programming can be employed to determine the optimal project cost and duration [37,38]. The financial analysis could also be conducted based on the progress payment percentage terms and the user’s financial position. The actual duration and cost during the execution of the project should be monitored and used as training data to improve future performance and adjust the schedule to meet its targets.

To summarize, the proposed system consists of six components: project analyzer, data warehouse, activity identifier and sequencing, activity magnitude and direct cost calculator, activity duration calculator, and schedule analyzer. The proposed method uses an approach where the dependencies of project elements are retrieved from the BIM files.

5. Conclusions

The study demonstrated that autonomous project management is feasible with the current AI technology available and proposed a blueprint for building an autonomous construction project schedule. ML, NLP, LLMs, and web scraping have revolutionized the world of AI, marking a new dawn for autonomous scheduling. Autonomous scheduling aids in cost reduction, progress supervision, delay and error prevention, and quality improvement. The proposed solution can analyze real-time data (e.g., cost, productivity, and progress) and make adjustments to optimize schedules accordingly. Moreover, the model would also be in a perpetual state of improvement, since completed projects could be used as training data.

Project managers can enjoy seamless communication because the cloud-based system would be able to synchronize inputs from all sources simultaneously. The proposed solution would thus enhance project performance through proactive planning and risk management. Contracting extra workers at the last minute, idle times, material shortages, and missing dependencies would become problems of the past. Integrating the autonomous project scheduling solution with the other components of an autonomous project management system would enable the surmounting of many of the challenges that project managers encounter and would result in greater cost efficiency and

fewer delays. Autonomous scheduling can change the course of the AEC industry and completely reshape its future.

References

1. Patel, D. *Revolutionizing Project Management with Generative AI*. ISJEM, 2023. <https://isjem.com/download/revolutionizing-project-management-with-generative-ai>. (accessed 2023-06-05)
2. Al-Sinan, M.; Bubshait, A.; Al-Dossary, S. A Framework for Embracing Construction 4.0. In *1st Int'l Conference on Challenges in Engineering, Medical, Economics and Education: Research & Solutions (CEMEERS-23)*, Lisbon, Portugal, June 21–22, 2023. <https://doi.org/10.17758/EIRAI19.F0623112>
3. Bryson, J.; Winfield, A. Standardizing Ethical Design for Artificial Intelligence and Autonomous Systems. *Computer*, 2017, 50 [5], 116–119. DOI: 10.1109/MC.2017.154
4. Hunt, J.; Osorio-Sandoval, C.A. Assessing Embodied Carbon in Structural Models: A Building Information Modeling-Based Approach. *Buildings* 2023, 13, 1679. <https://doi.org/10.3390/buildings13071679>
5. Hevner, A.R.; March, S.T.; Park, J.; Ram, S. Design Science in Information Systems Research. *MIS Q.* 2004, 28, 75–105. https://www.academia.edu/8153530/Design_Science_in_Information_Systems_Research
6. Vom Brocke, J.; Hevner, A.; Maedche, A. Introduction to Design Science Research. Design Science Research Cases. Springer, 2020
7. Kim, H.; Anderson, K.; Lee Sand Hildreth, J. Generating Construction Schedules Through Automatic Data Extraction Using Open BIM (Building Information Modeling) Technology. *Automation in Construction* 2013, 35, 285–295.
8. Wu, Z.; Ma, G. Automatic Generation of BIM-Based Construction Schedule: Combining an Ontology Constraint Rule and a Genetic Algorithm. *Engineering, Construction and Architectural Management Journal* 2022. DOI: 10.1108/ECAM-12-2021-1105.
9. Hong, Y.; Xie, H.; Agapaki, E.; Brilakis, I. Graph-Based Automated Construction Scheduling without the Use of BIM. *J. Constr. Eng. Manag.* 2023, 149, 05022020.
10. Faghihi, V.; Reinschmidt, K. F.; Kang, J. H. Construction Scheduling Using Genetic Algorithm Based on Building Information Model. *Expert Systems with Applications*, 2014, 41 [16], 7565–7578.
11. Kaplan, A.; Haenlein, M. Siri, Siri, In My Hand: Who's the Fairest in the Land? On the Interpretations, Illustrations, and Implications of Artificial Intelligence. *Business Horizons*, 2019, 62 [1], 15–25. https://www.researchgate.net/publication/328761767_Siri_Siri_in_my_hand_Who's_the_fairest_in_the_land_On_the_interpretations_illustrations_and_implications_of_artificial_intelligence
12. Michael, M. *Artificial Intelligence in Law: The State of Play*. Thomson Reuters Legal Research Institute, 2016. <https://britishlegalitforum.com/wp-content/uploads/2016/12/Keynote-Mills-AI-in-Law-State-of-Play-2016.pdf>
13. Bartneck, C.; Lütge, C.; Wagner, A.; Welsh, S. *An Introduction to Ethics in Robotics and AI*. Springer, 2021. <https://doi.org/10.1007/978-3-030-51110-4>
14. Allen, G. C. Understanding Artificial Intelligence Technology. DoD Joint AI Center, 2020.
15. Craglia, M. (Ed.); Annoni, A.; Benczur, P.; Bertoldi, P.; Delipetrev, B.; De Prato, G.; Feijoo, C.; Fernandez, M. E.; Gomez, E.; Iglesias, M.; Junklewitz, H.; López, C. M.; Martens, B.; Nascimento, S.; Nativi, S.; Polvora, A.; Sanche, I.; Tolan, S.; Tuomi I.; Vesnic Alujevic, L. *Artificial Intelligence - A European Perspective*. EUR 29425 EN, Publications Office, 2018.
16. Shah, C. *A Hands-on Introduction to Data Science*. Cambridge University Press, 2020.
17. Sarkar, D. *Text Analytics with Python: A Practitioner's Guide to Natural Language Processing*, 2nd Ed.; Bangalore, Karnataka, India, 2019. <https://doi.org/10.1007/978-1-4842-4354-1>
18. Zhao, Y. The State-of- Applications of NLP: Evidence from ChatGPT. *Highlights in Science, Engineering, and Technology* 2023, 49.
19. Council of the European Union, General Secretariat. *ChatGPT in the Public Sector—Overhyped or Overlooked?* Analysis, and Research Team, April 2023. https://www.consilium.europa.eu/media/63818/art-paper-chatgpt-in-the-public-sector-overhyped-or-overlooked-24-april-2023_ext.pdf (accessed 2023-8-15)
20. Jagdishbhai, N.; Thakkar, K. Y. Exploring the Capabilities and Limitations of GPT and ChatGPT in Natural Language Processing. *J. Manag. Res. Anal.* 2023, 10 [1], 18–20. <https://doi.org/10.18231/j.jmra.2023.004>
21. Rahaman, M. S.; Ahsan, M. M. T.; Anjum, N.; Terano, H. J. R. From ChatGPT-3 to GPT-4: A Significant Advancement in AI-Driven NLP tools. *Journal of Engineering and Emerging Technologies* 2023, 1, 50–60. 10.52631/jeet.v1i1.188.
22. Diouf, R.; Sarr, E. N.; Sall, O.; Birregah, B.; Bousso, M.; Mbaye, S. N. Web Scraping: State-of-the-Art and Areas of Application. In *2019 IEEE International Conference on Big Data (Big Data)*, Los Angeles, CA, USA, 2019, pp. 6040–6042. DOI: 10.1109/BigData47090.2019.9005594.

23. Mufid, M. R.; Basofi, A.; Mawaddah, S.; Khotimah, K.; Fuad, N. Risk Diagnosis and Mitigation System of COVID-19 Using Expert System and Web Scraping. *2020 International Electronics Symposium (IES)*, Surabaya, Indonesia, 2020, pp. 577–583. DOI: 10.1109/IES50839.2020.9231619
24. Dilmegani, C. *Top 3 Web Scraping Challenges Solved by AI in 2023*. AIMultiple, March 6, 2023. <http://research.aimultiple.com/smart-web-scraping/> (accessed 2023-06-23).
25. Kerner, S. M. Google Search Generative Experience (SGE). *Tech Target*. <https://www.techtarget.com/whatis/definition/Google-Search-Generative-Experience-SGE> (accessed 2023-06-23).
26. Lahmann, M.; Keiser, P.; Stierli, A. *AI Will Transform Project Management. Are You Ready?* 2018 PricewaterhouseCoopers AG.
27. Faghihi, V.; Nejat, A.; Reinschmidt, K. F.; Kang, J. H. Automation in Construction Scheduling: A Review of the Literature. *Int. J. Adv. Manuf. Technol.*, 2015, *81* (9–12), 1845–1856. DOI: 10.1007/s00170-015-7339-0.
28. ElMenshawy, M.; Marzouk, M. Automated BIM Schedule Generation Approach for Solving Time–Cost Trade-off Problems. *Engineering, Construction and Architectural Management* 2021, *28* [10], 3346–3367.
29. Hong, Y.; Xie, H.; Agapaki, E.; Brilakis, I. Graph-Based Automated Construction Scheduling without the Use of BIM. *J. Constr. Eng. Manag.* 2023, *149*, 05022020
30. Hong, Y.; Xie, H.; Bhumbra, G.; Brilakis, I. 2021. Comparing Natural Language Processing Methods to Cluster Construction Schedules. *J. Constr. Eng. Manage.*, 2021, *147* [10], 04021136. [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0002165](https://doi.org/10.1061/(ASCE)CO.1943-7862.0002165)
31. Amer, F.; Golparvar-Fard, M. Formalizing Construction Sequencing Knowledge and Mining Company-Specific Best Practices from Past Project Schedules. in *ASCE International Conference on Computing in Civil Engineering*, 215–223, Atlanta, Georgia, June, 17–19, 2019, doi:10.1061/9780784482421.028.
32. Faghihi, V.; Reinschmidt, K. F.; Kang, J. H. Construction Scheduling Using Genetic Algorithm Based on Building Information Model. *Expert Systems with Applications*, 2014, *41* [16], 7565–7578.
33. Amer, F.; Jung, Y.-H.; Golparvar-Fard, M. Construction Schedule Augmentation with Implicit Dependency Constraints and Automated Generation of Lookahead Plan Revisions. *Automation in Construction*, 2023, *152*, 104896. <https://doi.org/10.1016/j.autcon.2023.104896>
34. Prieto, S. A.; Mengiste, E. T.; García de Soto, B. Investigating the Use of ChatGPT for the Scheduling of Construction Projects. *Buildings* 2023, *13*, 857. <https://doi.org/10.3390/buildings13040857>
35. Faghihi, V. Automated and Optimized Scheduling Using BIM. PhD Dissertation, Texas A&M University, 2014.
36. Assaf, S. A.; Bubshait, A. A.; Atiyah, S.; Al-Shahri, M. [1999], Project Overhead Costs in Saudi Arabia. *Cost Engineering*, 1999, *41* [4], 33–38.
37. Al-Subhi Al-Harbi, K.; Selim, S.; Al-Sinan, M. A Multiobjective Linear Program for Scheduling Repetitive Projects. *Cost Engineering*, 1996, *38* [12], 41–44.
38. Singh, P.; Smarandache, F.; Chauhan, D.; Bhaghel, A. A Unit Based Crashing Pert Network for Optimization of Software Project Cost. *International Journal of Contemporary Mathematical Science* 2015, *10*, 29

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.