Final Report

[Industrial Based]
AI Bot to Make the Best Decision for the Customer

Course Information:

COMP4801 Final Year Project

Supervisor: Dr. Kenneth Wong

Author:

Tan Zhanwen, Francis 3035028518

Date of Submission: April 16th, 2017

Abstract

As an industry giant in IT products, Microsoft produces a lot of electronic products, like computers, tablets, etc. Some of these products are very alike in terms of hardware like CPU power, RAM size, which makes it difficult for customers without much knowledge in hardware to find the most suitable product. In addition, customers are not able to talk to salespersons at any time they want. This project is an industrial-based project initiated by Microsoft to solve the aforementioned problem by developing a product recommendation system, in the form of an online chat box which can talk to customers online and recommend the most suitable product.

Up to now, a functional Cantonese chat box has been successfully built and hosted on Microsoft Azure platform, which can understand human languages, recommend based on user characteristics, and handle purchase or complaint requests from customers. Also, it can be accessed through Skype and Facebook Messenger, which are two popular communication application in HK.

Acknowledgements

The progress of the project *AI Bot to Make the Best Decisions for the Customer* could not have been made without the help of following people. We would like to express our most sincere gratitude to them.

Dr. Kenneth Wong, supervisor of this project, for giving directions and support throughout the designing and implementation of this project.

Dr. Anthony Tam, second examiner of this project, for facilitating industrial connection and giving valuable comments during our preliminary presentation.

Mr. Samson Lee, contact person from Microsoft, for providing useful materials and helping us tackle technical problems.

Table of Contents

Abbreviation

AI	Artificial Intelligence
API	Application Programming Interface
CPU	Central Processing Unit
DBMS	DataBase Management System
LUIS	Language Understanding Intelligent Service
NLP	Natural Language Processing
RAM	Random Access Memory
SDK	Software Development Kit
SQL	Structured Query Language

1 Introduction

Back in the late 20th century, information technology was not advanced and there were few choices for digital products provided by Microsoft, such as laptops, smart phones. Therefore, customers did not bother to consider which model to buy during that period as the choices were limited. However, due to the advancement of technology and recent expansion of its market, nowadays Microsoft provides considerable choices for a specific type of product. Even for a particular series, customers are prompted to choose from customized specifications with different CPU, RAM, hard disk space, etc, making it difficult for customers without much technical background to select the most suitable product.

Three main constraints exist when customers of Microsoft would like to query the differences among models in order to find the most suitable one. Firstly, the transformation from on-site to online shopping renders it difficult for customers to seek helps from shop assistants. Secondly, it costs Microsoft much money to hire technicians for providing online support to customers. Thirdly, the service hours of online support are restricted to a certain period of time within a day, as a result of which customers cannot get helps online during other time slots.

This project aims to provide a solution by developing an artificial intelligent chat box as an extensible standalone module to major social network websites or applications which can understand the customers' queries in natural languages and respond correspondingly, so that customers can ask for recommendation when purchasing Microsoft products anytime while human resources are saved. There are two main challenges for us to overcome in order to successfully develop our final product. One is that the language model in Cantonese and the machine learning model need to be carefully designed to tailor to requirements from Microsoft. The other is how to collect sufficient data to train the models, so that the accuracy of the models is high enough to provide good recommendations.

The remaining of this interim report proceeds as follows. In the first place, personal contribution and project status will be presented. Then the previous work in the field and overall objectives of our product are provided, followed by project scope. Then we present our technical details, implementations, together with final results. Next, future improvements and limitations and difficulties encountered are clearly stated. Finally, we summarize this report with a conclusion.

2 Personal Contribution

My personal contributions to this group project include, all codes related to HTTP request sending and receiving, use Microsoft Bot Framework to build user interface like text, buttons, images, flow to guide user to select feature requirements and allow them to change at the end. Also, I created the LUIS language model, trained it (co-worked with Kevin), and connected it to the bot. Also, I made phone calls to Samson, the representative of Microsoft most of the time as I am only one in team who speaks Cantonese.

3 Project Status

Dates	Descriptions	Status
8 Sep – 30 Sep	· Confirm the Scope of Project · Produce the Simplest Type of Conversational Bot Demo being able to respond to multiple queries	Complete
1 Oct – 31 Oct	· Produce Conversational Bot that can respond with Waterfall model / Making use of State	Complete
1 Nov – 30 Nov	· Develop the language model in Cantonese	Complete
1 Dec – 31 Dec	· Train the Language Models with data	Complete
1 Jan – 28 Feb	· Integrate the trained Language Models with Bot Framework	Complete
1 Mar – 31 Mar	· Develop and connect the Bot with database (SQL / MySQL) · Hosted the Bot on Facebook / Web	Complete
1 Apr – 30 Apr	· Develop the Machine Learning Studio model	Complete
1 May – Project Ends	· Refine the Product · Potential further Features to be Added: Face Recognition / Voice Recognition	Complete

4 Existing Recommendation Applications

4.1 AI Bots

There are some other companies using AI recommendation bots to assist customers in buying specific products. *Figure 1* below is an AI bot for an online flower delivery store[1]:

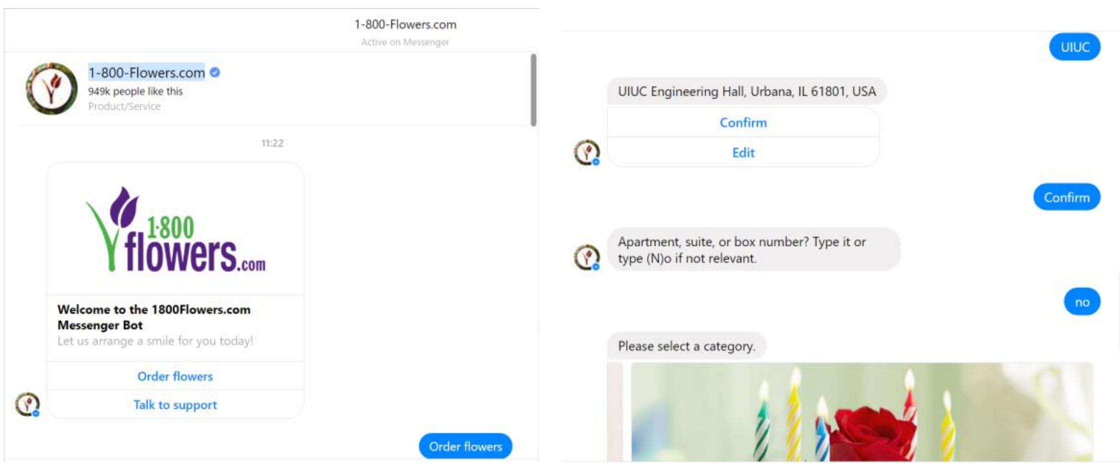


Figure 1: Conversation with another AI bot

The above chatbot is hosted on Facebook Messenger. The interface looks pretty nice and the instructions are clear, by prompting up buttons for use to click. However, users complete the order mostly by clicking the buttons, but are not able to type using their own expression, which makes the bot less intelligent and user-friendly. On the contrary, the bot we developed allows users to type in natural languages, so that it is more user-friendly.

4.2 Online Support

There are some existent chat boxes providing online support to customers for companies like Microsoft and Apple as shown in *Figure 2* [2] and *Figure 3* [3] below:

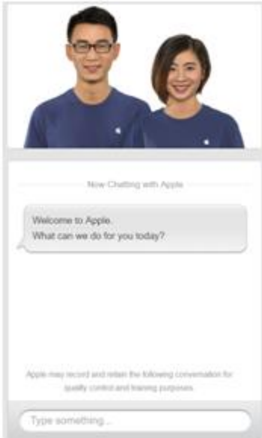


Figure 2: Apple online support

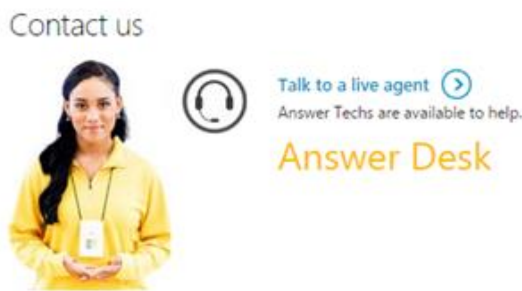


Figure 3: Microsoft online support

For both the chat boxes, the one talking to customers is a real person instead of an AI program. After we typed in some questions, it took around 10 to 20 seconds to get responses. In addition, the service hours are 9am-5pm and 9am-9pm for Apple and Microsoft respectively, which is inflexible for customers who purchase late at night.

Different from the two chat boxes above, our bot will be an AI program that is always online and can respond immediately, so that customers can query anytime and get responses in a timely manner.

5 Objectives

This project aims at creating an artificial intelligent chat box to be used by Microsoft Hong Kong Limited that can understand customers' queries in natural languages through online conversations, and then recommend the most suitable products to the customers. The chat box can be deployed on different channels. After successful implementation of the project, basic queries in Cantonese can be well understood by our chat box with satisfactory responses returned.

Our bot has three distinctive features. Firstly, instead of prompting customers to select from predefined multiple choices, as many other AI bots in the market do, our bot allows customers to query in their own languages. Secondly, machine learning techniques will be used so that our bot can predict what users like based on their personal characteristics. In addition, our bot can be deployed to various platforms, including Skype, Facebook Messenger, so that it can be easily accessible to customers.

6 Scope

The project scope is defined in six aspects, namely the language models, Machine Learning Studio model, SQL Database, interaction logics, products for recommendation, and host platforms.

6.1 Language Models

A Cantonese language model should be developed and trained using LUIS. The language model should be able to understand users' generic queries, like "I would like to use the PC at home", "I use the device mostly for browsing websites and emails", and returns a certain category of products to be recommended. Also, it should be able to identify complaints from users. Language models in other human languages are not considered for this project.

6.2 Machine Learning Studio Model

A model should be created in Azure Machine Learning Studio and trained using user data containing users' personal information and their ratings on different categories of products. When a user logs in our system, his personal information should be analyzed by the machine learning model to predict the suitable category of products.

6.3 SQL Database

An SQL database should be created in Microsoft Azure to host product data. Two tables should be included--- one for storing product information, the other for storing complaints. It should be able to be accessed by our AI bot at any time through online request.

6.4 Interaction logics

Firstly, if a user has registered in our system before, his personal information like age, occupation, and gender should be extracted from the database to be fed into the machine learning model to predict which products he may like. Then these products should be returned to users for recommendation.

Secondly, our bot should be able to handle some generic queries from customers, like “I would like to use the device for games”, “I always take the device with me when I go out.”. After analyzing these sentences, certain types of products should be returned to users.

Thirdly, after product recommendation, a button should be prompted up to ask the user whether he is satisfied with the recommendation. If not, the bot should ask users particular details, like brand and price, to search in the database and give more accurate results.

For each product recommended, it should include the product picture, product name, brief descriptions, and purchase button, so that the user can have a brief impression on what the product is like and directly go to the specific purchase webpage by clicking the button.

If the input from the user is classified as “Complaint”, it should be further categorized as “Product Complaint”, “Service Complaint”, or “General Complaint”, and then be stored in the online database with time, user name, user id, etc.

6.5 Products for Recommendation

For this project, products for recommendation will be confined to the products in the excel file provided by Microsoft, which includes around 200 products. Other products are not taken into consideration.

6.6 Host Platforms

The bot should be deployed on Facebook Messenger and Skype through Bot Framework portal from Microsoft. Other platforms, such as Slack are not specified by Microsoft and thus not in our scope.

7 Technical Details

7.1 Bot Framework

Microsoft Bot Framework [4] is primarily employed as the basis of our bot, which facilitates to build and deploy high quality bots. The framework provides tools to easily develop intelligent bots including basic I/O, language and dialogue skills as well as user connection interfaces.

Bot Framework consists of three major components : Bot Builder SDK, Developer Portal and Bot Directory. *Figure 4* gives a brief overview of these components. The framework also contains an emulator so that we can test our bot locally.

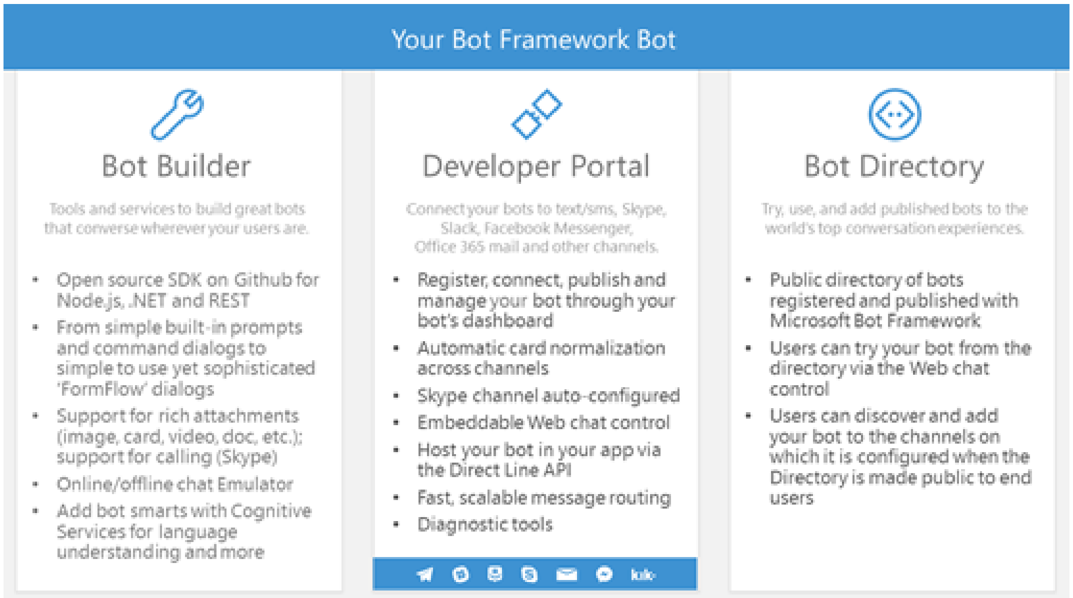


Figure 4: Bot Framework Overview

7.1.1 Bot Builder SDK

Bot Builder SDK provides a code template for us to start with. It provides us with a rich set of libraries, as in figure 5, to call for particular functions, like connecting to LUIS model as in 6.2, prompting up a button, which shortened our development time.

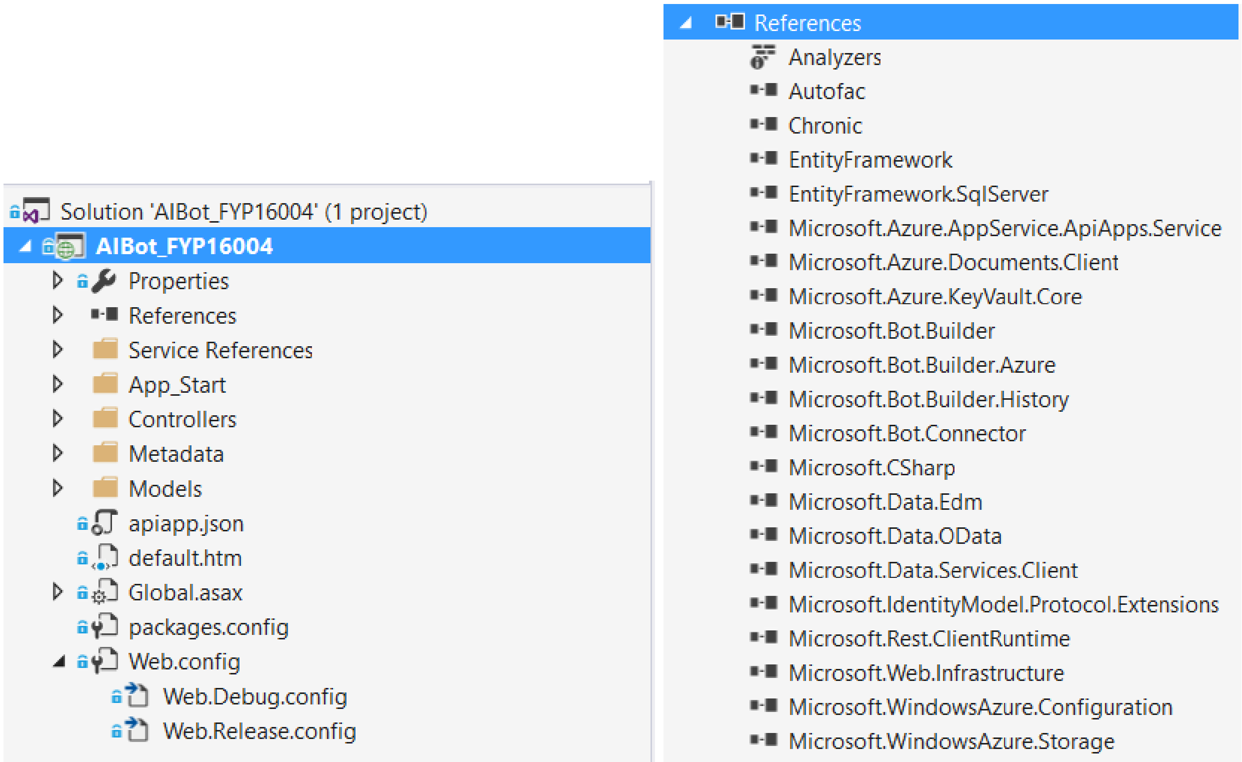


Figure 5: Bot Framework template and libraries

7.1.2 Developer Portal

Developer portal provides us with an interface to inspect the status and control the behavior of our bot. For example, by clicking the test button, a Ping request will be generated to our bot hosted on Azure to test its response. Also, we can inspect the exceptions in our bot by clicking the issues highlighted in red. In addition, we can use this portal to deploy our chat bot to different communication channels, like Skype, Messenger, as shown in “Channels” in the screenshot below. Furthermore, we can also directly talk to our bot using the chat box on the right for quick testing.

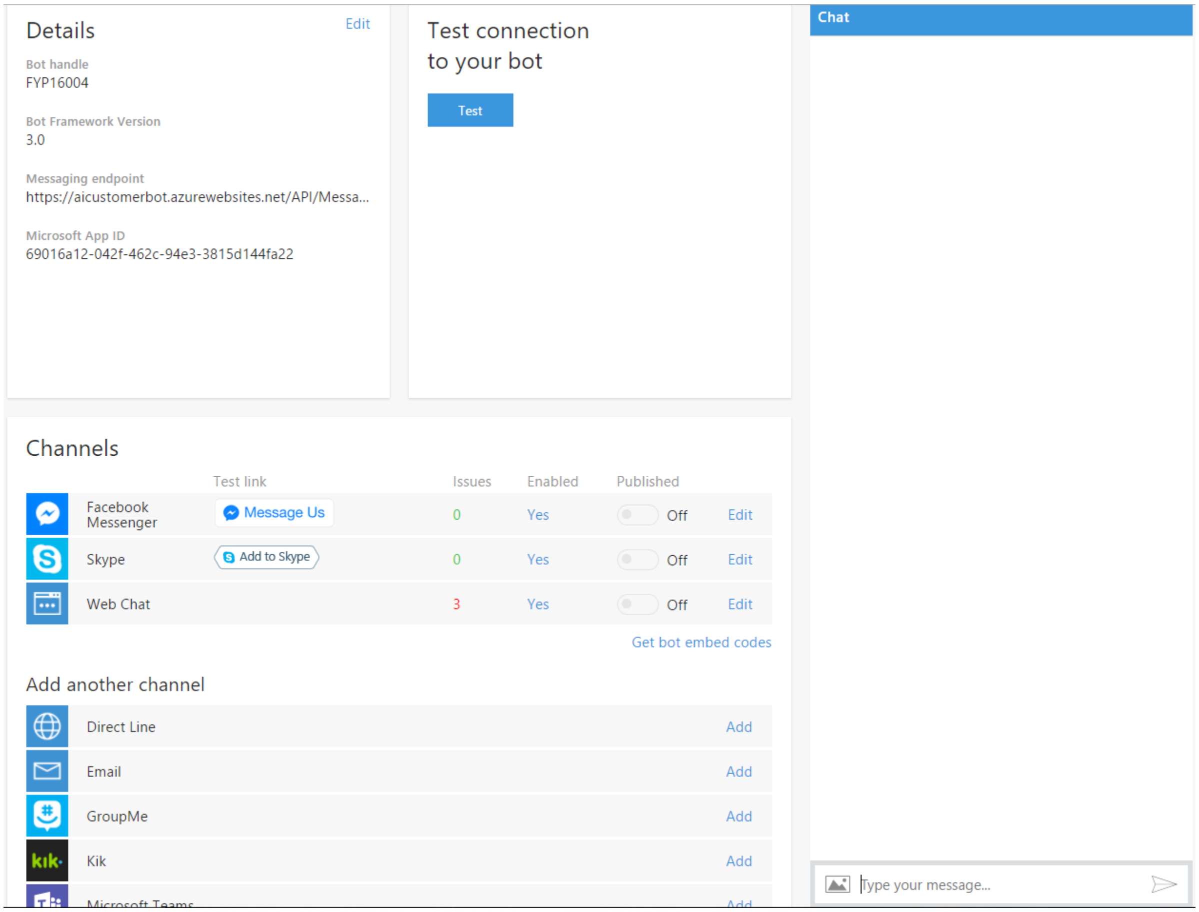


Figure 6: Bot Developer Portal

7.1.3 Bot Directory

Bot directory is the place where all kinds of bots developed through Microsoft Bot Framework are located. This serves to make the bot more easily found by the users and makes integration of the bot to specific channel like Skype more convenient.

7.2 Language Understanding Intelligent Service (LUIS)

In our project, we highly rely on the Microsoft's Language Understanding Intelligent Service(LUIS)[6] to analyze natural languages. In this section, we will firstly give a brief overview about LUIS, and then describe the model design. Finally, the difficulty of transcoding between traditional and simplified Chinese will be discussed.

7.2.1 LUIS Overview

LUIS is the language understanding service provided by Microsoft. The language understanding service is to analyze a sentence, identify and extract the linguistic elements. Through LUIS, developers can create a customized language model by defining intents and entities. Then the model can be trained by inputting labelled data. After training the model, if a sentence is input to the model, certain intent and entities of the sentence will be returned. Below is the interface of LUIS, through which we can find the numbers of intents and entities, number of training data, etc.

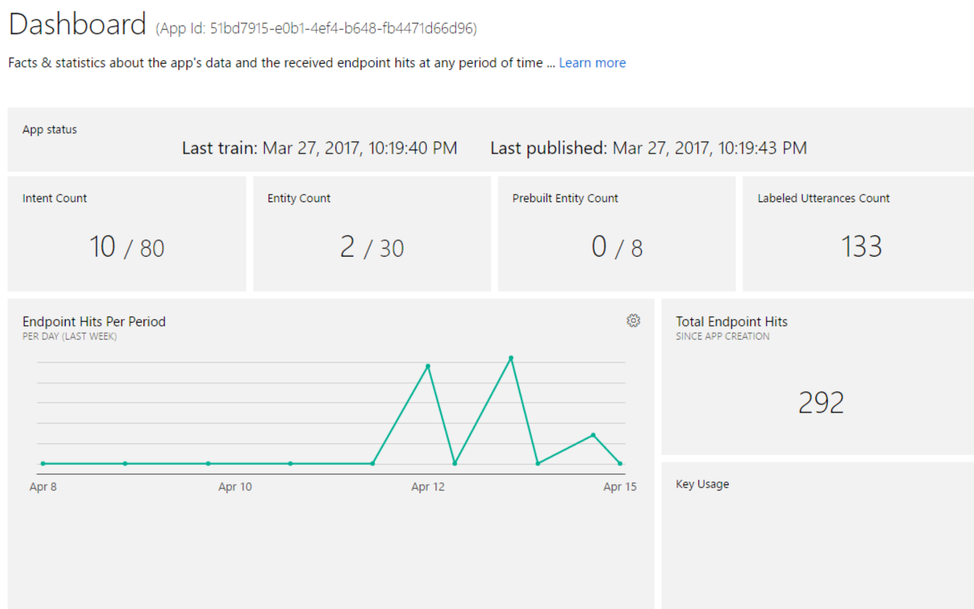


Figure 7: LUIS interface

7.2.2 LUIS Model Design

As mentioned in Scope 5.1, LUIS model should be able to understand some generic queries from users. Below are the sample questions from users provided by Microsoft. Each of the question should be matched to a suggestion, which is a category of products. The requirement is, if a user types a sentence, which is similar to any one listed in figure 8, then the corresponding suggestion 1 should be returned by the model.

These suggestions are “Desktop”, “Notebook”, “Tablet”, “2in1”, “GamingDevices”. In order to achieve the above goal, these 5 suggestions were created as intents in our language model. Questions on the left column were input as training data with the suggestions 1 as labels to train the model.

你會需要你的電腦方便攜帶?	建議 1	建議 2
我通常只會係屋企用	Desktop	Notebook
我有時都會想電腦跟住係身	Notebook	2in1
我會經常外出，電腦需要非常方便攜帶	Tablet	2in1
電腦效能上，你需要你嘅電腦做到什麼？		
我通常只會瀏覽網站同睇電郵	Tablet	2in1
我會上網，同埋用 Microsoft Word, Excel, Powerpoint..	2in1	Notebook
我會上網，同埋用 Photoshop, Illustrator	Notebook	2in1
你平時會點樣睇電影同埋改相		
我有時都會在電腦度睇電影同睇相	Tablet	2in1
我經常會在電腦睇我大量收集嘅數碼電影同睇相	Desktop	
我會用電腦睇高清電影同修改自己嘅相	Gaming Devices	Desktop

你平時會用電腦玩那類遊戲？		
我不會在電腦上玩遊戲	Tablet	2in1
我會玩悠閑遊戲，例如 Minecraft	Tablet	2in1
我會玩遊戲大作，例如 Gears of war 4, Call of Duty…	Gaming Devices	
你會喜歡你的電腦有什麼操作方式		
其實我唔需要鍵盤	Tablet	2in1
我鍾意用 Touch 功能同埋實體鍵盤打字	2in1	Notebook
我鍾意用傳統鍵盤滑鼠做嘢	Notebook	Desktop
你好，我係你哋嘅 Microsoft 產品顧問，今日過到嚟係咪想買新電腦？		
係呀，我想搵部輕啲嘅電腦	Tablet	2in1
我想搵部電腦做文書工作	2in1	Notebook
你想搵部電腦嚟打機定做野？		
我想搵部電腦打機	Gaming Devices	Notebook
我想搵部電腦做文書工作	2in1	Notebook
你用電腦嘅習慣，會唔會都想坐定喺度用？定係會想方便啲會拎出拎入？		
坐定喺度用	Desktop	Notebook
會拎出拎入	Tablet	2in1
平時工作上係咪都要做文書野？定係需要畫畫繪圖 autocad？老闆你做盛行？		
主要做文書工作	2in1	Notebook
需要畫畫繪圖 autocad	2in1	Notebook
做文書野多？老闆你之前 PC 用開乜嘢 window?用開邊個版本嘅 Microsoft Office? 之前 PC 大約幾多錢？		
Yes, 做文書野多	2in1	Notebook
你想搵部機做嘢之餘又可以玩 PC Game？老闆你平時玩開乜嘢遊戲啊？		
我會玩悠閑遊戲，例如 Minecraft	Tablet	2in1
我會玩遊戲大作，例如 Gears of war 4, Call of Duty…	Gaming Devices	

Figure 8: Question Catalog

Also, as our model should be able to identify complaints from users, an intent called "Complaint" was created in the model. As mentioned in the Scope, complaints need to be further classified into "Product Complaint", "Service Complaint", or "General Complaint". In order to achieve this, we created two entities in the model: "Product" and "Service" to identify the complaint type. The rules to decide the type of complaint are:

- Priority 1: If the query has entity "Product", then it is a product complaint, even if the query also has "Service" entity.
- Priority 2: A query that only has "Service" entity is a service complaint
- Priority 3: When there are no entities in the query, it is a general complaint

The source of complaints were from the Microsoft online forum (https://answers.microsoft.com/zh-hant/mobiledevices/forum?auth=1), where people posted issues to be solved.

In addition, intent "None" was created to identify the user queries that are irrelated, e.g. "Can you recommend a movie". "Greeting" intent was created identify greetings from users.

7.2.3 Traditional/Simplified Chinese Transcoding

As LUIS language model only supports simplified version of Chinese. However, people in Hong Kong use traditional Chinese instead during their daily life, which is a mismatch. In order to solve the inconsistency, we used the Microsoft Translator API to transcode the traditional

Chinese to simplified Chinese. The logic flow is, after the user types a query in traditional Chinese, we firstly use the translator service to transcode into simplified Chinese, and then input the simplified Chinese into LUIS model.

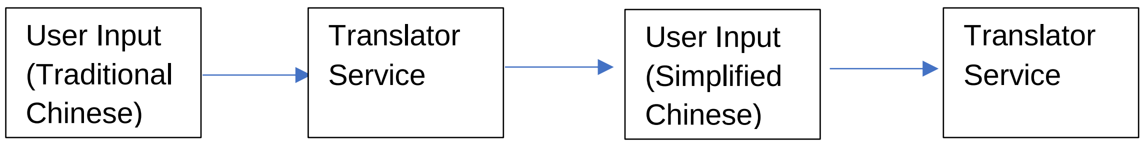


Figure 9: Workflow of translator service

7.3 Machine Learning Studio Model

7.3.1 Overview

Microsoft Azure Machine Learning Studio is a collaborative, drag-and-drop tool you can use to build, test, and deploy machine learning models on your data. After training the Machine Learning Studio model, it can be published as web services that can easily be consumed by custom apps. We need this in our project because we would like to predict what kind of product a user likes based on his personal characteristics. This is reasonable because when a user logins in his/her account, Skype or Messenger, there are generally some personal data on the user, like his age, gender, etc. These information can be used to predict the product users like.

7.3.2 Data Preparation and Pre-processing

To train the machine learning model, training data is a crucial part. However, Microsoft is unable to provide their user data, including user information, purchase history and product ratings to us. In order to incorporate this recommendation feature to our bot, we generated some user data on our own. Basically, we made up 60 people, with different ages, occupations, genders and ratings on different categories of products. Then we used these self-generated data to train the model. We joined these three tables, removed redundant columns, and fed the joined table into the machine learning model.

user_id	Age	Occupation	Gender
1	17	Student	M
2	18	Student	M
3	19	Student	M
4	20	Student	M
5	21	Student	M
6	22	Student	M
7	23	Student	M
8	24	Student	M
9	25	Student	M
10	26	Student	M

Figure 10: Workflow of translator service

user_id	device_id	rate
1	1	6
2	1	5
3	1	6
4	1	5
5	1	5
6	1	5
7	1	6
8	1	5
9	1	5

Figure 11. User ratings on device categories

id	name
1	Desktop
2	Noetbook
3	Tablet
4	2in1
5	Gaming Device

Figure 12. Device categories

7.3.2 Model Creation

Model creation in Microsoft Machine Learning Studio is by drag and drop pre-defined modules. For example, we can select the machine learning algorithm used in the model by drag and drop the box called Recommender User Info Based in the screenshot. “Naïve Bayes” algorithm is used in our project. The “Enter Data Manually” means the data input from self-generated data as in 6.3.1. “Web service input” means the model is also waiting for input from web for its prediction.

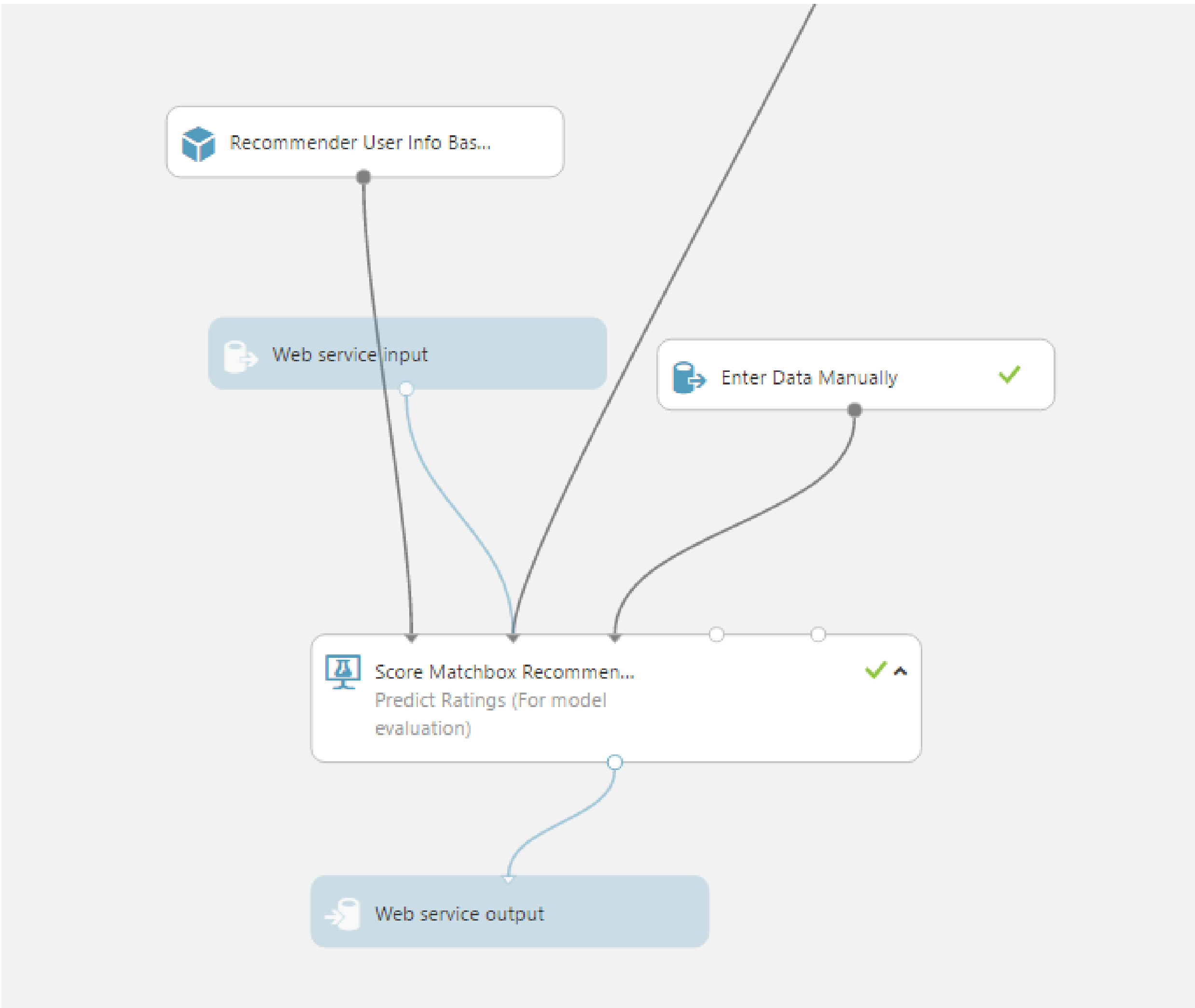


Figure 13. Machine learning model creation

7.3.3 Deploy Web Service

After training the model to a certain level, it can be published as a web service on the Internet. Then codes can be added into our bot to connect to the machine learning model for prediction.

7.4 SQL Database

SQL Database is created on Azure[7] and used for the bot to store and access data. The bot should be configured to connect to the database when it is published to Azure. Each table in SQL Database is converted as an ADO.NET Entity Data Model [8] in C# to allow programmatic access to the database. There are two tables in the SQL Database.One is the sample product catalogue with fields *Manufacture*, *Name*, *PriceBrand*, *FormFactor*, *DeviceHeadline*, *DeviceDescription*, *etc*. The other is to store complaints with fields Id, *UserID*, *UserName*, *Channel*, *Time*, *Type*, *Content*.

Manufactur...	Name	PriceBand	PriceUSD	DeviceColl...	FormFactor	DeviceCate...	DeviceHeadlineShort	DeviceHea...	DeviceDescription
ASUS	Zenbook 3 UX390UA	Premium (800+)	999	Catalog	Ultra-Slim Notebook	N/A	*NULL*	*NULL*	*NULL*
ASUS	Transformer 3 T305	Volume (500-799)	799	Catalog	2-in-1 Detachable	N/A	*NULL*	*NULL*	Est MSRP: $799-999Ultra-
ASUS	ROG GL702	Premium (800+)	1599	Catalog	Notebook	N/A	Compact for Gaming on the...	*NULL*	*NULL*
ASUS	ROG GL502VY	Premium (800+)	1499	Catalog	Notebook	N/A	Compact and lightweight fo...	*NULL*	*NULL*
ASUS	ROG GL552	Premium (800+)	899	Catalog	Notebook	N/A	*NULL*	*NULL*	Designed to dominate, tl
ASUS	ROG GL752	Premium (800+)	999	Catalog	Notebook	N/A	*NULL*	*NULL*	Designed to dominate, tl
ASUS	Zenbook Flip UX560	Premium (800+)	899	Catalog	2-in-1 Convertible	N/A	*NULL*	Stunning, 15...	*NULL*
ASUS	Zenbook Flip UX560UQ	Premium (800+)	999	Catalog	2-in-1 Convertible	N/A	*NULL*	Stunning, 15...	*NULL*
Samsung	Notebook 9 13	Premium (800+)	999	Catalog	Ultra-Slim Notebook	N/A	"1. Super light notebook (84...	*NULL*	*NULL*
Dell	Alienware Aurora r5	Premium (800+)	1500	Catalog	Mini Desktop	N/A	*NULL*	*NULL*	Premium Gaming Deskto
ASUS	Zenbook UX305CA (To...	Volume (500-799)	799	Catalog	Ultra-Slim Notebook	N/A	*NULL*	*NULL*	Fuel your everyday with t
ASUS	Zenbook Flip UX360CA	Volume (500-799)	699	Catalog	2-in-1 Convertible	N/A	Sleek & striking	*NULL*	Work it with this ultra-thi
ASUS	Zenbook Flip UX360UA	Premium (800+)	849	Featured	2-in-1 Convertible	N/A	Highend convertible	Outer beaut...	Treat yourself to the 2-in-
ASUS	Zenbook UX330CA	Volume (500-799)	699	Catalog	Ultra-Slim Notebook	N/A	Next-generation UX305 wit...	*NULL*	*NULL*

Figure 14. Database table storing product information

Id	UserID	UserName	Channel	Time	Type	Content
2	29:1MPH7iKQ-yik3UhHCK_UU0I3KOHVySCgKKACN...	Xusheng Ch...	skype	3/20/2017 2:49:49 PM	Product	Surface Pro 3 电池异常
5	29:1BU2zfqDNakc0qhU5_Gwa-nWxd9N3Jh__0Mm...	Francis Tan	skype	3/29/2017 3:31:33 AM	Product	windows phone打電話無聲音
6	29:1HK1128VSatk1fhU5_Gwa-nWxd9N3Jh__0MmLj...	Kai Huang	skype	3/29/2017 4:55:38 AM	General	手機無法安裝outlook點算？
7	29:1XCO13UTfKL3Jb1qhzjGsRs1ru1QobJkGffgAJbf3...	Samson Lee	skype	4/3/2017 1:02:59 AM	Service	點解宜家微軟服務電話無台灣？
8	2UTxbhlPeDv	You	webchat	4/13/2017 10:40:20 AM	Product	你好，我部Windows10今天突然無法使...
9	29:1BU2zfqDNakc0qhU5_Gwa-nWxd9N3Jh__0Mm...	Francis Tan	skype	4/16/2017 3:03:50 AM	General	我要投诉
10	29:1BU2zfqDNakc0qhU5_Gwa-nWxd9N3Jh__0Mm...	Francis Tan	skype	4/16/2017 3:04:01 AM	General	我要投诉

Figure 15. Database table storing user complaints

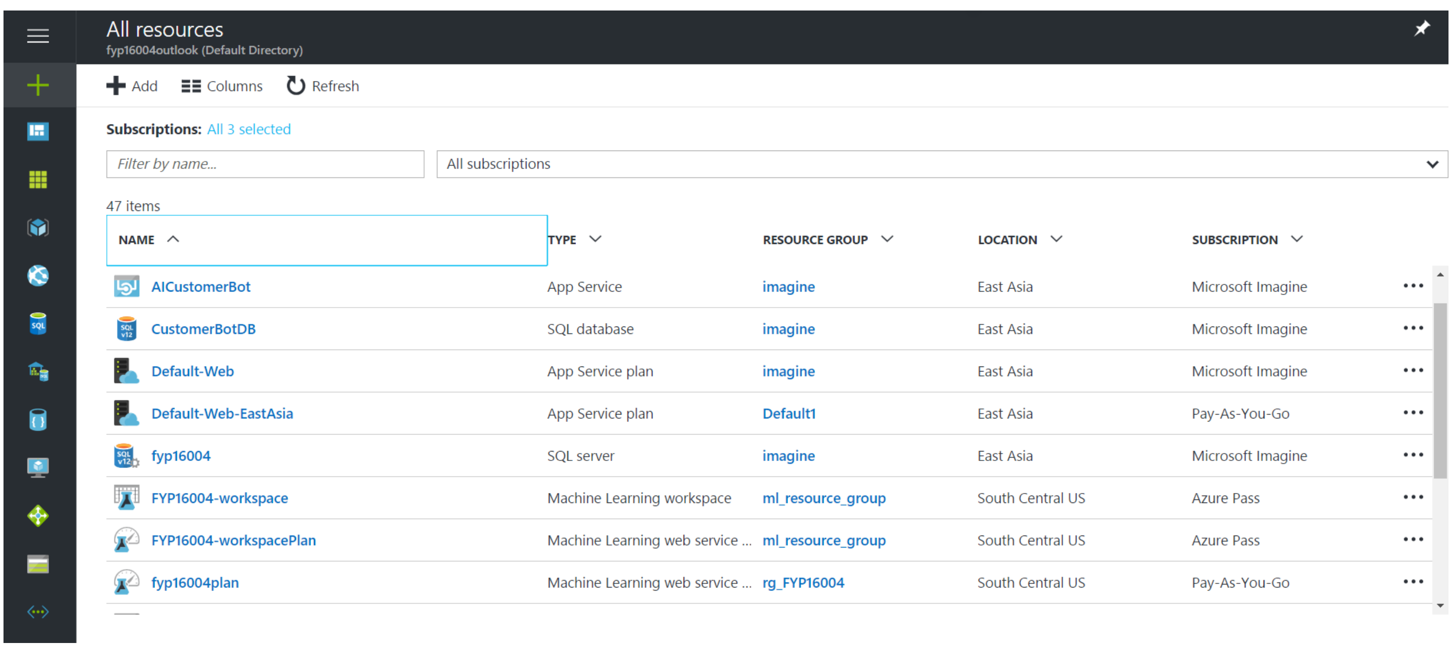
7.5 Microsoft Azure

7.5.1 Introduction

Microsoft Azure is a growing collection of integrated cloud services that developers and IT professionals use to build, deploy, and manage applications through its global network of datacenters.

7.5.2 Application deployment

As seen in the screenshot below, we have deployed several resources on Azure, including our bot developed using Bot Framework, SQL database, machine learning model etc. Through this portal, we can control the subscription plan of each application, and how many requests came in during a specific period of time.



8 Implementations

When a user inputs a sentence to our bot, the input is processed in the following flow:

8.1 Processed by the Machine Learning Model

When the user just logs in to talk to our bot, his personal information registered along with the account, like age and occupation, will be extracted and sent to the machine learning model established in Azure. These information will be treated as input to the model, and then the category of the products will be returned by the model and sent back to the bot. Based on this category, the bot will search in database to recommend products.

8.1 Processed by the LUIS language model

If a user inputs a sentence to our bot, possibly through Messenger or Skype, the input will be firstly sent to our trained language model for its labelling. Our language model will classify the input to one of the intents, and label the entities. Then the labelled result will be returned to our bot hosted on Azure in a JSON format.

8.2 Processed by Bot Framework

Based on the result returned from LUIS or machine learning model, our bot will handle differently. For example, if a product category is retuned, then our bot will search the database based on this category and return recommendations. If a complaint intent is received, our bot will decide the type of the content and store it into database. If an “unsatisfied” intent is received, our bot will prompt up features of products for users to select.

8.3 Database Query and Insertion

Finally the bot will interact with the database, either query the product catalogue for the products that meets user requirements then return the query result to the user or insert a record for the user complaint to be handled later.

9 Results

9.1 Project Webpage

We have established the project webpage [9], which provides a brief introduction of our project together with team members and contact details. One can also review the project progress from this webpage.

9.2 Detailed Project Plan and Final Report

The project plan mainly contains the motivation, objectives, scope, methodologies and tentative schedule of this project. The final report, namely this document, serves to discuss the methodology, implementation details, present our final product, and evaluate the product.

9.3 Cantonese AI Chat Bot

We have hosted a runnable Cantonese chat bot on both Skype and Messenger. Below is the guide on how to use the bot and its functionalities.

9.3.1 Greeting:

When users input greetings like “你好”, then simply a default sentence will be returned.

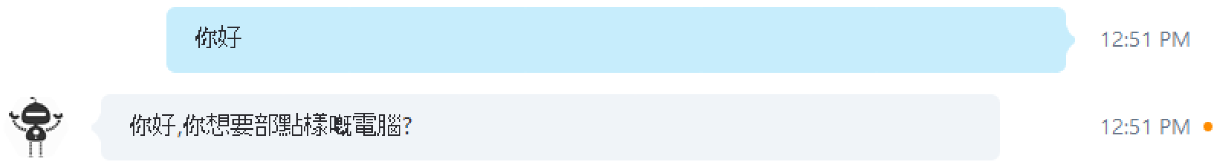


Figure 16. Bot-Greetings

9.3.2 Complaint:

When a complaint is received from users, we firstly categorize the complaint into either Product, Service, or General complaint, based on keywords extracted from user inputs through LUIS. We trained the LUIS using complaints from the forum mentioned in 6.2.2.

For example, when user input “Surface Pro 3 電池異常”, which is a product complaint as “Surface Pro” was mentioned.

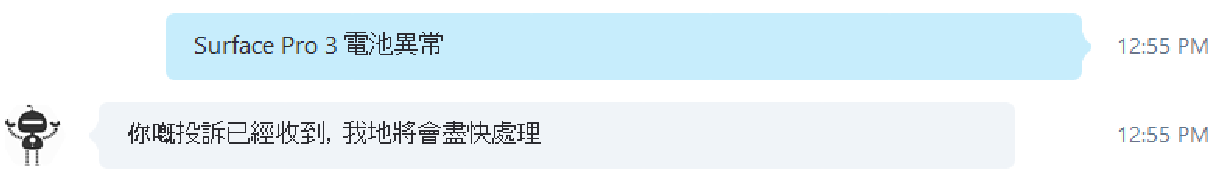


Figure 17. Bot-Product Complaint

Then the complaint is stored in the database as follow. As seen, the type of the complaints is “Product”. User id, user name, channel, time, complaint type and content of the complaint were stored in database to be processed by the staff.

Id	UserID	UserName	Channel	Time	Type	Content
6	29:1BU2zfqDNakc0qhU5_Gwa-nWxd9N3Jh__0MmLjhs8Uo	Francis Tan	skype	3/29/2017 4:55:38 AM	Product	Surface Pro 3 电池异常

Figure 18. Database Complaint Table

If the user just types a very general complaint, like “我要投訴”, without specifying what he’s complaining about, then the complaint is a general complaint and it will be stored in the database as a general complaint.

Id	UserID	UserName	Channel	Time	Type	Content
10	29:1BU2zfqDNakc0qhU5_Gwa-nWxd9N3Jh__0Mm...	Francis Tan	skype	4/16/2017 3:04:01 AM	General	我要投诉

Figure 19. Database Complaint Table

9.3.3 Recommendation based on user expression

After a user inputs a general sentence, usually stating what he/she is going to use the device for, we classify it as one of the categories in the form in 6.2.2. For example, if user inputs “我通常只會瀏覽網站同睇電郵” as in the form, our bot should categorize it as “Tablet”. Then based on this categorization, our bot searches in database and provide three products of “Tablet” of different manufacturers. For each product, we show the image (As there are no images for products in database provided by Microsoft, a default image is used at present), name, description, and button to the product page (As no product link is in the database, the button redirects to Microsoft online store by default).

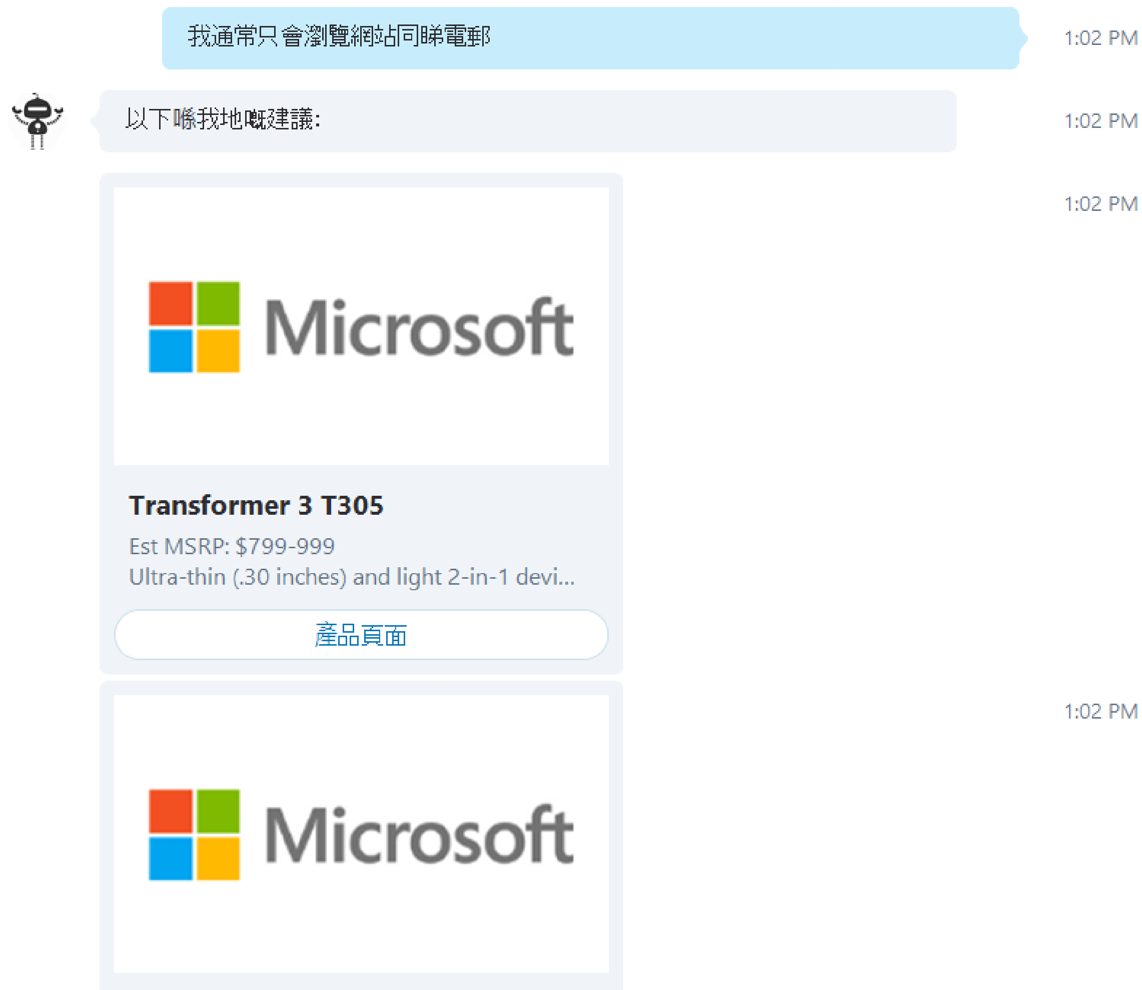


Figure 20. Bot-Input generic sentence

At the end of three recommendations, our bot will ask whether the user is satisfied with these recommendations or not.

Figure 21. Bot-Ask for satisfaction

Then users can click the buttons “滿意” or “唔滿意”, or they can type in using their own expressions as the input is processed by LUIS to determine user intent. If user clicks “滿意”, then a default thank-you sentence will be returned. If “唔滿意” is clicked, refer to 8.3.4

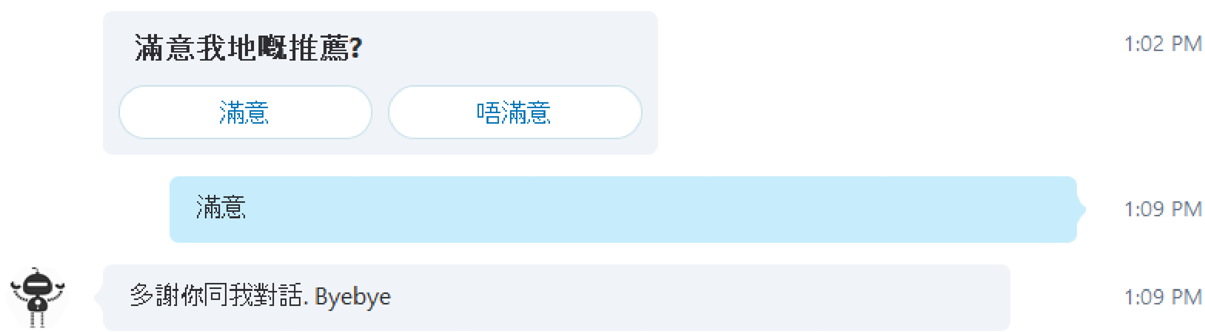


Figure 22. Bot-Satisfied

9.3.4 Deal with unsatisfied users

If user clicks "唔滿意", then boxes will be popped up to ask user to select manufacturer, price range and device type to match the products in db. For the example below, "Acer" for manufacture, "5000-8000hkd" for price rnage, "No preference" for device type were selected. Then corresponding products in database will be returned to users.

Figure 23. Bot-Select features

After the user selects all the features required, our bot will ask whether he/she would like to change the requirements just specified. If the user types 'no' at this stage, he/she will be prompted to choose the feature he/she would like to change.

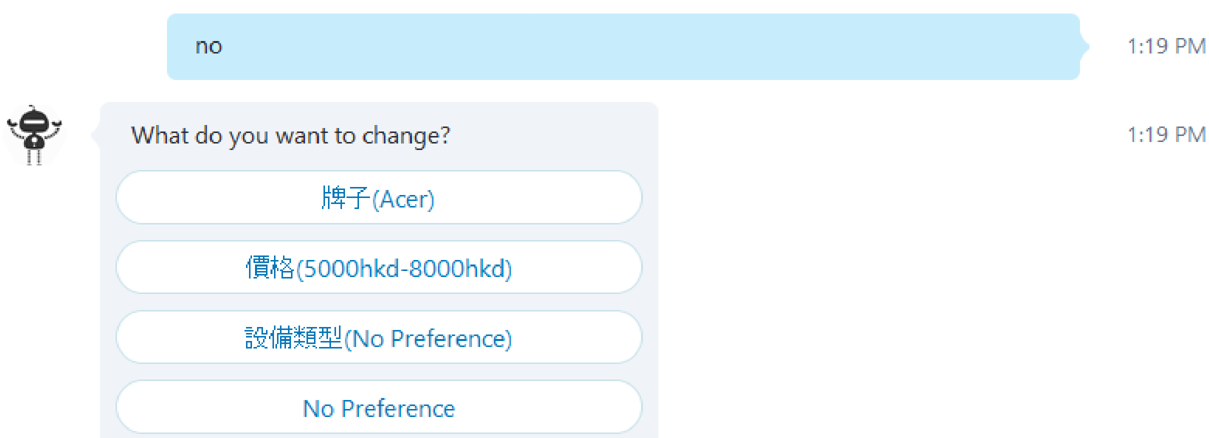


Figure 24. Bot-Change selected features

If the user types ‘yes’, then our bot will go on to search the database based on the features required.

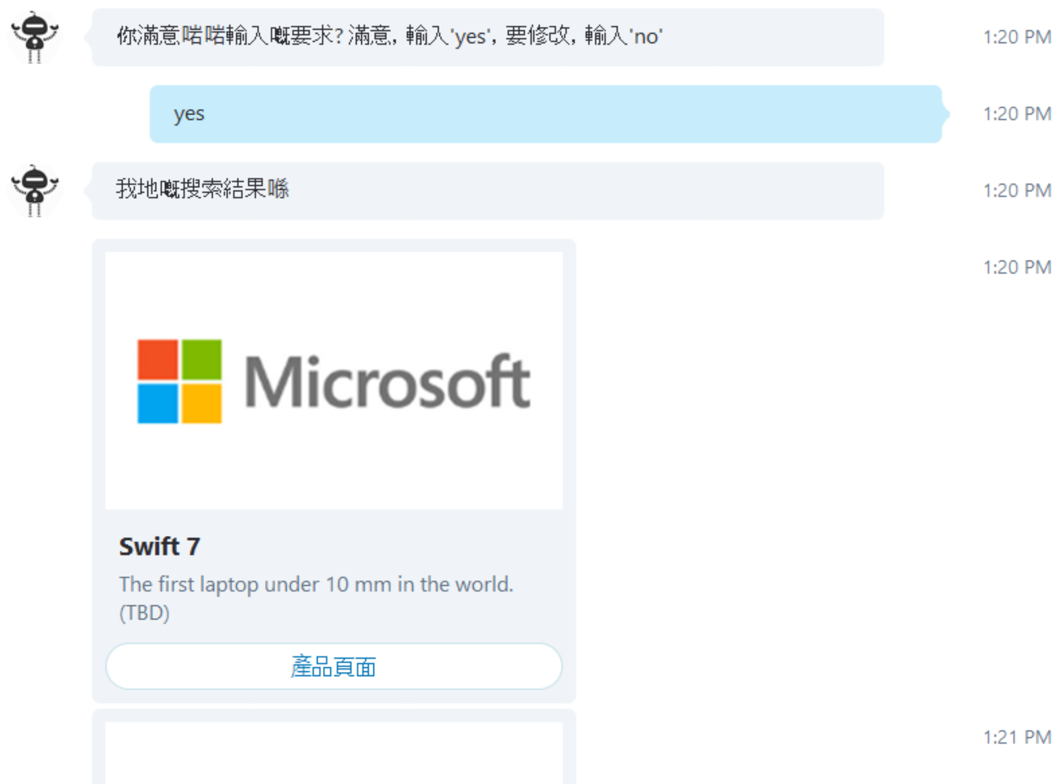


Figure 25. Bot-Search results if unsatisfied

9.3.5 Recommendation based on user characteristics

As mentioned above, machine learning model is used to predict what products he/she may like. We also implemented this feature in our bot. As we don’t have real user data from Microsoft, we used the data generated by ourselves. By typing “@” followed by an id ranging from 1 to 120, our bot will recognize it as log-in from one of the users defined at the backend. For example, if “@65” is input, then our bot would recognize that user with id 65 logs in.

user_id	Age	Occupation	Gender
61	17	Student	M
62	18	Student	M
63	19	Student	M
64	20	Student	M
65	21	Student	M
66	22	Student	M
67	23	Student	M

Figure 26. User information at backend

As in figure 26, user with id 65 is a male student at age of 21. The information will be sent to the machine learning studio model and the recommended category will be returned. Based on the category, we search in the database to return recommendations.

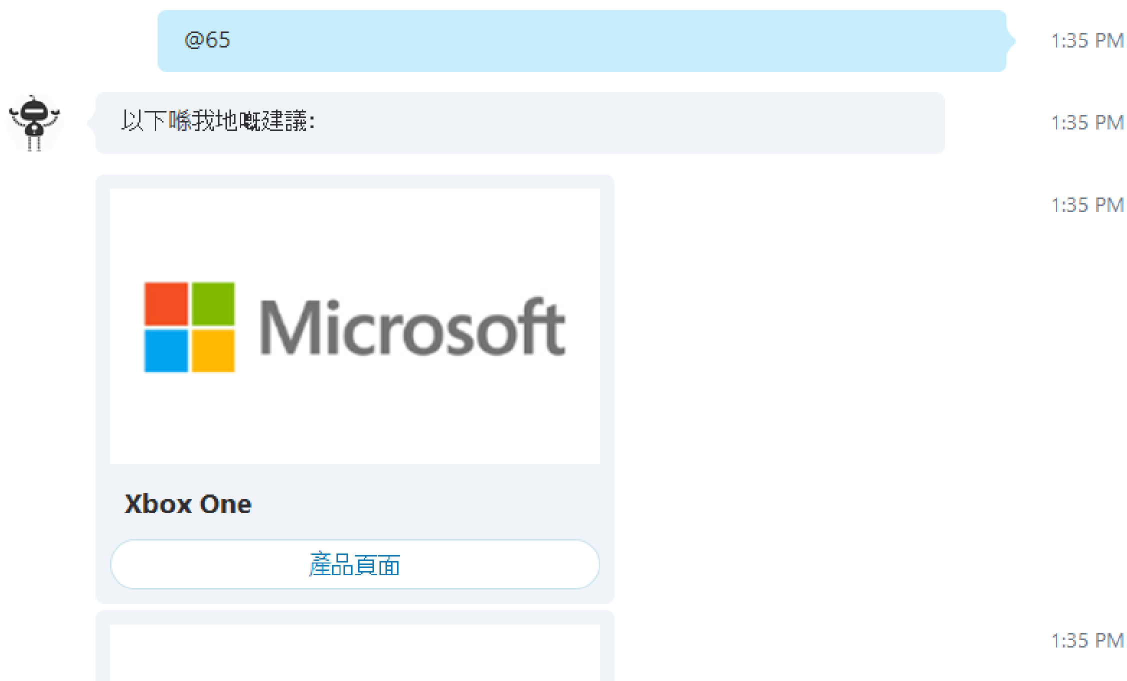


Figure 27. Bot-Recommendation based on user id

10 Result Evaluations

In order to evaluate the bot we developed, we invited 20 students from different faculties to try to talk to our bot. They were asked to give a rating on user interface and recommendation satisfaction from 1 to 10, respectively.

user id	faculty	year	user interface rating	recommendation rating	overall rating
1	Medicine	4	9	6	9
2	Social Science	4	9	8	6
3	Engineering	4	7	4	5
4	Business	1	7	7	8
5	Science	2	10	4	8
6	Medicine	1	8	4	5
7	Social Science	4	9	8	7
8	Social Science	1	8	7	7
9	Business	2	10	6	5
10	Business	4	7	8	6
11	Social Science	3	10	4	6
12	Engineering	4	10	4	5
13	Medicine	2	10	4	6
14	Social Science	1	9	7	7
15	Medicine	2	8	4	5
16	Art	1	10	4	6
17	Art	4	9	7	7
18	Medicine	1	8	4	8
19	Social Science	4	9	5	9
20	Business	2	8	4	7

Figure 28. User study

As in figure 28, these students seem to have a pretty high rating on the user interface, which may be due to the reason that our bot was integrated into those major communication applications, and device pictures, product links were shown in a card form. However, they seemed to have a lower rating on recommendation satisfaction, which may be due to several reasons. Firstly, maybe it’s due to the reason that the accuracy of our language model is not high enough as we don’t have much training data. Secondly, it may be the reason that the current pool of our products is not large enough, and therefore the results returned were limited. The overall rating of the bot was around 7, which was pretty goof and higher than expected.

11 Future Improvements

Although it is coming to the end of this project, there are some improvements to be made if this project is handed on to another team.

11.1 Language Model Improvement

At present, the accuracy of the language model is not high enough, possibly due to the reason that training data is insufficient. One solution might be to ask Microsoft to provide more training data. And we can produce more training data by changing one expression to another equivalent expression. Another solution is to change the scope: instead of deciding the category based on a generic sentence from users, we can have more detailed matching, like which kind of user expression should be matched to which feature of device. This involves changes on the language model design.

11.2 Machine Learning Model Improvement

At present, self-generated data is used for the machine learning model, as we cannot get real data from Microsoft. Two issues arose under this circumstance. The first is that the amount of data and number of features we generated are too small, which makes the prediction inaccurate. Second, self-generated data is biased, which cannot really represent users in real world. The future improvement would be to push Microsoft to provide real user data with sensitive information, like names, wiped out.

Another improvement is to select a better machine learning algorithm. As the amount of data is quite small at this stage, different algorithms don't make much differences. However, as amount of data increases in the future, selection of the algorithm should be more careful to produce the best performance.

11.3 User Interface Improvement

As the bot is potentially an industrial product of Microsoft aimed at serving its customers, Microsoft lays much emphasis on the bot's interaction with users. One possible improvement is to shorten the response time. At present, the response from our bot is quite slow, taking around 3 seconds for a request. The reason is unknown to us, which may be due to improper coding, or network settings of Azure applications.

Another improvement is to include correct pictures and purchase links for each product recommended. At present, default images and purchase links are used for each product as these information are absent in the database. The solution is to ask Microsoft to provide a more detailed database, possibly their real database in practice.

The final improvement might be to add state controls in the bot. Currently, there are no states in our bot, meaning that the bot doesn't memorize what you talked to it before. One solution is to record necessary information during each conversation and reuse it.

12 Difficulties and Limitations Encountered

This section discusses the four major difficulties we encountered and the possible solutions.

12.1 Changes of Scope

During this project, the requirements from Microsoft were changed largely, which made some of our previous efforts in vain. At the beginning, we developed a language model to parse users' requirements in his input. For example, when user types "I want to buy a computer with i5 processor, RAM size larger than 4GB", our previous language model should identify it as purchase intent, and extracts requirements as "CPU:i5", "RAM>4GB".

However, at the beginning of second semester, we were told that this language model was not needed. We were informed that we just need to assume that users will input a generic sentence, like "I mostly use the device to browse emails and website", then we directly categorize it into one of the product categories. This change made our previous model abandoned.

Also, the database was also changed during this project, so the database configuration had to be changed accordingly, which consumed us quite a lot of time.

12.2 Bot framework limitations

Although it is good for the framework to provide us with a lot of libraries to build up the interface, there are also limitations. For example, when using the "Formflow" class in the library to guide users to select product features, we cannot change the flow because it is fixed in the codes of this class, which made the development inflexible.

12.3 Azure account issues

Most of our applications are services on Microsoft Azure. As instructed by Microsoft, we used the plan called “Imagine”, which is a plan free for student developers. However, many services are not provided by this plan, like the translator service and machine learning studio. It took us a lot of time to handle the limitations caused by accounts, which was unnecessary.

12.4 Lack of Real Data

As Microsoft doesn’t provide us with much training data, like those user expressions for language model, and user data for machine learning model, we needed to generate on our own, which made the data biased and the models ill-performed.

13 Conclusion

In conclusion, we finished the scope from Microsoft by setting up a functional conversation bot. We developed the bot based on Bot Framework so that it can be easily deployed to Azure and integrated into Skype and Messenger. The bot can understand natural languages from users by using LUIS, which made it different from its counterparts in the market. Also, simple machine learning model is used to predict products users like based on their personal information. We report our progress to our supervisor and the representative from Microsoft on a regular basis and receive timely feedbacks from them. The current codes we have will be possibly handed on to Microsoft for their further development.

However, to be frank, we have achieved the scope of the project, but didn’t devote a lot of time to make it even better by considering adding features that are not in the scope. Future improvements might be adding more fancy features to the bot.

References

[1] 1-800-Flowers [Internet]. USA: 1800flowers.com Messenger Bot. [cited 2016 Dec 1] Available from: https://www.messenger.com/t/1800flowers/

[2] Apple Inc [Internet]. USA: Apple online chat support. [cited 2016 Dec 1]. Available from: https://aoschat.apple.com/Chat/getCustomerDetails.do

[3] Microsoft HK Ltd [Internet]. USA: Microsoft online chat support. [cited 2016 Dec 1]. Available from: https://www.microsoft.com/surface/en-hk/support/contact-us.

[4] Microsoft [Internet]. USA: Microsoft Bot Framework. [cited 2016 Oct 20]. Available from: https://dev.botframework.com/.

[5] https://github.com/Microsoft/BotBuilder/tree/master/CSharp/Samples.

[6] Microsoft [Internet]. USA: Language Understanding Intelligent Service. [cited 2016 Oct 20]. Available from: https://www.microsoft.com/cognitive-services/en-us/language-understanding-intelligent-service-luis.aspx

[7] Microsoft [Internet]. USA: Microsoft Azure. [cited 2016 Oct 20]. Available from: https://azure.microsoft.com/en-us/

[8] USMAN_MALINK [Internet]. Udemy Blog. [update 2014 May 29, cited 2017 Jan 22]. Available from: https://blog.udemy.com/ado-net-entity-data-model/

[9] https://i.cs.hku.hk/~fyp16004/