# ACCESSING CLASS MATERIALS FROM LINE: AN AI CHATBOT APPROACH FOR STUDENTS

Eduardo Carabez*, Takanori Sato

National Institute of Technology, Akita College / Department of Electrical and Information Engineering, Akita, Japan

Eduardo Carabez* (carabez@akita-nct.ac.jp)

**We present the development and implementation of an AI chatbot integrated with the LINE messaging application to facilitate student access to class materials, syllabi, and academic resources. Leveraging Google's Dialogflow, an online natural language understanding platform, the proposed chatbot aims to address the limitations of traditional information retrieval methods based on physical printouts and online repositories. A key focus of this research is the chatbot's ease of use to both teachers and students.**

**By utilizing LINE, a familiar and widely adopted messaging platform in Japan, students can intuitively interact with the chatbot using natural language. This eliminates the need for complex navigation through menus and printouts, simplifying the process of accessing vital course information. Dialogflow's machine learning algorithms allow the chatbot to continuously improve its understanding of student needs and refine its responses over time.**

**To setup the chatbot, teachers don't require AI knowledge and just need to upload their existing class materials (printouts, presentations, etc.) to Dialogflow's platform and provide a set of possible message interactions with the students and their desired outcome. The chatbot's core functionality relies on the robust capabilities of Google's Dialogflow. This platform is available 24/7 and enables the chatbot to understand and interpret student queries at their convenience, even with variations in phrasing and syntax, look for material that matches the query request and show such materials to the student on its entirety or in the way of a chat answer. This paper details the development process, including the design of the conversational flow and the integration of Dialogflow with the LINE messaging API. We also present preliminary findings from a pilot study, evaluating student perceptions of the chatbot's usability and effectiveness. We conclude by discussing the potential for future enhancements and broader applications of AI chatbots in education, particularly in leveraging messaging platforms to broaden access to academic resources.**

***Keywords:*** *AI, chatbot, dialogflow, accessibility*

## Introduction

The way students interact with classes and the materials provided in them is changing rapidly as technological developments allows a smoother integration of digitalization and accessibility both during classes and after them (Labadze, L. et al., 2023)

One example of this is the rapid spread of online class platforms such as Microsoft Teams and Google Classroom. Platforms such as these allow students to access class materials and information outside of school at the time they find it convenient. This is also beneficial to teachers, as they can rapidly communicate class updates to students, create assignments and evaluate their results, give feedback, etc. However, as more and more classes make use of online platforms, the negative aspects of having to navigate many online repositories to find class materials has begun to become a burden on the academic experiences of students.

When students want to, for example, check the syllabus for a specific class online, they can access the syllabus system, which in its current form doesn't have a smartphone-oriented website and thus provides a suboptimal browsing experience.

Another possible option to access the syllabus might be available if the teacher in charge of a class has uploaded the syllabus to the class platform. In this case, the students need to access the corresponding app or website, go to the page of the class, go through some menus, find the syllabus and either visualize it there or download it. This example illustrates a single case of material searching for a single class, but students constantly need to access and go through multiple classes and their related materials on a daily basis, so having to go through such process to access very simple but necessary and relevant information can become a significant detriment in their academic experience.

To address this issue, we implemented a chatbot powered by Dialogflow, a natural language understanding platform based on large language models, on the messaging application LINE that can provide information from the syllabus and other materials by naturally interacting with its users. The chatbot presented here does not require any level of understanding of AI for its implementation, as all processes involving AI are

already setup and automatically managed on the Google's server side.

In the following sections, we present the steps to necessary to implement the chatbot, from creating a business LINE account, to the process of feeding information to the chatbot platform and the results from a very simple survey with potential end-users.

### Creating and linking necessary accounts

To implement the chatbot we propose, a Google account to access the service called Dialogflow ES (Google's Dialogflow ES access, 2025) and a LINE business account are required.

**(1) The Google account** can be the one provided by the school or any other. Dialogflow ES is free to use at the time of writing of this work and there's a paid version with more features and better understanding capabilities called Dialogflow CX. It should be noted that the free version of Dialogflow used for this work limits the number of queries that can be done per minute, day, week and month as stated in the Dialogflow documentation. All conclusions, comments and descriptions provided in this work are made under the current environment of Dialogflow ES.

**(2) A LINE business ID** must be created and access to the LINE developers console and LINE official Account Manager must be confirmed with this account. In the LINE developer console home, create a provider first, access it, and create a channel of the "Messaging API" type. Access the created channel and look for the Webhook Settings (which are found under the Messaging API tab at the time of writing) and make sure the option to use webhook is on.

**(3) In Dialogflow**, create an agent and name it. After that, on the menu to the left, go to "Integrations" and search for LINE under the "Text based" category. After clicking it, you will be prompted with a floating window as the one seen in Figure 1. Following the instructions provided should be enough to get the following information from the channel created in (2) of this section: Channel ID, Channel secret (both of these were found in the basic settings) and Channel Access Token (found in the Messaging API tab and has to be issued).

Copy and paste these three pieces of information into their respective fields in the floating window. Then, copy the webhook URL provided in the floating window and paste it in the Webhook URL field found in the Messaging API tab of the channel created in your business LINE account (the same one from which the channel ID, channel secret and channel access token were taken from) by clicking "Edit". Once pasted, click "Verify" to see a message displaying "Success". If any error is displayed instead, even though the process was followed correctly, try refreshing the page for the floating window where the Webhook URL is provided in Dialogflow, then repeat the verification process.

By completing these three tasks, the core integration of Dialogflow and the LINE account is completed. Although it is not necessary for continuing, I recommend to access the LINE official account manager and turn off chats and auto-response messages to make sure that all interactions with the chatbot are solely determined by the Dialogflow service.

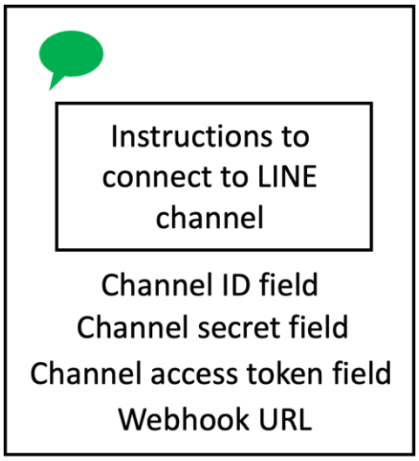


Figure 1. Representation of the floating window where the information to link the LINE channel and the Dailogflow agent is shown.

### Chatbot training and testing conditions

The most direct way to train the chatbot is through the creation of Intents, which are interactions that the chatbot manager expects to occur between the chatbot and the end-users.

To create an Intent, we access them from the "Intents" option on the main Dialogflow menu (to the left on the Dialogflow console at the time of writing) and then click the "CREATE INTENT" button.

In the created Intent, two main drop menus of interest are found: "Training phrases" and "Responses". Training phrases are suggestions of the possible inquiries the chatbot will get from the end-users and must be written by the chatbot manager. The chatbot will learn from these phrases and will be able to understand the intent of the end-users' request if the phrase received is of the same nature are the ones used for training without necessarily being the same phrase. After receiving an inquiry and understanding its intent as one it has been trained to recognize, the chatbot will then respond by using one of the options given to it in the "Responses" drop-down menu. The responses can be simple text messages or more complex application-oriented cards with options (visible after clicking the + button next to DEFAULT responses).

A single Intent should define a single type of interaction. For example, if the students are expected to inquire about the syllabus and also about school rules, an Intent that manages each of those topics should be created for easier management of the chatbot.

Another option to train the chatbot is by providing documentation with the information the chatbot can learn to interact with the end-user without creating specific intent scenarios. This can be done over the "Knowledge" option (to the left on the Dialogflow console at the time of writing). If this feature is in beta, it won't be accessible until turned on using the agent's settings and selecting the "Enable beta features and API's" option. After creating and naming a "KNOWLEDGE BASE", a message stating that no Knowledge has been created will be seen,

followed by a highlighted text to create the first one as seen in Figure 2. To create the first instance of Knowledge the chatbot manager needs to prepare a document based on the archetypes: FAQ or Extractive Question Answering. FAQ is a document formatted so a series of questions and answers are presented one after the other, while Extractive Question Answering can analyze text written without particular structure and extract information for its answers from what it understands of it. As a beta feature, this approach has a lot of limitations, for example, the formatting of the documents, which at the time of writing doesn't allow for tables or lists being used in the documents used to generate Knowledge.


Figure 2. Example of the message seen when no Knowledge has been created and the underlined option to be clicked to create one.

For this project, an intent that prompts a card when asked about the syllabus was created and the training phrases used for this are summarized in Table 1. The response when this intent is recognized is to ask the student who is the professor in charge of the class they are taking by means of an option card shown in the LINE chat. Additionally, we created Knowledge for a class called Network Basics and the information included is: class objective, grading method and overall description of its topics and scope. The document containing the class information was formatted to fit the Extractive Question Answering type, meaning it doesn't have any tables or lists, but it is not formatted as a series of questions and answers. The overall structure of the document can be seen in Figure 3.

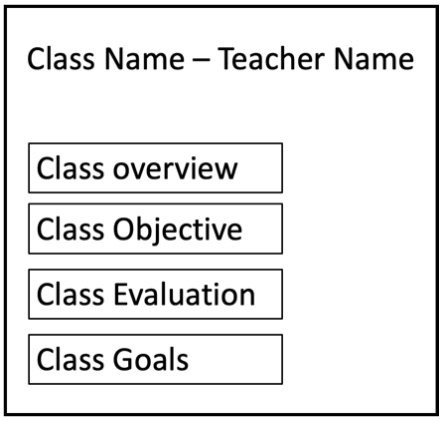


Figure 3. Overall structure of the document used for knowledge generation.

Default responses for the chatbot can be tested in the built-in cat box on the right of the Dialogflow window, which as the time of writing is signalled by a "Try it now" message. Writing in this text box will prompt the chatbot to write a reply from all of its currently created Intents and knowledge. This option only works for responses that are not app-dependant, so any response that uses cards within an app or more complex responses can't be verified using this tool.

Table 1. Training phrases and responses used in the created Intent.

| | |
|---|---|
| Training phrases | Syllabus, list of syllabi, please show me the syllabus, class syllabus |
| Responses | Card response with syllabi available and links to them |

To fully test our chatbot, we need to add it as a Friend on LINE and interact with it. The QR code to befriend the chatbot can be found in the LINE developer console under "Messaging API" settings.

Ten students added the chat as a friend and were asked to interact with the chatbot in no particular way to get the following information: name of the teacher in charge of the class, evaluation method and description of the class.

## Results and Discussion

Opening a chat dialog with the chatbot prompts an automatic welcome response that can be configured in the "Greeting message" option found in the LINE Official Account Manager. A series of control inquiries were made regarding the class to be answered by plain text messages. This is only to establish that the chatbot is working properly when the inquiry specifically targets information found in the document used for Knowledge generation.

After that, the chatbot was asked about the syllabus and the expected answers was to be in the form of a card with 2 options, one for each of the teachers in charge of the subject. However, the first answer was of text type, as shown in Figure 4(b), and contained information from the Knowledge base, which was not the intention. To solve this, the Knowledge Results Preference was adjusted in the Dialogflow console from 1 to -0.7, which prompted the expected response, as shown in Figure 4(c). When the user taps the option corresponding to their teacher, it gets directed to the syllabus website.

To further test the chatbot, 10 students from the 5$^{th}$ grade of the electrical department were asked to use the chatbot freely for no more than 3 minutes to retrieve information for a class and fill out a survey based on their experience (see Table 2).

All of the students succeeded in retrieving the information by performing very different inquiries. While some asked with more formal sentences such as "Who teaches this class?", others simply asked "Teacher?", with both cases ending in the same answer. Although the desired information itself was received, some students felt that the information was accompanied by unnecessary details. When asking for the name of the teacher, they would additionally receive a brief description of the class contained in the course "Class

Table 2. Survey results for the 10 students that tested the chatbot. 1 is for strongly disagree, 2 for disagree, 3 for agree, and 4 for strongly agree.

| | | Level of agreement | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| 1 | Writing to the chatbot was easy and felt natural | 0 | 0 | 1 | 9 |
| 2 | The results from the queries were satisfactory | 0 | 0 | 2 | 8 |
| 3 | Wider implementation of the chatbot could improve the academic experience | 0 | 0 | 0 | 10 |

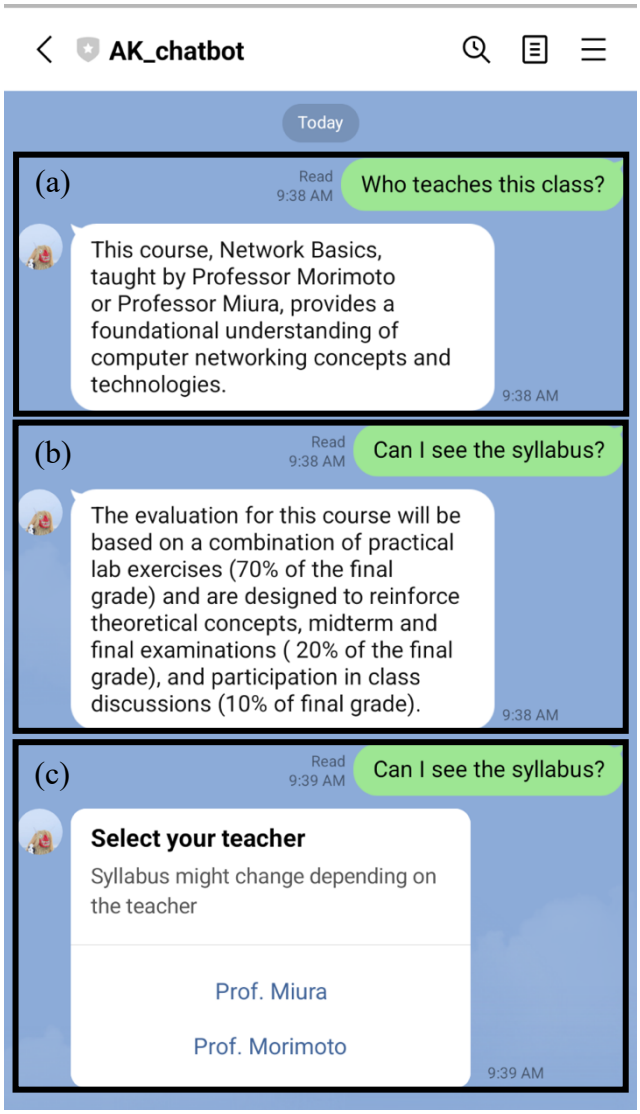


Figure 4. Chatbot responding to queries using Knowledge in (a) and (b), and using Intent (c) to prompt card style response.

overview block" of the document used for Knowledge generation (see Figure 3). On the other side, students were satisfied with the speed of the response and the ability of the chatbot to understand what they wanted to know without having to use specific phrases or writing style.

## Conclusions and future work

The work presented here goes through the creation and linking of the accounts and services needed to implement a chatbot in LINE powered by the Dialogflow platform. In our testing of the chatbot, students successfully retrieved specific information from it without receiving any kind of training. Furthermore, the students had overall a good experience interacting with the chatbot, and thought of it as a convenient way to get school information.

The chatbot implemented here runs on the free version of Dialogflow and has limited query quotas (Dialogflow documentation, 2025), which would impact the school-wide application and testing of the instance of the chatbot presented in this work. Opting for a paid Dialogflow service would allow not only for more students to access the chatbot, but also the access to more advance tools for training the chatbot that are not available in the free version.

Overall, we found that the proposed chatbot approach was promising for facilitating class materials and information. However, further testing is required to elucidate whether Dialogflow fits the needs of the KOSEN system on a broader scale. Furthermore, issues regarding privacy and the use of personal information have to be carefully considered, as in its current state, the chatbot can be befriended by any person with a LINE account and access to the friend QR code.

Going forward, we will provide the chatbot with Knowledge and Intents to cover more aspects of a class and test it real scenario to address how efficient this approach is in real life and assess the costs that deploying this model school-wide would incur. Additionally, alternative options to Dialogflow will be considered, with Microsoft Azure AI Bot being considered due to the possibility of using it with LINE in a similar way to the one presented in this work (Azure AI bot for LINE, 2025), as well as within Microsoft Teams, a service that has been widely adopted by the KOSEN system.

## References

Labadze, L., Grigolia, M. & Machaidze, L. Role of AI chatbots in education: systematic literature review. Int J Educ Technol High Educ 20, 56 (2023).

Dialogflow SE access. Retrieved on May 13, 2025 from https://dialogflow.cloud.google.com/

Dialogflow documentation. Retrieved on May 13, 2025 from https://cloud.google.com/dialogflow/

Azure AI Bot for LINE. Retrieved on June 13, 2025 from https://learn.microsoft.com/en-us/azure/bot-service/bot-service-channel-connect-line?view=azure-bot-service-4.0