Computational Linguistics and Chinese Language Processing
Vol. 28, No.1, June 2023, pp. 19-34

多領域任務導向一對一用戶對話收集系統

A Dialogue Collection System for One-to-One Multi-Domain Task Oriented Dialogs

葉丞鴻 *、李聿鎧*、張嘉惠*

Cheng-Hung Yeh, Yu-Kai Lee and Chia-Hui Chang

摘要

客服系統、聊天機器人、智慧音箱等對話系統需要有標記的對話語料進行模型訓練。 然而如何快速有效地收集對話語料，是建構對話系統必須面對的問題。現有任務導向系統主要以餐廳、旅館、機票訂位為主，尚無個人虛擬助理提供傳送訊息、建立活動等事務性服務之對話語料。本篇論文仿照 CrossWOZ 收集對話語料的方法，透過任務生成及對話網站介面設計，讓標記人員模擬用戶與虛擬助理對話情境，建立一個能夠處理電子郵件、管理行事曆、以及傳遞訊息等三種服務的對話語料集（稱為 MsgWOZ）。期望此語料為中文虛擬助理對話系統奠定發展基石。

標記系統和資料集已開源至 https:// github.com/TedYeh/messageWOZ。

關鍵字：任務導向對話系統（TOD）、對話語料庫建構、Wizard-of-Oz（WOZ）、Msg-WOZ

Abstract

Task-oriented dialog systems require labeled corpus for model training. However, in the face of new services, how to effectively collect dialogue corpus is a problem that must be faced in the construction of dialogue systems. Existing task-oriented systems mainly focus on reservations for restaurants, hotels, and airline tickets. There is no dialogue corpus for virtual assistants that could provide transactional services such as sending messages and creating events. This paper imitates the method of

* 國立中央大學資訊工程學系
Department of Computer Science & Information Engineering, National Central University
E-mail: kmes50215@gmail.com; yklee@g.ncu.edu.tw; chia@csie.ncu.edu.tw

collecting dialogue datasets from CrossWOZ to allow annotators to simulate user and virtual assistant dialogue scenarios through a dialogue website interface, creating a dialogue dataset that can handle three services: email management, calendar management, and message delivery. It is expected that this corpus will lay the foundation for the development of Chinese virtual assistant dialogue system. The annotation system and dataset have been open-sourced at https://github.com/TedYeh/messageWOZ.

Keywords: Task-orient Dialogue Systems (TOD), Dialog Corpus Construction, Wizard-of-Oz (WOZ), Msg-WOZ

1. 緒論 (Introduction)

透過自然語言和電腦進行溝通是人機互動長遠的目標，對話系統包含自然語言理解、及自然語言生成兩個部份。由於自然語言的本身的岐義性、以及個別衍伸的涵義，造成自然語言理解的困難。而對話生成牽涉到說話者的認知，更為複雜。因此，目前業界仍專注於建構任務型導向的對話系統 (Task-Oriented Dialogue Systems)，以幫助完成特定任務，例如飛機航班預訂(Seneff & Polifroni, 2000)或公車訊息(Raux, Langner, Bohus, Black, & Eskenazi, 2005)。而隨著智慧型系統及虛擬助理的普及，建構可跨不同應用領域處理任務的對話系統變得越來越重要。

依據(Chen, Liu, Yin, & Tang, 2017)的研究分類，對話系統本質必需理解人類語言時的歧義； 整合第三方服務和對話環境；最後，產生自然和引人入勝的回覆。現有任務導向對話系統將以上問題分為四個子任務來解決：自然語言理解(Natural Language Understanding, NLU)、對話狀態追蹤(Dialogue State Tracking, DST)、對話策略學習(Dialogue Policy Learning, DPL)及自然語言生成(Natural Language Generation, NLG)。

為了推動使用數據驅動方法建構對話系統的進展，過去已有數個對話語料庫的釋出。根據是否使用結構化標註方案來標註語義，這些語料庫大致可以分為兩類：帶有結構化語義標籤的語料庫((Hemphill, Godfrey & Doddington, 1990)、(Shah *et al*., 2018))；和沒有語義標籤但考慮到隱含用戶目標的語料庫((Ritter, Cherry, & Dolan, 2010)、(Lowe, Pow, Serban, & Pineau, 2015))。儘管做出了這些努力，但上述數據集通常在一個或多個維度上受到限制，例如缺少適當的註釋、僅在有限的容量中可用、缺乏多域用例或具有可忽略的語言可變性。

現今的任務導向語料庫中，大多以英文語料為主，且在語料蒐集的過程中，因為隱私和相關實體難以蒐集等因素，使得現有的語料皆以查詢資訊的意圖為主，如查詢天氣、查詢景點資訊等意圖，並沒有傳送、添加資訊等交易性(Transactional)意圖，造成了虛擬服務的侷限性。有鑑於上述提到的問題，本文提出了 MsgWOZ 對話語料集，這是一個基於雙人之間的中文多輪對話語料庫，且以電子郵件、行事曆及通訊軟體這三個領域來實現跨領域對話。藉由預先定義任務目標使標記人員能夠依據給定的槽與槽值來進行對

話，且每個對話都用一系列對話狀態和相應的系統對話行為進行標記，如此便能產生跨領域的對話目標以進行多領域對話及標記，解決無法產生多領域對話語料之問題。因此，MsgWOZ 可用於開發單獨的系統模塊作為獨立的分類任務，並作為現有基於模塊化方法的基準。最後，我們將 MsgWOZ 分為訓練、驗證及測試資料，並且將其輸入給 NLU 模型進行訓練及測試來分析其效能。

2. 相關研究 (Related Works)

任務導向型對話可以分成 Machine-to-Machine、Human-to-Machine 及 Human-to-Human 等三種語料蒐集方式，每種方式皆需要真人分別飾演使用者及系統，來模擬人機互動的對話情境。在 Human-to-Human 中以 Wizard-of-Oz 為主要蒐集方式，Machine-to-Machine 則以 Schema-Guided Dialogue 來使用機器模擬對話及蒐集，本章將回顧與探討 Wizard-of-Oz 和 Schema-Guided Dialogue 兩種資料蒐集方法的演進。

2.1 Wizard-of-Oz

在任務型對話中，研究人員常使用 Wizard-of-Oz (WOZ) (Kelley, 1984)來建構對話語料，其運作方式為一人當作機器，另一人作為人類，並模擬特定的情境及任務來進行對話，期望建立一 Human-to-Machine 對話的語料庫，用來觀察及訓練語言模型在任務導向對話上的效能和狀態。

ATIS (Hemphill *et al*., 1990)為最早使用 WOZ 來進行對話蒐集的語料庫，用於航班的口語理解任務上。WOZ2.0(Wen *et al*., 2017)改善了 Wizard-of-Oz 的方法，建立了餐廳訂位的任務型語料，在對話標記上，系統端除了需紀錄使用者的對話狀態及意圖外，也需同時對自己的對話進行標記。而這兩種語料雖然奠定了未來任務型導向對話的研究方向及基礎，但由於此二語料僅專注進行單一領域的對話，在多領域及跨領域的對話上仍有諸多限制。

New Woz (Ramadan, Budzianowski, & Gašić, 2018)為模擬使用者在遊客服中心的對話情境而建立的對話語料，增加了旅館、旅遊景點、餐館及交通工具等五種領域，並且每組對話需涵蓋一至五個領域以增加對話的複雜度及多元性。隨後，MultiWOZ(Budzianowski *et al*., 2018)遵循了相同的方法來蒐集語料。在 MultiWOZ 中，為了使對話標記人員更瞭解對話的主題及需要達到的任務，研究人員以基於模板的方式結合資料庫綱要中的槽來生成任務敘述。在對話階段，擔任使用者的人員需依據產生出來的任務依序進行對話，而擔任系統角色的操作人員則須對使用者提出的要求進行資料庫查詢，並將結果回報給使用者。

由於 MultiWOZ 產生了更為詳細的任務敘述，使得研究人員能對於任務需求來進行更具體的對話，藉此來提升對話語料的質量。但因為 MultiWOZ 使用 Amazon Mechanical Turk 進行人工對話標記，即使語料標記呈現了高度的一致性，但仍有同筆資料卻標記不同的情況。為減少對話標記所造成的不一致性，(Zhu, Huang, Zhang, Zhu, & Huang, 2020)

提出了 CrossWOZ 的中文對話語料，以自動標記來標記對話意圖及對話狀態，並且強調使用者在對話時，每輪所選擇的領域是彼此相依的，皆會影響到後續相關領域的選擇，期望如此設計能加強模型對上下文意的解析。

2.2 Schema-Guided Dialogue

Schema-Guided Dialogue (SGD)是以機器對機器(Machine-to-Machine, M2M)的方式來進行自動對話生成，M2M(Shah *et al*., 2018)利用 Self-Play 的自動化框架以來減少建構語料所需的成本。(Shah *et al*., 2018)抽取資料庫綱要(Schema)中的各個欄位來作為對話所需要的槽(Slot)和槽值(Slot Value)，將槽值隨機抽樣並插入至設計好的對話模板，使其成為粗略的任務型對話，最後再由眾包(Crowdsourcing)以人工改寫自動生成的對話。

(Rastogi, Zang, Sunkara, Gupta, & Khaitan, 2020)提出 Schema-Guided Dialogue(SGD)語料，以符合語音助理需要建立的大量服務之需求。(Rastogi *et al*., 2020)透過資料庫的 API(Application Programming Interface)來獲取綱要，每份綱要皆需包含服務、意圖及槽值。與 M2M 不同的是，SGD 利用兩個機率自動機(Probabilistic Automaton 來扮演系統及使用者，以對話代理組成對話模擬器(Dialogue Simulator)，並輸入綱要互相產生對話大綱，最後再以對話模板與大綱進行對話生成及改寫。由於大綱皆包含每輪對話的意圖及槽值，故以此方法能減少對話狀態及意圖標記等人力成本。即使 M2M 和 SGD 通過模擬器來模擬對話過程，再以模板與模擬器的輸出產生對話，但以此種方法仍需透過人工改寫來提升對話品質，且在對話改寫的連貫性仍會影響對話模型的效能。

隨著具有強大語意解析能力的預訓練語言模型出現，其在序列生成上的效果也逼近真實語料，故近期的研究也使用預訓練的對話模型作資料蒐集，藉此降低自然語言語料難以建置的難處。(Chiu, Li, Lin, & Chen, 2022)以預訓練模型模擬推銷員與顧客推銷為對話情境，建立同時包含閒聊(Chit-Chat)與任務導向的語料。在大多的對話中，使用者的回覆或請求經常是隱晦且不明確的，且會因為對話者的不同而夾雜開域(Open-Domain)的對話，這樣使的整個對話變的多元且複雜。(Chiu *et al*., 2022)以兩個 BlenderBots(Roller *et al*., 2021)來產生大量閒聊式對話，再利用意圖偵測模型來預測對話中隱含的意圖，並將下一輪對話搭配對話模板產生任務導向對話，最後再輸入模型作自動改寫，即可產生結合開域及任務導向的對話語料。

綜上所述，這些對話大多的情境皆是以使用者端進行對話的起頭，系統端再依據使用者提供的訊息進行資料查詢並進行回應。然而在大多現實的狀況中，系統會依據不同的情況先行與使用者進行對話，再依據使用者的提供的資訊請求相關的訊息以達成任務。因此本研究將汲取以上經驗，提出更符合真實情境的 Human-to-Human 語料蒐集方法。

3. WOZ-Style 對話語料收集 (WOZ-Style Dialogue Creation)

本研究將採用 Wizard-of-Oz 方法來建立對話語料收集環境，我們建立一個系統，模擬智能助理依據使用者需求來存取電子郵件、行事曆以及通訊軟體應用程式為對話情境，以

用戶(User)及助理(Assistant)兩個角色進行 Human-to-Human 對話及創建語料，且將領域分為郵件(Mail)、行事曆(Calendar)及通訊軟體(Message)。我們將語料蒐集分為資料庫及知識本體建構、目標生成、對話蒐集和對話標記。各步驟之概述如下:

1. 資料庫建構: 利用 Google API 擷取標記人員的 Gmail 和 Calendar 資訊，並依據獲得實體來建立資料庫。
2. 目標生成: 基於資料庫來設計一多領域目標生成器。對於橫跨不同領域的實體，我們使用特定條件來約束兩個彼此靠近的目標。而為了使對話標記人員更了解任務，我們創建一任務敘述模板來為每個目標產生逼近自然語言的描述。
3. 對話蒐集: 在正式對話開始之前，我們要求標記人員進行少量的對話，並依據他們的對話質量提供相對應的建議。等人員熟悉介面操作後，會將他們進行配對並根據給定的對話目標進行交談。
4. 對話標記: 我們定義一些規則來依據對話行為對用戶狀態、助理狀態和對話紀錄作自動標記。每個對話都包含一個結構化的目標、一個任務描述、用戶狀態、助理狀態、對話意圖和對話內容，如圖 1 所示。

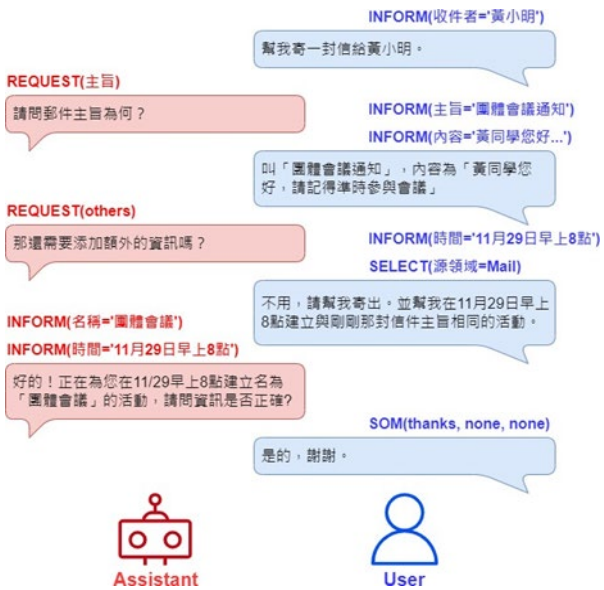

圖 1. 對話標記範例

[Figure 1. Example of Dialogue Annotation]

3.1 資料庫建構 (Database and Ontology Construction)

在過往 Wizard-of-Oz 方式建立的對話語料中，為了能使系統端能依據使用者的需求進行資料查詢，研究人員常在網路上蒐集大量跨領域的資料，並將擷取下來的實體(Entity)整理為資料庫綱要。在本研究中，為了取得真實的資料，我們從 Google API 來獲取標記人員的郵件及活動，並提取我們需要的實體來作為生成對話目標的依據。

3.2 目標生成 (Dialogue Goal Generation)

在 TOD 語料蒐集中，我們需要為每組對話建立多個對話目標。每個目標可分為多個子目標，子目標包含領域及相關槽與槽值。用戶需根據對話介面給定的子目標來進行對話。

為了符合實際對話情境，且避免產生過於複雜的對話，我們採用了一種方法，即每個目標最多生成三個子目標，且每個子目標可能屬於同一個領域。我們效仿(Zhu *et al*., 2020)的方法，使用 API 擷取的綱要來提取槽與槽值，並將目標表示為元組列表，其中包含(子目標 id、領域、槽、槽值)。子目標 id 用於區分同一個目標下的多個子目標。我們將槽分為兩種類型: 訊息槽(Informable Slots)和請求槽(Requestable Slot)。訊息槽用於告知助理的約束，以協助助理搜尋資料；請求槽代表用戶需要助理查詢的資訊。表 1 提供了一個對話目標的範例。在實際對話情況中，我們有時需要進行跨領域的任務(例如：建立完活動後需同時寄信通知與會人員)。為了使模型學習如何約束兩個跨領域的子目標，我們增加了跨領域的信息槽(在表 1 中以粗體字表示)，其槽值以目標 id 連結到不同的子目標，以達到跨領域約束的效果。

表 1. 對話目標範例
[Table 1. Example of Dialogue Goal]

Id	Domail	Slot	Value	Action
1	Message	User	user_name	Send
1	Message	Content	msg_content	Send
1	Message	Application	msg_app	Send
2	Calendar	Name	_____	Search
2	Calendar	Start Time	_____	Search
2	Calendar	Participant	**(id = 1)**	Search
2	Calendar	Content	_____	Search
3	Mail	Subject	**(id = 2)**	Send
3	Mail	Receiver	_____	Send
3	Mail	Content	mail_content	Send

在目標生成中，我們以下列四個步驟來生成目標。第一，對於信件、行事曆及通訊軟體這三個領域，使用機率 P 來產生相互獨立的子目標，且產生出來的子目標有可能來自相同領域。而每個領域的子目標需要有共同的訊息槽或請求槽，以達到跨領域目標生成使用，如表 2 所示。

其次，利用單一領域目標來產生跨領域子目標。例如，子目標(id=3)信件主旨之槽值將會代換為「出現在 id=2 的活動裡」的敘述，以方便標記人員進行跨領域對話。同樣地，我們也以相同的方式產生其餘領域的跨領域目標。

表 2. 每個領域中所使用到的插槽
[Table 2. Slots used in each Domain]

Domain	Slots
Mail (郵件)	Recipient, Subject, Sender, Content, Copy recipient, Bcc recipient
Calendar (日曆)	Name, Event time, Participant, Is all day, Content, Location, Message Domain
Message (訊息)	User, Content, Application

第三，決定產生出來的子目標需要進行的行為和動作。在本研究中，用戶不僅會查詢活動和郵件，還會依據自身需求建立活動或發送相關的郵件和訊息。在此階段，我們使用機率 *Psearch* 選擇子目標進行查詢動作，而使用機率 1 − *Psearch* 進行發送或創建的行為。由於缺乏通訊軟體相關的實體，故通訊軟體(message)僅具有發送訊息的動作。

最後，我們依據(Zhu *et al*., 2020)的做法，設計目標敘述模板，並將子目標之槽與槽值嵌入模板中，使其成為自然語言描述，方便標記人員了解對話目標，並根據對話上下文決定跨域訊息槽值，來為不同領域的目標添加了更多約束，從而引導對話。

3.3 對話蒐集 (Dialogue Collection)

我們以 CrossWOZ 所開發的對話標記系統為基礎，架設一對話網站來讓兩位使用者以 Wizard-of-Oz 的方式同時在線上進行對話及標記。圖 2 為對話網站之使用案例。在網站上，標記人員將自行選擇用戶(Client)或助理(Assistant)當作對話角色。標記人員將會兩兩一組進行配對並進入聊天室。在對話過程中，用戶需要通過對話來完成分配的目標，同時助理需搜尋資料庫以提供必要的信息並給出相對應的回覆。為了能得到現實生活中的資料，在本系統中，我們新增一個 Google 帳號使用者，讓標記人員可以透過 Google 提供的 API 擷取 Gmail 以及 Calendar 的資料。而在進行正式的對話收集前，我們先讓標記

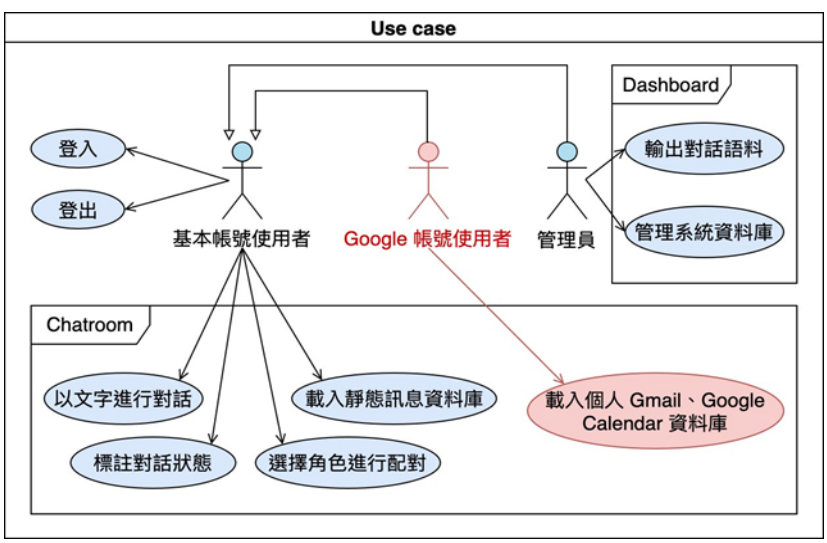

圖 2. 本對話蒐集系統之使用案例圖
[Figure 2. Use Case Diagram of the Dialogue Collection System]

人員完成少量的對話及依據對話質量給予反饋來做為訓練。以下將會描述用戶或助理進行對話的細節。

用戶端(Client Side)

圖 3 為本對話網站之用戶端的操作頁面。在用戶開始進行對話前，我們會給定用戶須達成的任務目標，其值也作為初始的對話狀態，也會給予自然語言的敘述，方便標記人員進行更準確的對話。在每一輪，用戶除了需要根據上一輪助理的回覆來修改狀態，也必須選擇在當前對話中的狀態，來表示當前已完成的對話子目標及對話狀態。舉例來說，當使用者對助理說「了解，再幫我用 Line 傳訊息給王正剛說志明下週五回台北」，由於此對話提到 Message 領域中的使用者、訊息及應用程式，故我們需在中間欄位的目標欄位勾選提到的資訊槽，並將所提及的槽值填上。一旦用戶目標狀態中的所有值都被填滿，即表示已完成此次的對話目標。

圖 3. 對話蒐集系統-用戶端，左: 標記任務描述；中: 需達成的目標；右: 對話視窗
[Figure 3. Dialogue Collection System - Client Side]

助理端 (Assistant Side)

圖 4 為本方法在蒐集對話的示意圖，為避免隱私問題，在進行對話時助理是透過讀取自己的資料來進行資料庫查詢。同時為了使本對話蒐集系統能夠自由擴充及更換任意領域的資料，我們也設計引入資料庫大綱(Outline)的設計，若要添加額外服務，更換大綱和引入資料庫檔案即可。

圖 5 為助理端的操作示例，中間欄位為查詢欄位，我們依據用戶提供的資訊進行查詢，而查詢倒的結果會顯示在左邊欄。我們將資料庫查詢結果作為系統狀態，且這些結果也表示每個當前對話用戶給予的約束條件。在每一輪對話，助理需要根據之前的用戶回覆查詢相關的結果。選擇檢索到的結果並依據選定結果來給出相應的回覆。以圖例來說，當使用者回答「了解，再幫我用 Line 傳訊息給王正剛說志明下週五回台北」，助理需將王正剛、志明下週五回台北及 Line 依序填入使用者、訊息及應用程式的欄位，以紀錄助理的對話狀態。當助理進行查詢動作時，可選擇中間欄的搜尋結果來顯示並紀錄詳細資訊，以作為對話決策學習(DPL)時加入資料庫查詢結果使用。而助理也可主動詢問用戶是否需要額外的訊息槽資訊，進而滿足對話的完整性。

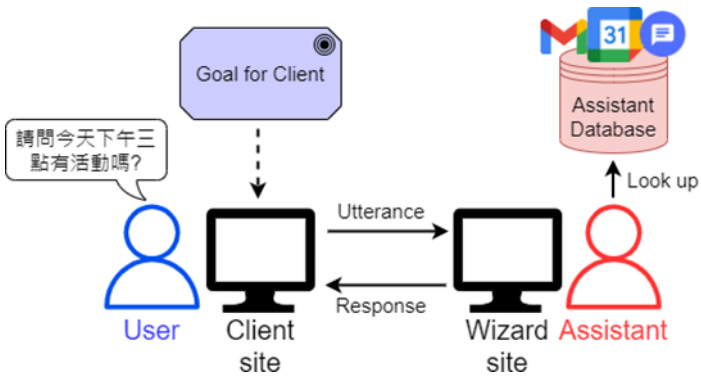

圖 4. 本研究方法之對話蒐集示意圖

[Figure 4. Schematic Diagram of Dialogue Collection in this research method]

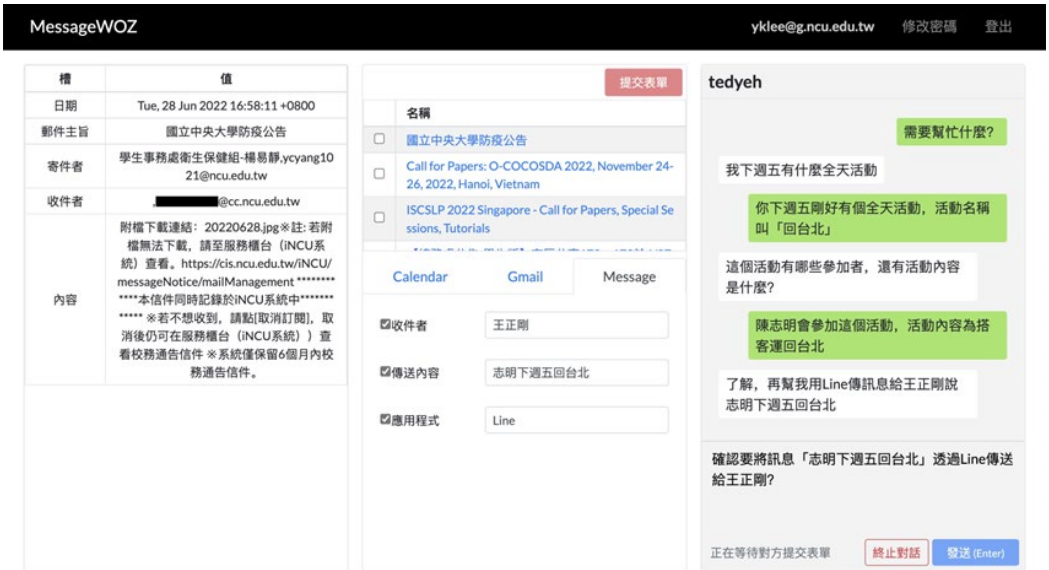

圖 5. 對話蒐集系統-助理端，左: 查詢結果；中: 查詢欄位；右: 對話視窗

[Figure 5. Dialogue Collection System - Assistant Side]

3.4 對話標記 (Dialogue Annotation)

我們收集對話數據後，定義了規則來自動標記對話行為。每段對話可以有多個行為，每組行 為由行為、領域、槽和槽值組成的元組。我們預先定義了五種行為，並以用戶和助理狀態的改動以及關鍵字匹配來取得對話行為。

在用戶端，對話行為主要來自當前子目標的選擇。例如，如果用戶選擇了表 1 中的(1, Message, User, user_name, Send)，則標記為(Inform, Message, User, user_name)。如果選擇(2, Calendar, Name, , Search)，則標記為(Request, Calendar, Name, none)。如果(3, Mail, Subject, (id=2)) 被選中，則標記為(Select, Mail, src_domain, Calendar)。這個行為是專門形成“與行事曆相關”的約束而設計。

在助理方面，我們主要將關鍵字匹配應用於標記對話行為。Inform 行為是通過將系統話語與所選實體的信息進行匹配而得出的，即通知或向用戶端確認已知的訊息。Request 行為則是將系統話語與提及之信息槽進行匹配所標記出來的，即向用戶端詢問用戶還未提及的訊息槽之值。當助理表示沒有結果滿足用戶約束時，將標記為 NotFound。

對於用戶端和系統端的社交言語(Social Obligation Management，SOM)，我們定義 Bye(再見),Confirm(確認), Done(完成任務), Greet(打招呼), Noneed(不需要), Reqmore(請求更多), Thanks(謝謝) 等一般行為，且使用關鍵字匹配進行標記。 除此之外，我們還為用戶狀態中的每個語義元組獲得了一個二進制標籤，它指示該語義元組是否已被用戶表達。這個註釋可以直接說明對話的進度。

我們以 JSON 格式來儲存標記完成的語料：其中 goal 及 final_goal 儲存該組對話需達成的目標及最後完成對話時的目標；messages 為一串列，儲存每輪對話所保存的訊息；content 為對話內容，role 為該輪次對話的角色，dialog_act 為該輪對話的對話行為，而 user_state 和 sys_state 則代表狀態，即儲存當前使用者或助理所提及的槽值。

4. Msg-WOZ 對話語料集實驗 (Dialogue Dataset Analysis & Experiments)

在研究過程中，我們對 8 位標註人員進行教學，每位人員會實際製作數組對話語料，並回饋對話標註系統之使用經驗。本研究總共標記了 339 組對話，並隨機將對話分為訓練、驗證及測試資料集。我們對此語料進行分析，以對話和標記二種統計面向進行討論，如表 3 所示。

表 3. 對話語料數量統計
[Table 3. Quantity Statistic of Dialogue Dataset]

	Train	**Dev**	**Test**	**All**
Dialogues	239	50	50	339
Turns	1,708	312	337	2,357
Tokens	78,409	14,109	16,195	108,713
Avg. turns	7.15	6.24	6.74	6.95
Avg. tokens	45.91	45.22	48.06	46.12
Avg. acts	4.01	3.96	4.01	4.00
Avg. u-acts	1.49	1.59	1.61	1.53
Avg. s-acts	2.51	2.36	2.39	2.48

在對話分析部份，由表 3 中上半部分我們得知訓練資料集共有 239 組且包含了 1,708 輪次的對話，而驗證及測試資料集則是各 50 組且分別包含 312 和 337 輪次的對話。我們也可從表中的 Avg. tokens 觀察到在每組對話平均都會有 45 個以上的字詞。在標記分析部分，我們計算了每個話語被自動標記的行為元組數量，並分別計算每個對話輪次中的平均行為數。而為了方便觀察使用者及助理在行為標記上的數量差異，我們也計算出使用者及助理每輪的平均行為數量。由於在對話中助理須主動向用戶確認多項來自用戶提出的訊息，故由表 3 中下半部分我們可得知三個對話語料之助理平均行為數(Avg. s-acts)皆高於用戶平均行為數(Avg. u-acts)。

為了更清楚了解 MsgWOZ 三個資料集的行為標籤及信息槽的分佈狀況，我們分析三個資料集中各個對話行為和訊息槽之數量，也繪製出對話行為的標記分佈，如表 4 及圖 6 所示。在對話行為標記部份我們發現，即使效仿 CrossWOZ 使用特定機率來生成特定行為的對話情境，Select 及 NoFound 這兩個對話行為之數量仍會受到當下標記人員的對話情境而影響。而在實際的對話情境中，我們常對虛擬助理傳達帶行為卻不帶任何訊息槽的話語 (如:幫我寄一封信、我想找一個活動...等)，為了確實將對話進行完整的分析，我們標記對話行為且給予未帶有任何槽及槽值的話語，並將其槽及槽值標記為”None”。

在社交言語（SOM）的行為中，我們標記了 7 種不同的社交言語。除了 bye、greet、thanks 三種常用之社交話語外，我們也定義了 confirm、done 及 reqmore 等助理請求用戶之話語標記。

表 4. 對話行為標籤及訊息槽數量統計
[Table 4. Dialogue Behavior Label and Message Slot Quantity Statistic]

		Train	Dev	Test	All
Dialogue actions	Inform	4,134	697	794	5,625
	Request	1,024	194	216	1,434
	Select	65	17	10	92
	SOM	1,611	320	328	2,259
	NoFound	16	6	3	25
Slots	Calendar				
	None	149	18	22	189
	Participant	268	34	40	342
	Name	436	56	68	560
	Is all day	50	6	7	63
	Content	245	37	54	336
	Location	290	34	40	364
	Event time	468	60	79	607
	Mail				
	None	28	4	3	35
	Subject	504	97	100	701
	Content	393	82	82	557
	Copy recipient	180	55	60	295
	Sender	122	24	22	168
	Bcc recipient	107	21	25	153
	Recipient	356	69	85	510
	Message				
	None	7	5	2	14
	User	505	99	95	699
	Content	581	110	121	812
	Application	550	103	118	771

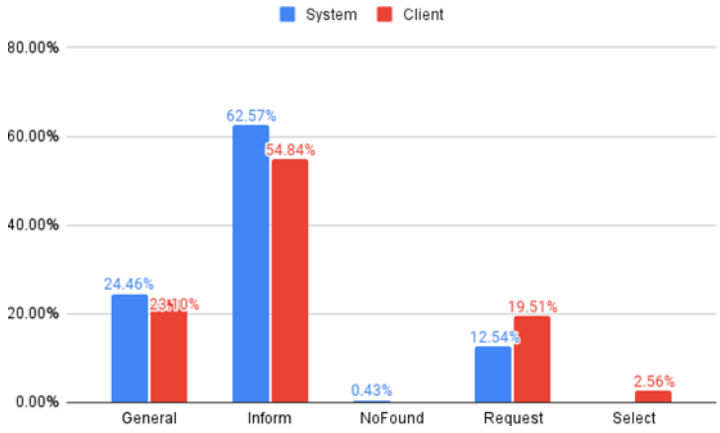

圖 6. MsgWOZ 之對話行為分佈
[Figure 6. Dialogue Behavior Distribution of MsgWOZ]

noneed 則為用戶向助理傳達不須提供額外訊息的回覆。表 5 為 MsgWOZ 中 7 種社交言語的數量統計。由於我們希望助理能夠主動且適時的向用戶確認任務訊息，故在表 5 能發現 confirm 和 reqmore 為較多的社交言語。

表 5. 社交言語 (SOM) 行為數量統計
Table 5. Quantity Statistic of SOM Behavior

	Train	Dev	Test	All
bye	5	3	1	9
greet	110	21	30	161
thanks	209	46	46	301
confirm	785	161	168	1,114
done	138	9	11	158
reqmore	296	69	65	430
noneed	68	11	7	86

為了進行 NLU 任務並評估資料可用性，我們使用 RoBERTa 進行訓練(Delobelle, Winters, & Berendt, 2020)。初始模型為中文預訓練的 RoBERTa(Cui *et al*., 2020)。NLU 任務包含意圖偵測和槽填充，我們使用 JointBERT 架構進行多任務學習(Chen, Zhuo, & Wang, 2019)，讓模型同時理解話語中的對話行為及具有意義的字詞。圖 7 展示了話語加上 CLS 和 SEP 標記後，輸入到 RoBERTa 中的流程，期望模型在 CLS 位置輸出話語的對話行為，其餘部分進行槽填充標記，預測各個字詞的起始(B)、中間(I)、其他(O 標記，以及所涉及的對話領域和資訊槽類型。在 MsgWOZ 訓練集上進行微調，考慮對話歷史，使用相同的 RoBERTa 模型，將對話歷史串接至 SEP 標記之後，模擬之前的對話上下文，並對

沒有上下文的情況進行了實驗。本實驗中，將模型的學習率設為 3e-5，Dropout 設置為 0.1。由於 BERT 預訓練模型最長輸入字串長度為 512 個字元，對訓練資料中過長的部分進行截斷，以避免超過模型的輸入長度限制。

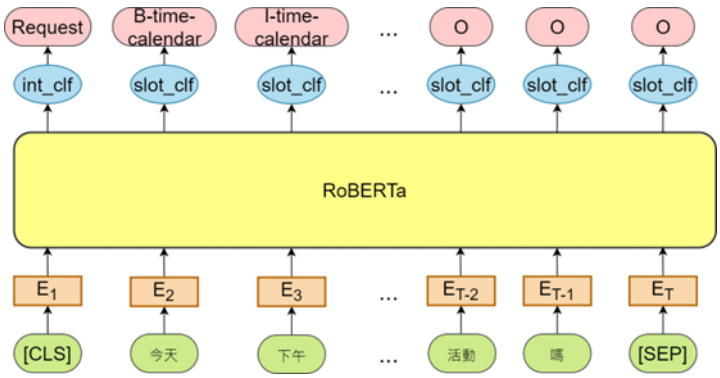

圖 7. 在本任務所使用到的 JointBERT 架構
[Figure 7. The BERT architecture in our NLU task]

我們使用 MsgWOZ 測試集對模型進行對話行為、對話領域及槽填充評估，結果如表 6 所示。此外，我們也針對(對話行為, 領域, 槽, 槽值) 的模型預測元組與正確答案計算 Overall F1 Score。在表 6 中可以清楚地發現使用對話上下文的模型(w/ context)表現得更好，而由 Slot, Overall F1 可得知，由於像郵件主旨、信件內容等槽值可能為帶有大量訊息的長序列，因此在預測效能上較其餘兩項指標差。

表 6. RoBERTa 模型在本語料上四項 NLU 任務之 F1 效能
[Table 6. F1 Performance of the RoBERTa model on four NLU tasks on this corpus]

Task	w/ context	w/o context
Dialogue Act (5)	90.56	89.33
Domain (3)	93.56	92.85
Dialogue slot (18)	80.52	78.44
Dialogue overall	84.78	83.27

5. 結論與未來展望 (Conclusion & Future Work)

在本研究中，我們以 Wizard-of-Oz 方式建立了 MsgWOZ 的中文對話語料，針對行事曆、電子郵件以及通訊軟體的三個領域進行了對話行為和槽值的標記，而此語料也是少數的中文任務型導向對話資料集。期望此資料集能夠為不斷增加新服務的虛擬助理提供訓練資料。另外，在標記系統中我們將資料庫及標記介面進行模組化，使標記人員可透過更換資料庫及任務目標來標記不同的資料。

整體說來，以 Wizard-of-Oz 方式建立對話語料，其對話內容完全由人類掌控，藉此可以產生較自然的對話語料；而其方式的缺點在於標註過程過於繁雜，標註人員不僅要在對話過程中同時對語料進行標註，且要顧慮某些實體是否於先前對話歷史中提及，因此標註人員對系統的熟悉程度會大幅影響整體資料集品質。

未來，我們將比較以 Schema-Guided 方式來產生任務型導向對話語料，希望透過自動標註，加速對話語料產生，避免人為標註煩瑣、確保標註的正確性，標註人員僅需針對機器生成的對話進行改寫，藉以讓語句變得更連貫，提升整體資料集品質。

Acknowledgements

本文為科技部產學合作研究計畫部分研究成果（計畫編號：MOST110-2622-E008-014）。

References

Budzianowski, P., Wen, T.-H., Tseng, B.-H., Casanueva, I., Ultes, S., Ramadan, O., & Gašić, M. (2018). MultiWOZ-a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, 5016-5026. Retrieved from https://aclanthology.org/D18-1547

Chen, H., Liu, X., Yin, D., & Tang, J. (2017). A survey on dialogue systems: Recent advances and new frontiers. *Acm Sigkdd Explorations Newsletter*, *19*(2), 25-35. https://doi.org/10.1145/3166054.3166058

Chen, Q., Zhuo, Z., & Wang, W. (2019). Bert for joint intent classification and slot filling. arXiv preprint arXiv:1902.10909.

Chiu, S., Li, M., Lin, Y.-T., & Chen, Y.-N. (2022). SalesBot: Transitioning from chit-chat to task-oriented dialogues. In *Proceedings of the 60th annual meeting of the association for computational linguistics (acl)*, 6143-6158.

Cui, Y., Che, W., Liu, T., Qin, B., Wang, S., & Hu, G. (2020). Revisiting pre-trained models for Chinese natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: Findings, 657-668. Online: Association for Computational Linguistics*. Retrieved from https://www.aclweb.org/anthology/2020.findings-emnlp.58

Delobelle, P., Winters, T., & Berendt, B. (2020). RobBERT: a Dutch RoBERTa-based Language Model. In *Findings of the association for computational linguistics: Emnlp 2020*, 3255-3265. Online: Association for Computational Linguistics. Retrieved from https:// aclanthology.org/ 2020.findings-emnlp.292

Hemphill, C. T., Godfrey, J. J., & Doddington, G. R. (1990). The atis spoken language systems pilot corpus. In *Speech and natural language: Proceedings of a workshop held at hidden valley*, pennsylvania, june 24-27, 1990.

Kelley, J. F. (1984). An iterative design methodology for user-friendly natural language office information applications. *ACM Trans. Inf. Syst.*, *2*(1), 26-41. Retrieved from https:// doi.org/ 10.1145/357417.357420

Lowe, R., Pow, N., Serban, I., & Pineau, J. (2015, September). The Ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the 16th annual meeting of the special interest group on discourse and dialogue*, 285-294. Retrieved from https://aclanthology.org/W15-4640

Ramadan, O., Budzianowski, P., & Gašić, M. (2018). Large-scale multi-domain belief tracking with knowledge sharing. In *Proceedings of the 56th annual meeting of the association for computational linguistics* (volume 2: Short papers), 432–437. Retrieved from https://aclanthology.org/P18-2069

Rastogi, A., Zang, X., Sunkara, S., Gupta, R., & Khaitan, P. (2020). Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(05), 8689-8696. Retrieved from https://ojs.aaai.org/index.php/AAAI/ article/view/6394

Raux, A., Langner, B., Bohus, D., Black, A. W., & Eskenazi, M. (2005). Let's go public! taking a spoken dialog system to the real world. In *proc. of interspeech 2005*.

Ritter, A., Cherry, C., & Dolan, B. (2010). Unsupervised modeling of Twitter conversations. In *Human language technologies: The 2010 annual conference of the north American chapter of the association for computational linguistics*, 172-180. Retrieved from https://aclanthology.org/N10-1020

Roller, S., Dinan, E., Goyal, N., Ju, D., Williamson, M., Liu, Y., … Weston, J. (2021). Recipes for building an open-domain chatbot. In *Proceedings of the 16th conference of the european chapter of the associa- tion for computational linguistics: Main volume*, 300-325). Online: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2021.eacl-main.24

Seneff, S., & Polifroni, J. (2000). Dialogue management in the mercury flight reservation system. In *Anlp-naacl 2000 workshop: Conversational systems*. Retrieved from https://aclanthology.org/W00-0303

Shah, P., Hakkani-Tür, D., Tür, G., Rastogi, A., Bapna, A., Nayak, N., & Heck, L. (2018). Building a conversational agent overnight with dialogue self-play. arXiv preprint arXiv:1801.04871.

Wen, T.-H., Vandyke, D., Mrkšić, N., Gašić, M., Rojas-Barahona, L. M., Su, P.-H., … Young, S. (2017, April). A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th con- ference of the European chapter of the association for computational linguistics*: Volume 1, long papers, 438-449).Retrieved from https://aclanthology.org/E17-1042

Zhu, Q., Huang, K., Zhang, Z., Zhu, X., & Huang, M. (2020). CrossWOZ: A large-scale Chinese cross-domain task-oriented dialogue dataset. *Transactions of the Association for Computational Linguistics*, 8, 281–295. Retrieved from https://aclanthology.org/2020.tacl-1.19