A Chatbot System for Construction Daily Report Information Management

J. Cho[a] and G. Lee[a]

[a]Department of Architecture and Architectural Engineering, University of Yonsei, South Korea
E-mail: jkun86@naver.com, glee@yonsei.ac.kr

Abstract –

Consistently updating, analyzing, and managing construction-related information is one of the key success factors in project management. Quite a few construction projects have recently started to utilize instant messaging (IM) applications such as Slack, WhatsApp, and WeChat as a communication channel among project participants to share daily construction information due to easy accessibility. However, general contractors are still required to manually extract and integrate the data from instant messages to compose daily reports. This is because the data inputted by subcontractors through IM applications are usually in an unstructured form and the IM application is not normally interoperable with the systems database especially developed for construction management. To solve this problem, this study proposes a chatbot-assisted construction daily report data management system. The chatbot in the proposed system collects and processes the required information through conversations with subcontractors, and automatically generates and shares a daily report for general contractors. A prototype system has been designed and implemented to prove the concept.

Keywords –

Construction daily report; Chatbot; Instant messaging application

1 Introduction

Consistently collecting and analyzing construction management related information, such as the schedule and cost, is very important for successful project results due to the inherent risks and uncertainty of the construction industry. Therefore, a systematic communication plan or process between the general contractor and subcontractors needs to be defined for efficient data collection.

However, it can be problematic if all participants do not utilize a unified communication channel or if a low level of technical understanding about the system prevents them from using it. Nevertheless, the use of everyday instant messaging (IM) applications at construction sites has been on the rise as a channel of information exchange. There are many advantages to using popular IM applications, such as interoperability with external systems, searching for information, and exchanging files.

In addition, chatbot technologies can be deployed. A chatbot is a service that provides a communication interface with users based on rules or artificial intelligence (AI). An AI-based chatbot answers based on the understanding the intentions of texts from users using natural language processing (NLP), while rule-based chatbot can answer only when detect predefined expressions[1]. When using chatbot technologies that understand intent and context from a user's utterances, and provide the necessary information through an interactive interface, there is the potential to support the user's efforts to collect and unify information. Therefore, this study presents a chatbot system model that collects construction daily report data through conversations with users, such as general contractors and subcontractors, and automatically generates daily report.

This paper is organized as follows. First, the requirements of the chatbot system are defined through the literature review. Then, the system architecture and dialog model for the two main functions of data extraction and report generation are explained. Finally, the functional feasibility was validated through the development of prototypes.

2 Literature Review

2.1 Limitations of Previous Studies on Construction Daily Report Systems

Many studies have suggested a system model for troubleshooting time-consuming and inefficient processes performed each day for collecting data and creating construction daily reports.

Russell et al. emphasized the importance of readily and reliably collecting data while presenting an early model of the computing system [2]. However, even with

the advantages of storing and utilizing information, the proposed early system model was not able to relieve the burden on the business because the manager still had to input all the information at construction field. In a case study lead by Yan-chyuan et al., a system model in which a daily report module extends its functionality by incorporating the estimation, pricing, and accounting system to enable cost management [3]. However, it was also not clearly defined for the main body to enter the necessary report data.

Subsequently proposed system models were designed to allow subcontractors to enter construction daily report data. The system model designed by Chin et al. would reduce a manager's tasks and enable plan-completion management by entering detailed level of work items by subcontractors [4]. More recently, a number of BIM-based cloud computing software programs, such as BIM360 and Procore, allow for effective construction project management by reflecting the system model explained above.

However, while this software provides strong management functionality, it has a major issue in that the systems require users to have a high level of technical understanding. Several factors, such as lack of education programs and low levels of subcontractor technical expertise, have been cited as the main cause of the problem with regards to using a unified platform [5].

2.2 Possibilities and Limitations in Current Use of IM Applications for Daily Report Information Collection

Recently, participants of quite a few construction projects have communicated using IM applications originally developed for nonbusiness purposes [6].

IM applications can provide better management capabilities when coupled with external management systems, such as a BIM-based construction management system explained above. However, due to the lack of connectivity, managers are currently required to spend additional time reformatting, integrating, and re-entering the same information.

This study proposes a "chatbot" as a solution to such redundant and manual data-input problems. In the construction domain, NLP has been mainly used for analysis of documentations for checking code compliance [7], retrieving similar project cases [8], and managing safety factors [9]. However, there are no studies utilizing NLP for construction daily report related data management.

In order to automatically manage information related to the construction daily report through communications with the chatbot, the specific model of the chatbot system needs to be designed to understand the messages from the users, to automatically extract the necessary information or store it in the database, and to automatically generate relevant documents for users.

3 Requirement and Coverage Definition of the System

3.1 Requirement of the System

The system requirements defined through the literature review are as follows:

1. The chatbot system should be able to extract the information related to the daily report from the messages or through the conversation with the chatbot and automatically store it in a linked database.
2. The chatbot system should be able to generate reports automatically based on information from the database.

With respect to the first requirement, it is necessary to define which types of information should be handled by this system. In this research, information types are defined based on prior research by Song [6]. Song has analyzed conversations that have taken place using IM applications on seven actual construction projects. Through the analysis, "date," "name of company," "types and number of labor," "types and number of equipment," and "work items" have been defined as primarily shared information types through IM applications. Figure 1 is an example message for daily reporting, including the five information types used in construction projects.

With respect to the second requirement, automatically created documents are defined as including "project name," "date," "total number of units per labor type," "total number of units per equipment type," and "work item list" as shown in Figure 2.

In order to calculate the total number of laborers and equipment used, the sum of the numbers entered by different subcontractors must be determined.

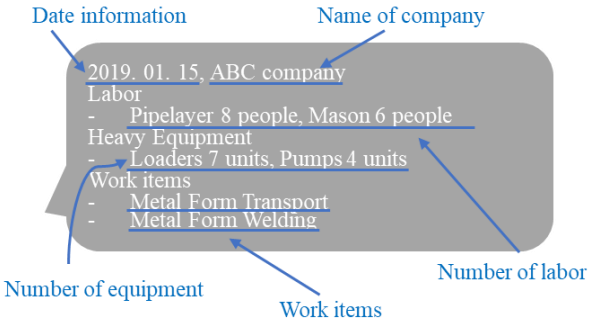

Figure 1. Mainly shared information types in an example user message for daily reporting

3.2 Coverage of the Defined Information Types in the System

In the case of labor and equipment, it has been determined that it is generally entered in a structure that follows the order of "type," "quantity," and "unit name," such as "Mason 6 people" in Figure 1. Therefore, in this study, a regular expression (regexp) approach is applied to detect the types "number of labor" and "number of equipment."

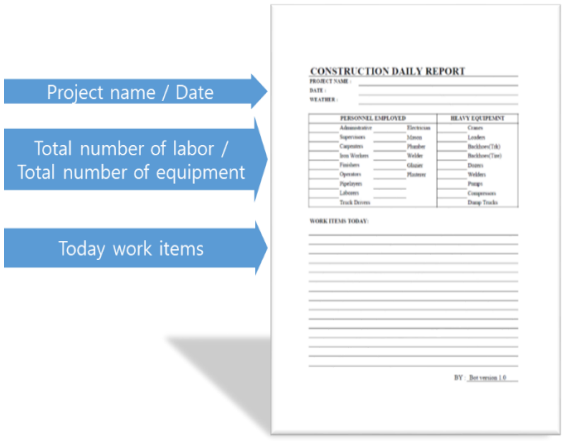

Figure 2. defined report format and included information types

In defining the regular expression for the "number of labor" and "number of equipment," because the detailed items of both types may vary depending on the breakdown structure applied, the items in Table 1 are temporarily defined as labor and equipment items that the system deals with.

Table 1. Temporary breakdown structure of labor and equipment covered by the chatbot system

Labor types		Equipment types
Carpenter	Electrician	Crane
Supervisor	Mason	Loader
Ironworker	Plumber	Backhoe
Glazier		Dozer
Operator		Welder
Pipelayer		Pumpcar
Truck driver		Compressor
Plasterer		Dump Truck

Individual input about work items for the day is designed to be made through a dialog with the chatbot because it is determined that a phrase about work items for the day follows a declaration expression such as "work item" in Figure 1 and is not entered in a structured form.

The architecture of the chatbot system to meet the defined requirements will be explained in the next section.

4 Design of the Chatbot System

The architecture of the system to meet the above requirements is designed as shown in Figure 3. This system consists of a web application, a chatbot module, and a back-end module that connects to IM applications.

The web application receives the text input from the user through the IM application and transfers it to the chatbot module. The chatbot module then analyzes the messages from the users and generates a responding sentence or call using several functions of the web application and utilizing the back-end module.

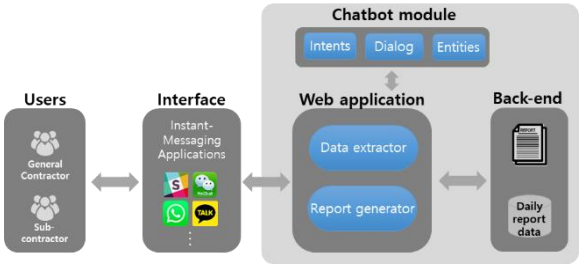

Figure 3. The architecture of the proposed system

The chatbot is an AI module based on Natural Language Processing (NLP) and it plays a role in understanding sentences that are received from users and then producing appropriate responses. The chatbot module is composed of "intent," "entity," and "dialog." Intent represents the purpose of the input by a user. All types of requests users made by users are defined as intent.. Entity means a term or object related to intent. Entity is used to provide a specific context for intent. Dialog is a branching conversation flow that defines responses to the defined intents and entities.

The web application consists of a "data extractor" and "report generator." Both modules are activated by function calls derived from the analysis results of chatbot modules.

The back-end system includes a database for storing extracted daily construction information obtained through message analysis and a file system for managing daily construction documents that are generated automatically upon user request.

In the following section, the design of the dialog model, intents, and entities for information input and report generation in the chatbot module are explained.

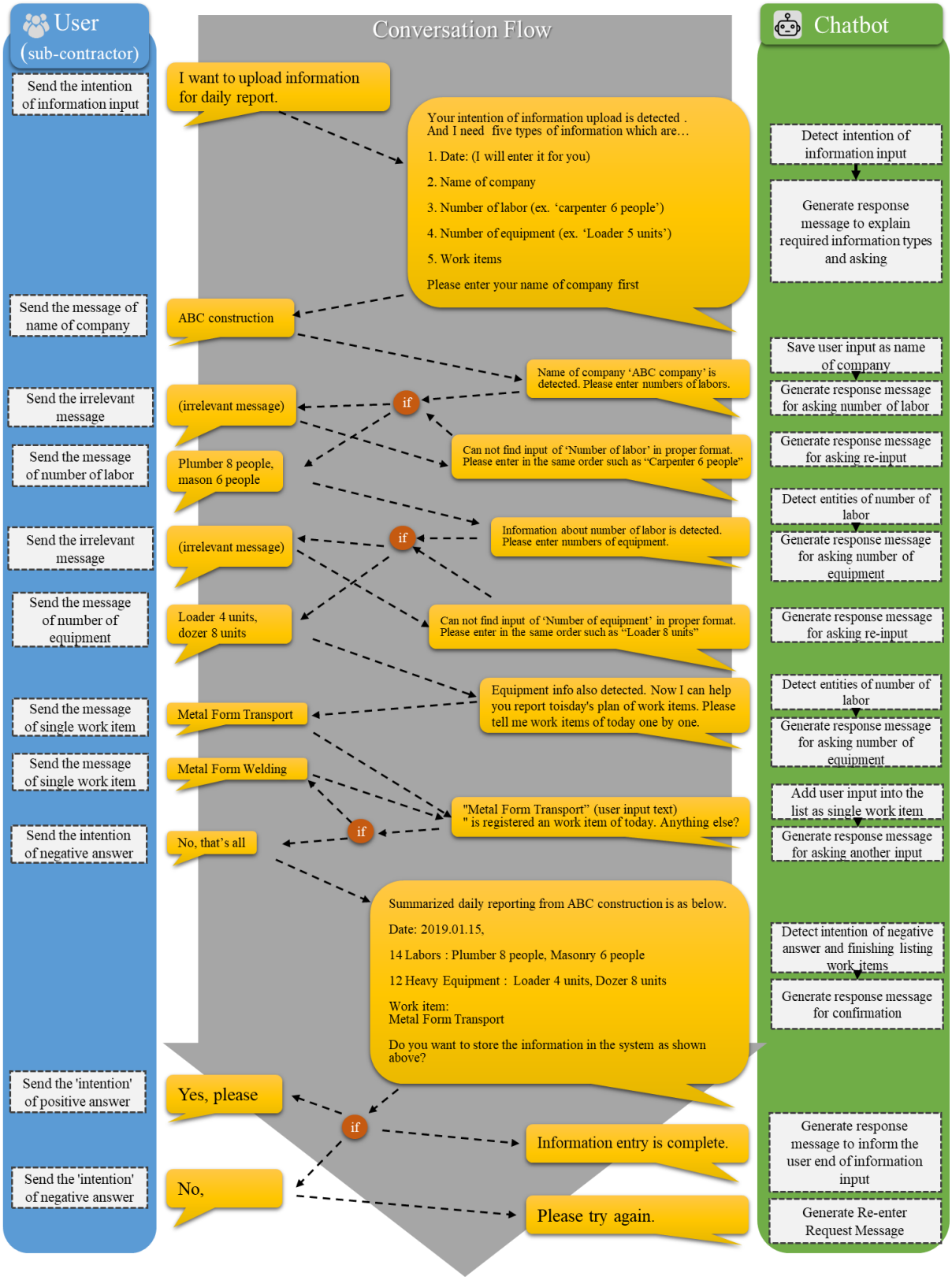

Figure 4. Dialog design for information input

The dialog model includes example messages between the user and the chatbot. Messages from the chatbot are generated by detecting one of the predefined "intents" and "entities" based on the context of the dialog.

4.1 Chatbot Module Design for Information Extraction

First, the dialog model for information extraction has been designed as illustrated in Figure 4.

The conversation begins with the chatbot detecting a message, representing the intention of the user to input information. Subsequent conversation flows are designed for chatbot to ask users for information in the order of labor, equipment, and work item. Finally, it is designed for the user to receive a final confirmation message generated by summarizing the information.

Table 2. Defined intents and examples of user utterances

Intent	Example expressions
Information input	"I want to upload information for the daily report" "Information input for daily report"
Positive answer	"Yes, please," "Great"
Negative answer	"No, it isn't," "No"

The types of intent to cover the dialog model defined above are given in Table 2. All the example expressions from users are necessary for the NLP training of the AI chatbot. The intent "information input" is designed to start a conversation process about information extraction with the chatbot. The "positive answer" and "negative answer" are designed for the confirmation step in the last part of dialog.

Table 3. Defined entities and value types

Entity	Value type	Structure description
Date	Patterns	Combination of year, month, and day
Name of company	Synonyms	Synonym from predefined list
Number of labor	Patterns	Combination of type, number of quantity, name of unit (Figure 5)
Number of equipment	Patterns	

The types of entity are defined in order to detect the types of information and extract them from the user messages as designed in Table 3. The remaining four information types were defined as "entities" for this dialog model, except for the non-structural form of work item mentioned previously.

The intent type "date" borrowed the most common form of structure, and the intent type "name of company" is defined as matching the text within a list of company names.

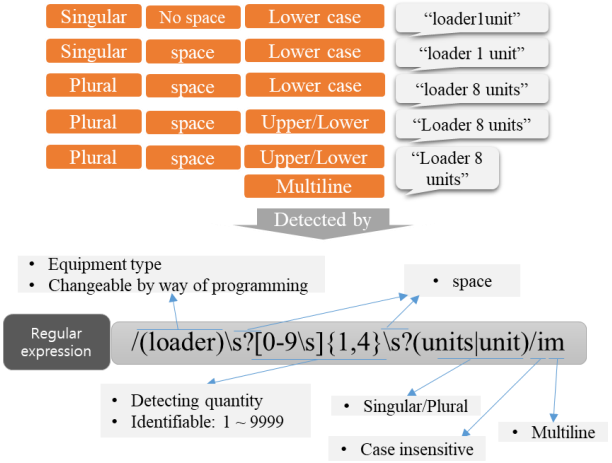

Figure 5. Designed regular expression approach for entity detection

Intent types "number of labor" and "number of equipment" takes the identical approach with regards to the pattern. Figure 5 describes the structure design of the regular expressions for recognizing the example text for the number of equipment. The pattern structure consists of "equipment type," "number of quantity," and "name of unit." The regular expression structure also includes consideration for "singular and plural form," "existence of space," "upper case and lower case," and "multiline" for the text from users.

Although Figure 4 explains the longest possible flow of conversation, based on the defined types of entities, it is also possible to have a scenario in which the chatbot module extracts all of the information for each of the four types from a single message sent by the user.

4.2 Chatbot Module Design for Report Generation

The dialog model for daily report generation has been designed as illustrated in Figure 6. The conversation begins with the chatbot detecting a message representing the intention of the user to request a document. Subsequent conversation flows are designed for the chatbot to ask for the user's e-mail address to send the generated document. The conversation is finished when the user receives an e-mail with the generated daily report.

This dialog model is designed for short conversations; only one type of intent "requiring document" is designed. Several example expressions, including the message

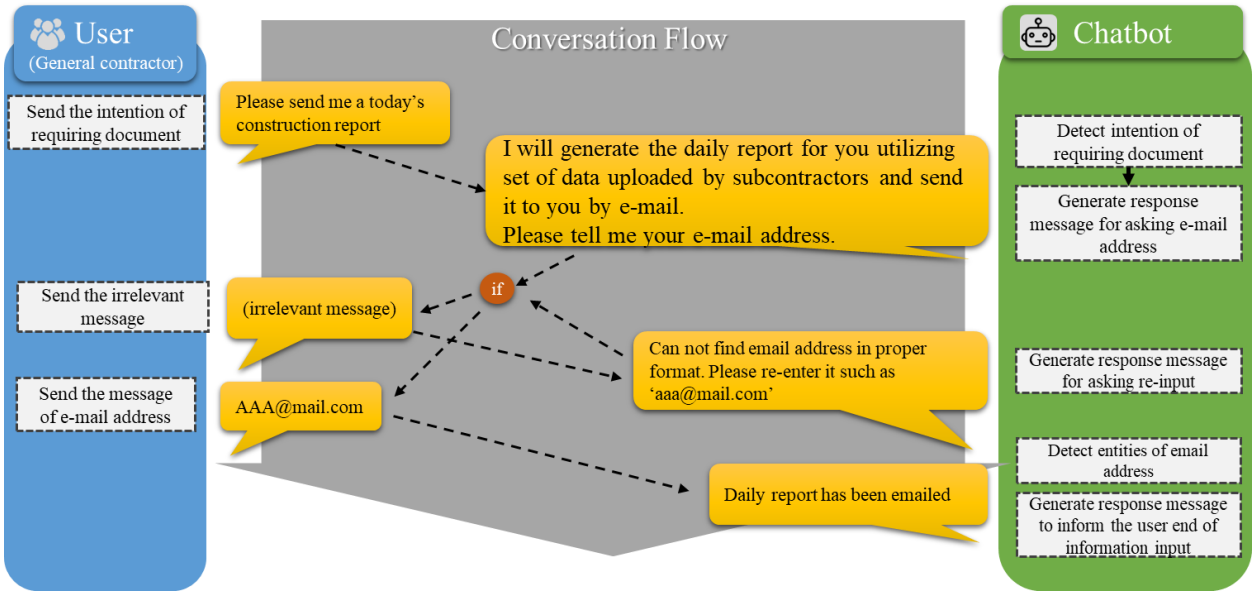

Figure 6. Dialog design for daily report generation

"Please send me today's construction report," are required for AI training as shown in Figure 6.

A single entity type is defined to obtain an e-mail address from a user in regular expression pattern.

It is also possible to skip this dialog in actual system implementation because once the system saves the email address, or has alternative information such as a phone number, the chatbot can automatically generate the documents and send them to the user on a regular basis at a defined time.

5 Prototype Implementation

5.1 Configuration of the Prototype System

A prototype of the proposed system as depicted in Figure 7 was developed to check the functional feasibility to determine how well it satisfies the requirements identified in the requirements analysis phase of system development.

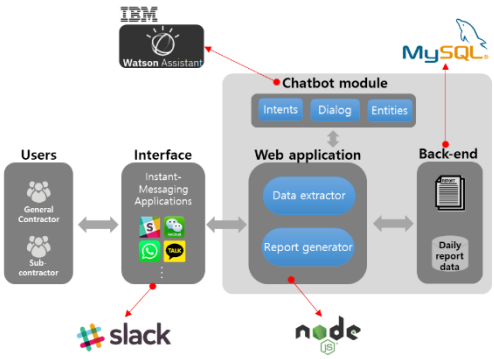

Figure 7. Configuration of the system prototype

First, the collaborative platform Slack, a well-established IM application, has been utilized for connecting users and the chatbot system.

The chatbot module has been implemented with Watson Assistant, which is an AI as a Service (AIaaS) provided by IBM for designing and implementing chatbots.

The web application has been implemented as a Node.js application. Node.js is an open source development platform for executing JavaScript code server-side. In the development of the Node.js web application, several packages shared through NPM (Node Package Manager) are included as dependencies of the system as explained in Table 4.

Table 4. Utilized packages for Node.js web application in the prototype

Name	Role
botkit	Integrating IM application and chatbot service
Express	Providing web application framework
Node-MySQL	Handling database
XLSL	Generating and updating excel file
Nodemailer	Sending e-mail

The "botkit" package has been utilized to integrate

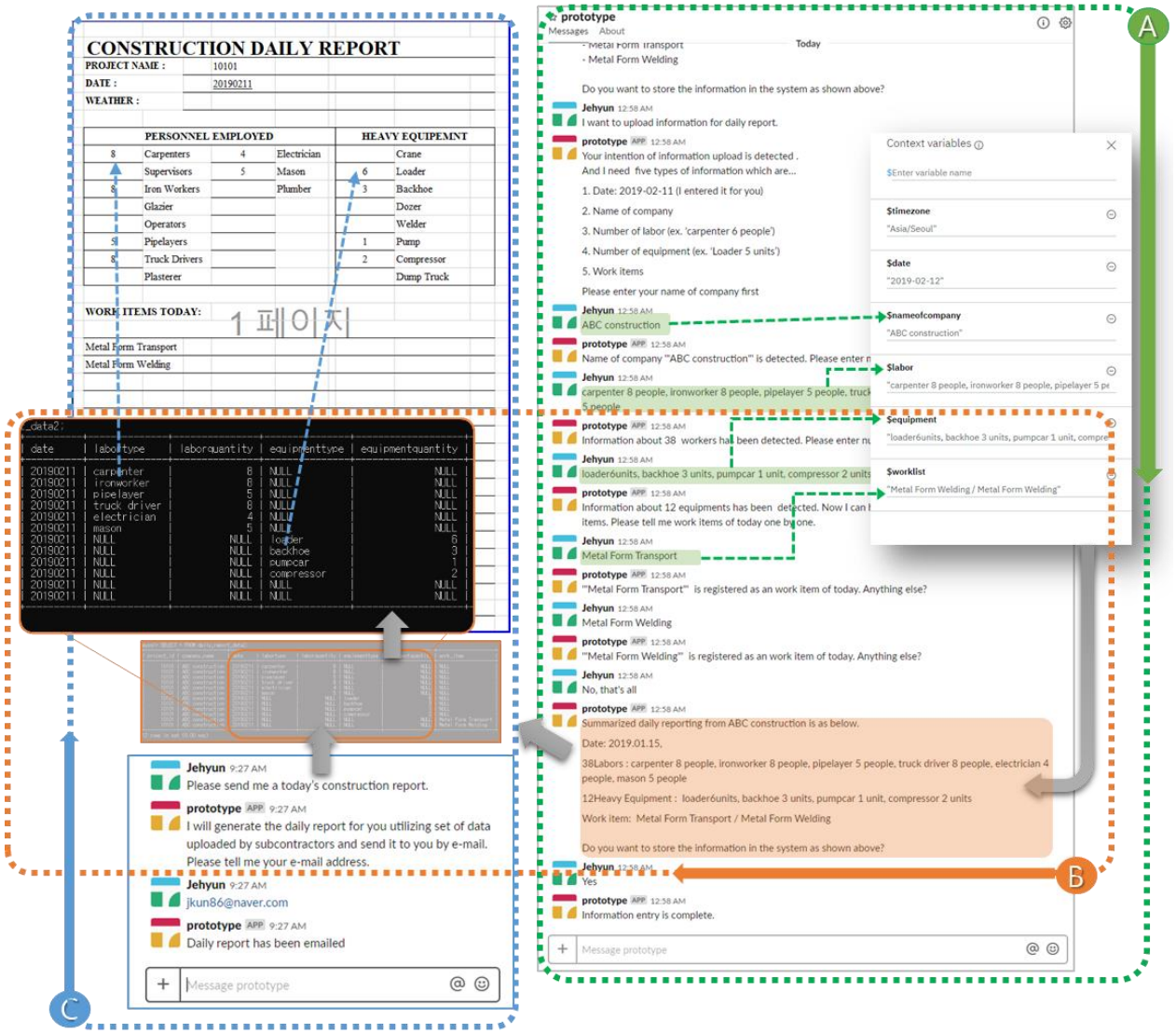

Figure 8. Implementation of the dialog models in the prototype system

the IM application and the chatbot service, which are Slack and Watson Assistant in the prototype. The "Express" package has been utilized for operating the server. The "Node-MySQL" package has been utilized for handling the database. The "XLSX" package has been utilized to generate and update the Excel file for the daily report. The "Nodemailer" package has been utilized to send emails to users.

The database system has been implemented with MySQL, which is a popular open source database.

5.2 Implementation of the Designed Dialog Model in a Prototype System

Data collection and automated report generation were tested using the prototype system based on predefined dialog models.

First, as depicted in area A of Figure 8, it is confirmed that, except for the "date" information, information related to the "name of company," "number of labor," "number of equipment," and "work items" was properly extracted from user messages, saved in the form of context variables during the process of conversation, and finally utilized for generating confirmation message.

Following this, as described in area B of Figure 8, user-confirmed data was parsed and stored by the web application as multiple rows in the database. In the case of "number of labor" and "number of equipment," quantity information was extracted and stored separately from the structured text such as "8" in the example "Carpenter 8 people,".

Finally, as shown in area C of Figure 8, a document was automatically generated by the web application by entering the information from the database into the

report's Excel form, which is predefined in advance when requested by the user.

5.3 Discussion

Through the experiments, the chatbot system has been identified as a functional and feasible system model that is expected to provide the following benefits:

- It can serve as a basis for enhancing management functions because IM application data is no longer volatile; it is now part of the system environment.
- This system model is beneficial to both user groups. The subcontractors do not require any special courses in the system input process, and the burden of data collection and documentation is reduced for the general contractors.

Nevertheless, the proposed system model has some problems to overcome at the current level.

- In the case of labor and equipment, the structure of text was borrowed from a previous study. However, for the "work items," because each subcontractor has a different work breakdown structure (WBS) and the text of the message is mixed with information about the work location in an unstructured form, it needs to incorporate a cognitive approach based on standardized WBS and LBS (Location Breakdown Structure).
- Due to the nature of the interactive interface, compared to the visual user interface, it is necessary to extend the dialog model in terms of convenience to meet various user scenarios. For example, it may be possible to modify only some of the total information entered, or to re-check the information entered.

6 Conclusion

This study presented a chatbot system model that automatically extracts information related to the construction work daily report exchanged in IM applications and automatically generates the construction daily report. The proposed system model includes a chatbot module, web application, and database. Several types of "intents" and "entities" are defined as needed for the conversation between users and the chatbot, along with dialog models. Detected information through the conversation between the user and chatbot could be utilized by the web application for data extraction and document generation using a database.

The prototype was developed and validated to determine whether the chatbot system can perform functions related to the requirements of data extraction and document generation.

The proposed system is expected to provide benefits for subcontractors in terms of ease of use and for general contractors with regards to reducing the effort to collect and document information. In addition, studies required for structural recognition of work items and functional range extensions for other management purposes, such as schedule and cost management, are further expected.

Acknowledgements

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.2017-0-00701(2nd year), Development of the Smart Field Platform for Planning and Managing Building Construction), grant (no. 19AUDP-B127891-03) from the Architecture and Urban Development Research Program funded by the Ministry of Land, Infrastructure, and Transport of the Korean Government and partially supported by the Graduate School of Yonsei University Research Scholarship Grants in 2017.

References

[1] Shridhar K. *Rule based bots vs AI bots. Online:* https://medium.com/botsupply/rule-based-bots-vs-ai-bots-b60cdb786ffa, Acessed: 2017, 03/22/2019.

[2] Russell A. D., Computerized daily site reporting, *Journal of Construction Engineering*, 119 (2):385-402, 1993.

[3] Shiau Y. and Wang W., Daily report module for construction management information system, *Proc. 20th Int. Symp. on Automation and Robotics in Construction*, 2003.

[4] Chin S., Kim K., and Kim Y.-S. J. J. o. c. i. c. e., Generate-select-check based daily reporting system, 19 (4):412-425, 2005.

[5] Construction M. H., "The Business Value of BIM for Construction in Major Global Markets." McGraw Hill Construction, Bedford, MA.," McGraw Hill Construction2014.

[6] Song T. S., "Extracting daily report information from text messages between the field managers," Master, Architecture & Architectural Engineering, Yonsei University, 2018.

[7] Zhang J. and El-Gohary N. M., Integrating semantic NLP and logic reasoning into a unified system for fully-automated code checking, *Automation in Construction*, 73:45-57, 2017/01/01/ 2017.

[8] Zou Y., Kiviniemi A., and Jones S. W., Retrieving similar cases for construction project risk management using Natural Language Processing techniques, *Automation in Construction*, 80:66-76, 2017/08/01/ 2017.

[9] Tixier A. J. P., Hallowell M. R., Rajagopalan B., and Bowman D., Automated content analysis for construction safety: A natural language processing system to extract precursors and outcomes from unstructured injury reports, *Automation in Construction*, 62:45-56, 2016/02/01/ 2016.