

THIẾT KẾ VÀ CHẾ TẠO CHATBOT GIAO TIẾP BẰNG NGÔN NGỮ VIỆT

DESIGN AND FABRICATION OF CHATBOT COMMUNICATING IN VIETNAMESE LANGUAGE

SVTH: Nguyễn Thành Tín, Trần Ngọc Linh, Phan Nhật Hạ

Lớp 15CDT2, Khoa Cơ khí, Trường Đại học Bách khoa – Đại học Đà Nẵng;

Email: thanhtin1997@gmail.com;

linhbosdaklak@gmail.com;

phanhatha1997@gmail.com

GVHD: Võ Như Thành

Khoa Cơ khí, Trường Đại học Bách khoa – Đại học Đà Nẵng;

Email: thanhvous@gmail.com

Tóm tắt - Giao tiếp giữa máy và con người đang đạt được những thành tựu đột phá. Ta có thể nhận ra điều này ở rất nhiều sản phẩm như Google Home, Amazon Echo hay Apple HomePod và được gọi chung là chatbot giọng nói. Tuy vậy, các chatbot giọng nói này chủ yếu sử dụng tiếng Anh, trong khi tiếng Việt vẫn còn chưa phổ biến. Bài báo này giới thiệu một Chatbot giọng nói Tiếng Việt mà ở đó, người dùng có thể dùng tiếng Việt để giao tiếp với Chatbot thông qua giọng nói. Từ đó Chatbot sẽ thực hiện các chức năng mà người dùng yêu cầu như hội thoại với các kịch bản soạn trước; thông báo thời gian, thời tiết; tra cứu thông tin trên Google và Wikipedia hay giao tiếp và điều khiển các thiết bị khác. Ngoài ra, bài báo còn giới thiệu ứng dụng của hologram để tạo nên một giao diện thân thiện hơn. Dự án sử dụng ngôn ngữ lập trình Python cũng như các nền tảng được xây dựng bởi điện toán của Google.

Từ khóa – chatbot; nhận diện giọng nói; nhận diện tiếng Việt; trí tuệ nhân tạo; SQL; hologram

1. Tổng quan về chatbot:

1.1. Chatbot là gì?

Chatbot về cơ bản là một hình thức thô sơ của trí tuệ nhân tạo. Nó hoạt động độc lập và có thể tự động trả lời các câu hỏi hoặc xử lý tình huống một cách tự nhiên. Độ chính xác, độ phức tạp và phạm vi sử dụng của chatbot được xác định bởi thuật toán của người tạo nên chúng.

Về cách thức hoạt động, Chatbot kết hợp các lời thoại được lập trình trước và tự học trong quá trình tương tác. Với các câu hỏi được đặt ra, chatbot sẽ dự đoán và phân hồi chính xác nhất có thể. Nếu tình huống đó chưa xảy ra (không có trong dữ liệu), chatbot sẽ bỏ qua nhưng sẽ đồng thời “bắt chước” để áp dụng cho các cuộc trò chuyện thường xuyên về sau

1.2. Chatbot giọng nói:

Sự khác biệt lớn nhất giữa các chatbot thông thường được thực hiện trên các nền tảng nhắn tin như Facebook Messenger và chatbot giọng nói là các chatbot giọng nói có thể được tích hợp trên các thiết bị như loa thông minh (Amazon Echo, Google Home), các thiết bị cầm tay (Airpod) và các thiết bị IoT khác và được kích hoạt cũng bằng giọng nói thay vì phải nhập lệnh từ bàn phím. Điều này cho phép chatbot giọng nói xuất hiện được trên nhiều môi trường sao cho phù hợp nhất với yêu cầu từ người sử dụng, đồng thời cũng tạo cho người dùng cảm

Abstract - Communication between machines and people is achieving dynamic achievements. In fact, we can recognize this fact in a wide range of products like Google Home, Amazon Echo or Apple HomePod. They are all called voice chatbots. However, these voice chatbots are mainly in English, while in Vietnamese is still not popular. This article introduces a Vietnamese voice Chatbot, in which users can use Vietnamese to communicate with Chatbot through voice. Chatbot can complete tasks from user's requests such as chatting with pre-written scripts; informing time and weather; searching on Google and Wikipedia or communicating and controlling other devices. In addition, the article introduces the hologram's application to create a friendlier interface. The project uses the Python programming language as well as the platforms built by Google.

Key words – chatbot; voice recognition; Vietnamese recognition; AI; SQL; hologram

giác chân thật và thân thiện hơn.

1.3. Tiềm năng phát triển:

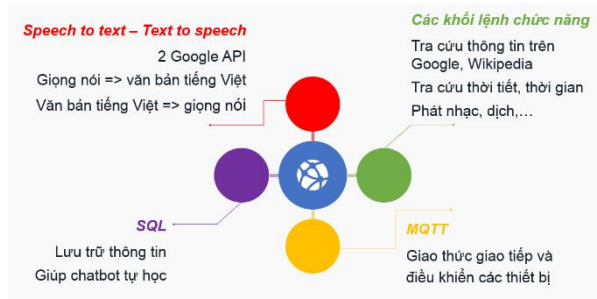
Chatbot nói chung và chatbot giọng nói nói riêng đang có tiềm năng phát triển rất lớn vì những công dụng to lớn của mình. Sự phát triển nhanh chóng của trí tuệ nhân tạo cho phép chatbot thực hiện được ngày càng nhiều chức năng hơn. Chatbot có thể đóng vai trò như một trợ lý ảo hỗ trợ người dùng thực hiện các nhiệm vụ hay tra cứu thông tin một cách tự động và tiện lợi. Thay vì phải cài đặt các ứng dụng cho từng công năng, chatbot có thể thay thế và hoàn thành tất cả mọi việc.

Ngoài lợi ích về mặt cá nhân, Chatbot đang dần trở thành một công cụ hữu ích cho các doanh nghiệp để tiếp xúc với khách hàng hay các dịch vụ hành chính, y tế để phục vụ người dân. Cùng với khả năng lưu trữ và tiếp cận được các nguồn thông tin về khách hàng cực kỳ lớn, Chatbot tỏ ra có ưu thế hơn con người trong vai trò của một nhà tư vấn, một nhân viên bán hàng khi mà mọi sở thích, yêu cầu của khách hàng đều được Chatbot sử dụng. Giảm được số lượng nhân viên cũng như đảm bảo việc tiếp xúc khách hàng hoặc người dân 24/24 cũng là những ưu thế to lớn của Chatbot trong các lĩnh vực này.

Xét riêng về chatbot giọng nói, theo số liệu chính thức của Amazon, hãng đã bán được hơn 20 triệu thiết bị Echo và hơn 20 ngàn thiết bị có sẵn trên Alexa Store. Mặt khác, Google đang cố gắng để làm cho những trợ lý

chatbot của họ có sẵn trên tất cả các thiết bị Android như smartphone, Androidwear, Smart tv, loa thông minh... Hiện nay cũng đã có nhiều nhà phát triển xây dựng chatbot bằng giọng nói như PullString hoặc Storyline và năm 2018 xu hướng này hứa hẹn sẽ tiếp tục phát triển hơn nữa.

2. Thiết kế Chatbot giọng nói tiếng Việt:



Hình 1. Các khối lập trình

2.1. Speech to text và Text to speech:

Đây là hai API được xây dựng bởi Google, hỗ trợ các nhà phát triển trong việc nhận dạng giọng nói và chuyển văn bản sang giọng nói. Hiện tại hai API này đã có hỗ trợ tiếng Việt, giúp cho Chatbot có thể giao tiếp bằng tiếng Việt với người sử dụng. Điều này còn khá mới với các chatbot giọng nói hiện nay khi mà chủ yếu sử dụng tiếng Anh.

Về cách thức sử dụng, giọng nói được thu qua bộ thu âm thanh và gửi lên Google Cloud để xử lý, sau đó trả lại một đoạn văn bản. Tương tự, nội dung Chatbot sẽ trả lời cũng được gửi lên Google Cloud để xử lý thành giọng nói để giao tiếp với người sử dụng.

Python đã cung cấp sẵn thư viện giao tiếp với 2 API này của Google. Ta cần khai báo như sau:

```
From googletrans import Translator
Import speech_recognition
From gtts import gTTS
Import sys
Import pyttsx
```

2.2. SQLite:

```
elif u"lịch bản" in str:
    #ai soạn kịch bản cho anh và nhớ nói chuyện
    c.execute("SELECT cauhoi FROM hoi")# WHERE cauhoi = "?"[str])
    data = c.fetchall()
    if str in [i[0] for i in data]:
        c.execute("SELECT ma FROM hoi WHERE cauhoi = '?'[str])
        dataa = c.fetchall()
        for j in dataa:
            straa = j[0]
        c.execute("SELECT name FROM traloi WHERE ma = '?'[straa])
        datab = c.fetchall()
        for k in datab:
            strab = k[0]
            print(strab)
            say(strab)
    else:
        c.execute("INSERT INTO hoi(cauhoi,ma) VALUES (?,?)", [str])
        conn.commit()
        print("Nhớ không biết câu trả lời, đây nhớ đi.")
        say("Nhớ không biết câu trả lời, đây nhớ đi.")
        r1 = speech_recognition.Recognizer()
        with speech_recognition.Microphone(sample_rate = sample_rate, chunk_size = chunk_size) as source1:
            r1.adjust_for_ambient_noise(source1)
            audio1 = r1.listen(source1)
            print("Done!")
            str1 = r1.recognize_google(audio1, language = "vi-VN")
            print("Tôi nghĩ rằng bạn đã nói: " + str1)
            c.execute("INSERT INTO traloi(name,ma) VALUES (?,?)", [str1])
            conn.commit()
```

Hình 2. Chương trình chạy SQL

SQL là viết tắt của Structured Query Language, một ngôn ngữ truy vấn mang tính cấu trúc và được thiết kế để quản lý dữ liệu trong một hệ thống quản lý cơ sở dữ

liệu quan hệ (RDBMS).

SQL có các chức năng:

- Tạo cơ sở dữ liệu, bảng và view mới.
- Chèn các bản ghi vào trong một cơ sở dữ liệu.
- Xóa các bản ghi từ một cơ sở dữ liệu.
- Truy vấn database theo nhiều cách khác nhau bởi các lệnh.
- Người dùng có thể truy cập dữ liệu từ RDBMS.
- Cho phép người dùng miêu tả dữ liệu.
- Cho phép người dùng định nghĩa dữ liệu trong một database và thao tác khi cần thiết.
- Cho phép người dùng tạo, xóa cơ sở dữ liệu và bảng.
- Cho phép người dùng tạo view, Procedure, hàm trong một Database.
- Cho phép người dùng thiết lập quyền truy cập vào bảng, thủ tục và view.

Tại sao phải sử dụng SQL? Khi Chatbot nhận được một câu hỏi không có trong hệ thống dữ liệu, Chatbot sẽ cập nhật câu trả lời bằng cách ghi thêm vào cơ sở dữ liệu câu trả lời vừa nhận được từ người dùng. Để làm được điều này, ta cần phải sử dụng SQL. Người dùng khi sử dụng từ khóa “lịch bản” trong câu nói của mình, nếu Chatbot nhận thấy câu trả lời chưa được lên kịch bản trước sẽ hỏi lại người sử dụng. Câu trả lời nhận được từ người sử dụng sẽ được lưu vào cơ sở dữ liệu của Chatbot.

id	cauhoi	ma
1	Xin chào tôi là Nguyễn Thành Tín	sinchao
2	Xin chào	sinchao
3	Xin chào bạn	sinchao
4	chào buổi sáng	sinchao
5	Bạn có khỏe không	hoi
6	Chào ngày hôm	hoi
7	chào mọi người	hoi
8	cho mọi người	hoi
9	ai soạn kịch bản cho anh và nhớ nói chuyện	hoi
10	ai soạn kịch bản cho anh và nhớ nói chuyện	hoi
11	soạn kịch bản cho anh và nhớ trả lời	hoi
12	lịch bản	hoi
13	lịch bản tính là tháng ngày	hoi
14	lịch bản tiếng việt là một tháng ngày	hoi
15	lịch bản tiếng hóa là một tháng ngày	hoi
16	lịch bản tiếng hoi	hoi
17	lịch bản lịch bản	hoi
18	lịch bản giọng dễ thật	hoi
19	lịch bản giọng dễ thật	hoi
20	lịch bản giọng dễ thật	hoi
21	lịch bản giọng dễ thật	hoi

Hình 3. Giao diện dữ dụng SQL

Lấy ví dụ với bảng “hoi” như Hình 3. Cột “id” có thể coi như số thứ tự và có thuộc tính tự tăng. Cột “cauhoi” và “ma” cho thuộc tính ta cần và không được bỏ trống. Để chatbot có thể trả lời câu hỏi sau mà không lặp lại câu trả lời của câu hỏi trước, ta gán mỗi bộ câu hỏi và câu trả lời với một mã riêng. Đó là tác dụng của cột “ma”.

Để lấy dữ liệu từ một bảng dữ liệu trong SQL, ta dùng lệnh “select”. Xét lại ví dụ trên. Đầu tiên, ta “select” cột “cauhoi” của bảng “hoi” và so sánh câu hỏi người dùng mới đưa ra với tất cả các câu. Nếu có, chatbot sẽ lấy mã ở cột “ma” và qua một bảng khác tên là “traloi” để lấy câu trả lời. Nếu không có câu nào giống, máy sẽ chèn câu hỏi đó vào bảng “hoi” với số id tăng dần, đồng thời phát ra câu “Nhớ không biết câu trả

lời, dạy nhỏ đi”. Người dùng đọc câu trả lời để máy thêm vào bảng “tralo”.

2.3. Các khối lệnh chức năng:

2.3.1. Tra cứu thông tin trên Wikipedia:

```
import wikipedia
import webbrowser

def wiki(str):
    try:
        wikipedia.set_lang("vi")
        result = wikipedia.summary(str, sentences=2)
        url = "http://vi.wikipedia.org/w/index.php?title="
        #webbrowser.open_new_tab(url+str)
        return result

    except wikipedia.exceptions.PageError:
        url1 = "http://www.google.com/?#q="
        webbrowser.open_new_tab(url1+str)
        return "I did not find any information regarding that query, here the related search results"
```

Hình 4. Khối lệnh tìm kiếm trên Wikipedia

Đối với tra cứu trên Wikipedia, cần sự hỗ trợ của MediaWiki action API. Để kích hoạt chức năng, người dùng cần nói “Nói cho anh về ABC”, với ABC là đối tượng cần tra cứu.

2.3.2. Tra cứu thời gian, thời tiết:

Chức năng tra cứu thời tiết cũng được hỗ trợ bởi Weather API. Chatbot sẽ cung cấp cho người sử dụng các thông tin về nhiệt độ, tốc độ gió, độ ẩm và áp suất không tại một thành phố nào đó. Người dùng cần nói “Nói cho anh nghe thời tiết của ABC”, với ABC là thành phố mà các thông tin về thời tiết được yêu cầu.

Để tra cứu thời gian, ta nhập các lệnh sau trong Python. Người dùng nói các câu nói “Bây giờ là mấy giờ”, “Mấy giờ rồi” để có được thông tin cần thiết.

```
elif str == "Bây giờ là mấy giờ" or str == "Mấy giờ rồi" or str == "what's the time" or str == "time" or str == "what is today" or str == "what's today" :
    print(localtime + "\n")
    tts = gTTS(localtime, lang='vi')
    tts.save("good.mp3")
    os.system("mpg321 -o alsa good.mp3")
```

Hình 5. Khối lệnh thời gian

2.3.3. Dịch:

Chatbot thực hiện chức năng này nhờ sự hỗ trợ của API Translation của Google. Vì vậy, người dùng có thể dịch nhiều ngôn ngữ khác nhau với độ chính xác ngày càng tăng lên. Người dùng có thể nói “Dịch cho tôi từ ABC”, với ABC là từ cần dịch.

2.3.4. Các khối lệnh if:

Khi phát hiện người lại gần nhờ cảm biến hồng ngoại, Chatbot sẽ được kích khởi và phát ra các câu chào hỏi phù hợp với hoàn cảnh và thời điểm giao tiếp như các lời chào vào các buổi sáng, trưa, chiều của một ngày.

Dự án xây dựng một số kịch bản để giao tiếp bằng các khối lệnh if. Nếu trong câu nói có các từ khóa đã được quy định, Chatbot sẽ nói những câu đã được lập trình sẵn.

2.3.5. Phát nhạc:

Chatbot có thể phát các bản nhạc bằng các lệnh có sẵn trong Python. Từ khóa người dùng cần nói là “Bài

hát ABC”, với ABC là tên bài hát.

```
import pygame

def music():
    global clip
    clip = pygame.mixer.Sound("hello.mp3")

def playsong():
    global clip
    clip.play()
    print("Playing music")

def pausesong():
    global clip
    clip.pause()
    print("Music is paused")

def unpausesong():
    global clip
    clip.unpause()
    print("Music is unpaused")

def stopsong():
    global clip
    clip.stop()
    print("Music is stopped")
```

Hình 6. Khối lệnh phát nhạc


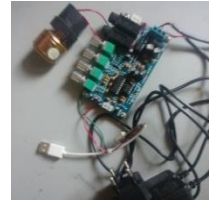
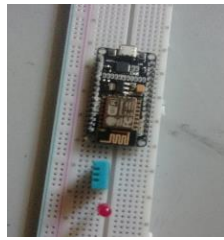
2.3.6. Đọc giá trị nhiệt độ, độ ẩm và điều khiển led:

Thông qua ESP8266, Chatbot có thể đọc được các giá trị của cảm biến nhiệt độ, độ ẩm và cho phép người dùng biết được chính xác các giá trị này tại vị trí của mình. Đồng thời, Chatbot cũng cho phép điều khiển led thông qua giọng nói vào bộ thu âm thanh nối liền với bộ xử lý chính.

2.4. MQTT:

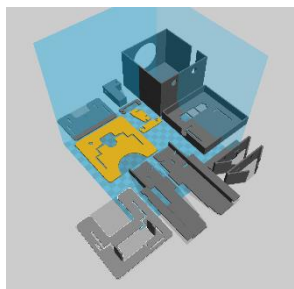
MQTT (Message Queuing Telemetry Transport) là một giao thức gửi dạng publish/subscribe được sử dụng trong các thiết bị IoT với băng thông thấp, độ tin cậy cao và khả năng được sử dụng trong mạng lưới không ổn định. Trong dự án này, MQTT cho phép giao tiếp giữa ESP8266 và các cảm biến hồng ngoại, nhiệt độ, module điều khiển led.

2.5. Các thiết bị sử dụng:

Stt	Thiết bị	Chức năng	Hình ảnh
1	Raspberry Pi	Bộ xử lý chính	
2	Bộ thu âm thanh	Thu giọng nói từ xa	
3	ESP8266	Đọc nhiệt độ, độ ẩm; điều khiển led	
4	Cảm biến hồng ngoại	Nhận biết có người lại gần	
5	Cảm biến nhiệt độ, độ ẩm	Thu thập giá trị nhiệt độ, độ ẩm	

Bảng 1. Các thiết bị sử dụng

2.6. Thân Chatbot



Hình 4. Các bộ phận khung Chatbot được in 3D

Các bộ phận của thân được chế tạo bởi máy in 3D. Phần thân giúp Chatbot trở nên đẹp mắt và thân thiện với người dùng. Đồng thời, đây cũng là nơi chứa Raspberry pi và micro để gọn gàng hơn.

3. Kết quả nghiên cứu và khảo sát:

Stt	Nội dung	Kết quả trả về	Kết quả (%)
1	Aloalo 123 321	Aloalo 123 321	100
2	Coi chừng nguy hiểm	Coi chừng nguy hiểm	100
3	Cha	Cha	100
4	Mẹ	Mẹ	100
5	Chó	Chó Không kết quả	60 40
6	Tró	Chó	100
7	Trường	Trường	100
8	Chường	Chưa Chường Không kết quả	20 30 50
9	Huynh	Khuynh Không kết quả Huỳnh	20 20 60
10	Huynh đệ	Huynh đệ	100
11	Huynh (sau khi đọc chữ "huynh đệ")	Huynh Huỳnh	40 60
12	Cắt	Cách Cắt Cách làm hoa đẹp	50 40 10
13	Cắt	C** Kích Cấp (Sau khi nói từ "Cắt giữ")	30 50 20
14	Cắt giữ	Cắt giữ	100

Bảng 2. Kết quả kiểm tra nhận diện giọng nói

3.1. Nhận diện giọng nói:

Sau khi cho hai giọng nói đến từ 2 vùng khác nhau thực hiện nói vào micro 10 lần, ta thu được kết quả nhận diện giọng nói và xử lý tiếng Việt như trong bảng 2.

Đối với các câu dài hơn 15-20 từ, độ chính xác đạt 70-97%.

3.2. Thông báo thời tiết:

Người dùng (N) hỏi Chatbot (C) về thông tin thời tiết Hà Nội, Đà Nẵng, Thành phố Hồ Chí Minh vào lúc 15h25-16h08 27/04/2018, thu được các kết quả giống nhau đối với mỗi thành phố. Cụ thể như sau:

Người: Nói cho anh nghe thời tiết Đà Nẵng

Đáp: (từ 15h25 đến 16h04) Dự báo thời tiết thành phố Đà Nẵng: nhiệt độ 30.16 độ C, tốc độ gió 6.7mét trên giây, độ ẩm 79%, áp suất không khí 1011atm.

(16h04) Dự báo thời tiết thành phố Đà Nẵng: nhiệt độ 30.19 độ C, tốc độ gió 6.7mét trên giây, độ ẩm 79%, áp suất không khí 1011atm.

Hỏi: Nói cho anh nghe thời tiết Thành phố Hồ Chí Minh.

Đáp: (từ 15h25 đến 16h04) Dự báo thời tiết thành phố Hồ Chí Minh: nhiệt độ 33.0 độ C, tốc độ gió 5.1 mét trên giây, độ ẩm 63%, áp suất không khí 1008atm.

Hỏi: Nói cho anh nghe thời tiết Hà Nội

Đáp: (từ 15h25 đến 16h04) Dự báo thời tiết thành phố Hà Nội: nhiệt độ 24.0 độ C; tốc độ gió 3.1mét trên giây, độ ẩm 78%, áp suất không khí 1014atm.

3.3. Kiểm tra SQL:

Dưới đây là một số đoạn hội thoại với Chatbot mà nếu câu trả lời chưa có trong cơ sở dữ liệu, Chatbot sẽ hỏi lại người dùng câu trả lời và sẽ trả lời vào lần hỏi tiếp theo. Lưu ý là cần phải có từ khóa "kịch bản" trong câu nói để Chatbot có thể thực hiện chức năng ghi dữ liệu mới của mình.

- Hội thoại 1:

N: Kịch bản Tín thật ngốc

C: Nhỏ không biết câu trả lời, dạy nhỏ đi

N: Cũng đúng ha

N: Kịch bản Tín thật ngốc

C: Cũng đúng ha

N: Kịch bản Tín không ngốc

C: Nhỏ không biết câu trả lời, dạy nhỏ đi

N: Đúng vậy

N: Kịch bản Tín thật ngốc

C: Nhỏ không biết câu trả lời, dạy nhỏ đi

- Hội thoại 2:

N: Kịch bản Giọng bạn dễ thương thật

C: Nhỏ không biết câu trả lời, dạy nhỏ đi

N: Cảm ơn bạn

N: Kịch bản Giọng cậu dễ thương thật

C: Cảm ơn bạn

N: Kịch bản Giọng cậu dễ thương thật

C: Nhỏ không biết câu trả lời, dạy nhỏ đi-

4. Bàn luận:

4.1. Từ các kiểm tra ở trên:

4.1.1. Nhận diện giọng nói:

a) Các từ càng thông dụng thì độ chính xác càng cao:

Có thể thấy rằng, với các từ phổ biến như cha, mẹ... thì độ chính xác có thể lên đến 100%.

Tuy nhiên, với các từ ít thông dụng hơn như chuông, tró... Chatbot sẽ quy về các từ thông dụng. Nếu nói các từ ghép chứa các từ này (như “huynh” trong “huynh đệ”) thì sau đó độ chính xác của các từ đơn sẽ cao hơn.

b) Độ chính xác của từ ghép hoặc một câu ngắn cao hơn của từ đơn:

Các từ ghép hoặc các câu ngắn có xu hướng đạt độ chính xác cao hơn. Điều này có thể lý giải như sau: có thể từ đơn là một từ ít được sử dụng, nhưng từ ghép hoặc câu ngắn chưa nó có thể thường xuyên xuất hiện. Vì vậy dựa theo ý a), ta có được kết luận trên.

c) Dễ bị lẫn lộn các từ có âm gần giống nhau:

Vì đặc trưng của từng giọng nói có thể khác nhau, vì vậy việc Chatbot có thể phân biệt các âm gần như nhau sẽ rất khó khăn và thường xuất hiện những nhầm lẫn.

4.1.2. Thông báo thời tiết:

a) Số địa điểm bị hạn chế:

Vì thông tin được lấy từ website openweathermap.org ở nước ngoài nên số lượng địa điểm bị hạn chế, cụ thể ở đây là ba thành phố Hà Nội, Đà Nẵng, Thành phố Hồ Chí Minh.

b) Các thông tin được cập nhật theo giờ:

Vì các thông tin được thu từ vệ tinh nên kết quả sẽ thay đổi theo từng khoảng thời gian.

4.1.3. Sử dụng SQL để tạo các kịch bản hội thoại:

a) Các câu hỏi phải tuyệt đối chính xác:

Từ đoạn hội thoại 1 và 2, ta có thể thấy rằng các câu hỏi phải tuyệt đối chính xác như những gì chúng ta đã nói với Chatbot, tức là đúng với những thông tin ghi trong cơ sở dữ liệu thì Chatbot mới có thể thực hiện vai trò của mình.

b) Kịch bản mới được ghi chồng lên kịch bản cũ:

Ta có thể thấy từ đoạn hội thoại 1, sau khi đã được dạy câu trả lời về câu nói “Tín không ngốc”, Chatbot trả lời câu “Tín thật ngốc” theo câu trả lời của “Tín không ngốc”. Từ đó có thể rút ra được, kịch bản vừa được thu nhận đã ghi đè lên thông tin của đoạn hội thoại trước trong cơ sở dữ liệu.

4.2. Các hạn chế chung của Chatbot:

4.2.1. Cần phải chứa từ khóa hoặc đúng tuyệt đối với các dữ liệu trong cơ sở dữ liệu:

Như ở trên các từ khóa có thể gây nhầm lẫn khi mà nó trùng với từ xuất hiện trong câu và có thể gây sự hiểu lầm.

Ngoài ra, cần phải đúng chính xác dữ liệu trong cơ sở dữ liệu thì Chatbot mới hoạt động hiệu quả.

Khắc phục: Áp dụng máy học cho Chatbot để Chatbot có thể xử lý tự nhiên, linh hoạt hơn. Tuy nhiên, để có thể làm được điều này cần có chi phí lớn để huấn luyện Chatbot trở nên thông minh hơn vì lượng dữ liệu và lệnh cần sử dụng là rất lớn.

4.2.2. Hiệu năng và tốc độ xử lý:

Sau một khoảng thời gian sử dụng, Chatbot có thể gặp rắc rối, thời gian xử lý lâu hơn hoặc có thể bị ngưng lại và cần kích hoạt lại để trở lại như cũ.

Ngoài ra, với các câu nói dài, khả năng chính xác sẽ giảm xuống. Đồng thời, thời gian đến khi bị đứng lại nhanh hơn.

Việc này có thể lý giải bằng khả năng hoạt động của mạng Internet, giới hạn bộ nhớ và khả năng xử lý của raspberry pi.

4.3. Phương hướng phát triển:

4.3.1. Kết hợp học máy:

Để Chatbot có thể giao tiếp tự nhiên và chính xác hơn, ta cần đến xử lý ngôn ngữ tự nhiên (NLP) cũng như các thuật toán giúp huấn luyện Chatbot.

4.3.2. Tạo giao diện thân thiện hơn:

Cùng với việc ứng dụng hologram, dự án sẽ ngày càng phát triển nhân vật để trở nên ngày càng xinh đẹp, tự nhiên và linh hoạt.

4.3.3. Thay đổi tần số âm thanh:

Để Chatbot tạo ra được giọng nói ở khắp mọi vùng miền, giúp người dân ở từng vùng miền có thể cảm thấy tự nhiên hơn khi giao lưu với Chatbot.

4.3.4. Giao tiếp và điều khiển nhiều thiết bị IoT:

Chatbot hướng tới mục tiêu là một trạm điều khiển các thiết bị ở nhà, văn phòng, nơi công cộng... Đọc các giá trị cần thiết của các thiết bị, từ đó điều khiển các hoạt động các thiết bị sao cho phù hợp.

5. Kết luận:

Bài báo này đã giới thiệu một Chatbot giọng nói tiếng Việt. Bài báo cũng đưa ra ba kiểm tra đối với các chức năng nổi bật so với các chatbot khác như nhận diện được giọng nói tiếng Việt hay tạo nên các kịch bản giao tiếp và được lưu vào database của Chatbot thông qua SQL. Chatbot giọng nói này cũng được giới thiệu với vai trò như một trợ lý ảo bởi nhiều chức năng cụ thể. Tuy vẫn còn nhiều hạn chế về kiến thức hay chi phí nhưng bước đầu, dự án đã xây dựng thành công một Chatbot giọng nói tiếng Việt và hứa hẹn sẽ còn nhiều cải tiến trong tương lai.

Tài liệu tham khảo:

- [1] <http://vietjack.com/sqlite/>
- [2] <http://vietjack.com/python/>
- [3] <https://unica.vn/lap-trinh-python-tu-zero-hero>
- [4] <https://github.com/vcharlieboi/ultron>
- [5] <https://github.com/nihal111/J.A.R.V.I.S>
- [6] <https://vietnambot.com/>