The artificial facilitator: guiding participants in developing causal maps using voice-activated personal assistant

by

Thrishma Reddy Shivarama Reddy
Lakehead University
Supervisor: Dr. Vijay Mago
A Thesis Presented to Lakehead University
in Partial Fulfillment of the Requirement for the Degree of
Masters of Science in Computer Science

Faculty of Science and Environmental Studies
Lakehead University

All rights reserved. This thesis may not be reproduced in whole or in part, by
photocopying or other means, without the permission of the author.

ABSTRACT

When it comes to any problem, their causes, and solutions, people often have very different perspectives, primarily due to the environment in which they were raised (culture, education, socio-economic status, and so forth). Complex problems often require coordinated actions from all stakeholders to achieve a resolution. Agreeing on the same course of action can sometimes be difficult, as the stakeholders might have a different perspective of the specific problem. Causal map is a way to capture different perspectives people have about any situation. Thus, we posed the following research question - is it possible to use conversational artificial intelligence to capture and store the thought process of a particular problem? In this research, we have conducted an experiment which consisted of two parts: 1) developing a model for a voice-activated personal assistant that interacts, captures, and converts the responses of the participant into causal maps and 2) a detailed pre-test and post-test questionnaire that focuses on assessing interactions and willingness of the participants to collaborate with the developed model. We were able to build an Alexa *skill* that could successfully capture participants thought process and transform it into a causal map that could be analyzed along with data from other participants. The results of our pre-test and post-test surveys conducted with ten researchers who participated showed that they rated the Alexa skill as a useful tool for capturing the thought process of a problem. In our view, understanding the human thought process is crucial for stakeholders to agree on the same course of resolution. The research concludes with a discussion of future uses and limitations.

Contents

List of Tables

List of Figures

ACKNOWLEDGEMENTS

To my supervisor, Dr. Vijay Mago, I would like to express my sincere appreciation and deep gratitude for his enthusiasm for research, effective guidance and continuous encouragement and support throughout this thesis. Vijay is one of the most intelligent professors I know. He played a key role in pushing me to apply for the MITACS Globalink Research Award and helped me throughout the process and was always available to guide me when I needed assistance. Working under him for the past year has been an absolute pleasure.

I would like to thank MITACS Globalink Research Award for financially supporting this research conducted at Furman University, USA.

I would like to thank Dr. Philippe J. Giabbanelli for hosting me at Furman University and guiding me throughout my research.

Finally, my sincere gratitude goes to my family and friends for their warm support and encouragement during my graduate studies.

I would like to dedicate this thesis to my parents, for always being my constant support in life.

Chapter 1

Background and Motivation

1.1 Introduction

Problems come in all forms, easy or difficult. The difficult ones such as ecological management or obesity are tedious to work with and are often labelled as *complex* problems. While the complexity sciences provide many definitions and tools to measure complexity[1], complex problems often share at least two traits which are central to this research. First, they are *multifactorial*. The traditional reductionist approach that attempts to fix the 'root' cause does not lend itself well to

[1]For an overview of the complexity sciences, see the map at `http://www.art-sciencefactory.com/complexity-map_feb09.html`

a complex problem [1], and may even cause harm through unintended consequences [2]. Instead, the importance is often on mapping [3] and navigating [4] the *complex system of interactions* between factors that contribute to a problem of interest and factors affected by it. Second, dissemination and implementation research emphasizes that solutions to complex problems often require *coordinated actions between stakeholders* from multiple sectors (i.e., a multi-actor view [5]). For instance, actions regarding population and obesity involve sectors as varied as food production, city infrastructure (e.g., to promote walkable cities and access to fresh food), mental and physical well-being of people [6] and so on. Coordinated actions should produce a coherent policy, which implies that stakeholders work together at least by sharing a mission [7].

Whether stakeholders share a mission when operating in a complex interaction system can be challenging. They may have different views or 'mental models' on how the factors interact, which may lead to very different perspectives on interventions. In the case of ecological management, one stakeholder may ignore the pressure of fishing and instead focus on the environment (e.g., enough nutrition for the fish, not too many predatory birds) while another may acknowledge that fishing reduces the fish population but downplay its importance [8]. Stakeholders may also have the same views but express them differently, for example by naming factors in different ways depending on their fields, which can create a communication gap [9, 10]. Consequently, complex problems involving multiple stakeholders often involve *participatory modeling*, which allows to externalize [11] and hence compare [8] the mental models of stakeholders. There are various approaches to participatory modeling, depending on whether the objective is to be able to *simulate* a system [13, 12] (e.g., to quantitatively assess *how much* effect an intervention would have) or to only capture its *structure* [14] (e.g., to qualitatively assess *what* an intervention would affect). In the example of obesity, qualitative approaches may be realized by systems dynamics or agent-based modelling [15] and generate 'systems maps' or 'diagrams' [16]. The creation of systems maps is particularly important either as an endpoint (for qualitative analysis of stakeholders' mental models) or as a step toward the creation of quantitative models [14] (e.g., starting with a

Causal Loop Diagram to produce a Systems Dynamics model). Causal maps are a widely used form of systems maps, in which concepts are represented as nodes, and their causal connections are captured through directed edges. In Figure.1.1 the core concept node is highlighted in red and the rest are nodes connected to it through causal connections.

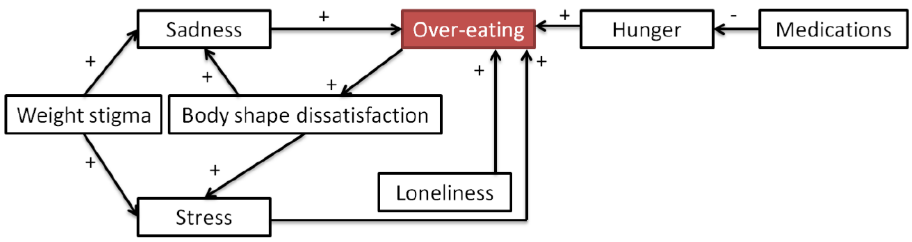

Figure 1.1: Sample causal map where *over-eating* is the problem of interest [17].

1.2 Background

1.2.1 Understanding how a human's brain reasons about a problem

Creating explanations of the things human's learn is an essential aspect of them. These explanations, in turn, depend on their understanding. This understanding is gained by individual experiences which are acquired through induction or deduction of a topic. Induction is drawing a conclusion based on previous experience [18]. Whereas a deduction is drawing a conclusion based on the models we have created in our minds [19] based on our experience.

If a person is unable to understand a topic by both induction or deduction process, then they are unable to explain it. If a person knows what causes something, how to stop it, why it exists, its internal structure, its defects, its relation to another topic, how to predict its occurrence in the future, why it happened, then to some degree the person can understand the topic [20]. According to *model theory* if a person understands the working of anything, for example — the inner workings of a laptop, digestion, marriage or death, then they have a created a mental model representation of their workings in their minds. These models created in one's head will then be used to formulate an explanation when asked why do you think a problem is caused or why does a particular thing work the way it works.

By constructing abstract models or systems that abridge and summarize their features [21], we understand the different characteristics of everyday life. We tend to call these models or systems as ideas in the purest form. For instance, our abstract concept of a bird is a model or system for thinking about actual birds to make sense of their behavior— as opposed to, say, the behaviour of cats, dogs, tortoises, beetles, and people. In short, our concepts provide our minds with systems for experiencing and thinking; our minds operate (reason) within them to investigate the world we experience with implications and consequences that are rich in meanings. Of course, a lot of this is done entirely automatically and subconsciously.

If humans are put in any situation, we start to give it meaning, try to figure it out with the logical structures we have at our disposal. So we make deductions/inductions quickly and automatically and as a result of how we shape the situation in our minds — that we don't typically notice these inferences. For example, we see dark clouds, for instance, and infer the chances of rain. We hear the slam of the door and assume that somebody has come. When we see a face that is frowning, we assume that the person is angry. If our friend is late, we conclude that she is inconsiderate. There are also subconscious biases that are included when deducing or inducing. For example when we meet a tall boy, and we infer that he's good at basketball or have assumptions based on ethnicity. We also listen to what people are saying and make an ongoing series of inferences about what they mean. Without these logical structures presented above, that draw our assumptions, we cannot explain things.

Many of our deductions or inductions are reasonable and justified [22], but of course, many are not. One of the most important critical thinking skills is the ability to notice and reconstruct the inferences we make and hence our experiences. Eventually, we realize that our point of view and the assumptions we make strongly influence the inferences we make. For this research, one thing to keep in mind is that many people have different opinions because they bring a distinct point of view to situations [22]. For example, if two people see a man lying on a footpath, one might infer, *There is a drunken person.* The other might infer, *There is a man in need of help.* These inferences are based on different assumptions about the conditions under which people end up on the streets, and these assumptions are connected to the point of view about people that each has formed. The first person assumes: "Only drunks are to be found on footpaths." The second person assumes: "People lying on the footpath need help." The first person may have developed the point of view that people are fundamentally responsible for what happens to them and ought to be able to take care of themselves. The second may have taken the view that people's problems are often caused by forces and events beyond their control. The two are persons mentioned above are modelling the situation differently, and therefore they are using a different system for experiencing it.

1.2.2 Identification of the root cause of a problem

During the formation of policy, either in a company or the government, solutions to problems are shaped and debated. These problems can vary from *climate change* to *obesity.* Instead of looking at its symptoms, it is essential to get to the root cause of the problem; because if not handled, the root cause will likely reoccur and make it challenging to handle. Hence to address this problem the policy must emphasize on finding the root cause of the problem first.

The root cause can be determined by finding the causal factors of a problem. Causal factors can be determined by asking "why" to the problem statement. For example: if we consider the problem statement — "why does poverty exist?" [23], Some of the causal factors can be — 1) Inadequate access to clean water and nutritious food. This is because if an individual doesn't get enough sustenance, they necessarily don't have the quality and vitality expected to work, while the lack of access to nourishment and clean water can likewise prompt preventable sicknesses like diarrhea. This leads to them spending the little money they have on medications which will knock them into extreme poverty. Regardless of whether clean water sources are accessible, they've frequently situated very far from where the families are located which leads to mothers and kids travelling a long way just for water when this time could have been utilized in getting an education to help secure an occupation further down the road [23]. 2) No access to jobs. Without a job or a way to make money, a person will fall into poverty. In any case, it's anything but difficult to accept that on the off chance that somebody needs an occupation, they could have one. That simply isn't valid, especially in developing and rural parts of the world [23]. 3) Poor education. Most of the people in poverty don't have an education. This is because families can't afford to send their kids to school because they need them to work to earn money to meet their daily family needs [23].

Once the causal factors are recognized the next step is to ask "why" questions again to the above-identified causal factors. Such as "why are there no jobs?" or "why is there inadequate access to clean water and nutritious food". The process of asking "why" needs to be continued until all the responses have been exhausted. You can refer to Figure 1.2 for a clear understanding.

This process requires good judgment and will go through a lot of trial and error. For example, if a causal factor has been identified as the root cause and was solved but even, so the problem continued to exist, then it is a clear indication that it is not the root cause and one needs to dig deeper to go beyond the causal factors. In the root causes for *poverty* can be identified as *no money with the government, drought, no education, and no jobs.* There are many ways to fetch the root cause of a problem, but for this research, we focus on finding the root causes using Why-tree process.

1.2.3 Why do we create causal maps?

A causal map is a *conceptual model*. In Modeling and Simulation (M&S), conceptual models are the first stage of model development before quantifying nodes and relationships (mathematical model [24])

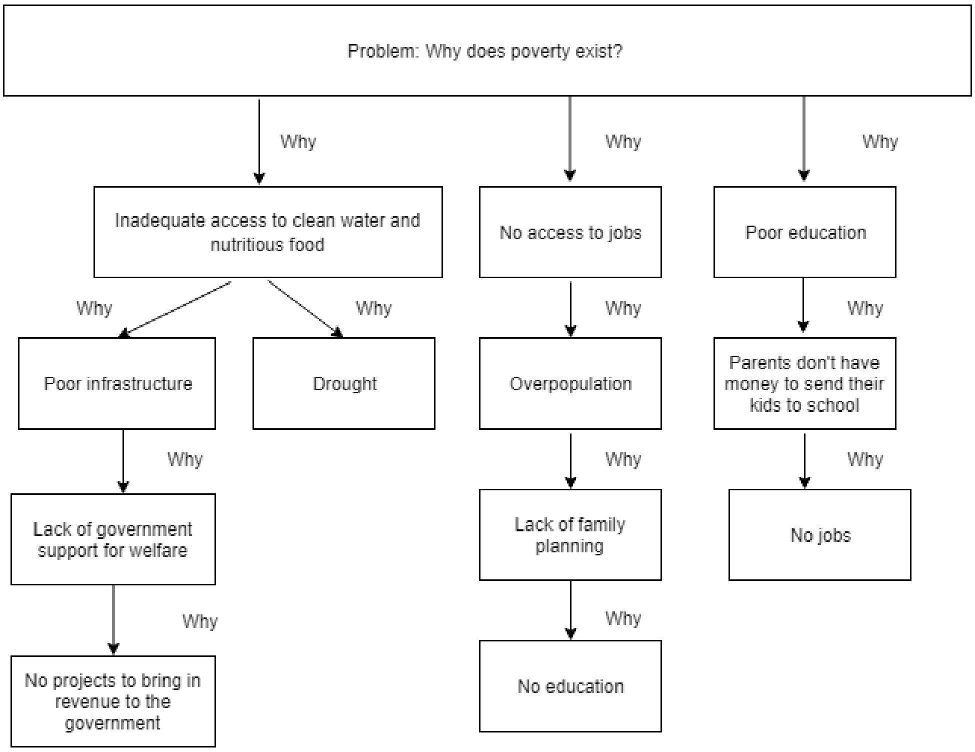

Figure 1.2: Using the Why-tree process to uncover the underlying reasons for poverty

and possibly implementing the model as code (computational model). Conceptual models serve multiple objectives such as identifying key elements and aspects (thus delineating the boundaries of a system) or externalizing hypotheses through a transparent list of expected relations [14]. These objectives may be sufficient to warrant the development of a conceptual model as a final product. In this case, the conceptual map is often analyzed using network theory[2]. A common type of analysis is the identification of clusters or communities to divide a complex system into broader themes, as exemplified by the Foresight Obesity Map [26, 27], maps for the Provincial Health Services Authority [28], or the recent work of Allender, McGlashan, and colleagues [30, 29]. Other analyses may include the centrality, to identify leverage points in a system [32, 31]; an inventory of loops, to better characterize and possibly change the dynamics of the system [34, 35, 33]; an exploration

[2]A conceptual model is an aggregate model in which *factors* or *concepts* are connected. This is different from a 'social network,' which is an individual model in which nodes represent individuals rather than factors. Although the methods are often similar (e.g., centrality, community detection), the application of network science to social networks is often presented as 'social network analysis' [25].

of disjoint paths between factors, to capture how a policy impacts an outcome in multiple ways [4, 35]; or a comparison of maps, to understand how different the mental models of participants are [36, 10].

Map-liked artifacts may be constructed solely from data, for instance as Structured Equation Models (SEM) or Fuzzy Cognitive Maps (FCMs) [37]. Alternatively, traces produced by an analyst in exploring the data can be structured in a map [39, 38], or the literature on a topic can be synthesized into a map [40]. It would be overly reductive to categorize such data-driven maps as 'objective' when compared to participant-driven maps being deemed 'subjective.' Data can also have "biases, ambiguities, and inaccuracies" and the inference process to build a map may not be perfect. In this research, the focus is on participatory modelling (PM), in which participants drive the development of causal maps. Participatory modelling serves a different (and sometimes complementary) purpose than data-driven modelling. As detailed elsewhere [17], data-driven modelling may strive for accuracy concerning the data whereas PM aims to be transparent and representative of the participants' mental models. PM can thus be employed in 'soft' situations that lack data and rely on human expertise [41], to support decision-making processes [42], or to understand what actions would be acceptable to various stakeholders [43].

The elicitation process consists of externalizing the mental model of a participant or group into a map. The elicitation process is first and foremost a *facilitation* process: we would like to support participants in expressing their views, rather than judging whether what they think is 'right' given our ideas. Research in cognitive sciences has long been concerned with how humans store mental models, or their "conceptualization of the world" [44]. This storage takes place in *semantic memory*, which provides functional *relationships* between objects. As we previously summarized, "if mental models are published and shared in the form of maps, it is owed to the fact that we seek to capture semantic memory whose structure is network-based" [8]. On one extreme, freeform approaches such as Rich Pictures pose no constraints on the creation of maps [45], which simplifies the process for participants but limits the analytical possibilities. At the other extremes, concept maps and mind maps have a very structured process that lists concepts (e.g., via brainstorming), groups them, links them and labels the links. However, this process precludes the presence of some structures (e.g., mind maps are trees so they cannot contain cycles) which are important to characterize the dynamics of a system. Causal maps occupy an intermediate position: the development process is more guided than rich pictures, less restricted than concept maps and mind maps, and any network structure can be produced by participants[3].

1.2.4 How do we create good causal maps?

The process to produce a map as shown in Figure 1.1 is relatively simple: participants create concept nodes and link them by indicating the causal relationship to be an increase ('+') or a decrease ('-') [49, 48]. However, at least three issues may arise if the facilitator does not provide further guidance[4]. First, participants need to choose node labels that have an unambiguous quantification: having 'more' or 'less' of this concept should be a straightforward notion. For instance, labelling a concept as 'weather' does not work, since having more or less weather is undefined. However, having more or less rain would be defined. A facilitator thus regularly ensures that labels are quantifiable, or prompts for clarifications that would change the label. Second, users may forget about concepts that they already have, and add one with a similar name. Facilitators thus continuously monitor the maps to either avoid creating a redundant concept or merge them once they are discovered. Given the tremendous potential for (subtle) variations in language, discovering similar concepts is a difficult problem, particularly as the number of concepts increases [10, 9]. Third, case studies have shown that cognitive limitations make

[3] There are at least two limitations to this representation. First, networks or graphs only represent binary relationships. However, participants may think of non-binary relationships, for instance when three concepts are directly involved together. While we have long been aware that cognitive structures could generally be represented by relations between any number of concepts (e.g. using a hypergraph), it has been common practice to limit the structure to a graph [46]. Second, the network is only used to represent what is true (i.e. the existence of a causal connection between two factors) rather than what is false. As noted by Johnson-Laird, mental models also include counterexamples, which are important in decision-making processes [47].

[4] Some of these issues are also addressed in our tutorials at `https://www.youtube.com/watch?v=OdKJW8tNDcM` and `https://www.youtube.com/watch?v=D-2Q2IHclo4`.

it difficult for participants to think of structures such as loops and disjoint paths [51, 50]. In particular, Ross observed how peculiar it was that "those who set policy think only acyclically, especially since the cyclical nature of causal chains in the real world has been amply demonstrated" [52]. Without paying particular attention to loops, participants may produce star diagrams with the one central problem at the core, and every other factor directly connecting to it. Facilitators may thus prompt participants extensively for relationships, to minimize the risk of missing loops or additional paths [28, 35].

1.2.5 Smart conversational agents

The term 'conversational agent' may be used loosely for any system that can carry on a conversation with a human. However, there are significant differences across systems. Unlike chat-bots, smart conversational agents are not limited to performing simple conversations. And unlike embodied conversational agents, they do not provide computer-generated characters to mimic the movements or facial expressions of a virtual interlocutor. Smart conversational agents are at the confluence of speech processing, Natural Language Processing (NLP), and artificial intelligence (AI). As detailed by Williams and colleagues [53], voice-activated devices such as Amazon's Alexa or Apple's Siri start by converting what a user said (i.e., an audio utterance) into text using automatic speech recognition. Words are then processed through Spoken Language Understanding (SLU) and passed onto a Dialog State Tracker (DST), which results in identifying an appropriate response. The words in the response are prepared by Natural Language Generation (NLG) and turned into audio by Text-to-Speech (TTS).

Smart conversational agents can be designed in many ways, as shown in the recent review by Laranjo *et al.* applied to health care [54]. A conversation may not be oriented toward the completion of a specific *task*, but takes place for its own sake. The flow of the discussion may be *controlled* by the system and the user. *Interactions* can be via spoken language and written language. Finally, the *dialogue management* may take the user through a sequence of pre-determined steps (i.e., a finite-state system), elicit input and parse it using a template to decide the dialogue-flow (i.e., a frame-based system), or take an agent-based approach to focus on beliefs and desires. In the specific health care context reviewed by Laranjo *et al.*, agent-based approaches were uncommon (1 study) while finite (6 studies), and frame-based systems (7 studies) were equally common [54]. However, when interactions rely on voice and a task has to be accomplished, then the frame-based design is so common that the system may be presented as a *slot-based dialogue system* [55].

1.2.6 Why Google Natural Language API?

Research [56] detailing a hybrid approach to Named Entity Recognition(NER), a combination of rule-based approach and machine learning techniques such as Conditional Random Fields(CRF), has been conducted previously. It uses a dictionary containing 20,000 words from the Telugu language. By going through various articles, newspapers, and Telugu grammar books, the rules for Noun identification are framed. Using this dictionary, nouns, some suffix and prefix are identified using a rule-based approach. In later stage CRF technique was used to improve the system's accuracy. In the paper [57] in order to identify nested entities, a neural model is proposed by dynamically stacking flat NER layers without relying on any external resources or linguistic features. Due to a sequence of words, the model uses a low-dimensional vector concatenated from its corresponding embedded word and character sequence to represent each word first [57]. The flat NER layer allows the capture of context representation by a long-term memory(LSTM) layer, taking the sequence of the word representation as input. In one of the research [58] it has been addressed that building dictionaries requires a great deal of human effort, and many types of named entities often find it difficult to obtain good coverage. The process is also very costly. This paper [58] describes an approach to the automatic construction of Named Entity Recognition(NER) dictionaries using large amounts of unlabeled data and a few seed examples to overcome this problem. First, for each entity type named, a high-recall, low-precision list of candidate phrases is collected from the large unlabeled data collection using simple rules. In the second step, by removing the noisy candidates from the list obtained in the first step, an accurate dictionary of named entities is constructed. This is done by learning a classifier using the lower-dimensional, real-evaluated Canonical Correlation Analysis(CCA) embedding as features of the candidate's phrases and training them using a small number of labeled examples. The classifier we are using is a binary support vector machine that predicts whether or not a nominated entity is a candidate phrase.

With the above mentioned research papers we could conclude a number of things such as - enormous amount of data does not mean great tagging, CRF and LSTM methods are used often; if we want to create a dictionary for our needs we have to create a massive data-set of every word possible since we plan to capture every problem stated by the user and not limit this to a particular set of words/problems. The existing solution is to use Stanford Named Entity Recognizer(SNER) [59] but since we are using AWS which runs on the cloud and is triggered by the user, this dramatically increases the delay in response during the conversation with Alexa. With all these conclusions in mind we found the best solution is to use an online service that does not occupy space on the AWS and is an existing technology that has a massive data-set of every word in the English dictionary. We found two API's that could do this - Paralleldots [60] and Google Natural Language API [61]. By using both of the API's for entity recognition of various words we found that the latter API is better and more reliable since it's a Google product and finalized on this API.

1.3 Motivation

Participants interested in developing causal models have often done it with the support of a trained facilitator, who elicits concepts and causal relations [62, 63, 64, 65, 66]. Alternatively, tech-savvy participants may receive training and independently develop causal models using software such as `cMap` (common in education research), `MentalModeler` (most used in socio-ecological systems), or `Vensim` (typical in health and systems engineering). However, both approaches(trained facilitator and available software) have limitations. A trained facilitator can provide ample guidance but may be costly or unavailable. The software may be free and accessible anytime, but it does not guide the participant through the process of building a causal map. Also, both approaches rely on a visual inspection of the map as it is built, which does not easily scale as participants start to have many concepts and interrelationships. For instance, a participant may add a concept that is synonymous with an existing concept. To notice this redundant concept, all other concepts should be examined manually by the facilitator and participant, which becomes prohibitive as the number of concepts increases.

Thus, there is a need for an approach to causal model building that can be available at any time, without costs, and scales easily. In this research, we address this need by leveraging voice-activated virtual assistants (`Amazon Alexa`) to design and implement a *virtual facilitator*. Our solution guides participants in developing a model through a conversation (like a human facilitator), but is available at any time without cost (as software) and continuously examines the map to avoid typical issues such as synonymy of concepts.

1.4 Structure of Thesis

This thesis is structured as follows. In Chapter 2, we discuss the methodology used for implementing an artificial facilitator using Alexa. The technologies used to develop the *Alexa skill* are detailed in Appendix A. We compare three causal maps generated by Alexa and published causal maps in Chapter 3. In Chapter 4, we test the *skill* with participants and conduct two surveys (provided in Appendix B.1 and B.2) to capture their response to see if the *skill* can be useful in real life to successfully convert a person's reasoning into causal maps. We conclude with a discussion, constraints, future work and conclusion in the final chapter. It is recommended to go through Appendix A before proceeding to read this Thesis if you do not know the terms used in Amazon Alexa development such as Amazon skill, intents and slots. For list of abbreviations used in this Thesis please refer to Appendix C

Chapter 2

Build a Model: An Artificial Facilitator

2.1 Overview

Using the technologies described in Appendix A, we combine all of them to create an artificial facilitator that can mimic a conversation of a participant and a facilitator but with Alexa. The Table 2.1 lists all the technologies we have used to develop our *Alexa skill*.

Technology	Version
Amazon Web Services	Accessed October 2018
Alexa Skills Kit (ASK)	
Dynamo DB	
Amazon S3	
Google Natural Language API	1.2 GA Release
Python	3.6
NetworkX	2.2
NLTK	3.3
Ubuntu	18.04.1 LTS

Table 2.1: List of technologies and versions.

2.2 Process in an artificial facilitation

Externalizing mental model through Alexa involves a conversation between the participant and Alexa. The conversation should be guided by Alexa which means Alexa should guide the participant through questions. As described in Chapter 1.2.4, the process needs to (i) obtain concept labels that are quantifiable and distinct from labels already used, and (ii) help participants provide relationships to minimize the risk of missing essential structures such as loops. To help participants track relationships, a map building process can be conceptualized as a *graph traversal*: we want to elicit/visit all of the concepts (i.e. nodes) that pertain to the user's mental model, and we move from a concept to another using a relationship. Unlike a *graph exploration* in which we typically come back to the first node, the map building process ends on an arbitrary node. Two typical approaches to a graph traversal are a depth-first search (DFS) and a breadth-first search (BFS).

2.2.1 Depth-first search

Starting from a root, a DFS follows *one* unexplored node, and from there visits another unexplored node, thus going as far as possible. When it cannot go further, it backtracks until it can branch in a new direction. This approach is potentially undesirable in a facilitated process for at least three reasons. First, it can take participants on tangents and cause them to quickly go away from the main topic until they realize that their factors are no longer relevant to the problem space. As a result, the map may be imbalanced, and a high cognitive load is placed on the individual who needs to frequently think of the problem's boundaries. Second, often going *back* to a node may feel less natural than going *forward*, possibly coming across as 'jumping' between ideas. Third, a DFS requires that the user only provides *one* new concept each time, and may thus ask many times about the same node. This is more cumbersome than providing all known concepts at once then moving on. These points are illustrated through an example of a DFS-based conversation in Box 1.

Box 1. Sample conversation between a user and our facilitator using a depth-first search.

"Let's discuss obesity. What is one cause of obesity?
- Obesity is caused by unhealthy foods.
- What do you think would be one reason for having unhealthy foods?
- They're cheaper than healthy ones.
- And why are they cheaper?
- Because of government subsidies for food production.
- What's one driver of agricultural subsidies?
- I think that's going off topic.
- Ok. Let's go back. You said unhealthy foods are cheaper than healthy ones, because of agricultural subsidies. Any other reason?
- Not that I can think of.
- That's fine. Let's go back. You said people have unhealthy foods because of prices. Any other cause?"

2.2.2 Breath-first search

Starting from a root, a BFS asks for *all* connected nodes. Intuitively, it acquires the complete layer of connected concepts at a distance one from the starting one. Then, it goes through all of these concepts and acquires all of their directly related concepts, thus completing the layer at a distance 2. By going through entire layers at a time, it avoids taking participants on tangents. By asking whether participants want to continue when an entire layer is done, it asks for conscious monitoring of the problem boundary at specific moments instead of offloading this responsibility onto the user at every question. When going through layers, it only goes forwards (i.e. uses a queue) instead of backwards (i.e. using a stack in the DFS). Finally, by asking for all connected concepts at once, users have the natural opportunity to share all of their thoughts instead of restricting themselves to a single new concept. For these reasons, our artificial facilitator uses a breadth-first search. The functioning of a BFS is illustrated via a conversation in Box 2.

Box 2. Sample conversation between a user and our facilitator using a breadth-first search.

"Let's discuss obesity. Can you share all of the contributors to obesity?
- Eating unhealthy foods, lack of physical activity.
- Alright, these are the direct drivers. Do you want to go further and share what do you think drives these?
- Sure.
- What do you think would be the reasons for eating unhealthy foods?
- Could be a coping mechanism for stress or depression, an affordability issue because they're cheap, or personal taste preference.
- And what about physical activity?
- Could be obesity itself, because it creates some barriers. Or a lack of access to facilities, or fear of engaging in physical activity.
- So we've now looked at indirect drivers. Do you think it'd be relevant to discuss their causes?"

2.2.3 Conversation flow diagram

While the BFS is meant to cover more concepts, the appearance of previous concepts can create loops. As illustrated in Box 2, we have a loop from obesity to a lack of physical activity, which itself contributes to obesity. As shown in Figure 2.1 and Figure 2.2, our process utilizes the layer-by-layer approach of the BFS. It also closely monitors the names of concepts, as shown in Figure 2.1(inset A). We actively prevent the creation of similar concepts, by informing the user that they are already present in the map under a possibly different name. We also

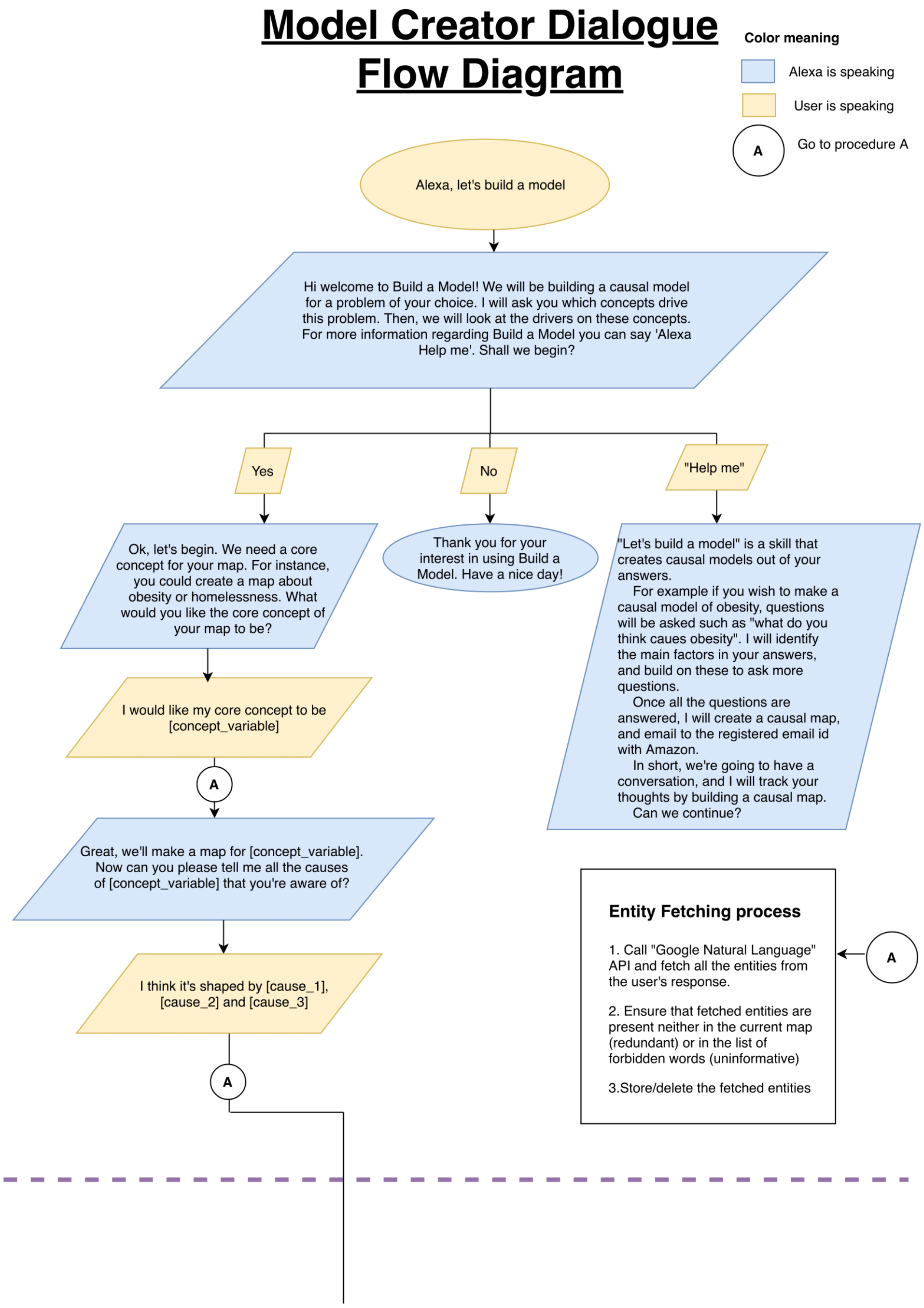

Figure 2.1: Process to start a model and provide the first causes.

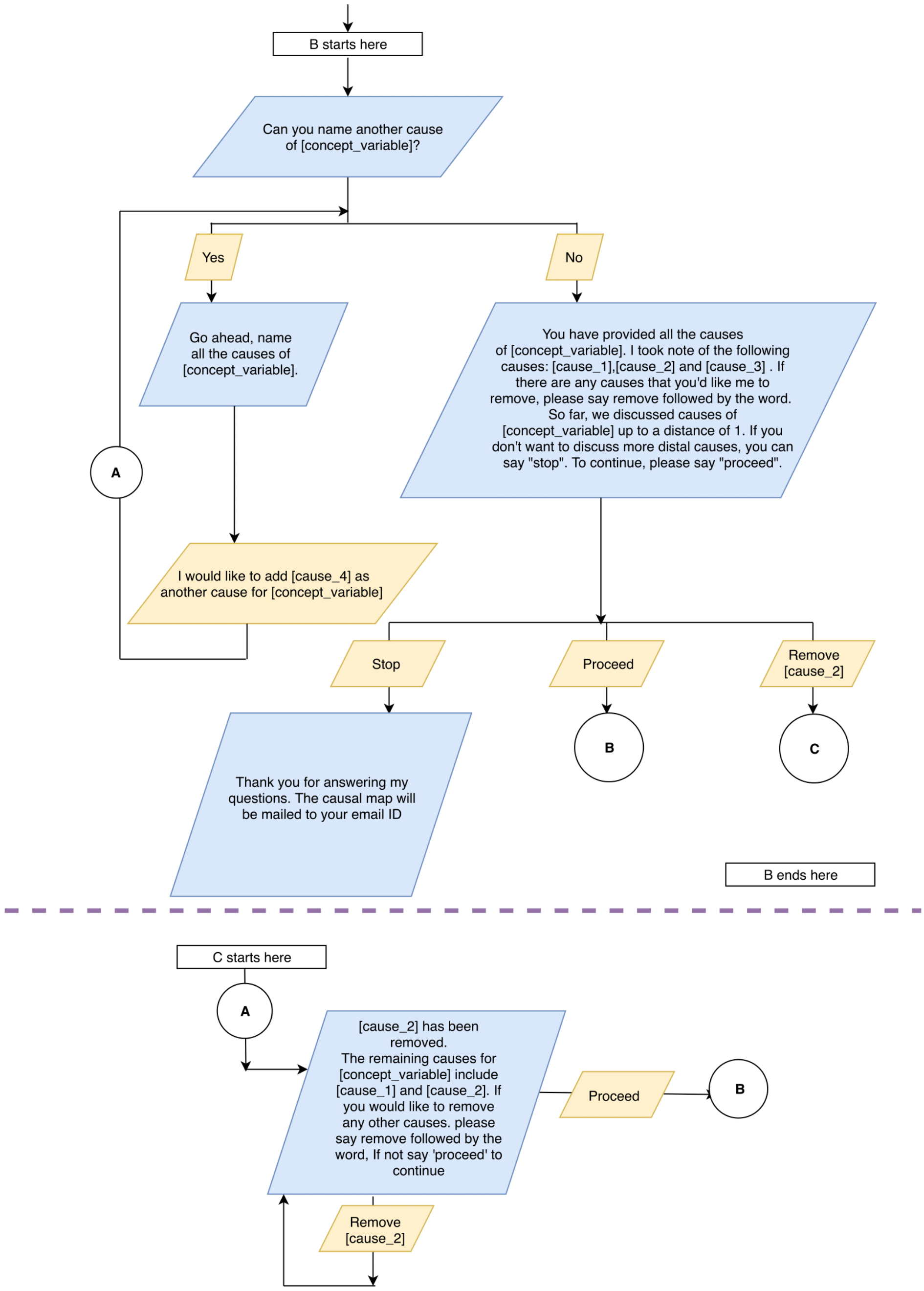

Figure 2.2: Continuation of the process, showing how to get additional causes, get another layer of causes, or removing a causal edge.

attempt to avoid the use of concepts that cannot be quantified, thus promoting more operational definitions of concepts.

2.3 Implementation

Our implementation is task-oriented as we seek to guide a participant in externalizing their mental model. The virtual facilitator controls the flow of the conversation by asking the questions. Interactions in the deployed version are exclusively vocal, but developers in Amazon Alexa also have access to a console that takes written input (for testing only). Dialogue-management uses a frame-based system. All of these technical choices were briefly discussed in Appendix A. Our code is provided at https://github.com/datalab-science/causalMapBuilder.

2.3.1 Entity fetching process

Alexa converts the user's voice response to text and sends it to our code, by applying Natural Language Processing on the text we can understand its structure and meaning by using machine learning. We can extract information about individuals, places, and events, and better understand their feelings surrounding social media and customer conversations. Before we dug deep into designing a natural language processing algorithm, we found that Google Natural Language API [61] can be used to serve our needs. We have used the Entity analysis (inset A from Figure 2.1) since we are interested in fetching the entities from the user's response.

Entity fetching is done to avoid capturing unnecessary words that do not relate to the core concept. For example, in the participant's response, "Excess weight can be caused due to excess food intake and lack of exercise". The entities we are interested in are "food intake and exercise". We identify these entities by passing the user's response to Google Natural Language API which identifies entities and returns them. Here the entities returned are; weight, food intake, lack of exercise. Only these entities are stored in our database and are used in the next questions. The entity fetching process usually works by fetching the nouns in the sentence, but when there are no nouns in the response, we fetch the adjectives in the response.

2.3.2 Checks before storing the entities in our database

Before storing the entities fetched from Google Natural Language API, the entities go through 3 different checks as shown below. As shown in Algorithm 1.

Algorithm 1 Entity Fetching

function $entity\_fetch(answer\_from\_user)$ **Input** : User's response
Output: Entities
$response \leftarrow answer\_from\_user$; `// 'Build a model' skill's intent fetches response from the user`
$entities \leftarrow GoogleNaturalLanguageAPI(response)$; `// using Google natural language API to fetch entities`
if *entities in entity_database* **then**

*[H]Three entity checks inform the user that the entity is already captured **else if** *entities not in database* **then**
  $synonyms \leftarrow getSynonym(entities)$; `// using WordNet NLTK fetch the synonyms of the new entities and store them`
  if *entities in synonyms* **then**
    inform the user that the entity *synonyms* is already captured
  else if *entities not in synonyms* **then**
    save the entities to entity_database
else if *entities in unquantifiable database* **then**
  do not capture the entities; `// unquantifiable database = [see Figure 2.3]`
return save the entities to entity_database

Identifying entities which are already captured

In the example mentioned in section 2.3.1 — when the participant responds - *Excess weight can be caused due to excess food intake and lack of exercise*, *weight* is captured as an entity. Since *weight* is already our core concept we remove it from the identified entities list. Also, if a participant uses the same answer to two different questions as shown in the example in Box 3, this will result in duplicates being stored in our database. To avoid this, we always check whether the previous captured entity was captured in any other response; if captured, we do not store these duplicate entities in our database. Instead, we link the first original entity to repeated entities. As seen in the below example we do not store *stress* as a new entity but link the previously captured entity to it.

Box 3. Sample conversation with duplicate entities
Alexa: What causes obesity?
Participant: Obesity is caused due to over-eating and stress.
Obesity = [over-eating, stress]
Alexa: What do you think causes over-eating?
Participant: Sometimes over-eating is caused due to stress
Obesity = [over-eating = [Obesity[1]], stress]

Identification of entities which is unquantifiable

A causal map is not supposed to have unquantifiable concepts, but users may lose track of this requirement. If Google Natural Language API identifies an unquantifiable entity, then our application can use it in nonsensical questions. For instance, 'excess' was identified as an entity although it is unquantifiable. The application may continue by asking "what causes excess?". We tested the *skill* with eight subjects over two months to identify such problematic entities. Since we cannot manually identify *all* such entities, we use the ones we identified as seeds to automatically fetch all similar entities, thus constituting a comprehensive dictionary of entities to ignore. The creation of this dictionary takes three steps performed using WordNet:

(1) We have a set of entities, identified during testing as both (i) fetched by the Google Natural Language API and (ii) unquantifiable. For instance, consider {lack, bunch}.

(2) For each word, we retrieve all its *hypernyms*, which are words with

a broader meaning (e.g., colour is a hypernym of red). Here, {lack, bunch} is transformed into {need, agglomeration, collection, cluster, gathering}.

(3) For each hypernym, we retrieve all its hyponyms, which are more specific words (e.g., hyponyms of colour would include red, blue, and green). In this example, {need, agglomeration, collection, cluster} would be expanded into a large set including {lack, necessity, urge, ..., bunch, pair, trio, hive, crowd, agglomeration, batch, block, ensemble, ..., population}.

The unquantifiable words fetched by the above process can be viewed here - `https://raw.githubusercontent.com/datalab-science/causalMapBuilder/master/remove_words.py`

Identification of synonyms

What if the user answers using synonyms of the same entities which has already been captured? If Alexa cannot identify the synonyms, then this creates a problem during the questions. Consider the example shown in Box 4. — stress and pressure are synonyms of each other.

Box 4. Sample conversation with synonyms
Alexa: What causes stress during school?
Participant: I think stress is caused due to exams
Alexa: What causes pressure during school?
Participant: Exams

Alexa should not ask two different questions for stress and pressure. Instead, Alexa should ask just one question about the first word that was answered by the participant. This is achieved by checking the synonyms in a dictionary we have created. This is achieved by retrieving the synonyms of each word from WordNet and constructing a dictionary of the synonyms for each identified entities. For every new entity identified we cross-check in this dictionary and if the entity is already present as the synonym of another entity, we do not save this as a new entity but link it to the existing entity in our database. If it is a new entity we save it as a new entity, find its synonyms and add it in our dictionary.

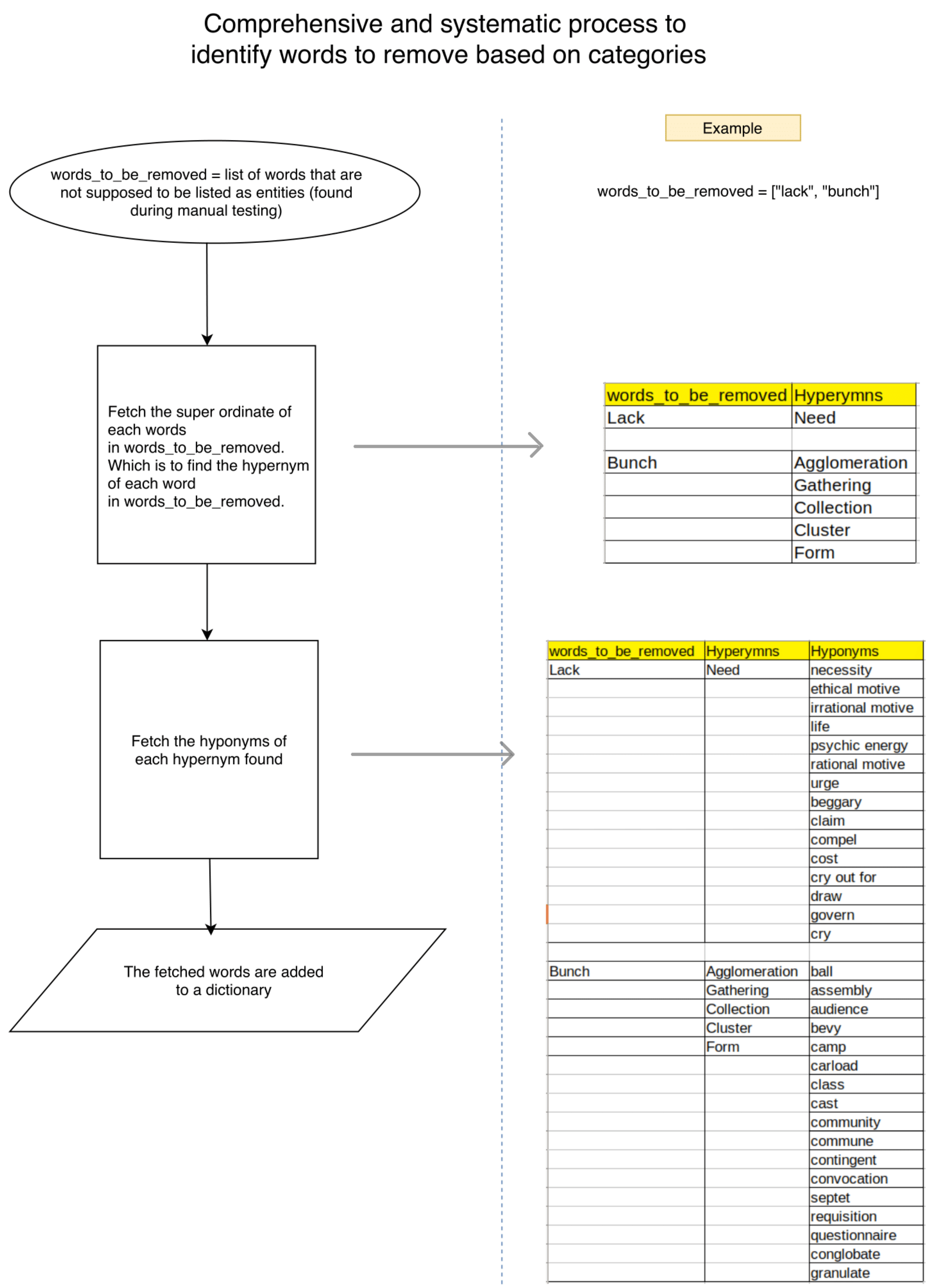

Figure 2.3: Identification of entities which cannot to be questioned.

2.3.3 Creation of causal maps through entities

We tried to create the causal map (Figure 1.1) using *Build a Model* and generated a causal map and shown in later chapter 3 (more specifically Figure 3.5.) The entities fetched during this session are as shown in Figure 2.4, entities mentioned in column B are connected to the entities in column A. This represents the relationships between two entities and we store these entities in the data structure format - dictionaries (dictionaries are unordered key-value-pairs). This dictionary is fed to python's NetworkX library [67] to form nodes and edges that replicate a causal maps.

We decided to use the [68] NetworkX library because we needed to convert entities into nodes representing their relationship with other nodes. Since NetworkX does an excellent job of converting words into graphs, we decided to select this library to create causal maps.

A	B
over-eating	hunger
over-eating	loneliness
over-eating	stress
over-eating	sadness
hunger	medications
stress	body shape
stress	weight stigma
sadness	body shape
sadness	weight stigma

Figure 2.4: Entities captured during the conversation with Build a Model when the core concept is over-eating

2.3.4 Summary

Once the conversation with Alexa is completed or ended by the user by using the stop word *stop* or sentence such as “I do not want to continue answering the questions." A causal map is then generated by Build a Model and sent to *datalab.science@gmail.com*, and the responses are then stored in JSON format in Amazon DynamoDb. Also, a conversation log containing the entire conversation from start to finish with the participant is stored in DynamoDb. The results and conversations logs are stored for future use for any researcher to analyze the data that has been collected. The email received contains the identified core concept and its causes, an attachment of the same in CSV format, an image of the causal map is also available for download. The screen-shot of the email is as shown in Figure 4.2.

Chapter 3

Case studies

3.1 Comparison of causal maps with the case study maps

We used three case studies to test our system. In the first two case studies: excess weight and unhealthy eating, we verified whether a participant could (re)create a previously developed causal map when using our artificial facilitator. Leveraging the broad variety of languages and accents supported by Alexa, we set the device to Indian English for these two cases, as it is the language spoken by our participant. In the third case: over-eating, the device was set to American English, and we tested additional features such as detecting redundant concepts or allowing the user to correct the map. All case studies were performed using an Amazon Echo Dot Device version 618571720. We recorded the discussion and the resulting map that our artificial facilitator emailed to the participant. To provide full disclosure, our three recordings can be viewed at `https://www.youtube.com/playlist?list=PL7UTR3EL44zrkwrcDkiSwV-7kL0Nv6fQ5`

3.1.1 Excess weight causal map

The map in Figure 3.2 without the green highlighted text shows the original map we used to test the *artificial facilitator* for *excess weight*. There are 5 layers of questions asked by Alexa (the layer questioning process is explained in Appendix B.3 in the user guide). The causes were answered as it is in Figure 3.2. Each node here is linked through directed arrows with positive (+), and negative (-) signs (which represents if the node affects positively or negatively, example: *exercise* negatively affects *excess weight* which means more of exercise reduces weight.) The entire conversation to replicate this map with Alexa can be viewed at `https://www.youtube.com/watch?v=57tq0w4OEPw&list=PL7UTR3EL44zrkwrcDkiSwV-7kL0Nv6fQ5&index=3&t=0s` and it took about 8 minutes. The Figure 3.1 is the map generated by Alexa for *excess weight*. The difference between the case study map and the Alexa generated map is highlighted in green in Figure 3.2.

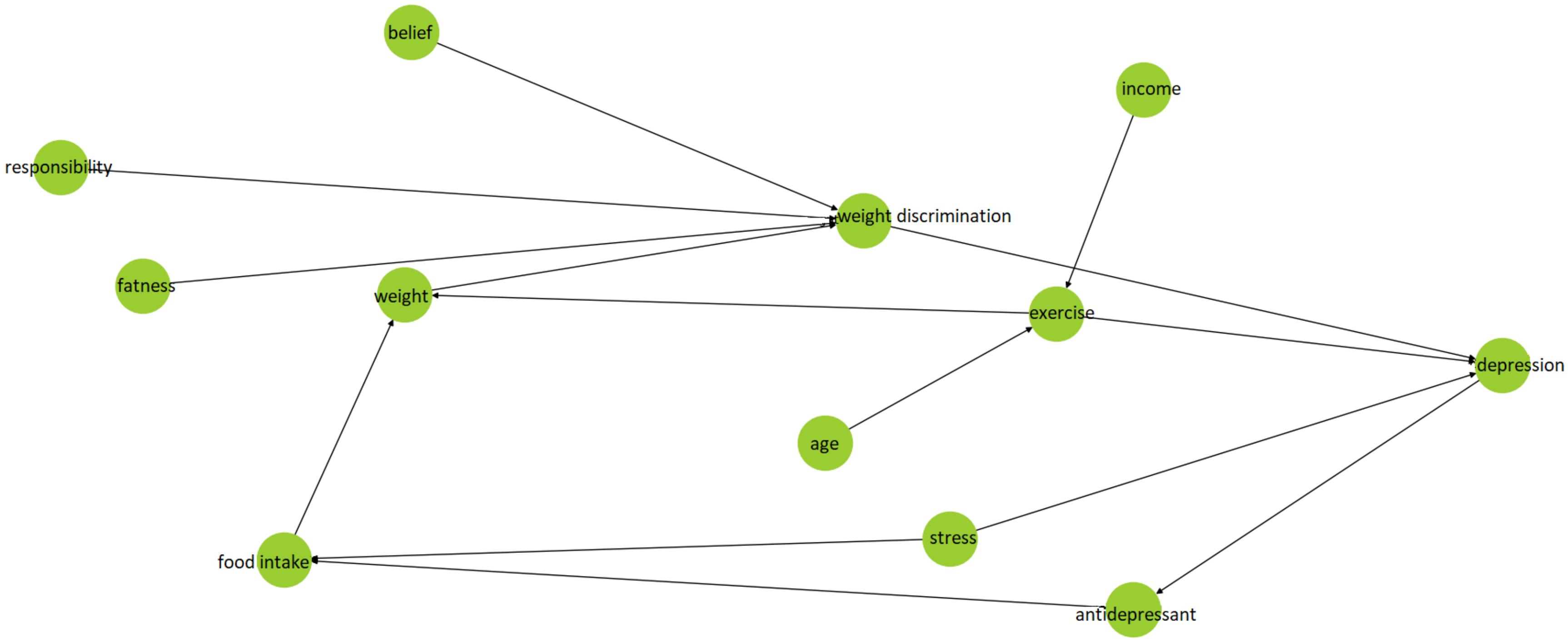

Figure 3.1: Causal map generated by Alexa for *excess weight*.

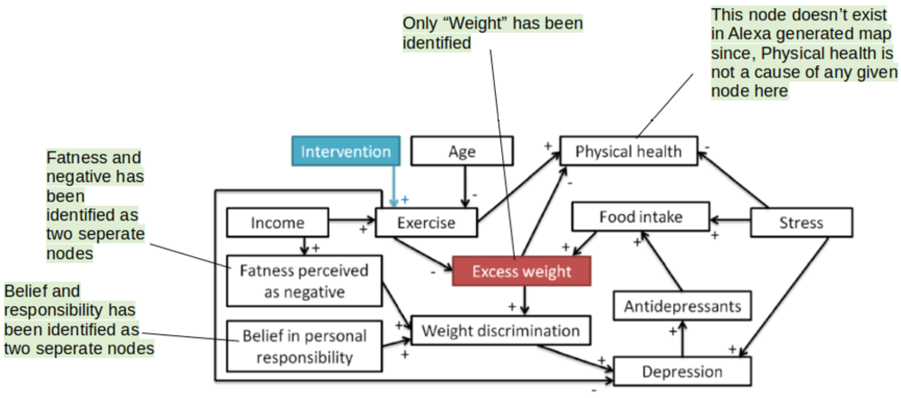

Figure 3.2: Difference between Alexa generated causal map and the original causal map for *excess weight*.

3.1.2 Unhealthy eating causal map

Similar to *excess weight* we tried to replicate the causal map in Figure 3.4 for *unhealthy eating*. The conversation can be viewed at and it took around 8 minutes 12 seconds to form the causal map shown in Figure 3.3. There were only 2 layers of questions asked by Alexa. The entire structure could be replicated with only two nodes begin identified differently compared to the original map. The difference is shown in highlighted text in Figure 3.2.

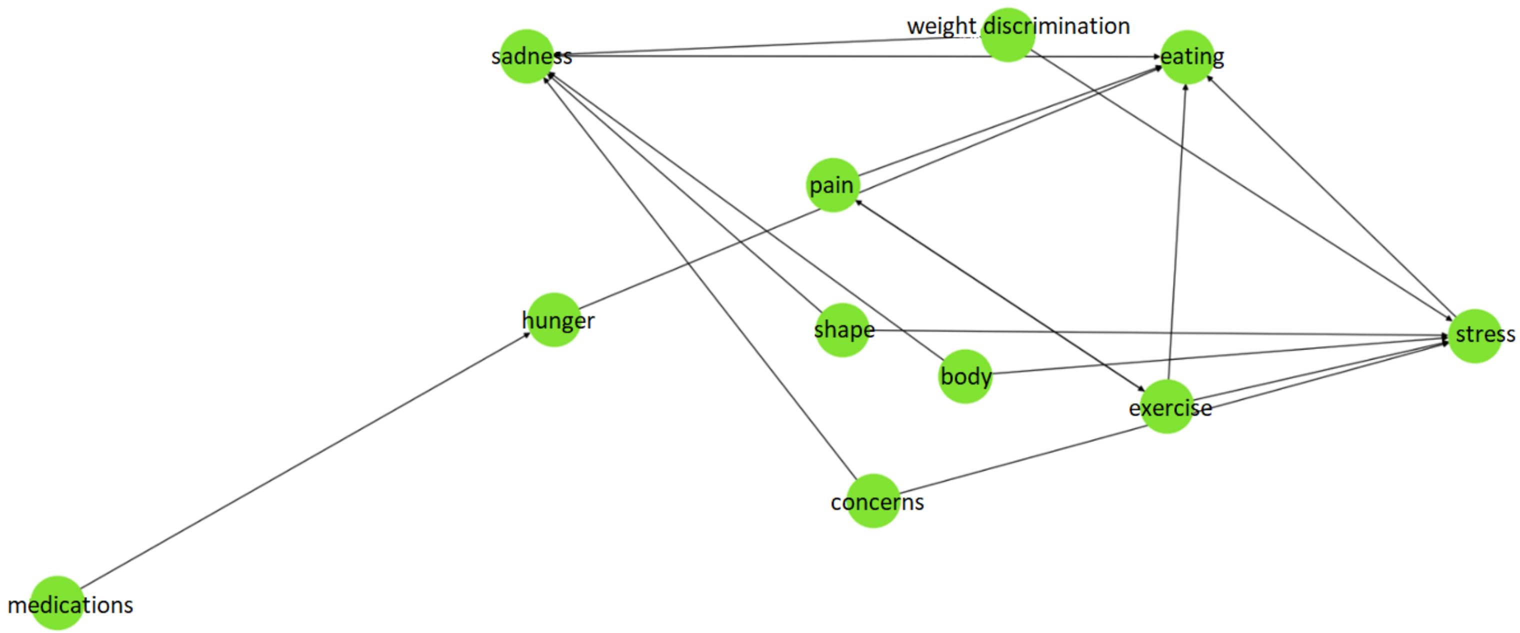

Figure 3.3: Causal map generated by Alexa for *unhealthy eating*.

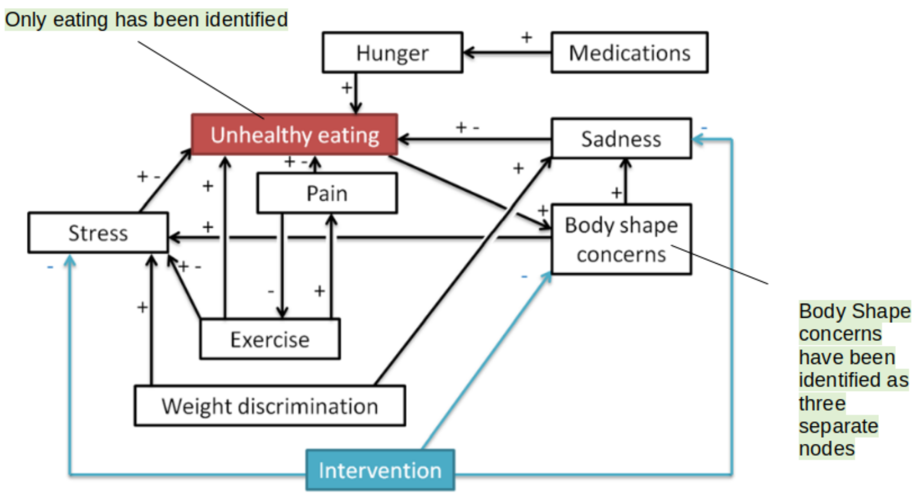

Figure 3.4: Difference between Alexa generated causal map and the original causal map for *unhealthy eating*.

3.1.3 Over-eating causal map

The third case study (Figure 1.1) involved demonstrating additional features of our artificial facilitator such as detecting redundant concepts, allowing the user to correct the map, requesting Alexa to repeat questions. To test the redundant identifying feature we stated that over-eating was caused by over-indulgence and noticed that the *skill* identified it as a redundant concept (since these two concepts are considered interchangeable as per WordNet) and informed the user, see `https://www.youtube.com/watch?v=U2mYkSLE9NE&t=40s`. We also confirmed that users were able to remove causes when they have been incorrectly captured (`https://www.youtube.com/watch?v=U2mYkSLE9NE&t=213s`). Finally, we verified that the virtual facilitator did repeat questions when prompted by the user (`https://www.youtube.com/watch?v=U2mYkSLE9NE&t=95s`).

3.2 Conclusion

Our first two case studies demonstrated that the *structure* of the maps could correctly be created using our artificial facilitator. We observed three issues due to the automatic detection of entities. First, it can

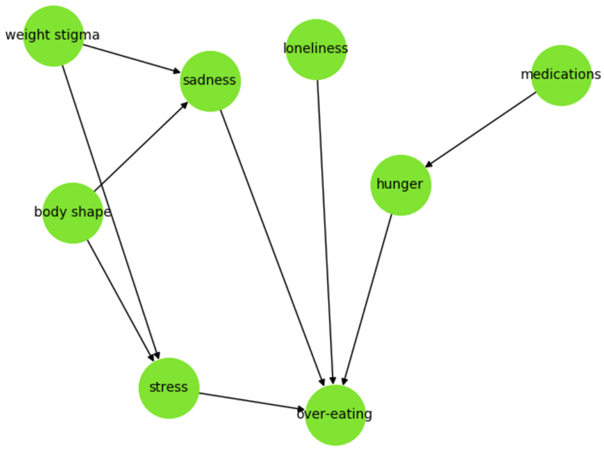

Figure 3.5: Causal map generated by Alexa for *over-eating*

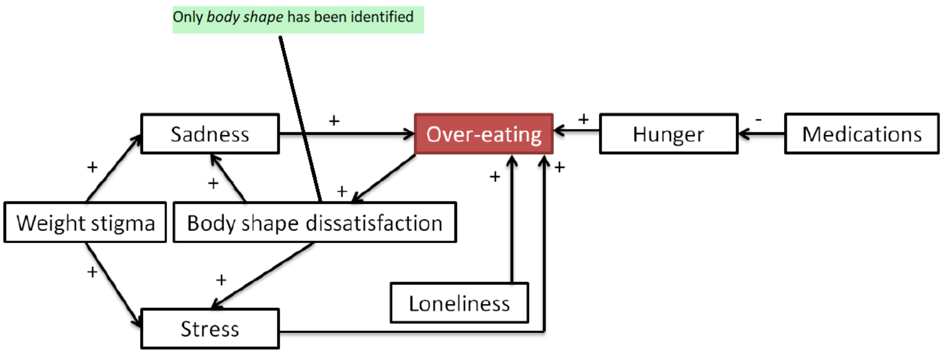

Figure 3.6: Difference between Alexa generated causal map and the original causal map for *over-eating*

lead to significantly shorter concept labels (`https://www.youtube.com/watch?v=57tq0w40EPw&t=324s`). The problem is aggravated when concepts that should be different are shortened such that they are indistinguishable. For instance, ‘cardiovascular diseases’ and ‘metabolic diseases’ are very different medical situations. However, Google Natural Language API recognizes both as ‘diseases’ and thus conflates them, which results in structural errors for the map. Second, entity recognition is a bottleneck of the application in terms of time: users have to wait for several seconds before entities have been processed silently. These awkward silences disrupt the flow of the discussion. Finally, accents can lead to very different performances in terms of entity recognition. Results are not only different between Indian and American participants, but also among Americans (e.g., from the South or the Midwest). As noted by Rachael Tatman, the training dataset for smart speakers results in working “best for white, highly educated, upper-middle-class Americans, probably from the West Coast, because that is the group that’s had access to the technology from the very beginning” [69].

Chapter 4

Testing the Artificial facilitator with Participants

4.1 Overview

We conducted a study with a small number of participants with researchers and friends located in Thunder Bay, Canada. The survey was conducted in DataLab at Lakehead University. Since the objective was to show VAPA can capture the thinking process of a person regardless of their English dialect, we did not foresee any specific requirement for selecting participants apart from the standard preconditions such as no psychological or neurological disorders and normal English speaking capabilities. Two surveys pre-test and post-test surveys (see the Appendix B.1 and B.2) were designed. Pre-test survey was to understand each participant's previous experience with VAPA, and their knowledge about entities and causal map. The post-test survey was designed to capture the participants experience with using Build a Model. We observed that during development phase of the tests a person found the questions asked by Alexa annoying when they had no idea regarding what information it was trying to capture from them. With this in mind, a user guide was designed (see Appendix B.3) which included — the workings of the skill, instructions on how to use and answer to the questions asked by Alexa, and an example conversation.

Note: No approval of ethics has been taken for testing with users as this research emphasizes the use of conversational artificial intelligence and NLP techniques as a proof of concept.

4.2 Hypothesis

The central goal of the experiments is to examine if it is possible to use VAPA to guide a participant in capturing their thought process of any problem without any guidance from a human facilitator. Based on this requirement, three null hypothesis was developed with respect to their experience with *Build a Model*:

$H_0{}^1$: There is no difference between participants with or without prior VAPA experience.
$H_0{}^2$: There is no difference between participants with or without experience with 3rd party skills/actions (Amazon/Google Home).
$H_0{}^3$: There is no difference between participants with or without the knowledge of causal maps and entities.

4.3 Method

4.3.1 Materials

The survey consists of two parts: a pre-test survey and a post-test survey along with a user guide (see Appendix B.1 and B.2). Both of these surveys were online surveys with a brief description. The user guide consisted of detailed instructions about the skill and its working. Each participant were required to enter their email id for two purposes: to use it as a unique key to link pre-test and post-test survey results and to email them their causal maps generated during testing.

The pre-test survey consisted of 15 questions which were aimed at gathering the participant's demographic information as well as their experience with VAPA on a scale from 1 (not likely to use) to 10 (extremely likely to use). These ratings were expected to be correlated with participants experience of the skill's usefulness in real life asked in post-test survey. The survey also required the user to rate if they have read the user guide and understood the workings of the skill, its questioning pattern and the way it communicated its results. These results were compared with the result of the post-test survey to find whether participants found the skill's to be useful. This was needed as the skill's success rate depended on the participant's understanding of the skill. Once the user has read the user guide and taken the pre-test survey, they will then be able to proceed to the post-test survey. The post-test survey aims to capture the participant's experience with the skill to find out if they find Alexa skill to be a useful voice activated tool for converting the user's thoughts into causal maps. The results of the post-test survey will be compared with pre-test results to see the conditions which affected their reactions. Both surveys had a question that asked the participant to give their concerns or feedback, if any, could be used to enhance future skills. It was expected that each participant would select a problem of their choice to form causal maps of their thinking about the chosen issue.

4.3.2 User Study

Data from 13 participants who agreed to participate in this study have been included. Also as mentioned before participants were researchers and students studying at Lakehead University and few of them had completed their education, even so, all the participants were from Thunder Bay, Canada. The skill was not deployed to the Amazon skill store and could be used only on one testing device which was handled by the student researcher. All the participants tested the skill and completed the survey under the supervision of the student researcher at DataLab at Lakehead University. The participants choose volunterialy to test the skill.

The participant's pools were roughly split into gender, with 38.46% of them being Female and 61.54% of them Male. Out of the female participants, 60% were native English speakers, and 40% were non-native English speakers. Out of the male participants, 25% were English speakers and 75% were non-native English speakers. A majority of 77% of the participants were pursuing their master's degree and aged between 22-27 years of age. For a complete profile of the participant's demographics, see Table 4.3.2

Demographic	N	%
Gender		
Female	5	38.46
Male	8	61.54
Age		
18-24	5	38.46
25-34	5	38.46
35-44	0	0
45-54	2	15.38
55-64	1	7.69
Native English Speaker		
Yes	5	38.46
No	8	61.54
Education		
Master's degree	10	76.92
Bachelor's degree	2	15.38
High school diploma or equivalent	1	7.69

Table 4.1: Participant's demographic information

4.3.3 Procedure

Participants who volunteered to test the skill were sent survey forms and the user guide. They were required to be present at the DataLab for testing the skill with the Alexa device which was enabled on it. As mentioned previously the procedure has four steps 1) read the provided user guide 2) take the pre-test survey 3) test Build a Model and 4) take the post-test survey. The estimated time for this was expected to be not more than 45 minutes.

4.4 Results

All 13 participants responded to all the questions. What we have noticed during testing with all these participants is that Alexa has been unable to correctly identify a few of the words spoken by the participants as each person has a different way of pronouncing words when using English of different dialects. This created unintended entities to be captured, creating questions that could not be answered. For example, a non-native English speaker from India wanted to form a causal map for *hate* but Alexa consistently identified the word as *hit*, because it is an entity identified by Google Natural Language API. The next question from Alexa to the participant was "can you name the causes of hit?". This question made no sense since the word *hate* was incorrectly identified as *hit*. Build a Model has been designed in five different English dialects, including USA, Canada, India, Australia, and the UK. During testing the corresponding English language of the skill was enabled depending on the English dialect of the person. These dialects in Alexa represent that the pronunciation of each person with different dialects can be correctly identified by Alexa when the correct dialect in Alexa is used. We tried the skill with the English(Indian) dialect with an Indian participant hoping that *hate* would be identified as *hate* and not *hit*, but it was misidentified on each trail. We also tested the same with a native English speaker with English(CN) and noticed that *hate* was identified as *hate* but more complicated words were misidentified such as cardiovascular disease, lymphoma and so on. What we have understood from this is that words in different dialects can be identified by Alexa as long as we have created *intent* slots in Alexa skill during development, which we did not. The intent slots act as Alexa's reference guide that says these are the words that can be said by participants. Since we don't give Alexa any idea of what to capture, this affects her word identification ability.

Since we wanted to capture every possible core concept chosen by a person, we developed the skill without any slots of intent. Using intent slots meant that we limited our scope to a specific topic and therefore encountered a major misidentification of words. Alexa worked better to identify native English speakers with their respective dialects compared to the non-native speaker and also failed to identify complicated words with native English speakers. We, therefore, felt that judging the usefulness of the skill developed through voice testing would not be a fair judgment as we did not develop voice recognition and depended on Alexa for it. The voice recognition will improve as technology improves with time, but for now we decided to have participants type-in their responses (typing is available only on Alexa development platform and hence the testing was performed on a Laptop), which meant that analysis based on native and non-native speakers was eliminated and the testing focused exclusively on if the developed model could convert the reasons or thoughts of any person into causal maps regardless of their English language competency.

The maps generated by the participants were mailed to their email id, and they confirmed it through visual inspection if they were created correctly or not. There were no errors in the generation of these maps because the participant had the option to correct any errors during the conversation with Alexa (option to remove wrongly identified entities is a feature of *Build a model*.

Question	**N**	**%**
Prior VAPA Experience		
Yes	11	84.62
No	2	15.38
VAPA experience with 3rd party skills/ actions		
Yes	4	30.77
No	9	69.23
Knowledge of entity		
Yes	12	92.31
No	1	7.69
Knowledge of causal maps		
Yes	12	92.31
No	1	7.69
Knowledge of relating problems to it causes		
Yes	11	84.62
No	1	7.69
Maybe	1	7.69
Is Build a Model a useful skill		
Yes	9	69.23
No	0	0
Maybe	4	30.77
Rate your experience with Build a Model(1-10)		
7-10	12	92.31
4-6	0	0
1-3	1	7.69

Table 4.2: Response of the participants

4.4.1 Hypothesis Testing

To test our three hypothesis, we employee a 2 x 2 cross tabulation analysis along with Chi-square and Fisher's exact test. The cross tabulation is a descriptive analysis of two given variables with percentage description of the row, column and the total.

			Post-test Q8: Do you think "Build a model" is a useful Alexa skill that can be used in the future? [Usefulness_Alexa_skill]			
			Yes	No	Maybe	Total
Pre-test Q6: Have you used voice-activated personal assistant before (such as Alexa, Google Home/mini, Apple Siri)? [VAPA_experience]	No	Count	1	0	1	2
		% within VAPA_experience	50.0%	0%	50.0%	100.0%
		% within Usefulness_Alexa_skill	11.1%	0%	25.0%	15.4%
		% of Total	7.7%	0%	7.7%	15.4%
	Yes	Count	8	0	3	11
		% within VAPA_experience	72.7%	0%	27.3%	100.0%
		% within Usefulness_Alexa_skill	88.9%	0%	75.0%	84.6%
		% of Total	61.5%	0	23.1%	84.6%
Total		Count	9	0	4	13
		% within VAPA_experience	69.2%	0%	30.8%	100.0%
		% within Usefulness_Alexa_skill	100.0%	0%	100.0%	100.0%
		% of Total	69.2%	0%	30.8%	100.0%

Table 4.3: Cross tabulation of previous VAPA experience of participants with their response to if they find *Build a model skill* useful

$\mathbf{H_0}^{\mathbf{1}}$ null hypothesis

Let us consider the $H_0{}^1$ null hypothesis: There is no difference between participants with or without prior VAPA experience in their experience with "Build a Model". To analyze this hypothesis, we need to cross-tabulate the participants previous VAPA experience and their experience with the skill after testing (Table. 4.3). We may notice that having previous experience with a VAPA was irrelevant to finding full use of the skill or not. Since each participant had to read the user guide, we can say that the instructions provided in it helped guide the participant with the skill. We carried out a Chi-square and a Fisher's test statistically to further test the null hypothesis. As you can see in Table 4.7, the p-value of Chi-square test is 0.522 which is above the significance value(0.05) considered. Similarly, p-value from Fisher's test is greater than 0.05.

Therefore we can accept the $H_0{}^1$ null hypothesis and say there is no significant relationship between the experience of a participant with VAPA and their experience with Build a Model. Which means if a person wants to have a positive experience with the skill, they don't need to have any VAPA experience.

$H_0{}^2$ null hypothesis

Similar to the first null hypothesis test, we wanted to see if any experience with installing skills from the store affected the experience of the participants with Build a Model. Experience from installing skills from the skills store tells us about if a person knows how to invoke a skill, follow the given instructions and give Alexa proper commands accurately. Figure 4.1 shows a skill called *Memory Bank* on Amazon skill store. Here you can see that if a person has had previous experience enabling this on their Alexa device and using it. Then they have an idea about how to use these skills which means that they can easily catch up with Build a Model's instructions with any trouble.

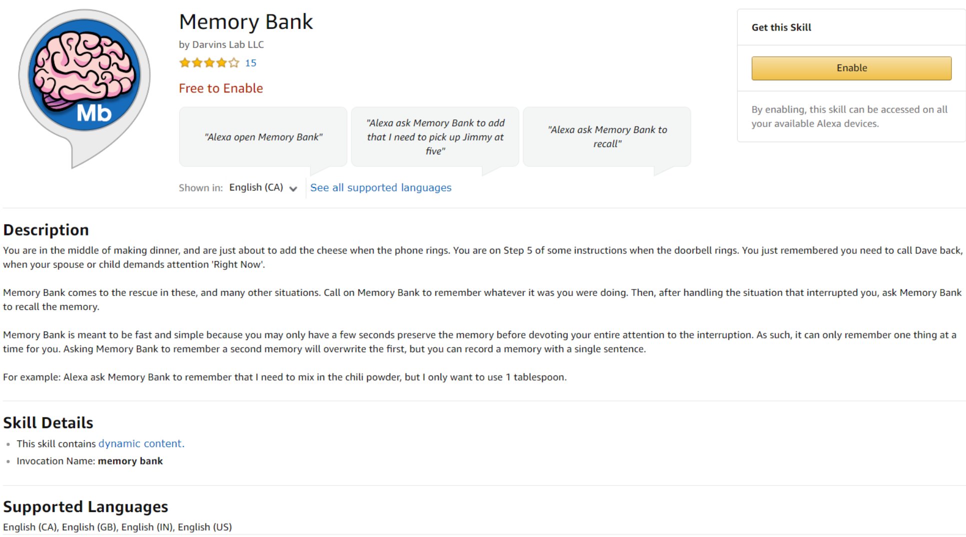

Figure 4.1: Snapshot of a *Memory bank skill* present on Amazon skill store

We performed a cross-tabulation of VAPA experience with skills and their Build a Model experience after testing (Table 4.4). What we could notice from the table is that most of the participants had never enabled a skill from the Alexa skill store which meant they did not know how to invoke a skill through an invocation name. For example: In Build a Model the invocation name is *Model Creator*. However, with the user guide provided they could follow along with the instructions properly. Most participants who have never enabled skills or participants who have enabled skills found Build a Model to be a useful skill in converting

			Post-test Q8: Do you think *Build a model* is a useful Alexa skill that can be used in the future? [Usefulness_Alexa_skill]			
			Yes	No	Maybe	Total
Pre-test Q7: If you have answered yes for the above question. Have you installed any skill for Alexa, or any action for Google home/mini? [VAPA_experience_with_skills]	No	Count	6	0	6	2
		% within VAPA_experience_with_skills	66.7%	0%	66.7%	100.0%
		% within Usefulness_Alexa_skill	66.7%	0%	66.7%	15.4%
		% of Total	46.2%	0%	46.2%	15.4%
	Yes	Count	3	0	1	4
		% within VAPA_experience_with_skills	75.0%	0%	25.0%	100.0%
		% within Usefulness_Alexa_skill	33.3%	0%	25.0%	30.8%
		% of Total	23.1%	0%	7.7%	30.8%
Total		Count	9	0	4	13
		% within VAPA_experience_with_skills	69.2%	0%	30.8%	100.0%
		% within Usefulness_Alexa_skill	100.0%	0%	100.0%	100.0%
		% of Total	69.2%	0%	30.8%	100.0%

Table 4.4: Cross tabulation of previous *VAPA experience* of participants with skills from the Amazon skill store and their response to if they find *Build a model skill* useful

thoughts through voice input into causal maps. We can see in Table 4.7 that both Chi-square and Fisher's exact test p-value is more than 0.05. Which says there is no correlation between finding Build a Model useful and having any experience with installing and using skills from the Alexa skills store. $H_0{}^2$ has been supported but with the condition that the participants be provided with a user guide which gives proper instructions and explanation of Build a Model working.

$\mathbf{H_0{}^3}$ null hypothesis

We wanted to see if there is a relation between "knowing what a causal map or an entity is" and "participants finding Build a Model helpful or not in converting their thoughts into causal maps just the way they wanted to". This was necessary because if the participant did not know what an entity is, he would rate the skill as not helpful as it depends on the entities being captured in the response of the participant. One of the participants, for example, responded *intrusive parents* as the cause of *stress* and only parents were captured from their response. If they did not know the skill captures only entities, then they will assume that the skill is not working accurately, which is not correct. During development testing of Build a Model, we came across this problem and added what an entity and a causal map is in the user guide for the participants to get an idea regarding them before testing.

We performed cross tabulations with "if they knew what an entity and

			Post-test Q8: Do you think *Build a model* is a useful Alexa skill that can be used in the future? [Usefulness_Alexa_skill]			
			Yes	No	Maybe	Total
Pre-test Q11: Do you know what an *entity* is? [Entity]	No	Count	1	0	0	1
		% within Entity	100.0%	0.0%	0.0%	100.0%
		% within Usefulness_Alexa_skill	11.1%	0.0%	0.0%	7.7%
		% of Total	7.7%	0.0%	0.0%	7.7%
	Yes	Count	8	0	4	12
		% within Entity	66.7%	0.0%	33.3%	100.0%
		% within Usefulness_Alexa_skill	88.9%	0.0%	100.0%	92.3%
		% of Total	61.5%	0.0%	30.8%	92.3%
Total		Count	9	0	4	13
		% within Entity	69.2%	0	30.8%	100.0%
		% within Usefulness_Alexa_skill	100.0%	0	100.0%	100.0%
		% of Total	69.2%	0	30.8%	100.0%

Table 4.5: Cross tabulation of whether the participant knew what an *entity* was with if *Build a Model* skills were found to be useful

a causal map is" against "if they found Build a Model useful", see Table 4.5 and Table 4.6. In these cross tabulations, we can see that the majority of the participants knew what a causal map (75.0%) and entity (66.7%) was and rated the skill as a helpful skill. 25% of participants who knew what a causal map is and 33.3% who knew what an entity is rated that the skill can maybe help convert the thoughts into causal maps. These participants left comments such as - entity capturing needs to improve and does not capture every entity they indented Alexa to capture. Participants who lacked entity and causal maps knowledge also rated it as a useful Alexa skill. Statistically, we can see in Table. 4.7 that p-value of both Chi-square(0.488 for entity*usefulness and 0.118 for causal map * usefulness) and Fisher's exact test(1.00 for entity * usefulness and 0.308 for causal map * usefulness) is greater than our assumed significance value 0.05. Which supports our $H_0{}^3$ that states that there is no relationship or dependency of knowing what a causal map or entity is to have a positive experience with Build a Model.

4.4.2 Summary of the results

As you can see, all three null hypotheses were conceptualized to see that "there is no significant relationship with any factors to have a positive relationship with Build a Model". It is confirmed to be true with statistical tests. We must note here, however, that this depends on

<table>
<tr><td rowspan="2"></td><td colspan="6">Post-test Q8:
Do you think Build a model
is a useful Alexa skill
that can be used in the future? [Usefulness_Alexa_skill]</td></tr>
<tr><td colspan="2"></td><td>Yes</td><td>No</td><td>Maybe</td><td>Total</td></tr>
<tr><td rowspan="8">Pre-test Q11:
Do you know
what a causal map is?
[Causal_map]</td><td rowspan="4">No</td><td>Count</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>% within Causal_map</td><td>0.0%</td><td>0.0%</td><td>100.0%</td><td>100.0%</td></tr>
<tr><td>% within Usefulness_Alexa_skill</td><td>0.0%</td><td>0.0%</td><td>25.0%</td><td>7.7%</td></tr>
<tr><td>% of Total</td><td>0.0%</td><td>0.0%</td><td>7.7%</td><td>7.7%</td></tr>
<tr><td rowspan="4">Yes</td><td>Count</td><td>8</td><td>0</td><td>3</td><td>12</td></tr>
<tr><td>% within Causal_map</td><td>66.7%</td><td>0.0%</td><td>25.0%</td><td>100.0%</td></tr>
<tr><td>% within Usefulness_Alexa_skill</td><td>88.9%</td><td>0.0%</td><td>75.0%</td><td>92.3%</td></tr>
<tr><td>% of Total</td><td>61.5%</td><td>0.0%</td><td>23.1%</td><td>92.3%</td></tr>
<tr><td colspan="2" rowspan="4">Total</td><td>Count</td><td>9</td><td>0</td><td>4</td><td>13</td></tr>
<tr><td>% within Causal_map</td><td>69.2%</td><td>0.0%</td><td>30.8%</td><td>100.0%</td></tr>
<tr><td>% within Usefulness_Alexa_skill</td><td>100.0%</td><td>0.0%</td><td>100.0%</td><td>100.0%</td></tr>
<tr><td>% of Total</td><td>69.2%</td><td>0.0%</td><td>30.8%</td><td>100.0%</td></tr>
</table>

Table 4.6: Cross-tabulation of whether the participant knew what a *causal map* was with if *Build a model* skill were found to be useful

	Chi-square statistic value	Fisher's statistic value	df(degree of freedom)	p-value (Chi-square)	p-value (Fishers Exact test)
VAPA experience * Usefulness	.410	NA	1	.522	1.000
VAPA experience with skills * Usefulness	.090	NA	1	.764	1.000
Knowledge of Entity * Usefulness	.481	NA	1	.488	1.000
Knowledge of Causal Map * Usefulness	2.438	NA	1	.118	.308

Table 4.7: Chi-square and Fisher's exact test using SPSS software

the user's first reading the user guide, to gain an understanding of the working of the skill. 11 out of the 13 participants rated between 8-10 for their overall experience with Build a Model which was a positive experience. Only one participant gave a low rating of 3, but we noticed that this participant had not gone through the provided user guide even though it was mandatory. This may be one of the reasons why he gave such a low rating. This participant did not provide any comments to support his rating.

4.5 Causal Maps created by participants using Build a Model

As mentioned before, when Alexa completes asking questions or if the user stops the conversation in the middle. The causal map generated during the conversation will be emailed, and the conversation will be stored in DynamoDb. The Figure 4.2 is a screen-shot of the email sent to *datalab.science@gmail.com* which contains congratulatory text, along with the core concept to which the causal map is formed. A list of entities and their related causes is also displayed. An image of the causal map and its causes in CSV format are attached.

Figure 4.3 and Figure 4.4 are few of the maps created by participants during testing. Build a Model. Participants also formed maps for obesity, climate change, hate, water scarcity, fashion, stress, smoking, depression and headache.

In Figure 4.3 there were two layers of questions asked by Alexa. *Drinking alcohol* has been sent to Google Natural Language API and *alcohol* has been identified as the entity, so Alexa has developed a causal map for *alcohol* even though the map was for *drinking alcohol.* There are two layers of questions asked by Alexa which is one for *alcohol* and then questions for the causes answered. Then the participant has requested to stop the questions once the second layer of questions was answered. Even though Build a Model identified *alcohol* as the core problem and proceeded to ask what causes alcohol? The participant knew why it was

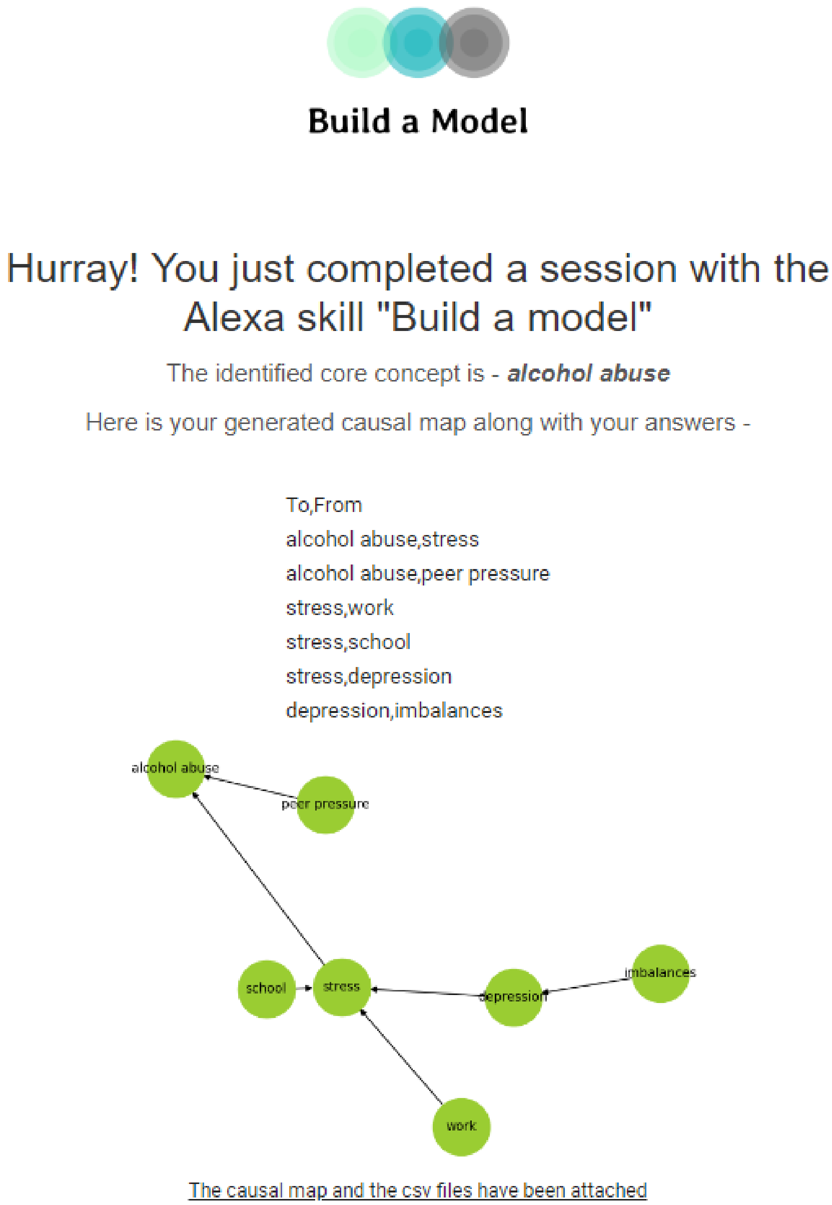

Figure 4.2: Snapshot of the email sent by *Build a model*.

asking such a question and proceed to answer without stopping the skill thinking it was not working properly. The causal map data is stored in JSON format in AWS DynamoDb table. Figure 4.5, 4.6, 4.7 are few of the causal maps generated by the participants during testing.

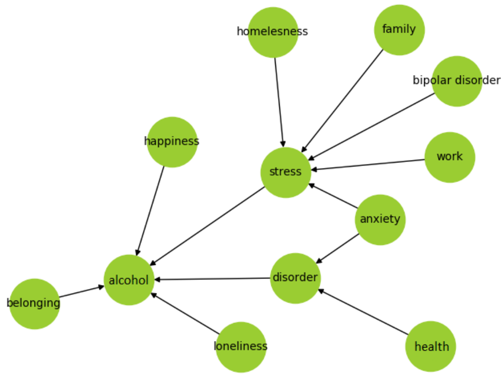

Figure 4.3: One of the participant's causal map created by *Build a model* for *Drinking Alcohol* as the core concept.

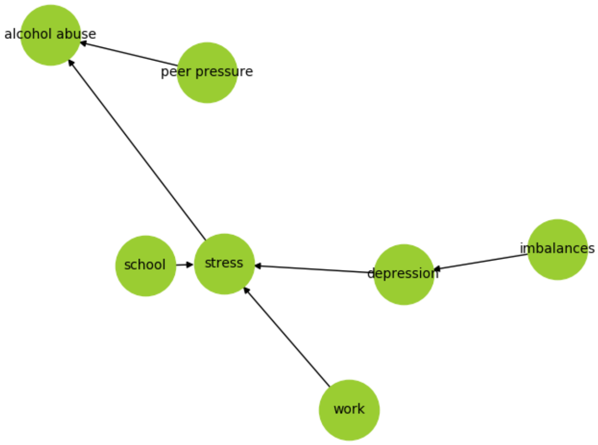

Figure 4.4: One of the participant's causal map created by *Build a model* for *Alcohol Abuse* as the core concept.

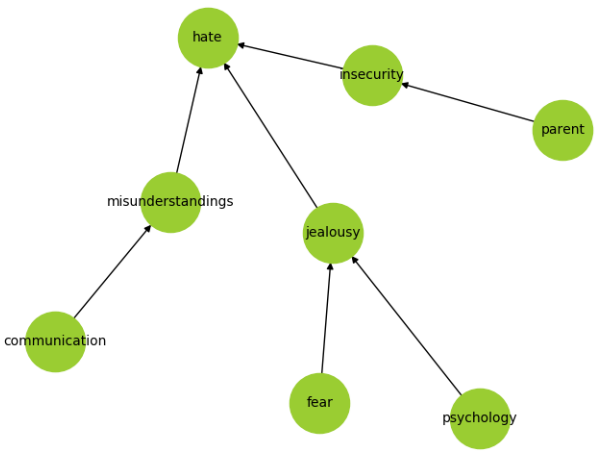

Figure 4.5: One of the participant's causal map created by *Build a model* for *Hate* as the core concept.

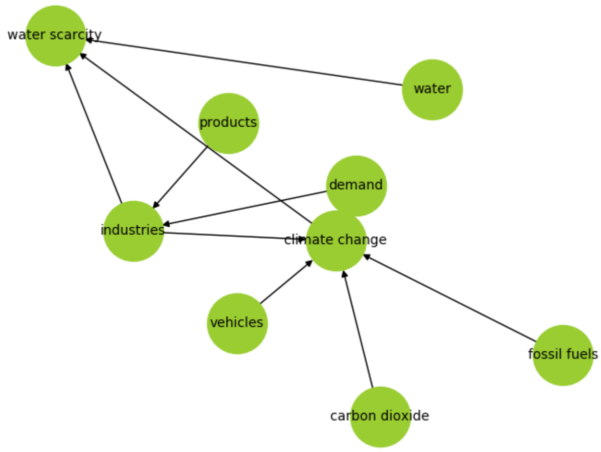

Figure 4.6: One of the participant's causal map created by *Build a model* for *Climate Change* as the core concept.

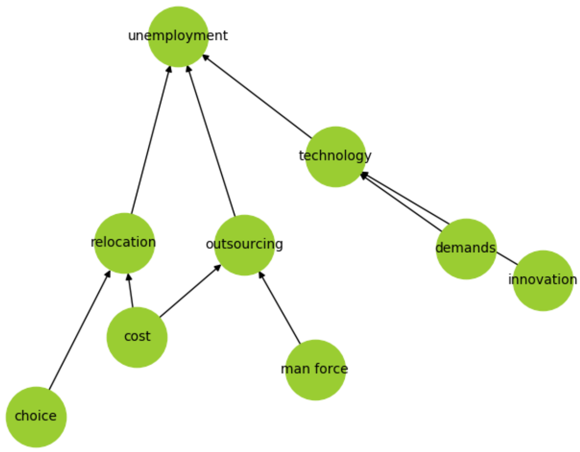

Figure 4.7: One of the participant's causal map created by *Build a model* for *Unemployment* as the core concept.

Chapter 5

Conclusion and Discussion

5.1 Discussion

In collaborative modelling, participants externalize their mental models into various artifacts such as causal maps. This externalization can be guided by a trained facilitator, but there may be associated costs, and availability is limited. Alternatively, free software can be used at any time to create causal maps, but they do not guide participants. Also, neither facilitators nor current software can easily cope with larger causal maps, for instance, to avoid the creation of redundant concepts. To address these limitations, we designed an artificial facilitator that leverages voice-activated technologies. We implemented the prototype via Amazon Alexa and demonstrated its features through 3 case studies as discussed in Chapter 3.

In Chapter 4 we developed a study to see if participants found the artificial facilitator to be useful. In this study, Participants reported that the developed skill — *Build a Model* can successfully capture their thought process of a problem and its causes without the need for some-one to physically be present to guide their conversation. The study proved that the skill we have developed can be used to replace a facilitator in real life. Findings from this experiment prove that VAPA can be used for decision-making during policy formations by externalizing each stakeholders thought process into causal maps. These causal maps can then be analyzed separately for a better understanding as to which policy would best support to address

a problem.

5.2 Limitations

As our system constitutes the first use of voice-activated technologies to build causal maps in participatory modelling, we are at the early stage of building causal maps with VAPA. There are several opportunities to improve the system or address additional research questions in the short- and medium-term. In the short-term, our prototype faces two limitations. First, we used hand-crafted rules, which is more in line with early spoken dialogue systems than with current ones. Other approaches use generative methods (e.g., Bayesian networks) which often involve hand-crafted parameters or discriminative methods where parameters are inferred by machine learning from the data. As stated by Henderson, discriminative machine-learned methods are now the state-of-the-art in dialogue state tracking [55]. However, machine learning requires data to learn from, and there is currently no corpus of the model building involving a facilitator and one participant. Such sessions are often conducted with *many* participants and the recordings are not released as the consent forms generally include an anonymity clause. Designing a better artificial facilitator will thus start by assembling a large set of recordings between a facilitator and a participant, for instance by modelling a system in which participants would be comfortable in publicly sharing their perspectives.

Second, our approach extensively relies on Alexa followed by Google Natural Language API to identify entities. Our prototype struggled with creating causal maps with specialized terms (e.g., from the medical domain) as Alexa could not identify them in speech and the API would not see them as relevant entities. The API may improve over time, and it may also be assisted with ontologies to identify (i) which specialized terms may be used, and (ii) which term is likely to be used following another one. Similarly, improvements in the API would reduce the processing time which currently results in many awkward seconds of silence. We note that improvements in the API or Alexa Skill Kit will automatically benefit the quality of our application, without changes in our design or implementation.

Future researchers may explore how an artificial facilitator can guide in aspects that are necessary yet challenging for trained facilitators. The structure of causal maps is usually analyzed *after they have been built*, for instance by identifying leverage points via centrality [32, 31] or inventorying loops that drive the dynamics of the map [34, 35, 33]. However, a large map of a complex system that contains no loops may already be identified as problematic, suggesting that some causal edges are potentially missing. Consequently, the artificial facilitator can leverage network algorithms to analyze the structure of the map as it is built, thus informing participants of potential issues and approaches to address them. The artificial facilitator can also build on natural language processing in many ways that go beyond the identification of entities. Causal maps sometimes start with a brainstorming process, in which many concepts are generated and then *grouped*. Our artificial facilitator can use the semantic relatedness of concepts to inform the user about potential themes, which may result in combining several overly-detailed concepts into a more abstract category.

Alexa's ability to correctly identify words in any given dialect is the main factor. As people use Alexa more, this will populate Alexa's database which then will increase its ability to identify more complex words. Since voice recognition is a relatively recent introduction, significant improvements in the processing of natural language and related technology in the future should dramatically reduce errors.

During testing of the skill, it was required that the participants go through the user guide provided to them. Understanding *Build a Model's* process is essential for successfully converting user's thoughts into causal maps. As a result, the usage reports of the participants may reflect use in the best case scenario rather than actual use.

5.3 Future work

Build a Model can be used by the government/any company as a survey tool to extract reasoning for a specific problem. Data from a specific location can produce causal maps of an issue that can be used to understand what people at that location think about a particular problem. For example, Anti-vaxxers movement can be analyzed using this tool if people participate in answering Alexa's questions. We can look at questions like - What if we were able to capture what goes on in people's head? and then aim at such people to educate them if they have a wrong understanding of a particular problem's causes.

We noted that for voice recognition, Google Home performs much better than Alexa. We can create an action (similar to Amazon's skill) that will significantly enhance the ability to recognize words spoken by a user and thus identify complex words.

To validate our causal maps with the maps generated by a human facilitator, we can bring in a trained facilitator and have a session with the participants for a particular problem and use the same set of participants to create causal maps using *Build a Model*. We can then compare both, the human facilitator generated map and Alexa's generated map to validate our map creation process.

5.4 Contributions

Creating causal maps using voice conversation has never been done before. We have used the available technologies to create *Build a Model* which can replace a human facilitator in creating a causal map for a particular problem. In short we use Alexa to capture the user's response, converts it to text and this text is converted into causal map.

We performed a test with the participants to see if what we have created is a useful skill that can be used in real life to convert a human's thoughts into causal maps. With the test data we could confidently say that participants who tested the skill found it a useful skill.

Supplementary Materials

Our code is available at https://github.com/datalab-science/causalMapBuilder. Our three case studies, as well as a video overview, can be accessed at https://www.youtube.com/playlist?list=PL7UTR3EL44zrkwrcDkiSwV-7kL0Nv6fQ5

Appendix A

A.1 Background of Technologies

In the modern world of digitization and voice-activated personal activated systems, companies such as Amazon, Apple, Google, and Samsung have introduced their native voice-activated personal assistants(VAPA) such as Alexa, Siri, Google home and S voice. These personal assistants are capable of performing various tasks such as calling a contact, sending a message through voice input, playing music, answering questions related to weather, playing news and so on. We wanted to integrate this technology with the externalization of thoughts of a human. To do this, we have chosen to develop an Alexa skill (Alexa's abilities to perform a task) since the numbers of Alexa devices sold is relatively more compared to Google home [70]. It also provides an easy development platform for developers to develop customized skills.

A.2 Software development frameworks

As you can see in Fig. A.1. We use an Alexa device and Amazon Web Services(AWS) such as S3, Lambda and, DynamoDb. The technologies we have used to develop Build a Model is as shown in Table 2.1. The *skill* is programmed in Python 3.6 using ASK which is hosted on Amazon S3.

A.2.1 Amazon Alexa

Alexa is the cloud-based voice service by Amazon which lets users speak to its devices such as Echo Dot, Amazon Echo and other smart home devices. Alexa comes with more than a few capabilities: music

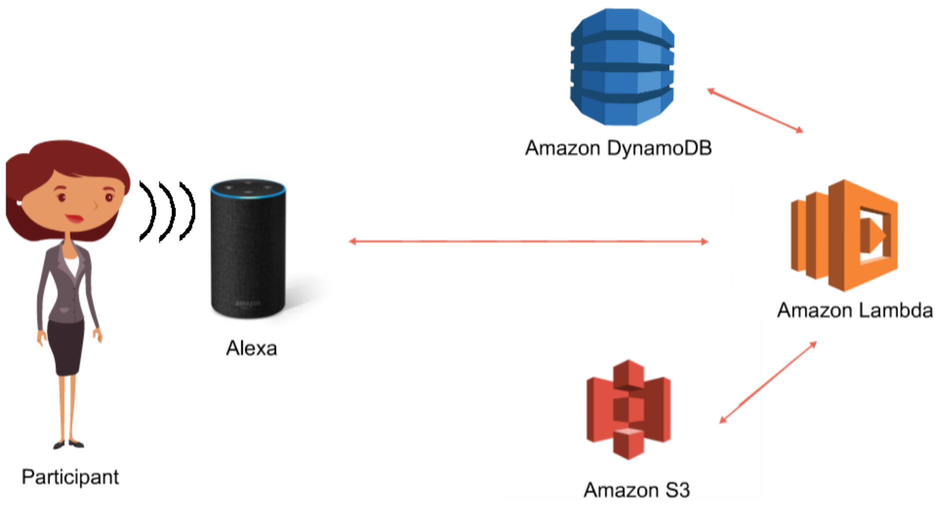

Figure A.1: Interaction of Amazon services with Alexa

playing, weather updates, or even reading the news. These built-in capabilities are limited but can be extended by installing *skill* from Alexa skill store. Alexa *skills* are apps that add even more capabilities to Alexa. Alexa also provides a platform for anyone to build these *skills* that make customers life faster, easier, and more delightful for them. It provides an Alexa Skills Kit (ASK) [71] which is a set of self-service APIs, tools, documentation, and samples of code.

Alexa acts as the front end of an application by being the contact point for the user by converting voice to text (to send the user's response to the back end for processing) and text to voice (to respond to the user's request after processing in the back end). In the Figure A.2 we can see the skill structure created for *Build a Model*. The back end or the end point is the code we have written, which is hosted on AWS Lambda. Alexa requires an intent, slots and sample utterances when a *skill* is created in ASK. Intents are the requests a *skill* can handle; slots are arguments for those intents and sample utterances map the intents to the words and phrases users can say to interact with a *skill*. After much trial and error, we have found that the best way to create an Alexa *skill* for this research is to create just one intent with no slots. By doing this, we force Alexa to trigger a single intent for any response from the user. This means that we can capture anything the user says and apply Natural Language processing to it.

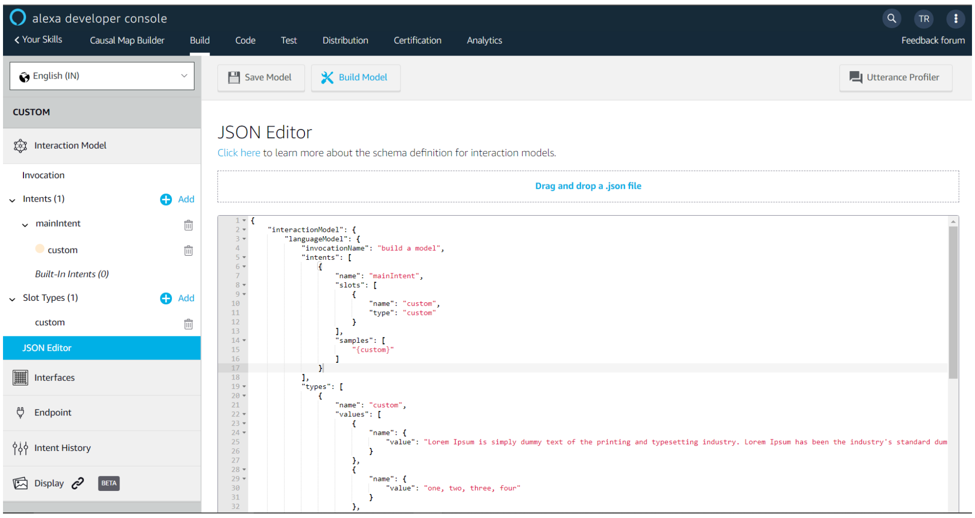

Figure A.2: Screen-shot of Amazon Skill - *Build a Model*

Terminology	Meaning	What we have named when building Alexa skill
Skill	Alexa capabilities are referred to as skills	Build a Model
Invocation name	Name provided by the developer which when used will trigger the corresponding skill	Model Creator
Intent	An intent represents an action that fulfills a user's spoken request. Intents can optionally have arguments called slots.	Main Intent
Slot	Slots are basically variables in utterances. These can have predefined values but are by default empty	Custom Slot

Table A.1: Terminologies used during the development of Alexa skill: *Build a Model*

A.2.2 Amazon Web Services

Amazon Web Services(AWS)[72] is a cloud computing platform from Amazon that provides a metered pay-as-you-go service to individuals, companies or governments. We are using three services of AWS which are - S3 [73], Lambda [74] and DynamoDb [75].

Amazon S3 or Amazon Simple Storage Service is a "simple storage service" that provides storage of objects via a web service interface. It uses the same scalable storage infrastructure that Amazon.com uses to run its global network of e-commerce[73]. A bucket(which are similar to file folders, store objects, which consist of data and its descriptive metadata) must be created in S3 which then holds any type of data. One main thing we noticed is that if the file size is higher than 50 Mb, only Chrome browser will allow uploading files higher than 50 Mb to the S3 bucket. If we use any other browsers like Explorer or Safari, the file will not be uploaded, and S3 will throw an error.

AWS Lambda is an event-driven, serverless computing platform which provides computing services that runs code in response to events and automatically manages the computing resources required by that code[74]. When an event is triggered, a lambda function can be used to run the code which fetches the text from Alexa, process it, and sends back the result in text format to Alexa. In our case, the code hosted on S3 can be executed when the user responds to Alexa's questions. We found that increasing the memory space of the Lambda function to 1024 Mb decreases the response latency to Alexa and setting the timeout to 5 minutes helps the function not to end when processing a significant input which takes more time.

Amazon DynamoDb is a proprietary, fully managed NoSQL database service that supports key value and data structures [76]. The data extracted during a conversation with a user can be stored in a table created in DynamoDb. To capture the successful conversational logs and not so successful conversational logs, we created two tables success and error. In this case, the successful conversation here refers to creating a causal map through the responses of the user; this conversation log is stored in the success table. Sometimes when Alexa does not correctly identify words, and we have not handled it in our code, Alexa throws an error and stops abruptly. That is called unsuccessful conversation because it did not lead to the causal map being created; the conversation log is then stored in the error table. The conversation log includes information such as captured entities, questioned entities, and so on, which will be required in future for analysis.

A.2.3 Google Natural Language API

Google Natural Language API is a powerful pre-trained model developed by Google which allows developers to work with natural language understanding features including sentiment analysis, entity analysis, entity feeling analysis, content classification, and syntax analysis. Because we wanted to create a *skill* that captures any word without any boundaries, we had to zero down on the Google Natural Language API as it was the only API that offered such a vast entity analysis database. Google Natural Language API is queried extensively to find entities. Consider that the artificial facilitator asks “what causes obesity?” and the user responds “I believe that obesity is caused by an excess in eating and not enough exercise” Google Natural Language API will extract the entities obesity, eating, and exercise. Since an answer often includes a repetition of the subject, we automatically ignore user-provided entities that were part of the question. In this example, obesity would be ignored. Thus there are only two new concepts: eating and exercise.

A.3 Integration of technologies

Using ASK, we create a custom *skill* which interacts with the user. We write a program that retrieves from ASK the user’s response, fetches entities using Google Natural Language API, questions the entities to capture their reasoning (as explained in section 1.2.2) and finally converts the entities into causal maps using *NetworkX* library and emails the causal map and the list of entities to *datalab.science@gmail.com*. The code for this will be uploaded to a bucket in S3. Lambda acts as an endpoint for the Alexa *skill* and program which when triggered gets the code from S3. For future analysis of causal maps created during the conversation, DynamoDB is used to store the conversation logs.

We have chosen to integrate the currently best available technologies(instead of developing a new one) in order to see if we can create a voice-activated application that converts thoughts into causal maps.

Appendix B

B.1 Pre-test Questionnaire

This is a pre-test survey that attempts to capture your voice-activated personal assistant (Alexa) knowledge before testing the ability-Build a Model. You must have gone through the user guide to "Build a model" before you take this survey.

The purpose of this survey is to understand your knowledge with - a) Personal Assistant activated by Voice. b) User guide provided to you for "Building a Model." c) Map of the entity and the cause.

== Email address
——————————-

===What is your status?
1) Student
2) Faculty
3) Researcher
4) Other(please specify)——————————-

=== Are you a Native English speaker?
1) Yes
2) No

=== What is your age?
1) 18-24
2) 25-34
3) 35-44

4) 45-54
5) 55-64

=== What is your Gender?
1) Female
2) Male
3) Prefer not to say

=== What is your level of education?
1) High school diploma or equivalent
2) Bachelor's degree
3) Master's degree
4) Doctoral degree

=== Have you used voice-activated personal assistant before (such as Alexa, Google Home/mini, Apple Siri)?
1) Yes
2) No

=== If you have answered yes for the above question. Have you installed any skill for Alexa or any action for Google home/mini?
1) Yes
2) No

=== How comfortable are you with using a voice-activated personal assistant (1 being the lowest and 10 being the highest)?
Rate from 1 to 10 ——————————-

=== Have you read the user guide for "Build a model" skill?
1) Yes
2) No

=== Was the user's guide provided to you easy to understand?
1) Yes
2) No

=== Do you know what an "entity" is?

1) Yes
2) No

=== Do you know what an "causal map" is?
1) Yes
2) No

=== Can you relate a problem with its causes? (For example - Obesity is caused due to stress and overeating)
1) Yes
2) No
3) Maybe

=== Are you confident you can create a causal map successfully using Alexa skill - "Build a model"?
1) Yes
2) No
3) Maybe

=== Do you have any questions, comments or concerns?

B.2 Post-test Questionnaire

This survey is to be taken once you have tested "Build a model" Alexa skill.

Your feedback is essential to us as we hope to use this feedback to improve the skill. This survey aims to capture your experience with "Build a model".

(Please provide the same email address which you have provided in the pre-test questionnaire for "Build a model")

=== Email address

——————————-

=== What was your approach when testing the Alexa skill?
1) Voice
2) Typing

== Did the skill capture all of your answers correctly??
1) Yes
2) No
3) Few

== If you have answered the previous question as ”no” then please provide us with all the answers which you think we’re not captured.?
——————————-

== Were you able to successfully create a causal map with entities? 1) Yes
2) No
3) Maybe

== What was your core concept used to create the causal map (example: Obesity)? ——————————-

== Were you able to understand and answer to all of the questions asked by Alexa??
1) Yes
2) No
3) Few

== How much time did it take you to create the causal map by conversing with Alexa? (Example: 5 minutes)
——————————-

=== Do you think ”Build a model” is a useful Alexa skill that can be used in the future?
1) Yes
2) No

3) Maybe

=== How was your overall experience with "Build a model"? (1 being the lowest and 10 being the highest)
Rate 1- 10
____________________-

=== Do you have any questions, comments or concerns?
____________________-

B.3 User Guide

Figure B.1: User guide of Build a Model provided to the participants before testing - page 1.

"Build a model" is a custom Amazon Alexa skill developed to capture the user's knowledge about any problem (chosen by the user) and its causes by the user's responses to the questions asked by Alexa. The obtained information is then modelled into a causal map.

The way the core concept and its causes are captured is as shown below -

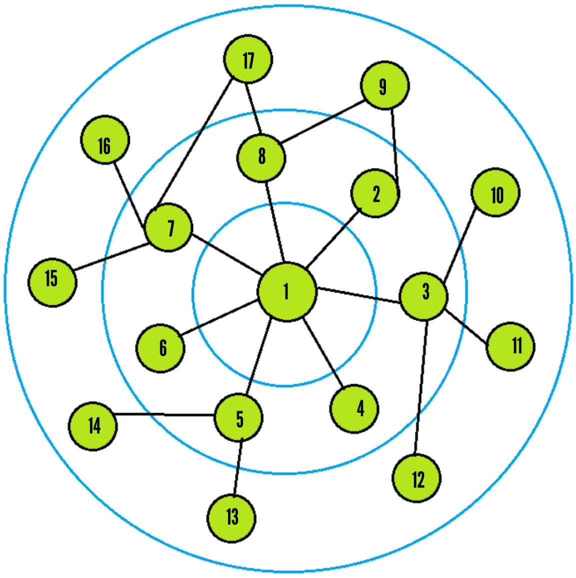

Fig 1: Working of how "Build a model" captures problem and its causes

Figure B.2: User guide of Build a Model provided to the participants before testing - page 2.

Step 1: The core concept of the user's choice is captured. The core concept must be an entity (1 is the core concept in Fig. 1).

Step 2: The causes of the core concept are captured(2,3,4,5,6,7,8 are the causes of 1 in Fig. 1). This is called the first layer.

Step 3: The causes of the causes (2,3,4,5,6,7,8) which are captured in the above step are obtained in the next layer or the second layer. For example causes of 8 are 17 and 9, causes of 3 are 10,11 and 12.

This process continues until the user requests Alexa to stop or till all the causes of every entity are captured.

Don't worry if its too confusing, examples are provided at the end of this document for better understanding.

Figure B.3: User guide of Build a Model provided to the participants before testing - page 3.

If you have never used any voice-activated assistant or don't know the terms - Alexa skill, causal map or entity, then this section is a must read for you. If you know them, then skip ahead to the next section.

1) What is Voice activated personal assistants?
It is a software that lets people give instructions to it through voice rather than traditional ways of input such as typing. The software uses natural language processing to understand your instructions to performs them.

2) What are the different types of voice-activated assistants available in the market?
Amazon Alexa, Google Home/Mini, Siri and Cortona.

3) What is an Alexa skill?
Alexa skill here is like an app on the app store.

4) What is an entity?
An entity is any object we want to model and store information about. Entities are usually recognizable concepts, either concrete or abstract, such as the person, places, things. Example: Obesity, over-eating, stress

5) What is a causal map?
A causal map is a type of concept map in which the links between nodes represent causality or influence.

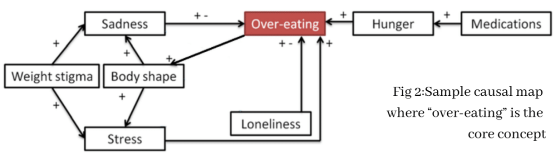

Fig 2:Sample causal map where "over-eating" is the core concept

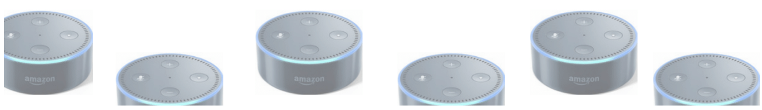

Figure B.4: User guide of Build a Model provided to the participants before testing - page 4.

Instructions to use “Build a model”

1) Triggering the skill
The skill must be triggered through an invocation name. Here the invocation name is “causal map builder.” This name is used to trigger the skill. For example – By saying “Open causal map builder” to Alexa will trigger “Build a model” skill.
(Invocation name: Is the name that Alexa uses to identify and trigger a particular skill. This is similar to opening an app on your phone by selecting it)

2) Answering to Alexa questions.
Relevant answers are to be given to Alexa based on the questions asked. Try to keep your answers short and as relevant as possible.

3) Stopping the session
During the proceedings of a session with Alexa, you can stop the session any time by saying “Alexa stop” or “stop.”

4) Remove an entity
If you want to remove any entity that has been captured by Alexa, you have the option to remove entities by using the keyword - "'remove" followed by the entity. For example - "Remove stress". Here stress is the entity that has to be removed.
(Note: This option is available only when Alexa gives you the option to remove an entity)

Please see the below example of a conversation with the user and “Build a model” skill.

Figure B.5: User guide of Build a Model provided to the participants before testing - page 5.

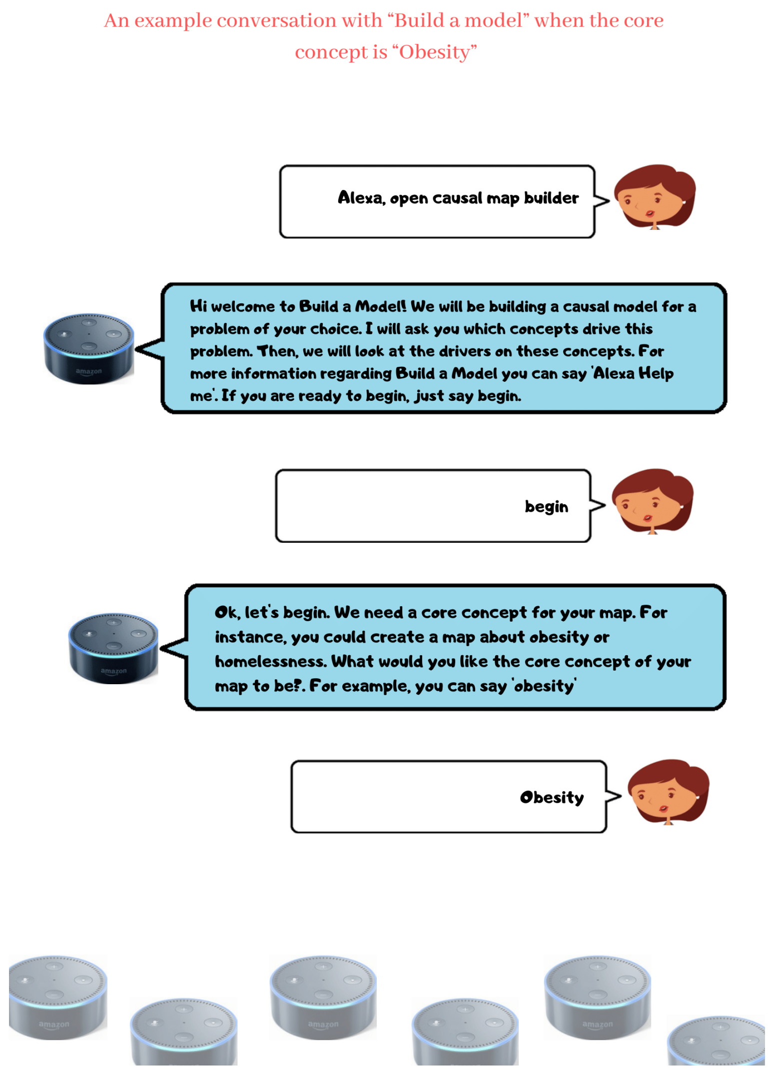

Figure B.6: User guide of Build a Model provided to the participants before testing - page 6.

Appendix C

ASK	Alexa Skill Kit
VAPA	Voice Activated Personal Assistant
NLP	Natural Language Processing
SLU	Spoken Language Understanding
DST	Dialog State Tracker
NLG	Natural Language Generation
CRF	Conditional Random Fields
NER	Named Entity Recognition
LSTM	Long-term memory
CCA	Canonical Correlation Analysis
SNER	Stanford Named Entity Recognizer
TTS	Text-To-Speech
STT	Speech-To-Text
SEM	Structured Equation Models
FCM	Fuzzy Cognitive Maps
PM	Participatory Modelling
M&S	Modelling and Simulation
BM	Build a Model
AWS	Amazon Web Services
DFS	Depth First Search
BFS	Breath First Search
API	Application Program Interface
JSON	JavaScript Object Notation
CSV	Comma-Separated Values

Table C.1: List of abbreviations used

Bibliography

[1] Heitman et al. “Reductionism at the Dawn of Population Health”. In: *Systems Science and Population Health* (2017), 9 to 24.

[2] David S Fink et al. “Wrong answers: when simple interpretations create complex problems”. In: *Systems Science and Population Health* (2017), 25 to 36.

[3] Philippe J Giabbanelli. “Analyzing the Complexity of Behavioural Factors Influencing Weight in Adults”. In: *Advanced Data Analytics in Health*. Springer, 2018, 163 to 181.

[4] Philippe J Giabbanelli and Magda Baniukiewicz. “Navigating complex systems for policymaking using simple software tools”. In: *Advanced Data Analytics in Health*. Springer, 2018, 21 to 40.

[5] Barbara Riley et al. “Systems Thinking and Dissemination and Implementation Research”. In: *Dissemination and Implementation Research in Health: Translating Science to Practice* (2017), p. 143.

[6] Tanner Verigin et al. “Supporting a systems approach to healthy weight interventions in british columbia by modeling weight and well-being”. In: *Proceedings of the 49th Annual Simulation Symposium*. Society for Computer Simulation International. 2016, p. 9.

[7] Laurette Dube et al. “From policy coherence to 21st century convergence: a whole-of-society paradigm of human and economic development”. In: *Annals of the New York Academy of Sciences* 1331.1 (2014), 201 to 215.

[8] Eric A Lavin et al. “Should we simulate mental models to assess whether they agree?” In: *Proceedings of the Annual Simulation Symposium*. Society for Computer Simulation International. 2018, p. 6.

[9] Vishrant K Gupta et al. "An Online Environment to Compare Students' and Expert Solutions to Ill-Structured Problems". In: *International Conference on Learning and Collaboration Technologies*. Springer. 2018, 286 to 307.

[10] Philippe J Giabbanelli et al. "Overcoming the PBL Assessment Challenge: Design and Development of the Incremental Thesaurus for Assessing Causal Maps (ITACM)". In: *Technology, Knowledge and Learning* (2017), 1 to 8.

[11] Alexey Voinov et al. "Tools and methods in participatory modeling: Selecting the right tool for the job". In: *Environmental Modelling & Software* 109 (2018), 232 to 255.

[12] Paul P Maglio et al. "Agent-based models and systems science approaches to public health". In: *American journal of preventive medicine* 40.3 (2011), 392 to 394.

[13] Philippe J Giabbanelli et al. "Using Agent-Based Models to Develop Public Policy about Food Behaviours: Future Directions and Recommendations". In: *Computational and mathematical methods in medicine* 2017 (2017).

[14] Philippe J Giabbanelli et al. "Using visual analytics to support the integration of expert knowledge in the design of medical models and simulations". In: *Procedia Computer Science* 51 (2015), 755 to 764.

[15] H Xue et al. "Applications of systems modelling in obesity research". In: *Obesity Reviews* 19.9 (2018), 1293 to 1308.

[16] Helen de Pinho. "Generation of systems maps: mapping complex systems of population health". In: *Systems Science and Population Health* (2017), 61 to 76.

[17] Philippe J Giabbanelli et al. "Creating groups with similar expected behavioural response in randomized controlled trials: a fuzzy cognitive map approach". In: *BMC medical research methodology* 14.1 (2014), p. 130.

[18] *How to Conduct a Root Cause Analysis*. `https://www.thecompassforsbc.org/how-to-guides/how-conduct-root-cause-analysis`. Accessed: 2019-04-01.

[19] Phil Johnson-Laird. *An Enquiry Concerning Human Understanding*. Oxford University Press, 1748.

[20] Sangeet S. Khemlani et al. "Causal reasoning with mental models". In: *Frontiers in Human Neuroscience* 8 (2014), p. 849. ISSN: 1662-5161. DOI: 10.3389/fnhum.2014.00849. URL: https://www.frontiersin.org/article/10.3389/fnhum.2014.00849.

[21] Mohammed K. Khalil et al. "Applying learning theories and instructional design models for effective instruction". In: *The American Physiological Society* 40.2 (2016), 147 to 156.

[22] *Stanford Encyclopedia of Philosophy*. https://plato.stanford.edu/entries/induction-problem/. Accessed: 2019-04-25.

[23] *The top 9 causes of global poverty*. https://www.concernusa.org/story/top-9-causes-global-poverty/. Accessed: 2019-04-25.

[24] Philippe J Giabbanelli et al. "A fuzzy cognitive map of the psychosocial determinants of obesity". In: *Applied soft computing* 12.12 (2012), 3711 to 3724.

[25] S Zhang et al. "Applications of social network analysis to obesity: a systematic review". In: *Obesity Reviews* 19.7 (2018), 976 to 988.

[26] Diane T Finegood et al. "Implications of the foresight obesity system map for solutions to childhood obesity". In: *Obesity* 18.S1 (2010), S13 to S16.

[27] SA Jebb et al. "Executive summary: foresight tackling obesities: future choices project". In: *Obesity Reviews* 8 (2007), vi to ix.

[28] Lydia Drasic et al. "Exploring the interactions between physical well-being, and obesity". In: *Canadian Journal of Diabetes* 39 (2015), S12 to S13.

[29] Jaimie McGlashan et al. "Comparing complex perspectives on obesity drivers: action-driven communities and evidence-oriented experts". In: *Obesity science & practice* (2018).

[30] Steven Allender et al. "A community based systems diagram of obesity causes". In: *PLoS One* 10.7 (2015), e0129683.

[31] Emily A Knapp et al. "A network approach to understanding obesogenic environments for children in Pennsylvania." In: *Connections (02261766)* 38.1 (2018).

[32] Jaimie McGlashan et al. "Quantifying a systems map: network analysis of a childhood obesity causal loop diagram". In: *PloS one* 11.10 (2016), e0165459.

[33] Agnes Semwanga Rwashana et al. "Advancing the application of systems thinking in health: understanding the dynamics of neonatal mortality in Uganda". In: *Health research policy and systems* 12.1 (2014), p. 36.

[34] Brynle Owen et al. "Understanding a successful obesity prevention initiative in children under 5 from a systems perspective". In: *PloS one* 13.3 (2018), e0195141.

[35] PJ Giabbanelli et al. "developing technology to support policymakers in taking a systems science approach to obesity and well-being". In: *Obesity Reviews* 17 (2016), 194 to 195.

[36] Giabbanelli et al. "Learning Analytics to Support Teachers Assessment of Problem Solving: A Novel Application for Machine Learning and Graph Algorithms". In: *Utilizing Learning Analytics to Support Study Success*. Springer, 2019, 175 to 199.

[37] Manpreet Singh et al. "Building a Cardiovascular Disease predictive model using Structural Equation Model & Fuzzy Cognitive Map". In: *Fuzzy Systems (FUZZ-IEEE), 2016 IEEE International Conference on*. IEEE. 2016, 1377 to 1382.

[38] Simon F Pratt et al. "Detecting unfolding crises with visual analytics and conceptual maps emerging phenomena and big data". In: *Intelligence and Security Informatics (ISI), 2013 IEEE International Conference on*. IEEE. 2013, 200 to 205.

[39] Venkata Sai Pillutla et al. "Iterative generation of insight from text collections through mutually reinforcing visualizations and fuzzy cognitive maps". In: *Applied Soft Computing* (2019).

[40] Sachiko Ozawa et al. "Exploring pathways for building trust in vaccination and strengthening health system resilience". In: *BMC health services research* 16.7 (2016), p. 639.

[41] Jetter et al. "Fuzzy cognitive maps for engineering and technology management: What works in practice?" In: *PICMET* (2006).

[42] Rebecca Jordan et al. "Twelve questions for the participatory modeling community". In: *Earth's Future* 6.8 (2018), 1046 to 1057.

[43] Philippe J Giabbanelli et al. "Feasibility and framing of interventions based on public support: leveraging text analytics for policymakers". In: *International Conference on Social Computing and Social Media*. Springer. 2016, 188 to 200.

[44] Jeffrey R Binder et al. "The neurobiology of semantic memory". In: *Trends in cognitive sciences* 15.11 (2011), 527 to 536.

[45] PJ Lewis. "Rich picture building in the soft systems methodology". In: *European Journal of Information Systems* 1.5 (1992), 351 to 360.

[46] Erdmute Sommerfeld et al. "Operations on cognitive structures to their modeling on the basis of graph theory". In: *Knowledge structures*. Springer, 1994, 151 to 196.

[47] Philip N Johnson-Laird. "Mental models and human reasoning". In: *Proceedings of the National Academy of Sciences* 107.43 (2010), 18243 to 18250.

[48] Vijay K Mago et al. "Fuzzy cognitive maps and cellular automata: An evolutionary approach for social systems modelling". In: *Applied Soft Computing* 12.12 (2012), 3771 to 3784.

[49] Vijay K Mago et al. "Analyzing the impact of social factors on homelessness: a Fuzzy Cognitive Map approach". In: *BMC medical informatics and decision making* 13.1 (2013), p. 94.

[50] Robert Axelrod. "Results". In: *Structure of decisions: The cognitive maps of political elites* (1976), 221 to 250.

[51] Robert Axelrod. "Decision for neoimperialism: The deliberations of the British Eastern Committee in 1918". In: *Structure of decisions: The cognitive maps of political elites* (1976), 77 to 95.

[52] Stanley Ross. "Complexity and the presidency". In: *The Structure of decision: Cognitive maps of political elites* (1976), 96 to 112.

[53] Jason Williams et al. "The dialog state tracking challenge series: A review". In: *Dialogue & Discourse* 7.3 (2016), 4 to 33.

[54] Liliana Laranjo et al. "Conversational agents in healthcare: a systematic review". In: *Journal of the American Medical Informatics Association* 25.9 (2018), 1248 to 1258.

[55] Matthew Henderson. "Machine learning for dialog state tracking: A review". In: *Proc. of The First International Workshop on Machine Learning in Spoken Language Processing*. 2015.

[56] Dr. M. Humera Khanam et al. "Named Entity Recognition using Machine Learning Techniques for Telugu language". In: *IEEE* (2016).

[57] Meizhi Ju et al. "A Neural Layered Model for Nested Named Entity Recognition". In: *Association for Computational Linguistics* NAACL-HLT (2018), 1446 to 1459.

[58] Arvind Neelakantan et al. "Learning Dictionaries for Named Entity Recognition using Minimal Supervision". In: *European Chapter of the Association for Computational Linguistics* 14 (2014), 452 to 461.

[59] *Stanford Named Entity Recognizer (NER)*. `https://nlp.stanford.edu/software/CRF-NER.html`. Accessed: 2018-08-27.

[60] *Parallel Dots (NER)*. https://www.paralleldots.com/.

[61] *Google Natural Language API*. `https://cloud.google.com/natural-language/`. Accessed: 2018-08-27.

[62] Kristen Hassmiller Lich et al. "Extending systems thinking in planning and evaluation using group concept mapping and system dynamics to tackle complex problems". In: *Evaluation and program planning* 60 (2017), 254 to 264.

[63] Leah Frerichs et al. "Mind maps and network analysis to evaluate conceptualization of complex issues: A case example evaluating systems science workshops for childhood obesity prevention". In: *Evaluation and program planning* 68 (2018), 135 to 147.

[64] Steven Gray et al. "The structure and function of angler mental models about fish population ecology: the influence of specialization and target species". In: *Journal of Outdoor Recreation and Tourism* 12 (2015), 1 to 13.

[65] Deanna R Befus et al. "A Qualitative, Systems Thinking Approach to Study Self-Management in Women With Migraine". In: *Nursing research* 67.5 (2018), 395 to 403.

[66] Jennifer Yourkavitch et al. "Interactions among poverty, gender, and health systems affect women's participation in services to prevent HIV transmission from mother to child: A causal loop analysis". In: *PloS one* 13.5 (2018), e0197239.

[67] *Tutorial to use NetworkX*. `https://networkx.github.io/documentation/stable/tutorial.html`. Accessed: 2018-08-27.

[68] Aric Hagberg et al. "Exploring network structure, dynamics, function using networkx". In: *Exploring network structure, dynamics, function using networkx* 40.2 (2016), 147 to 156.

[69] Drew Harwell. "The accent gap". In: *The Washington Post* (2018).

[70] *Amazon says 100 million alexa devices have been sold, what is next?* `https://www.theverge.com/2019/1/4/18168565/amazon-alexa-devices-how-many-sold-number-100-million-dave-limp`. Accessed: 2019-03-27.

[71] *Alexa skills kit*. `https://developer.amazon.com/alexa-skills-kit`. Accessed: 2018-08-27.

[72] *Alexa skills kit*. `https://developer.amazon.com/alexa-skills-kit`. Accessed: 2018-08-27.

[73] *Amazon Simple storage device*. `https://aws.amazon.com/s3/`. Accessed: 2018-08-27.

[74] *AWS Lambda*. `https://aws.amazon.com/lambda/`. Accessed: 2018-08-27.

[75] *AWS DynamoDb*. `aws.amazon.com/dynamodb/`. Accessed: 2018-08-27.

[76] *AWS DynamoDb*. `https://aws.amazon.com/dynamodb/faqs/`. Accessed: 2018-08-27.

[77] Castellani et al. "Map of complexity science". In: *Art & Science Factory* (2009).

[78] Tatiana Lukoianova et al. "Veracity roadmap: Is big data objective, truthful and credible?" In: *Advances In Classification Research Online* (2014).

[79] Canalys. *Amazon reclaims top spot in smart speaker market in Q3 2018*. 2018. URL: https://www.canalys.com/newsroom/amazon-reclaims-top-spot-in-smart-speaker-market-in-q3-2018.

[80] *Create the Interaction Model for Your Skill*. https://developer.amazon.com/docs/custom-skills/create-the-interaction-model-for-your-skill.html. Accessed: 2018-08-27.

[81] *SPSS IBM Software*. https://www.ibm.com/analytics/spss-statistics-software. Accessed: 2019-04-25.

[82] *Network X*. https://networkx.github.io/. Accessed: 2019-04-25.

[83] Manpreeth Singh and other. "Building a Cardiovascular Disease predictive model using Structural Equation Model amp; Fuzzy Cognitive Map". In: *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. July 2016, pp. 1377–1382. DOI: 10.1109/FUZZ-IEEE.2016.7737850.

[84] Andrew Hepner et al. "Automating Articulation: Applying Natural Language Processing to Post-Secondary Credit Transfer". In: *IEEE Access* 7 (2019), pp. 48295–48306. ISSN: 2169-3536.

[85] Atish Pawar et al. "Challenging the Boundaries of Unsupervised Learning for Semantic Similarity". In: *IEEE Access* 7 (2019), pp. 16291–16308. ISSN: 2169-3536.

[86] Sahib Singh Budhiraja. "Extracting specific text from documents using machine learning algorithms". dissertation. Lakehead Univeristy, 2009.

[87] Atish Pawar et al. "Calculating the similarity between words and sentences using a lexical database and corpus statistics". In: *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING* (2018).

[88] Sahib Singh Budhiraja et al. "A Supervised Learning Approach For Heading Detection". In: *Cornell University* (2018).

[89] Zoumpolia Dikopoulou et al. "A new approach using mixed graphical model for automatic design of fuzzy cognitive maps from ordinal data". In: *IEEE Access* (2017). DOI: 10.1109/FUZZ-IEEE.2017.8015607.

[90] Vijay Mago et al. "Analyzing the impact of social factors on homelessness: a fuzzy cognitive map approach." In: *BMC Medical Informatics and Decision Making* (2013). DOI: `https://doi.org/10.1186/1472-6947-13-94`.

[91] Vijay Mago et al. "Fuzzy cognitive maps and cellular automata: An evolutionary approach for social systems modelling". In: *Applied Soft Computing* 12 (2012), 3771 to 3784.

[92] Vijay Mago et al. "Supporting meningitis diagnosis amongst infants and children through the use of fuzzy cognitive mapping". In: *BMC Medical Informatics and Decision making* 12 (2012), p. 98.