Qi, R., Xu, J., & Zheng, R. (2026). A multi-agent data-to-model workflow for make-ready assessment. In F. Drevland, E. Forcael, F. Hamzeh, M. Poshdar, & I. D. Tommelein (Eds.), *Proceedings of the 34th Annual Conference of the International Group for Lean Construction* (IGLC34) (pp. 238–249) https://doi.org/10.24928/2026/0266

A MULTI-AGENT DATA-TO-MODEL WORKFLOW FOR MAKE-READY ASSESSMENT

Ruixuan Qi[1], Jinying Xu[2], & Ruiyan Zheng[3]

ABSTRACT

The Last Planner System (LPS) is widely recognized as an effective lean approach for improving workflow reliability. Although the theoretical framework of LPS has been well established, its practical implementation remains challenging in real projects. A major barrier lies in initiating and maintaining Lookahead planning, which is intended to connect phase scheduling with commitment planning. Current practice often shows gaps, including lack of detailed data required for Make-Ready assessment in Lookahead planning.

This paper proposes a multi-agent, data-to-model workflow that lowers the entry barrier to Lookahead planning by extracting project-specific Lookahead rules and readiness assessments directly from commonly available progress schedule data. Instead of training a generalized predictive model, the workflow coordinates specialized agents to sequentially normalize tasks, infer dependencies, reconstruct production states, and support readiness scoring with an interpretable learning component.

The workflow is demonstrated using real data from a high-rise building project. The case suggests that meaningful Make-Ready assessments can be derived from standard project schedule data. It may help practitioners reduce reliance on subjective judgment and fragmented information in Lookahead planning, while offering a practical method to push near-term work commitments further in LPS adoption.

KEYWORDS

Last Planner System, lookahead planning, make-ready assessment, multi-agent systems, data-to-model workflow.

INTRODUCTION

The architecture, engineering, and construction (AEC) industry continues to face a persistent productivity challenge, while lean production and digital technologies have improved performance in many other sectors. Within Lean Construction, the Last Planner System (LPS) is widely recognized as an effective approach for improving workflow reliability through collaborative and commitment-based planning (Hamzeh et al., 2012; Koskela et al., 2020). Among its core functions, Lookahead planning is especially important because it connects longer-term plans with near-term work commitments and helps determine whether upcoming tasks are truly ready to be executed.

[1] Graduate student, National University of Singapore, Singapore, e1538237@u.nus.edu
[2] Assistant Professor, Department of the Built Environment, National University of Singapore, Singapore; jinying.xu@nus.edu.sg, orcid.org/0000-0001-9589-6396
[3] PhD student, Department of the Built Environment, National University of Singapore, Singapore; zhengruiyan@u.nus.edu, orcid.org/0009-0009-8421-0614

In practice, however, Lookahead planning remains difficult to start and sustain. Many projects still rely on conventional schedules, such as master schedules or summary bar charts, that record dates and durations but do not clearly represent production logic, prerequisite work, or constraint status (Ballard, 1997; Hamzeh et al., 2012). This means that Make-Ready assessment often depends on manual interpretation and tacit knowledge rather than on a consistent and verifiable process. When planners must judge design readiness, material availability, workspace conditions, and trade interfaces across many upcoming tasks, the process can easily become fragmented and unreliable, increasing the risk of "making-do" and its associated waste, rework, and safety problems (Lagos et al., 2024).

Recent studies have explored how automation and artificial intelligence can support construction planning through task decomposition, constraint identification, and predictive analysis (Amer et al., 2021; Lagos et al., 2024). These studies show useful technical possibilities, but they often focus on isolated functions or depend on large, highly structured datasets that are rarely available in everyday project settings. As a result, an important practical question remains open: how can commonly available planning data be translated into a usable Make-Ready assessment process for Lookahead planning?

This paper addresses that question by proposing a multi-agent, data-to-model workflow for Make-Ready assessment. Here, "multi-agent" does not refer to autonomous robots or fully independent decision-makers; it refers to a set of coordinated analytical functions, each responsible for one part of the assessment process, such as interpreting task descriptions, inferring dependencies, reconstructing production status, checking constraints, and estimating readiness. Framing the method in this way keeps the focus on the planning problem itself: supporting Lookahead planning with a workflow that is transparent, repeatable, and compatible with the data conditions of real AEC projects.

The workflow is examined through a retrospective case study of a high-rise building project, where repetitive task patterns make it possible to infer production logic from standard schedule records. Rather than presenting AI as a separate research topic, the study uses it as an implementation mechanism for operationalizing Make-Ready assessment under realistic data limitations. The paper therefore advances the proposition that Make-Ready assessment can be better supported when ordinary planning data are reorganized through a transparent, project-level workflow rather than treated as input for a black-box prediction task.

LITERATURE REVIEW

THE LAST PLANNER SYSTEM AND LOOKAHEAD PLANNING

The Last Planner System (LPS) organizes project planning into levels of decreasing time horizon and increasing commitment, commonly including master scheduling, phase scheduling, Lookahead planning, and Weekly Work Planning (WWP) (Hamzeh et al., 2012). Within this structure, Lookahead planning serves as the bridge between longer-range intentions and short-term commitments. Its purpose is not simply to list upcoming activities, but to screen them so that only work that is executable under current conditions enters the weekly plan (Ballard & Tommelein, 2016; Lagos et al., 2024). For this reason, the quality of Lookahead planning depends heavily on whether task readiness can be assessed in a systematic way before commitments are made. This section is kept brief because the present study does not aim to revisit LPS theory in general; it focuses on the specific information and decision problem embedded in Make-Ready assessment.

INFORMATION LIMITATIONS IN LOOKAHEAD PLANNING

A central difficulty is that the information needed for Make-Ready assessment is often not represented clearly in the planning documents that projects already maintain. Conventional

schedules are mainly designed to communicate milestones, durations, and contractual sequencing. They are much less effective at showing detailed production logic, prerequisite relationships, or the current status of constraints (Ballard, 1997; Hamzeh et al., 2012). In that sense, the challenge is not simply a lack of data. It is a mismatch between the purpose of existing planning artefacts and the richer situational judgment required in Lookahead planning.

This mismatch creates a practical gap between the intent of Lookahead planning and the information base available to support it. In many projects, planners must infer readiness from fragmented records and personal experience, which makes the process hard to repeat, explain, or scale across teams (Koskela, 2000; Power et al., 2023). Compared with later AI-oriented studies, this stream of work is valuable because it defines the decision context clearly: the issue is not only whether a task will finish as planned, but whether it should be allowed to enter a reliable near-term work plan in the first place.

CONSTRAINT MANAGEMENT AND "MAKING-DO" IN LPS

Constraint management is central to the flow-based logic of Lean Construction. Koskela (2000) and Kim and Ballard (2000) describe task execution as depending on the convergence of several prerequisite flows, such as prior work, information, materials, labor, equipment, workspace, and external conditions. If one of these flows is missing or unstable, the task is not reliably ready for execution.

On site, however, identifying these constraints is often still a manual and experience-based activity. Screening upcoming work under time pressure can vary greatly in rigor from one planner or meeting to another, especially when multiple trades, locations, and prerequisite conditions must be considered simultaneously (Power et al., 2023). When this screening is incomplete, work may start before all requirements are in place, leading to "making-do" and the resulting loss of workflow reliability.

ARTIFICIAL INTELLIGENCE IN CONSTRUCTION PLANNING

Research on AI in construction planning has expanded rapidly, covering areas such as progress monitoring, schedule analysis, delay prediction, and planning support (Hatami et al., 2022; Moragane et al., 2024). These studies demonstrate that construction data can be processed in more systematic ways, but they differ substantially in what they support. Monitoring-oriented approaches are strong at detecting status or deviation after work has begun. Predictive approaches are useful for estimating likely delay or non-compliance from historical patterns. By contrast, Lookahead planning requires an earlier and more intervention-oriented judgment: whether upcoming work is sufficiently ready to be committed. This difference in decision timing is important because a model that predicts problems after they emerge is not equivalent to a workflow that helps planners prevent premature commitment beforehand.

A smaller group of studies is more directly related to Lookahead planning. Prior work has explored machine learning for functions such as task decomposition, analysis of Reasons for Non-Compliance, and constraint identification from historical project data (Amer et al., 2021; Lagos et al., 2024). These studies are valuable, but they usually automate one task at a time and often rely on structured execution records or richer datasets than many projects actually possess during routine planning. Other approaches, such as natural language processing for extracting constraints from text, can improve information access but still leave the final feasibility judgment largely with human planners. In comparative terms, existing approaches can be roughly distinguished along three dimensions: the planning stage they support, the data richness they require, and whether they automate an isolated function or a linked decision process. Most prior work is strongest in one dimension but weaker in the others, which helps explain why useful AI functions have not yet translated into a widely usable Make-Ready workflow.

FROM ISOLATED AI FUNCTIONS TO MULTI-AGENT WORKFLOW SUPPORT

The literature therefore points to two linked gaps. First, Lookahead planning is widely recognized as essential to LPS performance, yet it remains difficult to operate with the information and tools commonly available in practice (Hamzeh et al., 2012; Power et al., 2023). Second, existing AI-based studies tend to support isolated planning functions rather than the full process of Make-Ready assessment as it is carried out in project settings. This is where a multi-agent perspective becomes useful. In related digital planning research, multi-agent systems are typically valued not because they make a single prediction more accurate, but because they allow different reasoning tasks to be decomposed, coordinated, and updated across a workflow. That logic is particularly relevant here, since Make-Ready assessment itself consists of several linked judgments rather than one single classification step.

This study responds to those gaps by treating Make-Ready assessment as a workflow problem rather than as a standalone prediction problem. Instead of seeking a universal model, it proposes a project-level, multi-agent workflow that organizes available planning data into a sequence of interpretable steps. The value of the multi-agent framing in this paper is therefore practical and organizational: it separates task interpretation, dependency reasoning, state reconstruction, constraint checking, and scoring into connected functions that can be inspected and refined individually. In this way, AI is positioned as a support mechanism within Lookahead planning rather than as a separate objective detached from AEC practice.

A MULTI-AGENT DATA-TO-MODEL WORKFLOW FOR MAKE-READY ASSESSMENT

This research introduces a multi-agent, data-to-model workflow for Make-Ready assessment within Lookahead planning. The emphasis is not on AI for its own sake, but on how commonly available project records can be translated into a more consistent readiness assessment process. The workflow breaks the overall task into several coordinated functions so that the logic behind Make-Ready assessment becomes easier to trace, reuse, and improve under realistic project data conditions.

OVERVIEW OF THE MULTI-AGENT WORKFLOW

Figure 1 illustrates the overall architecture of the proposed workflow. The workflow operates on a shared project data structure and consists of six coordinated functions, described here as agents because each one handles a distinct analytical role in the overall process:

1. Task Normalization Agent, which converts varied schedule task descriptions into a consistent task vocabulary that can be used for Lookahead analysis.
2. Dependency Reasoning Agent, which infers task-to-task dependencies and minimum spacing rules from project data and domain knowledge.
3. State Reconstruction Agent, which rebuilds the near-term production situation from schedule updates and resource-related information.
4. Constraint Checking Agent, which checks whether the known prerequisite conditions for upcoming tasks have been satisfied and identifies likely blockers.
5. Readiness Labeling Agent, which assigns readiness labels by comparing inferred feasibility with observed execution outcomes.
6. Learning-Based Scoring Agent, which fits a lightweight project-specific model to estimate readiness likelihood and help planners prioritize attention.

The agents exchange structured data tables rather than informal judgments, which helps keep the workflow transparent and allows individual steps to be refined without changing the whole process. This decomposition also reflects the argument developed in the literature review: the

practical value of a multi-agent design lies less in technical novelty alone and more in its ability to connect several planning judgments into one inspectable workflow.

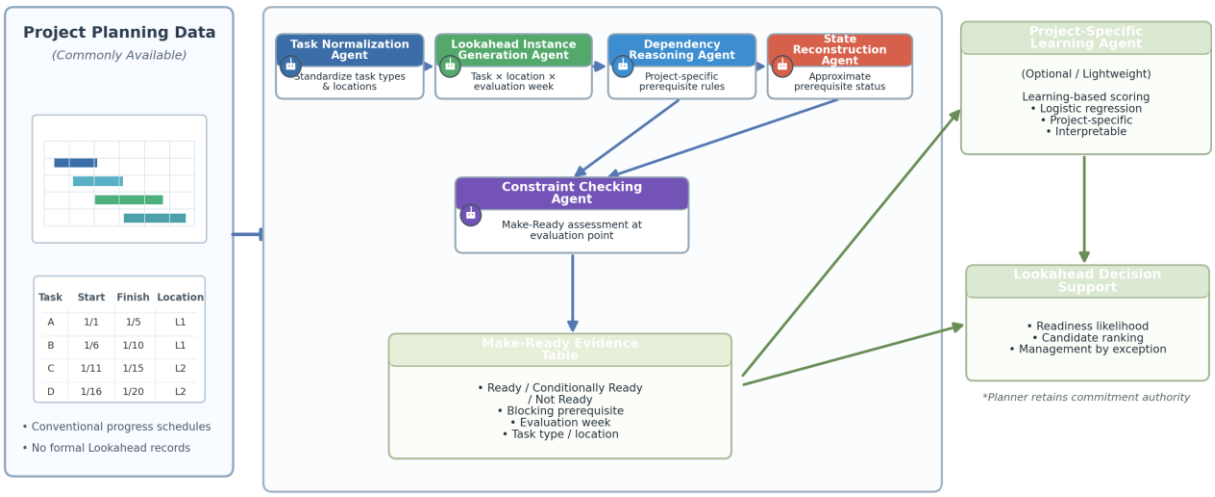

Figure 1: Overview of the proposed multi-agent workflow for operationalizing Make-Ready assessment

TASK NORMALIZATION AND LOOKAHEAD INSTANCE GENERATION

The workflow begins by processing conventional progress schedules, which typically contain free-text task descriptions organized by location or phase. The Task Normalization Agent maps these descriptions to a finite *TaskTypeCatalog*, capturing essential production characteristics such as discipline, trade, and Lookahead role.

It is worth noting that in this study, the standardization of tasks followed a two-step process designed to achieve reproducibility. First, a limited "task type directory" was manually constructed as a project-specific standardized code table. This directory was developed based on the repetitive work packages observed in the case schedules and general structural distinctions such as discipline, industry, stage, and forward-looking roles. Second, each original task description was mapped to this directory through explicit cues contained in the schedule text, including keywords for work content, stage expressions (e.g., first stage, second stage), and location identifiers. Only when a task description was ambiguous, complex, or not specific enough to meet the rule-based consistent allocation would it undergo manual review.

Based on normalized task types and scheduled dates, the workflow generates specific formatted Lookahead instances, each of which represents a candidate task evaluated at a specific planning time within the Lookahead window. This transformation converts static schedule information into a dynamic representation of planning decisions.

DEPENDENCY REASONING AND PRODUCTION STATE RECONSTRUCTION

For Lookahead instances, the Dependency Reasoning Agent determines the necessary pre-task execution and the minimum completion time interval. These dependencies may stem from observed execution sequences in historical data, from the inherent engineering logic embedded in task types, and from the input of specific rules by humans.

In parallel, the State Reconstruction Agent infers the current production status at each forward time point based on the progress information. Even if the data at a given point in time are unavailable, a conservative estimate of the completion status can be derived from progress, thereby yielding an approximate yet practically useful reconstruction of the state.

CONSTRAINTS CHECKING AND READINESS LABELING

The Constraint Checking Agent assesses the Lookahead instance on the basis of the inferred requirements for the Lookahead instance, as compared to the reconstructed state of the production. In the above process, the state of readiness is defined as a time-specific planning condition rather than an outcome-based completion condition. As such, the raw task is not used as an individual labeled data point. Rather, as the task is expanded to multiple instances within the weekly Lookahead, the task is represented multiple times within the Lookahead window with changing readiness as the prerequisites are satisfied.

From this comparison, each case is assigned a reconstructed readiness status. In the workflow, three states relevant to the practical application were identified as Ready, Conditionally Ready, and Not Ready. Ready means the main prerequisite conditions are met at the point of evaluation. Not Ready means the main prerequisite conditions still have unresolved blocking prerequisites. Conditionally Ready is the middle case where there are still unresolved prerequisites, but they are short cycle and/or trivial prerequisites. They should not be considered blocking prerequisites. It is important to note that in the real world, not all prerequisites should be equally blocking. It is also important to note the three-state model is a much more accurate representation of the Make-Ready logic than the two-state model. Blocking prerequisites are included with the readiness status so the basis for the decision can be inspected.

PROJECT-SPECIFIC LEARNING FOR READINESS SCORING

After the readiness labels are generated, the labeled Lookahead instance table is used to train a lightweight, project-specific scoring model. The purpose of this component is not to replace rule-based Make-Ready checks or planner judgment, but to provide an interpretable probability score that can help prioritize candidate tasks within weekly planning windows.

To improve learning stability under constrained data conditions, the three-state readiness labels were consolidated into a binary modelling target. Instances labeled Ready or Conditionally Ready were grouped as ready-like, whereas Not Ready instances were treated as blocked. This formulation reflects the practical objective of distinguishing tasks that are sufficiently executable for near-term consideration from those still affected by more substantive blockers.

The model is trained using structured features derived from upstream workflow outputs, including task descriptors, timing variables, and aggregated dependency-state information. To reduce label leakage, the main evaluation adopts a stricter feature configuration that excludes variables that are near-direct encodings of the reconstructed label. In this way, the model learns a compact approximation of the project-specific readiness logic rather than merely reproducing the rule-based decision itself. The resulting score is intended to support ranking, management by exception, and the reuse of project-specific planning experience over time. For evaluation purposes, this score was interpreted as the estimated probability of a ready-like instance and was mapped to a binary prediction using a fixed classification threshold.

CASE STUDY AND IMPLEMENTATION

CASE PROJECT OVERVIEW AND DATA AVAILABILITY

The case study was conducted using planning data from the Wuhan Yangtze River Center project (hereafter, the Case Project), a large-scale high-rise development. An overview of the Case Project is presented in Table 1. This Case Project is representative of complex urban construction environments in which Lookahead planning is both critical and difficult to sustain.

Table 1: Overview of planning data scale and structure in the Case Project

Category	Description / Value
Project name	Wuhan Yangtze River Center
Project type	High-rise mixed-use development (office tower + commercial podium)
Main structures	One 83-storey office tower and one 7-storey commercial podium
Total gross floor area	349,055 m² (271,579 m² above ground; 77,476 m² underground)
Underground levels	4 basement levels, including a partial mezzanine level (B1M)
Project schedule	1 September 2020 – 29 August 2024
Primary planning data source	Master construction schedule (Gantt chart)
Planning granularity	Phase- and location-based tasks
Lookahead schedules	Not available
Constraint logs / RNC records	Not available

The available data reflects typical conditions in current construction projects rather than idealized LPS implementation. The primary source of information was a conventional master schedule prepared in Gantt chart format. This schedule specified task names, planned start and finish dates, durations, and location references, but did not include explicit Lookahead schedules, constraint logs, RNC records, or WWP. In addition, limited resource-planning information was available in the form of aggregated labour-allocation tables with a coarse time granularity.

These data-related limitations are not accidental; they reflect common obstacles encountered when implementing LPS in practical applications. In many projects, forward planning is conceptually recognized, but in practice it may be hindered by limited experience with LPS implementation and historical data, as well as by unstructured task descriptions and the absence of a formal constraint-tracking mechanism. Therefore, decisions about preparatory work mainly rely on planners' experience and informal reasoning or on systematic evidence. Hence, this case study is intentionally conducted under these limitations. It is not assumed that there are mature LPS masterpieces; rather, the widely available data are used to test whether forward reasoning can be reconstructed and realized by combining domain logic and lightweight learning mechanisms with standard planning data.

DATA FLOW: FROM RAW SCHEDULE DATA TO MAKE-READY ASSESSMENT

The following section explains how real-world progress data from the case project are gradually transformed into probabilistic outcomes that support Make-Ready planning through the proposed multi-agent workflow.

Agent 1: Task Normalization

The original task descriptions in the construction schedule include multiple types of information simultaneously, including construction stage (e.g., first layer, second layer), location (e.g., B1-1 zone), work content (excavation, support installation, curing, removal), and quantitative details (depth, area, duration). As a result, the schedule contained hundreds of syntactically

distinct task names that were semantically similar from a production-flow perspective. Through agent 1, a large number of heterogeneous task descriptions were consolidated into a compact, reusable TaskType vocabulary, as shown in Table 2.

Table 2: Examples of task normalization from raw schedule descriptions to standardized TaskTypes

Original Schedule Task Description (examples)	Normalized TaskType	Lookahead Role	Discipline / Trade
First / Second / Third / Fourth-stage earth excavation	EXCAV_STAGE_1–4	Production task	Earthworks
Soil excavation in the access ramp (haul road) area	EXCAV_SPECIAL_ZONE	Production task	Earthworks
First / Second / Third-stage support installation (including curing)	SUPPORT_INSTALL_STAGE_1–3	Constraint-enabling task	Temporary works
Support curing (Stage 1); slab concrete curing (7 days)	CURING_SUPPORT / CURING_SLAB	Constraint-enabling task	Temporary works / Concrete
First / Second / Third-stage support removal	SUPPORT_REMOVE_STAGE_1–3	Constraint-releasing task	Temporary works
Basement bottom cleaning; blinding concrete placement	BASE_PREPARATION / BLINDING_CONCRETE	Production task	Concrete works
Foundation formwork installation; foundation waterproofing and protection layer	FORMWORK_MUDSILL / WATERPROOFING_PROTECTION	Constraint-enabling task	Concrete / Waterproofing
Foundation and pile cap reinforcement fixing; foundation concrete pouring	REBAR_FIXING_FOUNDATION / CONCRETE_POUR_FOUNDATION	Production task	Reinforcement / Concrete

Agents 2–5: From Normalized Tasks to Lookahead Readiness Evidence

Once tasks are standardized, downstream agents can operate from the same work representation. This is important because it minimizes ambiguity that typically occurs when different planners interpret schedules differently. Agent 2 then reviews the standardized schedule and generates candidate task–location–period combinations within the upcoming window, creating a continuous set of future instances. Essentially, the master schedule becomes an input reflecting on-site Lookahead decisions. Next, dependency logic is added. Agent 3 combines general engineering precedence and observed sequencing patterns across zones to determine project-specific task relationships. It produces dependency rules that specify prerequisites and minimum conditions for each task type. Since reliable completion data was unavailable, Agent 4 reconstructs weekly production status from planned dates and conservative offsets. With this information, Agent 5 performs the Make-Ready check: verifying prerequisites against the reconstructed status and dependency rules. Each forward instance is assigned a reconstructed readiness status, which in the implemented case was later consolidated into a binary learning

target of ready-like versus blocked for model training. Table 3 illustrates a typical task sequence from the case project, showing how each instance is transformed through Agents 2–5.

Table 3: Example of a task progressing through the proposed multi-agent Make-Ready assessment workflow

Workflow Stage / Agent	Data Representation	Key Fields Generated	Example Value from Case Project
Raw Schedule	Schedule row	Task name; planned start; planned finish; location	"Second-stage support installation (including curing)"; 2020-09-26; 2020-10-15; B1-1
Agent 1 – Task Normalization	Normalized task record	TaskType; stage; location_id; attributes	SUPPORT_INSTALL_STAGE_2; stage=2; loc=B1-1; includes_curing=1
Agent 2 – Lookahead Instance Generation	LookaheadInstance	Lookahead week; evaluation window; candidate task	Week 39; 2020-09-21–10-18; SUPPORT_INSTALL_STAGE_2 @ B1-1
Agent 3 – Dependency Reasoning	DependencyRule	Prerequisite TaskType; target TaskType	EXCAV_STAGE_2 → SUPPORT_INSTALL_STAGE_2
Agent 4 – Production State Reconstruction	Weekly production state snapshot	TaskType; completion status estimate	EXCAV_STAGE_2 @ B1-1 = not completed
Agent 5 – Constraint Checking & Labeling	Make-Ready assessment record	Dependency satisfied; blocking prerequisite; readiness label	Dependency satisfied = No; blocked by EXCAV_STAGE_2; Not Ready

Agent 6: Learning-Based Readiness Scoring in the Case Project

In the case project, labeled Lookahead instances provided by Agents 2 through 5 were used to train a lightweight readiness scoring model. Following the logic of building up planning experience over time, the training dataset was split along temporal, rather than random, dimensions, with earlier weeks used for training and later weeks used for validation/testing.

The final solution takes a binary learning target, where instances classified as Ready or Conditionally Ready are treated as a single category of 'ready-like' instances, while those classified as Not Ready are treated as blocked instances. A lightweight logistic regression model was used, which was trained on structured features at the instance level, based on the workflow. The primary evaluation used a strict feature setting, in order to avoid near-direct label encodings and provide a more realistic assessment of potential usefulness. The model output was then treated as a readiness score in the form of the estimated probability that a given instance belonged to the ready-like class.

Quantitative Evaluation

The quantitative evaluation examined whether the proposed workflow generated sufficiently stable and learnable project-specific readiness patterns for later weekly planning windows. It did not aim to recover universally true historical readiness labels, because the case project did not contain formal Lookahead plans, constraint logs, or reasons-for-non-compliance records. Instead, the evaluation tested whether the reconstructed readiness logic produced by the workflow could be learned and reused in a temporally consistent manner. The scoring model output was interpreted as the probability of a ready-like instance. For classification-based evaluation, these probabilities were converted into binary predictions using a fixed threshold of 0.5, with instances above the threshold classified as ready-like and those below it classified as blocked.

The resulting dataset comprised 778 training instances, 109 validation instances, and 128 test instances. Model performance was assessed using accuracy, precision, recall, F1-score, Brier score, and ROC-AUC. On the training set, the model achieved an accuracy of 0.961, precision of 0.940, recall of 0.964, F1-score of 0.952, Brier score of 0.043, and ROC-AUC of 0.985. On the validation set, the model achieved an accuracy of 0.862, precision of 1.000, recall of 0.634, F1-score of 0.776, Brier score of 0.131, and ROC-AUC of 0.919. On the held-out test set, the model achieved an accuracy of 0.844, precision of 0.900, recall of 0.321, F1-score of 0.474, Brier score of 0.140, and ROC-AUC of 0.836.

These results indicate that the model retained meaningful discriminatory power on unseen later planning windows, although performance declined from training to testing, as expected under temporal generalization. The test-set pattern is especially informative: the model showed high precision but relatively low recall, meaning that when it identified a task as ready-like it was usually correct, but it also missed a substantial share of ready-like cases. In practical terms, the model behaved conservatively. Such conservatism may be acceptable, or even desirable, in Make-Ready support, because avoiding premature commitment of blocked tasks can be more valuable than aggressively admitting all potentially executable work into the weekly plan.

DISCUSSIONS

RESULTS AND KEY FINDINGS

This study suggests that historical construction planning data, even limited conventional schedule records, can be reorganized into a useful evidence base for Make-Ready assessment. In the case project, the workflow produced interpretable readiness indicators by combining task normalization, dependency reasoning, state reconstruction, and learning-based scoring. This matters because many projects already hold such data, even when they do not maintain richer digital twins or highly structured execution logs.

The evaluation further suggests that experience-based readiness judgment can be learned in a limited, project-specific sense. The resulting model should not be interpreted as universally transferable, since the logic it captures depends on recurring task patterns, local planning practices, and the quality of available schedule records. Its value lies more modestly in making implicit project knowledge more explicit and reusable within the same planning environment.

OUTCOME ANALYSIS

Rather than producing a universally generalizable predictive model, the proposed workflow enables project-specific readiness assessment that evolves as planning data accumulates. The learning is localized, contextual, and iterative, which is closely aligned with the situated and iterative logic of LPS practice.

The key point is that these results suggest that useful decision support can be achieved without assuming perfect or highly standardized data. That observation is important for practice

because many construction teams operate precisely under such constraints. It also helps explain why a workflow-oriented design may be more realistic than approaches that depend on extensive data engineering before any planning support becomes possible.

CONTRIBUTIONS TO LEAN CONSTRUCTION RESEARCH AND PRACTICE

This work makes three primary contributions to Lean Construction research. First, it operationalizes Make-Ready assessment as a sequence of explicit reasoning steps rather than leaving it as a largely tacit planning activity. Second, it shows how AI can be positioned as workflow support for Lookahead planning instead of as a detached predictive exercise. Third, it offers an example of project-specific learning that may help structure planner judgment under ordinary data conditions, while also clarifying the limits of such learning.

CONCLUSIONS

This paper presented a data-to-model workflow for operationalizing Lookahead and Make-Ready assessment from ordinary project schedule data. Rather than assuming that richer digital infrastructure is already available, the study focused on how existing planning records can be reorganized into a more transparent support process for near-term commitment decisions.

Rather than pursuing a universally generalizable predictive model, the proposed workflow separates the assessment problem into linked analytical functions that can be inspected and improved individually. The case study suggests that this approach can generate practical readiness evidence under constrained data conditions, but it also shows that the resulting logic remains project-specific and dependent on data quality. In addition, the present study is based on retrospective reconstruction and temporal validation rather than live deployment in ongoing Lookahead meetings. The findings therefore support the feasibility of the workflow as a decision-support approach, but they do not yet demonstrate how planners would adopt, interpret, or act on these outputs in real-time production settings. Future research should examine implementation in live planning environments and assess its effect on actual commitment decisions, exception handling, and workflow reliability.

ACKNOWLEDGMENTS

This research is funded by the NUS Start-up Grant Project: “Digitally Empowered Project Cognition for Sustainability”.

REFERENCES

Agrawal, A. K., Zou, Y., Chen, L., Abdelmegid, M. A., & González, V. A. (2024). Moving toward lean construction through automation of planning and control in last planner system: A systematic literature review. *Developments in the Built Environment, 18*, 100419. https://doi.org/10.1016/j.dibe.2024.100419

Amer, F., Jung, Y., & Golparvar-Fard, M. (2021). Transformer machine learning language model for auto-alignment of long-term and short-term plans in construction. *Automation in Construction, 132*, 103929. https://doi.org/10.1016/j.autcon.2021.103929

Ballard, G. (1997). Lookahead planning: The missing link in production control. In Proceedings of the 5th Annual Conference of the International Group for Lean Construction (IGLC).

Ballard, G., & Tommelein, I. D. (2016). Current process benchmark for the Last Planner® System. Lean Construction Journal, 57–89. https://doi.org/10.60164/32b7a3h4d

Hamzeh, F., Ballard, & Iris, G., & Tommelein, D. (2012). Rethinking Lookahead Planning to Optimize Construction Workflow. Lean Construction Journal, 2012, pp. 15–34. https://doi.org/10.60164/g3c3a1f5c

Hatami, M., Franz, B., Paneru, S., & Flood, I. (2022). Using deep learning artificial intelligence to improve the foresight method in the optimization of planning and scheduling of construction processes. In *Computing in Civil Engineering 2021* (pp. 1171–1178).

Kim, Y. & Ballard, G (2000). Is the Earned-Value Method an Enemy of Work Flow?. 8*th Annual Conference of the International Group for Lean Construction.*

Koskela, L. (2000). *An exploration towards a production theory and its application to construction* (Doctoral Dissertation). VTT Technical Research Centre of Finland.

Koskela, L., Vrijhoef, R., & Dana Broft, R. (2020). Construction supply chain management through a lean lens. *Successful construction supply chain management: Concepts and case studies*, 109–125. https://doi.org/10.1002/9781119450535.ch6

Lagos, C. I., Herrera, R. F., Mac Cawley, A. & Alarcón, L.F. (2024). An AI Copilot for Make-Ready Planning in the Last Planner System. In D. B. Costa, F. Drevland, & L. Florez-Perez (Eds.), *Proceedings of the 32nd Annual Conference of the International Group for Lean Construction (IGLC32)* (pp. 365–376). https://doi.org/10.24928/2024/0155

Power, W., Sinnott, D., Lynch, P. & Solorz, C. (2023). An Evaluation of the Lookahead Planning Function in Last Planner® System. *Proceedings of the 31st Annual Conference of the International Group for Lean Construction (IGLC31)* (pp. 1337–1349). https://doi.org/10.24928/2023/0217

Moragane, H. P. M. N. L. B., Perera, B. A. K. S., Palihakkara, A. D., & Ekanayake, B. (2024). Application of computer vision for construction progress monitoring: a qualitative investigation. *Construction Innovation*, *24*(2), 446–469. https://doi.org/10.1108/CI-05-2022-0130