PathBot: An Intelligent Chatbot for Guiding Visitors and Locating Venues

Katlego Mabunda and Abejide Ade-Ibijola

Formal Structures, Algorithms, and Industrial Applications Research Cluster
Department of Applied Information Systems
School of Consumer Intelligence and Information Systems
University of Johannesburg, Auckland Park Bunting Road Campus
Johannesburg, South Africa
e-mail: mabunda.katlego@gmail.com, abejideai@uj.ac.za

***Abstract*—This article reports the development of an intelligent Chatbot called `PathBot`. `PathBot` is used to guide visitors to locate venues on the University of Johannesburg's Bunting Road campus. The conversation of `PathBot` happens through a Mobile Application client. `PathBot` processes user's text using the DialogFlow API. We have designed a Finite Automaton that represents a sub-map of the campus. Extracted tokens from the DialogFlow API is sent to this Automaton, and an algorithm is used to generate descriptions of paths from where a user is, to where they want to go. An evaluation of the user's opinion about this tool and its userability was conducted and most users agreed that this Automaton-driven tool is helpful in locating venues on the mapped campus.**

Keywords-artificial intelligence; finite automata; Chatbot, natural language processing; path-finding algorithm

I. Introduction

Chatbots have been increasingly adopted by most organisations to run some of their operations, either by communicating with customers, their employees or with other business units. Generally, AI technology has been seen as a concept that will dramatically transform the business world and human society [29]. Ransbotham, Kiron, et. al [33] stated that over 85% of global executives perceive AI as an enabler for competitive advantage and will allow companies to gain or sustain a competitive advantage. The driver behind the development of Chatbots is due to the desire for a human to express their interest, wishes, or queries directly and naturally, by speaking, typing, and pointing [5].

A Chatbot can be defined as a computer program designed to interact with users using natural language. It is interactive and seeks to engage in human-like conversations. It can also be referred to as the conversational agent, chatters, or artificial conversation entities [1]. Chatbots have now become a crucial part of the corporate strategy by providing better customer services to the automated IT help desks and other areas in the business and provide solutions to a variety of industries [29].

There are few Mobile Applications (Apps) designed for navigating through the campus such as *Deakin Scout* designed by the Deakin University in Australia [30] and *Find My Class* mobile App used by several universities in the United States of America (USA) [19]. These Apps serve the same purpose of helping students find their way around the campus [19] [30], however, they are not Chatbots. The existing Mobile Apps are not AI Chatbots, they follow the old known method of selecting the origin and destination from a drop-down list and it gives you directions. There is a need for a Chatbot for locating venues at University of Johannesburg Bunting Campus. Mobile Apps such as Google Maps or other developed Maps cannot assist as they can only direct you up to the campus gate but cannot navigate you inside the campus and walk you inside the buildings.

This study combined existing methods and frameworks developed in Information Technology related fields to develop an AI Chatbot called `PathBot` that uses a Finite Automaton to introduce AI in universities as the world is headed to a 4th Industrial Revolution. `PathBot` enables students to be able to navigate to areas at the university that they were initially struggling to locate, without asking around for directions. `PathBot` follows the approach of many Chatbots, but the difference is it uses a Finite Automaton. `PathBot` follows the process shown in Figure 1 which involves a user interacting with the `PathBot's` User Interface (UI). The user will request for a location, and the `PathBot` will retrieve the path or directions constructed by a Finite Automaton and reply to user with a list of directions.

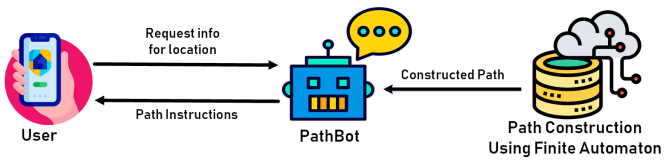


Figure 1. Process flow for PathBot.

This study made the following contributions. We have:

1) Developed a Finite Automaton that constitutes of states and transitions. States serve as a list of buildings at the

978-1-7281-4577-8/19/$31.00 ©2019 IEEE

UJ Bunting Road campus and transitions are paths to navigate from one state to another,
2) Developed an AI Chatbot called `PathBot` for navigating from one building to another at the UJ Bunting Road campus, and
3) Evaluated the `PathBot`, and the results shows that 90% of students find it difficult to navigate around the campus and 100% find `PathBot` to be a useful tool.

The rest of this paper is structured as follows: Section 2 represents the background and related work in the field. Section 3 describes the Finite Automaton and the Graph Traversal Algorithm. Section 4 outlines the implementation of `PathBot` and depicts real examples. Section 5 discusses the evaluation and Section 6 discusses the conclusion and the future work to further the study.

II. Background and Related Work

A. Background

Chatbots are the most fascinating computer user interfaces that are rising in popularity. Examples of Chatbots include but not limited to ELIZA, which is a program that simulates a psychotherapist [41], PARRY which simulates a paranoid patient by studying the nature of paranoia [11]. Other forms of Chatbots are available online, such as a multi-award-winning Mitsuku, it is a Chatbot designed for entertainment and not designed such as other Psychotherapist Chatbots that can solve users problems [32], it is used for chatting when one needs someone to talk to.

The recent development of a Chatbot is Apples SIRI, a virtual assistant capable of speech recognition and spoken communication. What all these Chatbots seek is to be as close as natural human language as possible [25]. The type of communication humans has with machines or Chatbots is the kind that is desirable and resembles the coherence of natural communication between human [20]. Park, Aiken, et. al [32] articulated that a Chatbot has good conversational abilities as it is good with emulating a human conversation, and it learns through knowledge.

As Mobile Apps keep rising, smartphone users are increasingly interested in the method of communication offered by a Chatbot [22]. Gartner [27] articulated that there will be an increase in acceptance of Mobile Apps and that by 2020 the use of these Mobile Apps will be used to seek virtual assistance from Chatbots [27]. Another estimation made by Gartner [16] stated that by 2020, 25% of customer service communications will be handled by Chatbots and by 2021 more than 50% of businesses will spend on Chatbots [17].

In addition to that, Business Insider [8] articulated that 80% of businesses will automate most of their processes in the form of Chatbot technology. In the banking and insurance sectors Foye [13] predicted that by 2022 there will be $8 billion cost-saving per year when Chatbots are utilised. Other financial institutions such as Absa Group Limited (AGL) have even started with the adoption of this concept and have introduced chat services on WhatsApp for their customers to perform their transactions. The transactions are the same as the ones on their mobile banking app and internet banking, a customer can get their balance, buy electricity, pay beneficiaries, check credit score, and other transactions that the bank offers [8].

B. Benefits of Using Chatbot

This section discusses some of the benefits of Chatbots.

1) Chatbots used for customer services can result in productivity as Chatbots reply quickly and are convenient because they are available 24/7. They also provide quick access to information [6].
2) On the messenger system, chatbots can be accessed by a broad number of people and personalised messages can be automated [3].

C. Drawbacks of Using Chatbots

This section highlights some of the disadvantages of using Chatbots.

1) Chatbots Vs Human: Thompson, Gallacher et. al. [38] examined how suitable are Chatbots as conversation partners and also to examine the interest of learners in the English language when they interact with a Chatbot versus when they interact with human partners. The results showed that other students have no problem with using technology, however, others prepare the traditional way of learning. There was a decline in students' performance and that is due to the Chatbot responding inappropriately and the conversation not being natural.
2) Request-Response: Cho and Lee [9] articulated that Chatbots work on a request-response pair description, meaning a user has to ask first for the robot to react and respond accordingly. The robots do not work without the user prompting first, it cannot be designed in such a way that the robot interrogates first [9]. When it is designed appropriately it will then be able to respond accordingly, however, it will not be able to go back to the original conversation afterward.

D. Analysis of Related Work

In this section, two types of Chatbots are discussed, Rule-based and AI Chatbots. Rule-based Chatbots uses the if/then statements which are used to hold basic conversations and AI Chatbots can understand idioms and slang words since humans sometimes do not speak in a logical way.

1) Rule-based Chatbots: Rule-based Chatbots also called decision-tree bots are Chatbots that use the if/then rules statements [37]. The defined rules use logical steps and direct-action buttons. Rule-based Chatbots are useful when the Chatbot is for resolving simple queries, as the questions must be direct and simple [37]. The rules are as follows:

```
Rule 1: If 1 and 2 then 3
Rule 2: If 2 and 3 then 5
Rule 3: If 5 then 6
Rule 4: If 6 then 5
```

2) AI Chatbots: AI Chatbot seek to be as natural as possible when interacting with users. It learns the natural language as it interacts with the user to provide better answers. It updates a Machine Learning engine with every new word from its interaction with the user in order to be able to respond better next time. The more you interact with it the better it becomes with responses. The examples of AI Chatbots include but not limited to NLP and Machine Learning. Sentences contain few elements that can be extracted as entities which will form as basis for the Chatbot to give answer to. For instance, in a sentence that says:

```
"Hi, I am looking for a one bedroom
apartment in Waterfall Johannesburg.
My budget is R7000.00 and it should
have a Gym."
```

Words such as “one apartment”, “Oakdene”,”R5000.00” and “Gym” are what the Chatbot will extract using tokenisation and return the answer based on what it has stored in the NLP database.

3) Features of Chatbots: AI and Rule-based Chatbots are developed differently. This section describes the features of the three Chatbots and shows a list of different Chatbots that serve different purposes in Table I.

a) Artificial Linguistic Internet Computer Entity (ALICE): ALICE is a web-based Chatbot which stands for Artificial Linguistic Internet Computer Entity [4] [40]. It uses pattern-matching algorithms, where a large number of basic categories or rules are used to match input patterns to output templates [2]. In its architecture, it ensures that language knowledge models are plugged and played properly, and this is caused by separating the Chatbot engine and the language knowledge model. The knowledge-base is called the Graphmaster, which is a graph with vertices and edges. Pattern matching is conducted using a Depth-First Search (DFS) on the knowledge-base [36].

It uses AIML files which stands for Artificial Intelligence Mark-up Language to store English conversation patterns knowledge. AIML is a feature of Extensible Mark-up Language (XML) developed by the Alice free software community and Dr. Richard S. Wallace between 1995 and 2000[3]. AIML is composed of data object made up of tags which includes topics, aiml, patterns, template, categories etc [39]. The format of AIML is depicted below in algorithm 1:

```
<?xml version = "1.0" encoding = "UTF-8"?>
<aiml version = "1.0.1" encoding = "UTF-8'?
    <topic>
        <category>
            <pattern> </pattern>
            <template> </template>
        </category>
    </topic>
```

Algorithm 1: AIML example [1]

Although there are good things that ALICE as a Chatbot has, there is however few things it could improve on. ALICE uses a simple pattern template to constitute the user’s input and the corresponding output. It uses a technique called recursive which is considered as a power point of the system, which the technique used to simplify the user’s input by calling categories that match recursively [36]. The strongest and most important issue that ALICE has is with its pattern matching algorithm, which is conspired to be too easy and that it depends on DFS which finds the longest pattern matching [36]. With its many benefits, it however still fails the Turing test, which is designed to test natural language, reasoning, knowledge and learning in Chatbots. ALICE displays poor grammar matching, it does not show extensive knowledge because it sometimes skips an answer in favour of a random word, which then disqualifies it to be considered an NLP tool [26].

b) Brisbot: Brisbot is a Chatbot designed by the Swedish helpline BRIS using Kik and Facebook Messenger’s Application Programming interface (API) to help troubled children with the problems they are experiencing in their day-to-day lives. BRIS is a non-profit organisation running a helpline for children until a teenage level nationally [34]. This Chatbot is used to give advice and answers the questions that the children of today face. The answers on the Chatbot are written by professional child counsellors [31]. It was designed in such a way that only pre-programmed questions are selected by the user in order for them to get answers, meaning it does not respond to open questions and has a free-text feature [34].

Brisbot is like a Frequently Asked Questions (FAQ) used in a format of a Chatbot which uses general pre-written questions and answers to maintain anonymity of users, no personal information can be added. BRIS uses this Chatbot as an addition to their regular support line. With little research done on Brisbot, there is currently no idea how the Brisbot was designed. However, most Chatbots that uses Facebook Messenger’s API, Slack, and Telegram are designed or built using Chatfuel, DialogueFlow, Botsify and many other Machine Learning tools. These tools help with building a Chatbot without having to code, they are web-based tools with a user interface editor to help with designing a Chatbot.

This study concludes by arguing that although Brisbot is a great tool to help the children figure out answers to the problems they are experiencing and that it can be easily accessible on the Facebook Messenger page, it however cannot be considered an NLP tool. It does not allow communication to flow such as a normal human-like conversation because of the pre-loaded questions and answers.

c) Replika: Replika is an AI Chatbot designed on a Mobile App by San Francisco-based AI startup team called Luka in 2017 [28]. It was designed for the purpose of acting

TABLE I: Examples of Chatbots [42]

Name	Year	AI or Rule-Based	Purpose
ALICE	1995	Rule-based	It was designed for entertainment. A user can chat with the Chatbot on any topic.
Brisbot	2016	Rule-Based	Designed to help troubled children with the problems they are experiencing in their day-to-day lives.
Mitsuku	2013	AI	It was designed for entertainment and not designed such as other Psychotherapist Chatbots that can solve users problems.
Replika	2014	AI	It was designed for the purpose of acting as one's best friend or companion, but users must first feed it with information as the App ask questions about the user's life in order to build a library of information.
NBC Politics Bot	2016	Rule-based	It was designed help people find different news that they would want to read.
Casper	2016	AI	Designed to help users to get through the night when they cant sleep..
HandM	2016	Rule-based	Designed to be your personal stylist, it helps you come up with an outfit for the day.
Roof.ai	2016	Rule-based	It assist real estate agents to connect with their customers and provide them with required information.
Domino Pizza	2016	Rule-based	Designed to help customers customise the pizza and order.
UNICEF	2017	Rule-based	Assists people raise their issues especially on the needs of their communities.

as one's best friend or companion, but users must first feed it with information as the App ask questions about the user's life in order to build a library of information that will run through the Neural Network and also for the Chatbot to know the user better [23]. The Chatbot will learn about the user's interests, linguistic syntax and habits over time and will give emotional support to the user without judging them [23]. 30% of the responses that Replika give comes from a script and the remaining 70% comes from a Neural Network [12]. Neural Networks are a variety of deep learning technologies used to process information between inputs and responses which are outputs in parallel [18]. Replika has gained so much popularity and many people find it interesting. It is a remarkable AI Chatbot, however, such as any other invention it has its own pitfalls. The Chatbot only relies on user's words and cannot discern and infer on the information the user provides [12].

E. Terms

The following are terms used in this paper:

1) **Finite Automaton**: Finite Automaton is a computational model of a system [15] and sometimes referred to as a robot [14]. It has finite number of states with edges that transition from one state to another in a sequential manner guided by a discrete input [15].
2) **Breadth-First Search (BFS)**: BFS is a search algorithm that uses a queue that follows a first-In-First-Out (FIFO) system to traverse a graph through level-by-level or layer-by-layer of nodes before going to the depth of the graph [7].
3) **Natural Language Processing (NLP)**: NLP is a computational technique for analysing naturally occurring human-like texts using linguistic analysis with the purpose of achieving human-like language [21].

Despite hundreds of Chatbots being developed lately, it is, however, difficult to find one that is specifically designed for the purpose of navigating through different venues using a Finite Automaton. Students can study at a campus for years until they get their degrees without having been to some other areas of their campus and the justification is that they do not know how to get to those areas as campuses might be big. New students constantly get lost and ask for directions as they maneuver from one lecture class to another or going to dean or admin offices. It also becomes worse when visitors visit the university, they will get lost because of not being familiar with the place. They will either ask for directions, read the directions board, or take a paper-based route map that is difficult to follow.

III. Design and Algorithm

`PathBot` cannot be successfully implemented if the core elements are incorrect. The most important aspect of `PathBot` is finding directions or a path by first identifying the origin and then the destination. Once that is done, it also must select the shortest path available between the origin and the destination. To identify states and paths, `PathBot` uses a Finite Automaton and a BFS algorithm to identify the shortest path.

A. Finite Automaton

The Finite Automaton in Figure 2 was created using Automatic Graph Layout (AGL) which is a Microsoft Research tool built on the principle of the Sugiyama Scheme. It produces hierarchical layout that naturally applies flow of information to graphs [24]. A Finite Automaton or the graph that was designed contains all the information `PathBot` uses to give directions to users. Finite Automaton is a computational model of a system [15] and sometimes referred to as a robot [14]. It has finite number of states with edges that transition from one state to another in a sequential manner guided by a discrete input [15]. Finite Automaton is also defined as

$$M_1 = (Q, \Sigma, q, F, \delta)$$

, where Q is a set of all states, Σ is for inputs, q is the initial state, F is a set of final states, and δ is a transition function from $Q * \Sigma \longrightarrow Q$ [14].

Finite Automaton only accepts regular languages which are either Deterministic Finite Automaton (DFA) or Non-

Deterministic Finite Automaton (NFA) [15]. DFA differs with NFA in a way that if given the current state the next state will always be known, whilst in NFA there could be multiple next states. DFA has only one unique state, has no choices or randomness, and it's also simple and easy to design. Unlike DFA, NFA's state may be chosen at random and all the next states may be chosen in parallel [10]. [14] indicated that in a graph a Finite Automaton or robot has to explore a graph's unlabelled nodes and edges that are labelled at each node.

The automaton transverse all the edges of the graph without any prior knowledge [14]. The Finite Automaton for this study's methodology transition from one state to another guided by the input of the user to be able to output the required results. Finite Automaton stores its vertices and edges in a file system. The vertices file consists of names of states and the edges files consist of directions to use moving from one state to another. To determine the path to be taken by the user, this study adopted BFS method to determine the shortest path.

Figure 2 has states `1`, `2`, `3`, `4`, `9`, and `11`,which are called **Canary Gate**, **Biokinetics**, **Block A**, **Block B**, **UJ Gymnasium**, **Ontdekking**, sequentially. Between the states the are edges that defines a path a user has to take, for instance, between **Ontdekking (`11`)** and **Block A (`3`)** the path to take is $a_{87}.a_{88}.a_{89}$. These symbols have meaning, they list a set of directions that has to be followed from the origin to the destination. There are also other symbols used such as **y**, **d**, **x**, **z**.

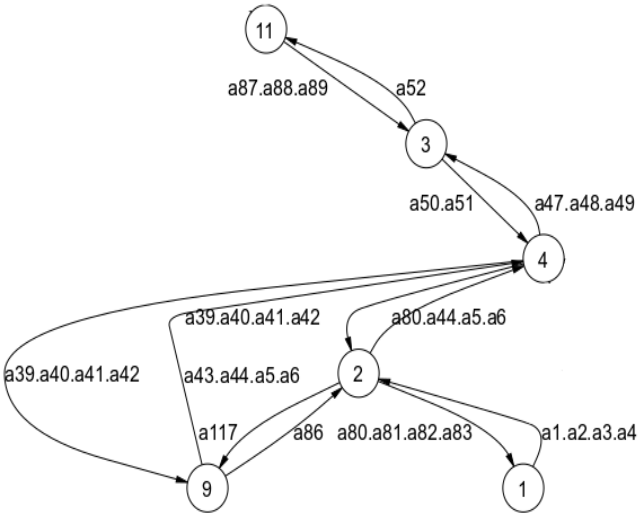


Figure 2. Finite Automaton: https://rise4fun.com/Agl/Ycmp

B. Graph Traversal Method

In order for the user to quickly navigate to their desired destination, there must be an algorithm applied to the Finite Automaton for it to choose the shortest or the quickest path that leads to the destination. The shortest path algorithm adopted by this study is the BFS, which is a search algorithm that uses a queue that follows a FIFO system to traverse a graph through level-by-level or layer-by-layer of nodes before going to the depth of the graph [7]. The exploration of a graph starts at the root node then move to the adjacent nodes or all nodes directly linked to it and proceeds recursively [35]. Algorithm 2 shows the Pseudo-code adopted by this study to traverse the Finite Automaton developed. The input is a Graph and a starting vertex root of the Graph, and the output is the tracing of the shortest path back to the root by the parent.

```
Procedure BFS(G, start_v):
    let Q be a queue
    label start_v as discovered
    Q.enqueue(start_v)
    while Q is not empty
        v = Q.dequeue(
        if v is the goal
            return v
        for all edges from v to m
                in G.adjacentEdges(v) do
            if m is not labeled as
                discovered
                label m as discovered
                m.parent = v
                Q.enqueue(m)
```

Algorithm 2: BFS Algorithm [7]

C. Lookup Table of Symbols

In order to understand the given directions below, one has to understand the symbols indicated on the Finite Automaton and other symbols used for directions. The Finite Automaton on Figure 2 depicts the directions of the following states:

X_1: Bunting Road
X_2: Outside Walk path
X_3: Inside Walk path
X_4: Walk down the stairs
X_5: Walk up the stairs
X_6: Rooms A046 – A048
X_7: Curved Road
X_8: Assembly point I
X_9: Parking area
X_{10}: Room A046
X_{11}: Room A052 – A064
X_{12}: Room A064
X_{13}: Room A126 – Room A134

Symbols such as **y**, **d**, **z** have their own meaning. **y** indicate landmarks that you can come across in the campus, **d** and **q** indicate egocentric coordinates, and **z** indicates different entrances.

Y_1: Board written 1 Bunting Road, OFFICE OF THE Campus Director: Auckland Pack (Kingsway & Bunting road)
Y_2: Netcare Education Board
Y_3: White building written Biokinetics & Aquatic Centre with silver letters
Y_4: Block B written on an orange board on a building
Y_5: Absa Auditorium written on a board
Y_6: College of Business and Economics building

Y_7: Library & Information Centre written on an orange board on a building
Y_8: Gentlemen Bathrooms
Y_9: Ladies Bathrooms
Y_{10}: Biokinetics Parking
Y_{11}: Gymnasium board written on a dark brown face brick building
Y_{12}: Stop Sign
Y_{13}: Block A cashiers written on an orange board on a building
Y_{14}: Ontdekking building written with silver letters on a building
Y_{15}: Block K written on an orange board on a building
Y_{16}: Student Centre written on an orange board on a building
Y_{17}: Waste/dumping area
Y_{18}: School of Tourism and Hospitality written with silver letters on top on a building
Y_{19}: Campus Health Services written on an orange board on a building
Y_{20}: Impala Court written on an orange board on a building
Y_{21}: Centre for Psychological Services and Career Development (PsyCaD) written on an orange board on a building

Y_{22}: Block C board on a wall
Y_{23}: Technopreneurship centre room
Y_{24}: Neutral bathrooms
Y_{25}: Lecture hall B20
Y_{26}: Lecture hall B21

d_1: left
d_2: straight
d_3: right
d_4: go inside or enter Block B
d_5: go outside or exit
d_6: far left
d_7: far right

q_1: First turn
q_2: Second turn
q_3: First right
q_4: second right

z_1: Main Entrance
z_2: Right door
z_3: Entrance 2

D. Finite Automaton Directions

Using the symbols discussed above, you can traverse Figure 3 and 4 with the example of directions below:

1) ***From Block B (4) to UJ Gymnasium (9):***
a_{39} : From `4` z_1 and turn d_3,
a_{40} : Continue walking d_2 on x_7,
a_{41} : Pass q_1 on your right,
a_{42} : Turn d_1 when you see y_{10} and y_{11} on your left.

2) ***From UJ Gymnasium (9) to Block B (4):***
a_{43} : From `9` Walk to y_{10},
a_{44} : Turn d_3,
a_{45} : Continue d_2 and pass y_{12} and q_1 on your left,
a_{46} : Continue d_2 until you see y_4,

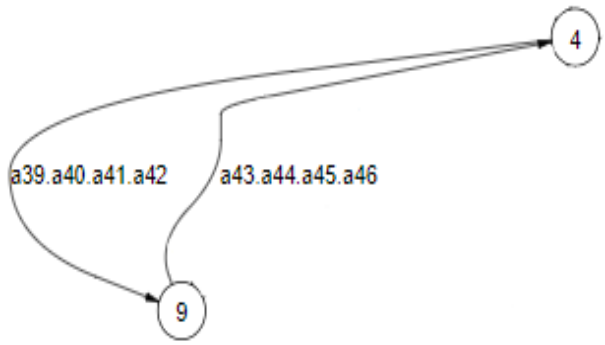


Figure 3. Block B to Library and Return.

3) ***From Ontdekking (11) to Block A (3):***
a_{87}**:** From **`11`** x_5,
a_{88}**:** Turn d_3,
a_{89}**:** Continue d_2 until you see y_{13}.

4) ***From Block A (3) to Ontdekking (11):***
a_{52}**:** From **`3`** Continue d_2 until you see y_{14} and y_{15} on your left.

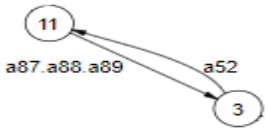


Figure 4. Block A to Ontdekking and Return.

Remark (The Symmetry of Finite Automaton)**.** *The developed Finite Automaton is not symmetrical, meaning the directions from point A to point B will not be the same as the directions from point B to point A. For instance in Figure 2, if you want to go to Block B (`4`), you cannot use the path from Ontdekking (`11`) to Block A (`4`) then to Block B (`4`), as the directions are customised for only two states that connects to one another.*

IV. Implementation

`PathBot` was created using React-Native and Dialogflow, for both iOS and Android. There is a list of predefined campus locations saved in Dialogflow as entities. In Dialogflow an entity is a list of categorised data that will be identified and extracted as useful data from the natural language input. For example there is `campus_locations` as an entity type and Block C stored as one of the entity entries, so if natural language is provided as an input containing Block C, then Block C will be identified and extracted as a parameter. To improve the agent's ability to effectively respond to different end user requests, what is referred to as an intent was added in a Dialogflow. Intents categorise an end user's intentions for one conversational turn.

For each intent few training phrases that the user might say were added and then Dialogflow expands on the list. One or more predefined responses to the intent were also added.

The two most important intents are the ones used to extract the user's destination and origin. One intent is setup to handle all end user's requests to get to certain destinations on the campus, from which destination is extracted. The other intent is setup to handle all requests related to a user trying to provide their current location, from which the origin is extracted. Other intents are setup to handle small talk, unknown locations and things that might be said out of context.

On the App an API service has been setup to communicate with Dialogflow. To initiate the greeting sequence, a message "Hello" is sent in the background to Dialogflow and from then the conversation starts. Payload data from Dialogflow is used to pull the information needed to help find the right directions for the user. From the payload,the intent type with some parameters(if any) is received in which Dialogflow may have identified as useful data. From the DESTINATION intent you find the destination and from the ORIGIN intent you find the user's current location.

Once the user's destination and origin is received, the shortest path is found using BFS, which may return a string of connected edges or null for a non existent path. When the path does not exist the agent is notified which in turn responds to the user with the message alerting them of the situation. If `PathBot` receives a string of connected edges back from the BFS algorithm, then it will process the string through a parser which is supposed to construct a list of directions in the order that the edges are connected.

The examples below show the directions with the example of iOS screens. The screen will be the same for Android. When you open `PathBot`, you will get the message "Hello! How can I help you?", refer to top message in Figure 5 and Figure 7. The user will have to reply with their destination, and the origin will be requested by `PathBot`, then return the directions. Once the user clicks on the message "Click here, and follow these directions to Block B", then the screen with a list of directions will appear, such as in Figure 6 and Figure 8.

A. *Example 1:**From Block A (3) to Block B (4)***

a_{50}**:** From **3** turn d_3,
a_{51}**:** Continue d_2 until you see y_4 on your right.

B. *Example 2:**From Block B (4) to Block A (3)***

a_{47}**:** From **4** z_1 facing **9**, turn d_1,
a_{48}**:** Continue d_2,
a_{49}**:** You will see y_{13} on your left, then immediately after passing q_1 turn on your left.

V. Evaluation

This section represents the outcome of the evaluation done with UJ students on their experience in using `PathBot`. `PathBot`

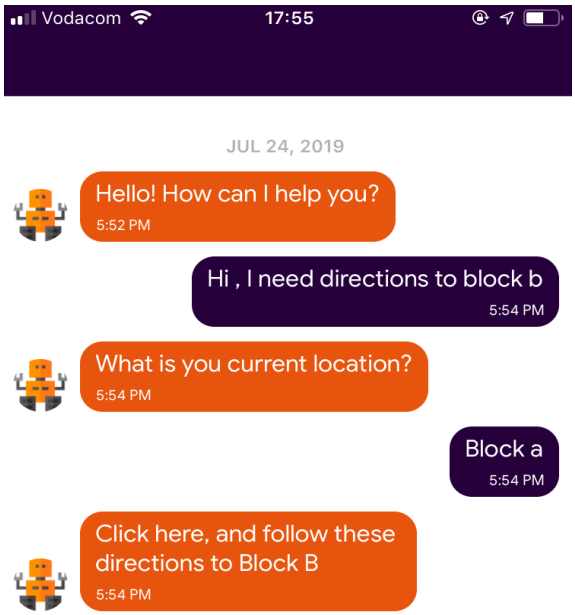


Figure 5. Block A to Block B chat screen.

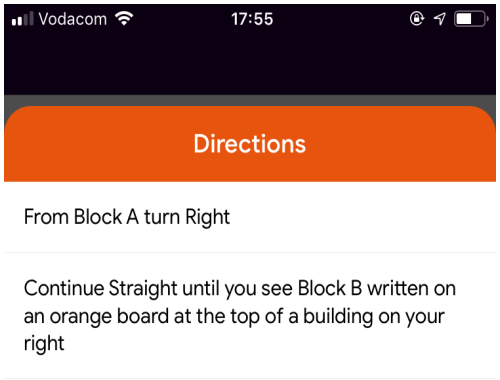


Figure 6. Block A to Block B directions screen.

was assessed and an online survey was conducted with 10 students studying Information Systems. The students were asked if `PathBot` is easy to use and if they could consider it as a useful tool to use every day at the campus and the results are as shown on Figure 9.

Navigating around the campus has always been a challenge for many students, and 90% of the students who did the evaluation attested to this. From the evaluation only 10% indicated that they are familiar with all areas at the Bunting Road campus, however, the results further show that all the students have needed assistance to navigate around the campus before. All students indicated that they have used a Chatbot before, therefore, it made it extremely easy for 50% of students to use `PathBot`, whereas 30% said it was average and 20% said it was okay. For usefulness, all students indicated that `PathBot` is a useful tool and can assist them when they are

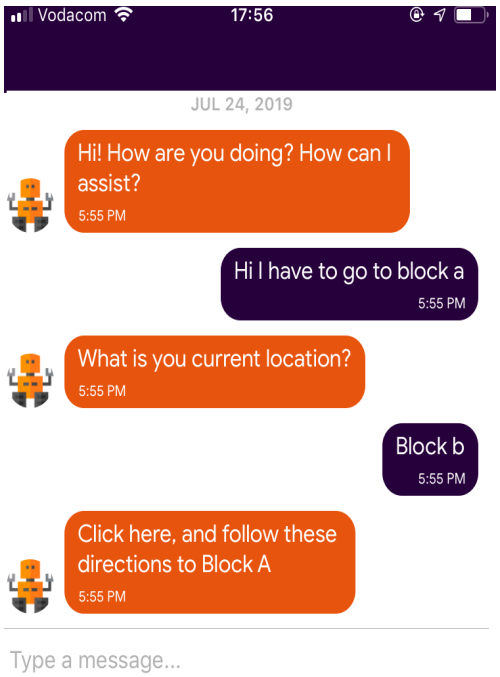


Figure 7. Block B to Block A chat screen.

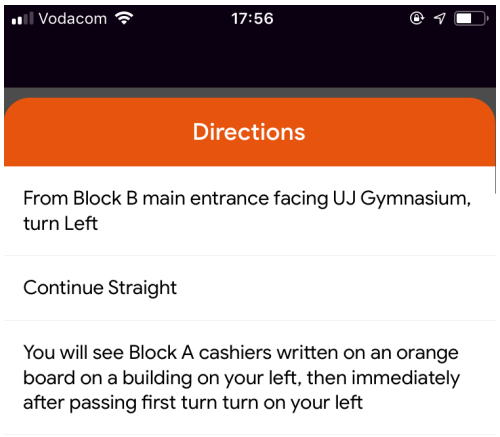


Figure 8. Block B to Block A directions screen.

looking for venues around the campus. 30% of students rated the accuracy of the directions given by the PathBot to be exceptionally good, 30% said it is okay, another 30% gave it an average score, whereas 10% said it is poor.

In the comment section, the evaluation shows a high need for a visual reference of the map of Bunting Road campus and for the PathBot to give the predictive location as the user might not know where they are. Other students suggested that PathBot should have audio with a guiding instructor and include more location than what it currently covers.

VI. Conclusion and Future Work

A. Conclusion

This paper presented the development of PathBot as a tool to assist students and visitors to navigate through the campus venues. To ensure accuracy of PathBot, a Finite Automaton which has a number of states and transitions was developed. The states are selected areas that serve as origin or destination and transitions are paths you can take when navigating from one state to another. BFS algorithm was used on PathBot to generate the shortest path for the user. PathBot was also evaluated with a number of students which showed a 90% agreement on the usability and 80% agreement on usefulness.

B. Future Work

Currently PathBot returns a list of direction all at once. It does not handle rerouting and cases where the user is lost. In future, a visual map will be added for user friendliness and the app will re-route as the user gets lost. More locations or areas will be added for PathBot to cover more areas. Interviews will also be performed with more users to test the usability, usefulness, and accuracy of PathBot.

References

[1] Sameera A Abdul-Kader and JC Woods. Survey on chatbot design techniques in speech conversation systems. *International Journal of Advanced Computer Science and Applications*, 6(7), 2015.

[2] Bayan AbuShawar and Eric Atwell. Alice chatbot: Trials and outputs. *Computación y Sistemas*, 19(4):625–632, 2015.

[3] Nahdatul Akma Ahmad, Mohamad Hafiz Che, Azaliza Zainal, Muhammad Fairuz Abd Rauf, and Zuraidy Adnan. Review of chatbots design techniques. *International Journal of Computer Applications*, 181(8):7–10, 2018.

[4] Alice. A.l.i.c.e ai foundation. 2002. url - https://home.pandorabots.com/home.html.

[5] Aggeliki Androutsopoulou, Nikos Karacapilidis, Euripidis Loukis, and Yannis Charalabidis. Transforming the communication between citizens and government through ai-guided chatbots. *Government Information Quarterly*, 36(2):358–367, 2019.

[6] Petter Bae Brandtzaeg and Asbjørn Følstad. Why people use chatbots. In *International Conference on Internet Science*, pages 377–392. Springer, 2017.

[7] Aydin Buluç and Kamesh Madduri. Parallel breadth-first search on distributed memory systems. In *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*, page 65. ACM, 2011.

[8] BusinessInsider. 80% of business want chatbots by 2020. 2016. url - http://www.businessinsider.com/80-of-businesses-want-chatbots-by-2020-2016-12.

[9] H.K. Cho and D.H. Lee. A chatbot-based robot control architecture for conversational human-robot interactions. *HRI 2018 Workshop on Social Human-Robot Interaction of Human-Care Service Robots*, 2018.

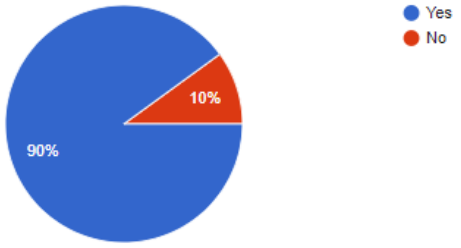
[10] Daniel IA Cohen and Daniel IA Cohen. *Introduction to computer theory*, volume 2. Wiley New York, 1991.

[11] Kenneth M Colby. Simulations of belief systems. *Computer models of thought and language*, pages 251–286, 1973.

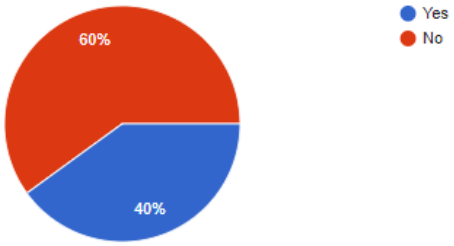
[12] Forbes. Replika chatbot. 2018. url - https://www.forbes.com/sites/parmyolson/2018/03/08/replika-chatbot-google-machine-learning/2a3106d74ffa.

[13] L Foye. Chatbot conversations to deliver $8 billion in cost savings by 2022. juniper research. analyst xpress, 2018.

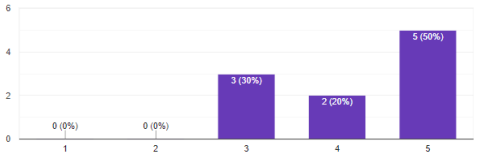
[14] Pierre Fraigniaud, David Ilcinkas, Guy Peer, Andrzej Pelc, and David Peleg. Graph exploration by a finite automaton. *Theoretical Computer Science*, 345(2-3):331–344, 2005.



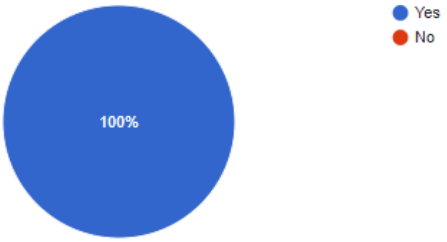
(a) Do you find it difficult navigating around the Bunting Road campus?



(b) Have you seen or used a Chatbot that works like PathBot before?



(c) How easy is it to use PathBot?



(d) Do you consider PathBot as a useful tool?

Figure 9. Evaluation Results.

[15] William B. Frakes and Ricardo Baeza-Yates, editors. *Information Retrieval: Data Structures and Algorithms*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1992.

[16] Gartner. Gartner says 25 percent of customer service operations will use virtual customer assistance by 2020. 2018. url - https://www.gartner.com/newsroom/id/3858564.

[17] Gartner. Gartner top strategic predictions for 2018 and beyond. 2018. url - https://www.gartner.com/smarterwithgartner/gartner-top-strategic-predictions-for-2018-and-beyond/.

[18] John J Hopfield. Artificial neural networks. *IEEE Circuits and Devices Magazine*, 4(5):3–10, 1988.

[19] Kinetic. Find my class. 2019. url - http://www.schmidtyapps.com/.

[20] Cheongjae Lee, Sangkeun Jung, Seokhwan Kim, and Gary Geunbae Lee. Example-based dialog modeling for practical multi-domain dialog system. *Speech Communication*, 51(5):466–484, 2009.

[21] Elizabeth D Liddy. Natural language processing. 2001.

[22] Ju Long, Juntao Yuan, and Hsun-Ming Lee. How to program a chatbot–an introductory project and student perceptions. *Issues in Informing Science and Information Technology*, 16:001–031, 2019.

[23] Martino Mensio, Giuseppe Rizzo, and Maurizio Morisio. The rise of emotion-aware conversational agents: threats in digital emotions. In *Companion Proceedings of the The Web Conference 2018*, pages 1541–1544. International World Wide Web Conferences Steering Committee, 2018.

[24] Microsoft. Microsoft automatic graph layout. url - https://www.microsoft.com/en-us/research/project/microsoft-automatic-graph-layout/.

[25] Juho Miettinen et al. Utilization of chatbots in customer interface. 2018.

[26] James H Moor. The status and future of the turing test. *Minds and Machines*, 11(1):77–93, 2001.

[27] Gustavo Marques Mostaço, Ícaro Ramires Costa De Souza, Leonardo Barreto Campos, and Carlos Eduardo Cugnasca. Agronomobot: a smart answering chatbot applied to agricultural sensor networks. In *14th international conference on precision agriculture*, volume 24, pages 1–13, 2018.

[28] M. Murphy and J. Tamplin. Machines with brains. 2017. url - https://classic.qz.com/machines-with-brains/1018126/lukas-replika-chatbot-creates-a-digital-representation-of-you-the-more-you-interact-with-it/.

[29] Quynh N Nguyen and Anna Sidorova. Understanding user interactions with a chatbot: a self-determination theory approach. 2018.

[30] University of Deakin. Deakin scout. url - https://www.deakin.edu.au/life-at-deakin/why-study-at-deakin/deakin-scout.

[31] Shikhar Prateek Pandey. An anonymous virtual friend: Chatbot. *Journal Current Science*, 20(04), 2019.

[32] Mina Park, Milam Aiken, Laura Salvador, et al. How do humans interact with chatbots?: An analysis of transcripts. *International Journal Of Management & Information Technology*, 14:3338–3350, 2018.

[33] Sam Ransbotham, David Kiron, Philipp Gerbert, and Martin Reeves. Reshaping business with artificial intelligence: Closing the gap between ambition and action. *MIT Sloan Management Review*, 59(1), 2017.

[34] C. Rishabh and J. Anuradha. Counsellor chatbot. volume V, pages 126–136. IRJCS: International Research Journal of Computer Science, 2018.

[35] Stuart J Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,, 2016.

[36] Bayan Abu Shawar and Eric Atwell. *A comparison between Alice and Elizabeth chatbot systems*. University of Leeds, School of Computing research report 2002.19, 2002.

[37] Peter Sydenham. *Handbook of Measuring System Design*. 07 2005.

[38] Andrew Thompson, Andrew Gallacher, and Mark Howarth. Stimulating task interest: human partners or chatbots? *Future-proof CALL: language learning as exploration and encounters–short papers from EUROCALL 2018*, page 302, 2018.

[39] Tutorialspoint. Aiml introduction. url - https://www.tutorialspoint.com/aiml/.

[40] Richard Wallace, Hideto Tomabechi, and Doubly Aimless. Chatterbots go native: Considerations for an eco-system fostering the development of artificial life forms in a human world. *Published online: http://www. pandorabots. com/pandora/pics/chatterbotsgonative. doc*, 2003.

[41] Joseph Weizenbaum et al. Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45, 1966.

[42] Wordstream. 10 of the most innovative chatbots on the web. 2019. url-https://www.wordstream.com/blog/ws/2017/10/04/chatbots.