

# **The Virtual Suspect Meets Alexa**

Designing a Brand New Natural Language Interaction

**Gonçalo Miguel da Silva Baptista**

Thesis to obtain the Master of Science Degree in

**Information Systems and Computer Engineering**

Supervisor: Prof. Rui Filipe Fernandes Prada

## **Examination Committee**

Chairperson: Prof. Francisco António Chaves Saraiva de Melo

Supervisor: Prof. Rui Filipe Fernandes Prada

Member of the Committee: Prof. Maria Luísa Torres Ribeiro Marques da Silva Coheur

**October 2020**



# Acknowledgments

First of all, I want to thank Diogo and Rui, who made this work possible. Not only for creating the Virtual Suspect and laying the foundation of the work, but also for all the guidance, help and support they provided through this last year and a half.

Secondly, I want to thank my family, for all the love and support they have given me throughout the years, for trusting me and letting me forge my own path.

I want to thank Ana, for standing beside me on this journey, for all the love, support, laughter, memories and everything else you have given me. I want to thank Beatriz, Alexandra, Joana, Francisco, Pedro, Manuel, Tomás, Nádia, Catarina, Francisco, and everyone else at CPLEIC, for all the joy, laughter, melancholy, tears, sweat, and all the memories we have made. I will forever cherish the bonds we have made, and I am proud to call you my home and my family.

Thank you all for carrying me across this finish line, I could not have done it without you.



# Abstract

Interactive Agents need a good interaction in order to showcase their abilities and fulfill their roles. Rato et al. created a Virtual Suspect capable of lying, but its interaction was limited. We took advantage of the tools provided by Amazon Alexa to create a new natural language conversational interaction with the Virtual Suspect. We used an iterative, user-centered approach when designing the new interaction, collecting feedback and data from User Studies in order to improve the interaction with the Virtual Suspect. After we managed to create a good interaction we did another User Study with the original lying algorithm and concluded it still needs improving.

## Keywords

Virtual Suspect; Alexa; Natural Language; Conversational Agent; Conversational Interaction

# Resumo

Agentes Interativos precisam de uma boa interação para demonstrar as suas capacidades e desempenhar o seu papel. Rato et al. criaram um Suspeito Virtual capaz de mentir, mas a interação era limitada. Nós aproveitámos as ferramentas disponibilizadas pela Amazon Alexa para criar uma nova interação conversacional em língua natural com o Suspeito Virtual. Usámos uma abordagem iterativa e centrada em utilizadores para criar a nova interação, recolhendo feedback e dados através de Estudos com Utilizadores para melhorar a interação com o Suspeito Virtual. Depois de criarmos uma boa interação, fizemos outro Estudo com o algoritmo da mentira original e concluímos que este ainda precisa de ser melhorado.

## Palavras Chave

Suspeito Virtual; Alexa; Língua Natural; Agente Conversacional; Interação Conversacional

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Motivation . . . . .	3
1.2	Problem . . . . .	4
1.3	Objective . . . . .	4
1.4	Document outline . . . . .	4
<b>2</b>	<b>Related Work</b>	<b>5</b>
2.1	Virtual Suspects . . . . .	6
2.1.1	Psychological Based Virtual-Suspect for Interrogative Interview Training . . . . .	6
2.1.2	A Virtual Suspect Agent's Response Model . . . . .	7
2.2	Conversation Agents . . . . .	7
2.2.1	Conversational Agents for Game-Like Virtual Environments . . . . .	7
2.2.2	An Embodied Conversational Agent for Interactive Videogame Environments . . . . .	8
2.2.3	PICA: Proactive Intelligent Conversational Agent for Interactive Narratives . . . . .	9
2.3	Interrogation in Video Games . . . . .	10
2.4	Alexa Prize . . . . .	11
<b>3</b>	<b>Background</b>	<b>12</b>
3.1	Virtual Suspect . . . . .	13
3.1.1	Story Representation . . . . .	13
3.1.2	Interaction . . . . .	15
3.1.3	Architecture . . . . .	16
3.1.4	Use Case: Peter Barker . . . . .	17
3.1.5	Prototype . . . . .	18
3.1.6	Implementation . . . . .	20
3.2	Amazon Alexa . . . . .	21
3.2.1	Interaction Model . . . . .	22
3.2.2	Skill Service . . . . .	25

<b>4</b>	<b>Solution Architecture</b>	<b>26</b>
4.1	A Virtual Suspect Skill . . . . .	27
4.1.1	Interaction Model . . . . .	27
4.1.2	Skill Service . . . . .	28
4.1.3	Architecture . . . . .	28
4.2	Implementation Plan . . . . .	29
<b>5</b>	<b>Development</b>	<b>31</b>
5.1	First Steps . . . . .	32
5.1.1	Recreating the Original Functionality . . . . .	32
5.1.2	Preparing for the First User Study . . . . .	36
5.1.3	First User Study . . . . .	38
5.2	Improving the Interaction . . . . .	38
5.2.1	Virtual Suspect Response Model . . . . .	39
5.2.2	Virtual Suspect Story . . . . .	40
5.2.3	Natural Language Generator . . . . .	42
5.2.4	Interaction Model . . . . .	42
5.2.5	Skill Service . . . . .	43
5.2.6	Second User Study . . . . .	44
5.3	Last Adjustments . . . . .	45
5.3.1	Turning On the Lie . . . . .	46
5.3.2	Third User Study . . . . .	46
<b>6</b>	<b>User Studies</b>	<b>47</b>
6.1	First User Study . . . . .	48
6.1.1	Objectives . . . . .	48
6.1.2	Scenario . . . . .	48
6.1.3	Materials . . . . .	50
6.1.3.A	Preamble . . . . .	50
6.1.3.B	Annotations . . . . .	51
6.1.4	Procedure . . . . .	53
6.1.5	Results . . . . .	54
6.1.6	Discussion . . . . .	55
6.2	Second User Study . . . . .	56
6.2.1	Objectives . . . . .	56
6.2.2	Scenario . . . . .	56
6.2.3	Materials . . . . .	57



6.2.3.A	Instructions . . . . .	57
6.2.3.B	Questionnaire . . . . .	58
6.2.3.C	Annotations . . . . .	61
6.2.4	Procedure . . . . .	61
6.2.5	Results . . . . .	62
6.2.6	Discussion . . . . .	64
6.3	Third User Study . . . . .	64
6.3.1	Objectives . . . . .	65
6.3.2	Scenario . . . . .	65
6.3.3	Materials . . . . .	65
6.3.3.A	Instructions . . . . .	65
6.3.3.B	Questionnaire . . . . .	66
6.3.3.C	Annotations . . . . .	66
6.3.4	Procedure . . . . .	68
6.3.5	Results . . . . .	68
6.3.6	Discussion . . . . .	70
<b>7</b>	<b>Discussion</b>	<b>72</b>
7.1	Limitations . . . . .	73
7.1.1	Interaction Model . . . . .	73
7.1.2	Virtual Suspect Architecture . . . . .	74
7.1.3	Lying Algorithm . . . . .	74
7.2	Results . . . . .	75
<b>8</b>	<b>Conclusion</b>	<b>76</b>
8.1	Future Work . . . . .	78
<b>A</b>	<b>User Study Materials</b>	<b>82</b>
A.1	Instructions . . . . .	82
A.1.1	Second User Study . . . . .	82
A.1.2	Third User Study . . . . .	83
A.2	UEQ Results . . . . .	84
A.2.1	Second User Study . . . . .	84
A.2.2	Third User Study . . . . .	84
A.3	Conversation Flow Results . . . . .	84
A.3.1	Second User Study . . . . .	84
A.3.2	Third User Study . . . . .	84
A.4	Questionnaire for the Third Study . . . . .	84

# List of Figures

3.1	Entity Examples . . . . .	13
3.2	An event with references to the relevant entities . . . . .	14
3.3	Query Examples . . . . .	15
3.4	Virtual Suspect Architecture . . . . .	16
3.5	First Virtual Suspect Prototype [1] . . . . .	18
3.6	Second Virtual Suspect Prototype [1] . . . . .	18
3.7	Virtual Suspect Implementation . . . . .	20
3.8	Workflow of an Alexa Skill . . . . .	21
3.9	Alexa Developer Console . . . . .	22
3.10	Intents and Responses . . . . .	23
3.11	The Utterance Profiler in the Developer Console allows us to see the identified intent for a specific phrase, and other possible intents . . . . .	23
3.12	Slots . . . . .	24
4.1	Solution Architecture . . . . .	29
5.1	Entity Distribution in Events . . . . .	37
5.2	Knowledge Roles in Entities . . . . .	41
6.1	Question Classification System . . . . .	52
6.2	Answer Classification System . . . . .	53
6.3	UEQ Scales and Items [2] . . . . .	59
6.4	UEQ Benchmarks for Second Study . . . . .	63
6.5	UEQ Benchmarks for Third Study . . . . .	69
6.6	Comparison of the UEQ Results . . . . .	71
A.1	Mean Value of each UEQ Item for the Second Study . . . . .	85
A.2	Means and Confidence Intervals of UEQ Scales for the Second Study . . . . .	85

A.3	Mean Value of each UEQ Item for the Third Study . . . . .	86
A.4	Means and Confidence Intervals of UEQ Scales for the Third Study . . . . .	86
A.5	Mean Value of each Conversation Flow Question for the Second Study . . . . .	87
A.6	Means and Confidence Intervals of Conversation Flow Scales for the Second Study . . . . .	87
A.7	Mean Value of each Conversation Flow Question for the Third Study . . . . .	88
A.8	Means and Confidence Intervals of Conversation Flow Scales for the Third Study . . . . .	88
A.9	First Page of Questionnaire for the Third Study . . . . .	89
A.10	Second Page of Questionnaire for the Third Study . . . . .	90
A.11	Third Page of Questionnaire for the Third Study . . . . .	91
A.12	Fourth Page of Questionnaire for the Third Study . . . . .	92
A.13	Fifth Page of Questionnaire for the Third Study . . . . .	93
A.14	Sixth Page of Questionnaire for the Third Study . . . . .	94
A.15	Seventh Page of Questionnaire for the Third Study . . . . .	95
A.16	Eighth Page of Questionnaire for the Third Study . . . . .	96
A.17	Ninth Page of Questionnaire for the Third Study . . . . .	97

## List of Tables

5.1	Original Questions and corresponding Queries . . . . .	33
5.2	Streamlined questions . . . . .	34
5.3	Translation to the Interaction Model . . . . .	36
6.1	Exchange Classification . . . . .	54
6.2	Question Problems Breakdown . . . . .	54
6.3	Answer Problems Breakdown . . . . .	54
6.4	Conversation Flow Evaluation Questions . . . . .	60
6.5	Question Problems Breakdown . . . . .	62
6.6	Answer Problems Breakdown . . . . .	62
6.7	New Conversation Flow Evaluation Questions . . . . .	67
6.8	Question Problems Breakdown . . . . .	69
6.9	Answer Problems Breakdown . . . . .	69

# Listings

3.1 Excerpt of the JSON sent by the Skill . . . . .	25
3.2 Skill Response . . . . .	25

# Acronyms

<b>ASK</b>	Alexa Skills Kit
<b>AWS</b>	Amazon Web Services
<b>DLL</b>	Dynamic-link library
<b>HTTP</b>	Hypertext Transfer Protocol
<b>JSON</b>	JavaScript Object Notation
<b>UEQ</b>	User Experience Questionnaire
<b>UX</b>	User Experience
<b>WOZ</b>	Wizard-of-Oz

# 1

## Introduction

### Contents

---

1.1 Motivation . . . . .	3
1.2 Problem . . . . .	4
1.3 Objective . . . . .	4
1.4 Document outline . . . . .	4

---

Interactive Agents can perform a wide variety of roles in our modern world. You can have an Interactive Agent in a video game, in a museum, in an online shop, or in a classroom. Systems like this can be a useful teaching tool because they are less expensive, more accessible and offer increased control of the environment [3]. Whatever their function or context, the quality of an Interactive Agent depends mainly on the quality of its decision-making and of its interaction. An agent can have very efficient and complex programming but if it cannot establish communication with an user, it will not serve its purpose. Conversely, you can have an agent with the most amazing natural language model and an internal model that does not support the rich interaction mechanisms. An Interactive Agent needs both good functioning to fulfill its role (whether as a museum guide or a shopping assistant), and good interaction to be able to understand and be understood (whether by a visual interface or natural language communication).

These Interactive Agents can sometimes be used for training a specific role or job. One such example is a Virtual Suspect [4,5], an agent that inhabits the role of a suspect in a police interrogation, that can be used to train police in interrogative techniques. The same technique can also be used for entertainment purposes as part of an investigative video game.

Another technology that has been consistently evolving is Natural Language Understanding and Voice Interaction, with Voice Assistants becoming more and more ubiquitous, being present in our homes and phones. One example of this is the Alexa, a voice assistant developed by Amazon, which is not only an extremely versatile voice assistant, but it also allows users to create third-party applications and functionality. It is now easier than ever for people to create interactive and conversational agents.

## 1.1 Motivation

In 2016, Rato et al. [1, 6] designed and developed a model of a Virtual Suspect with the ability to autonomously create parallel stories to the one initially stored in its memory, and thus allowing it to lie. Most research of this work centered around how to model the agent's memory and its interaction with the user through questions and answers, in a way that would allow it to lie on the spot, without pre-programmed responses. In order to test their Virtual Suspect, they developed a simple and limited visual interface that allowed users to select questions to ask the Virtual Suspect.

In their conclusion, Rato et al. [1] posited that their interface was too limited and that an approach using Natural Language Processing would highly improve the interaction between the user and the Suspect.

## 1.2 Problem

The original interaction with the Virtual Suspect [1] was too limited. It only had a pre-defined small number of questions that users could choose from, and did not allow them to ask anything else. It did not cover or showcase the full capabilities of the Virtual Suspect model designed by Rato et al., as it omitted certain types of questions altogether. It did not have a sense of progression or finality, as the users could only ask the same questions over and over again. The order did matter, and sometimes the answers could change, indicating that the agent had previously lied, but there was no natural conversation flow.

## 1.3 Objective

Our goal is to create a better interaction with the Virtual Suspect designed by Rato et al. [1], by creating a new Natural Language conversational interaction. We will take advantage of the tools provided by Amazon Alexa to create that new interaction. We want to provide a better User Experience (UX) for interacting with the Virtual Suspect, by creating a natural and fluid conversation with the agent, where users can naturally flow from one question to the next at their own pace. Our interaction will give users freedom in interacting with the Virtual Suspect, and will fully showcase all its the capabilities. We want to create an interesting, meaningful and user-driven interaction with the Virtual Suspect.

## 1.4 Document outline

In Chapter 1 we have the motivation and goals behind our work with the Virtual Suspect [1] and Alexa. Chapter 2 presents other works with some relation and similarities to our own. We look at other Virtual Suspects, Conversational Agents, Interrogation in Video Games, and the Alexa Prize.

Chapter 3 expands upon the two concepts introduced in this chapter, the Virtual Suspect and the Amazon Alexa. We analyse how the Virtual Suspect was designed and developed, and we explain what Alexa is and how the tools that will allow us to create our new interaction work.

In Chapter 4 we combine the information learned in Chapter 3 to define our solution. We also explain the process by which we built our new interaction, which is defined in Chapter 5 and Chapter 6. We followed a user-centered approach with which we iteratively improved the interaction with the Virtual Suspect using feedback from User Studies. Chapter 5 details all the work and changes we made to the interface and the Virtual Suspect, and Chapter 6 details the three User Studies that were made during that development.

Chapter 7 contains a discussion of the limitations we faced while developing the new Virtual Suspect interaction and an analysis of our results. Chapter 8 contains our conclusions and ideas for the future.



# 2

## Related Work

### Contents

---

2.1 Virtual Suspects . . . . .	6
2.2 Conversation Agents . . . . .	7
2.3 Interrogation in Video Games . . . . .	10
2.4 Alexa Prize . . . . .	11

---

In this section we will review works from two different academic areas. We will study existing models for Virtual Suspects, and we will take a look at Conversational Agents. We will also take a look at video game mechanics related to our problem, and at the conversational agents produced in the context of the Alexa Prize competition.

## 2.1 Virtual Suspects

### 2.1.1 Psychological Based Virtual-Suspect for Interrogative Interview Training

This paper [5] is similar to our work in the sense that it is a virtual suspect, an agent designed to play the role of a suspect in a police interview. This work is based more on real police cases and focuses more on the psychological and emotional state of the suspect, having the ability to model the personality of the suspect before the interview begins, which influences how the agent will respond to the questions. The questions and answers themselves are made according to defined templates, and all the responses the suspect can give are previously set. It has a long and a short term memory, as well as an internal state vector. The long and short term memory are used to fill the response templates, the long term memory with facts from the database, like personal information and past events; the short term memory to determine if a question is a follow-up. The internal state vector saves how the agents responds to the questions asked.

One interesting aspect of this paper is the methodology they used to test their virtual suspect. The authors compare it both with a human performance and a random response selection mechanism. First they had three subjects interact with the Virtual Suspect system, one had the responses selected by a human, another by their Response Model, and the other had the responses randomly selected. They collected the transcripts of the three interactions and showed them to 24 other participants, and asked whether the participants thought the answers had been selected by a computer or a human, as well as other similar questions/statements. The results show that their response model performed very similarly to the human interaction, and both the human and their response model performed better than the random response selector. This indicates that their response model behaves in a similar manner to a human.

The idea of testing the response model against a human performer and a random agent is an interesting one, especially to show that the response model behaves similarly to the human, even if it is not the most relevant to us, since we are going to be mainly working on the interaction of the Virtual Suspect and not the Response Model. Still, the approach followed can be adjusted to evaluate the conversation's flow in a similar manner.

### **2.1.2 A Virtual Suspect Agent's Response Model**

This paper [4] focuses on creating a Virtual Suspect with the express purpose of training police officers, and the focus of the work is creating an agent that is able to portray different personas and having the users try to identify different personalities. The authors propose a model for intelligent agents with three main components: the ability to interpret the actions of the user (e.g. the user says "Confess, criminal!" which is interpreted in the abstract terms *dominant* and *aggressive* behaviour), a response model that provides the reasoning of the agent based on the input of that interpretation (e.g. the user is dominant and aggressive which makes me sad and angry), and a selection of the most appropriate behaviour in the agent's repertoire, based on the state of the response model (e.g. the abstract state of the response model is sad and angry, so make a sad face and say "You're not nice!"). They developed a response model that can *play* a suspect that has a "personality" (a persona). It simulates a persona and models the interpersonal aspects of an interaction in an abstract manner. It calculates the interpersonal properties that the response of the subject should have, based on the interpretation of the contribution by the user. They base a lot of their work on the theories and techniques behind real police interrogations.

The focus of this work is mostly around the agent being able to play different personalities. The authors include a lot of variables into their model that help build slight differences across the multiple personalities. They have access to a corpus of police interviews, and since the main focus is the training of police officers, the authors strive for accuracy. In order to select an appropriate response from the available actions, they break down both the inquiry and the personality of the agent into several factors. This, combined with all the details of the agent's model, allows for the selection of the most adequate response to the inquiry. Even though they include a lot of variables and details in their model, they do not go into detail what those possible actions are and how they are defined. That is, they do not explain how the agent lies and how it handles lying for example.

So, this work is similar to ours in the sense that it is a Virtual Suspect agent, but their approach is different, focusing more on modelling different personalities for the agent than the process of coming up with lies. Still, the three main components that they propose provide a good baseline for how an agent should behave: first, interpret the question, second, update out beliefs according to what the user said, and third, come up with an answer.

## **2.2 Conversation Agents**

### **2.2.1 Conversational Agents for Game-Like Virtual Environments**

In this paper [7], Morris proposes a model for a conversational agent that incorporates shallow models of emotion and personality to create a virtual character capable of believable conversational behavior,

all inside a Cluedo-style game. An important part of this work is a conversational agent architecture that builds on an existing natural language generation system to incorporate conversational behaviours that are dependent upon the agent's personality, current emotional state and the agent's beliefs about the social relationship between the conversants. The key development of this architecture is the introduction of a "Language Use Manager", which determines the relationship between the emotional status and personality type of the agent's character and the behavioural choices made in generating natural language. Each character has an episodic memory structure, as well as unique models of personality, emotion, temperament and social role.

While Temperament and Social Role do not seem to be as relevant to our work, the Personality Model and Emotional Model (as well as the Emotional Affect Manager, which determines exactly how each interaction affects the agent's emotional state) seem like ideas more useful to us. Having certain personality traits and tracking the emotional state of the agent could affect how he responds to certain interactions with the user. The previously mentioned Language Use Manager is a rule-based module that determines what the agent says, such as word choice, sentence structure and turn taking, which includes deciding when to reveal or hide information, as well as change the subject. The agent also keeps track of the emotional state of the other character he is talking with (the user) and uses that in the process of deciding what to say. This creates a model of what the agent knows of the user and what it knows that the user knows.

Unfortunately this is all just a proposal and no work has actually been implemented, so there are neither results to analyze nor examples to base ourselves on. Still, the idea presented is solid and can be useful when designing our agent.

## **2.2.2 An Embodied Conversational Agent for Interactive Videogame Environments**

This paper [8] is about an Embodied Conversational Agent in the Half-Life domain and discusses how it handles commands by the user and how it handles the identification of entities known as Referring Expressions. For example, if the user says "get the gun", the agent has to know which gun the user is referring to. Existing techniques that deal with reference resolution deal only with entities in the discourse context, that is entities that have been mentioned in the conversation with the user. But in this work, they want to be able to reference all kinds of entities, not just ones mentioned in the conversation. Entities such as objects on the user's, the agent's, or both, field of view; items in the agent's inventory; actions that the agent has previously performed; etc.

They start with building a thorough context, considering everything that surrounds the agent, its inventory, the user's field of view, etc. Then, every action is mapped to a competence module, which contain the primitive action, a list of preconditions, an add list, a delete list and a language module. In

the current implementation, the conversational actions only happen at the same time as other actions, such as confirming an action. When the user asks the agent to do something, first it parses the utterance and identifies which action and target the user is talking about. When that is identified, the correct competence module is selected. If all the preconditions are met, then the action executes. If not, the system propagates the competence module and tries to fulfill all the preconditions. So, for example, if the user asks the agent to “kill the enemy”, first the agent identifies the action as KILL(OBJECT ENEMY), then the agent activates the KILL\_ENEMY module, the KILL\_ENEMY module has the precondition OWNS\_GUN, if the agent does not have a gun, it will seek to fulfill that by activating the GET\_GUN module, and so on, until all conditions are met.

Although the specifics and domain of this work are not very relevant to our work, what is relevant is the importance of modeling the context, so that the user can use pronouns and Referring Expressions to refer to entities in the domain. While they do not go into much detail into this, and even admit to not having put much work into this aspect (which explains its simplicity), the idea behind it is important. If we are thorough in modeling the context, and are careful when storing the history of the conversation, then we can improve the quality of the dialogue with our agent by improving its ability to recognize and identify Referring Expressions.

### **2.2.3 PICA: Proactive Intelligent Conversational Agent for Interactive Narratives**

PICA [9] is a conversational agent designed to guide an user through an interactive narrative. It monitors user engagement to see if the user is paying attention and/or misses any important part of the story. It is able to proactively talk with the user when it believes the user missed something, or is about to miss something, to bring it to their attention. It is also able to listen to the user and respond to them. It has a Symbolic Knowledge Base that maintains information about the story, and a Temporal Knowledge Map that combines a story graph and a neural network, which saves both story information and the user’s belief model. It has a Proactive Discourse Model, which has both an Active and Reactive module, allowing PICA to interject and talk with the user, as well as listen and react to the user’s inputs. This is enabled by the Natural Language Knowledge Interface, which parses the user’s inputs and extracts the intention behind them, between a set of seven possible intentions.

The authors’ main contribution is the way they represent and interact with the narrative. They combined several approaches, a symbolic knowledge base, a story graph and neural networks, to create a model that is able to represent both the actual events of the story and the user’s beliefs about them. The symbolic knowledge base allows the definition and representation of the story in a way that is easily readable and understandable. It is good both for authoring and for interacting with the story structure. The time graph is a way to organize the different events that comprise the story, especially since some of them can occur in parallel. It is an easy way to visualize the progression of the story. Although the

neural network is a bit too complex for our necessities, since we do not need to map each state change to a different neuron, it is what results of the neural network that is interesting: the user belief model. They are able to map a value between -1 and 1 (-1 means the user definitely does not know about it, 1 the user definitely knows, and 0 means that PICA is unsure about the user knowing) to each event and therefore map the user's knowledge of the story. Not only that, but they have the capability of storing belief models for several users at the same time.

## 2.3 Interrogation in Video Games

In their video essay "What Makes a Good Detective Game"<sup>1</sup>, Game Maker's Toolkit analyze, as the title suggests, what makes a good detective game, by comparing it with other detective media, like movies. They break down the process of an investigation into distinct steps: Gather Information, Expose Lies, Follow Leads, Find Connections, Make Deductions, and finally Make an Accusation. They talk about all sorts of mechanics in existing games that emulate these steps of an investigation, and they review how well they work within the game and how much they contribute to the interaction and the gameplay.

Of these different steps, the one most relevant to our work is Expose Lies. The Make an Accusation could also be considered similar to our work, but since this is defined in the broader context of a detective game, it involves analyzing all the available evidence and finding the culprit of the crime, while in our game we can already assume we are talking with the right suspect, we just need to catch them in a lie. So, in order to Expose Lies, the most important mechanic is the ability to present the interviewee with a piece of evidence that contradicts their statement. This has been executed well in a number of games.

One of the main points of the video is that games that prompt the player give away the mystery. Whether by letting them know when they picked up relevant clues, or when to confront a witness, and so on. Similar to prompting, when you give your player a multiple choice question, with few options, you make it too easy for the player to guess their way through the game, no detective work required. One way to combat this is to give the player so many options that they cannot brute-force their way through the game, they actually have to think about the clues they have and figure out a solution.

This is similar to our approach for the Virtual Suspect, by not giving the player limited dialogue options, we hope to make it possible for him to ask whatever he wants of the Virtual Suspect, and figure out the case (based on the evidence he has available at the start of the interrogation) on his own.

---

<sup>1</sup>[https://youtu.be/gwV\\_mA2cv\\_0](https://youtu.be/gwV_mA2cv_0)

## 2.4 Alexa Prize

The Alexa Prize Socialbot Grand Challenge is a competition created by Amazon for universities dedicated to accelerating the field of conversational AI. The competition is focused on creating an open-domain socialbot, an Alexa skill that converses coherently and engagingly with humans on popular topics and news events. Participating teams advance several areas of conversational AI including Knowledge Acquisition, Natural Language Understanding, Natural Language Generation, Context Modeling, Commonsense Reasoning and Dialog Planning. Throughout the competition, Alexa users test the socialbots, and the university teams have access to that user data and can use it to improve their socialbot.

In their paper *Conversational AI: The Science Behind the Alexa Prize* [10] the Alexa Prize team themselves recommend that a socialbot has the following characteristics: 1. Dialog Manager (DM) 2. Natural Language Understanding (NLU) and Knowledge Module 3. Response Generation 4. Conversational User Experience (CUX) Handler 5. Ranking and Model Selection Policy Module. Some of these can be applied to our Virtual Suspect, the Dialog Manager is the Virtual Suspect architecture from the previous work, the Natural Language Understanding will be done by Alexa, the Knowledge Module is the equivalent to the Knowledge Base, and the Response Generation is also handled by the Virtual Suspect architecture. The last two points, the Conversational User Experience Handler and the Ranking and Model Selection Policy Module are not really applicable to our Virtual Suspect because they are more indicated for the open domain conversations of the Alexa Prize.

We also looked at the work done by the participants of the 2018 Alexa Prize for inspiration. Gunrock [11], from the University of California, was the winner of the competition. Per their description, Chen et al. developed a context-aware hierarchical dialog manager to handle a wide variety of user behaviors, such as topic switching and question answering. In addition, they designed a robust three-step natural language understanding module, which includes techniques such as sentence segmentation and automatic speech recognition (ASR) error correction. Furthermore, they improved the human-likeness of the system by adding prosodic speech synthesis.

Their model is a lot more complex than what we need to accomplish. Most of the things they are doing can and will be accomplished by Alexa's own functionalities in our work. Even the code structure will not be so complicated, since we do not need databases to be able to handle a variety of topics (since our problem is closed-domain).

In the end, we realized that the work done for the Alexa Prize, while important for the development of the field of Conversational AI, is too far removed from what we are trying to do to be useful to us.

# 3

## Background

### Contents

---

3.1 Virtual Suspect . . . . .	13
3.2 Amazon Alexa . . . . .	21

---



Before we can work on a solution, there are two fundamental components that must be understood.

1. The Virtual Suspect framework, as defined by Rato et al. in Virtual Suspect - A Lying Virtual Agent [1].
2. Amazon Alexa and the Alexa Skills Kit (ASK), the tool which will allow us to redesign the interaction with the Virtual Suspect.

### 3.1 Virtual Suspect

Rato et al. [1] set out to design and develop an autonomous agent that would be capable of lying in the context of a police interrogation, as well as create a model for how the agent would come up with those lies. They created an agent that had an internal memory of its own story, was capable of answering questions about that story, and was capable of coming up with lies and alternative stories when answering those questions.

#### 3.1.1 Story Representation

The Virtual Suspect possesses an internal memory of its own story, composed by *entities* and *events*.

An *entity* is the most basic memory fragment. It represents a concept in story, like an object, a person or a location. It is defined by the tuple  $\langle Identifier, Type, Value \rangle$ , where:

**Identifier** is an unique identifier for each entity;

**Type** is used to group similar entities together;

**Value** is what represents the actual concept.

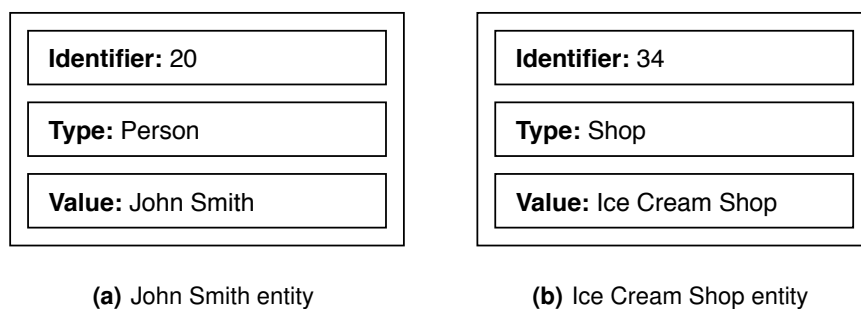


Figure 3.1: Entity Examples

Figure 3.1 contains two examples of entities, where Figure 3.1(a) represents a person named John Smith, and Figure 3.1(b) represents an ice cream shop.

Different entities can be combined to create an *event*. It represents a discrete episode in the agent's story. It is defined by the tuple  $\langle Identifier, Real, Incriminatory, Action, Time, Location, Agent, Theme, Reason, Manner \rangle$ , where:

**Identifier** is a unique identifier for each event;

**Real** is a flag that indicates if the event actually happened or not;

**Incriminatory** is a value from 0 to 100 which indicates how compromising the event is;

**Action** is a verb that represents the action corresponding to the event;

**Time** is an entity that represents the time frame in which the event took place;

**Location** is an entity that represents the location where the event took place;

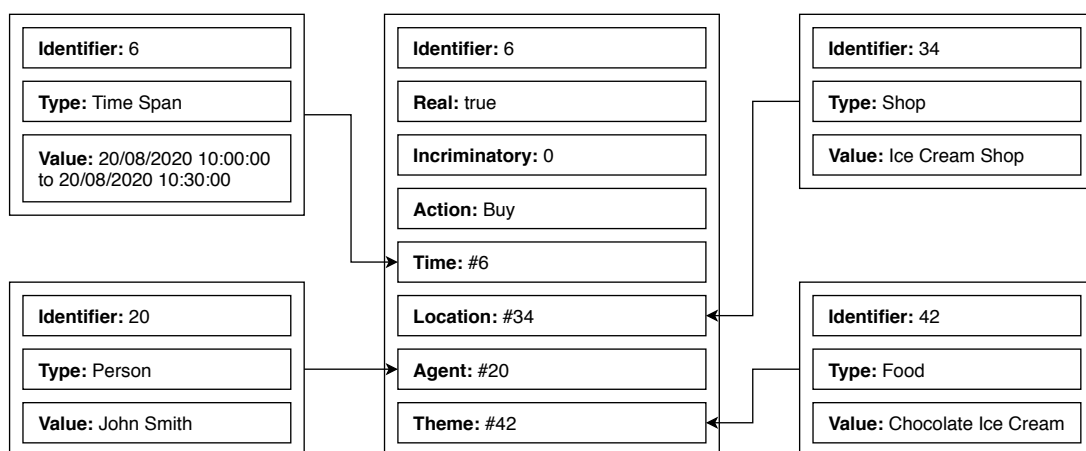
**Agent** is optional and it represents people involved in the event, can contain multiple entities;

**Theme** is optional and it represents the target of the action, can contain multiple entities;

**Reason** is optional and represents the reason for the action to occur, can contain multiple entities;

**Manner** is optional and represents how the action occurred, can contain multiple entities.

An event does not directly contain entities, it merely references them, and the same entity can be referenced by various events.



**Figure 3.2:** An event with references to the relevant entities

Figure 3.2 shows an example of an event, which can be described as “John Smith bought a chocolate ice cream at the ice cream shop”. We can see that this event was real and was not incriminatory, and that it took place between ten and ten thirty am on August 20th. This event does not have any entities corresponding to the Reason or Manner fields because they are not relevant to this event.

The agent's memory contains all of the entities and events, and the sequence of Real events can be considered the agent's story.

### 3.1.2 Interaction

Interaction between the user and the Virtual Suspect is performed through questions and answers. The user asks questions about the agent’s story and the agent computes the appropriate responses.

Questions are represented by a structure called a *query*, which is defined by the tuple  $\langle QuestionType, QuestionFocus, Conditions \rangle$ , where:

**Question Type** is the category of the question, and can be *Validation* or *Information Gathering*;

**Question Focus** represents the event field to be returned, similarly to the questions “Where?”, “Who?”, “When?”, etc;

**Conditions** is a list of constraints that restrict which events the question can be referring to.

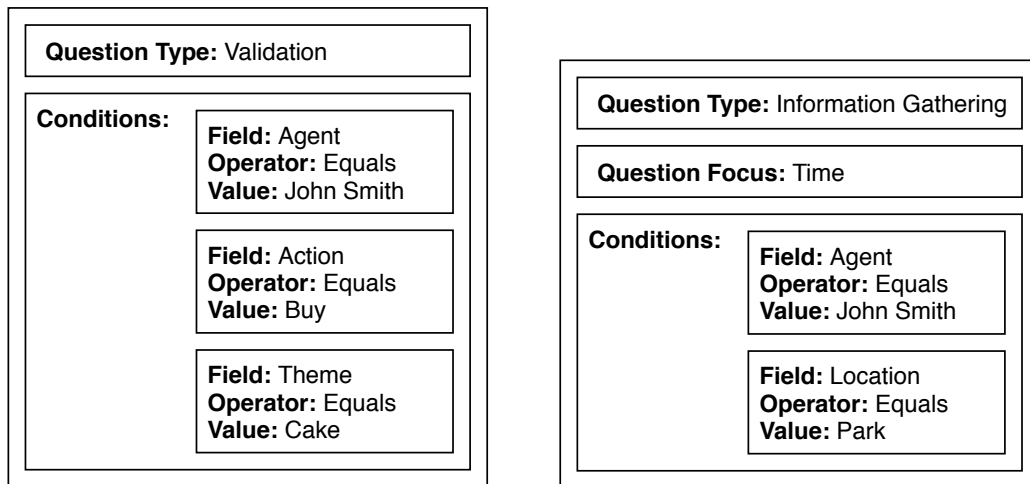
A Validation question is a yes-or-no question, and it seeks to validate the conditions in the query, and it does not need a question focus. An Information Gathering question seeks to retrieve a specific field from an event, determined by its question focus, while satisfying the query conditions. Each *condition* can be defined by the tuple  $\langle Field, Operator, Value \rangle$ , where:

**Field** refers to the event’s field to be tested, like Location or Agent;

**Operator** is the comparator used in the comparison, with the only ones used in the original prototype being Equal and Between;

**Value** is the value to be compared to.

For example, a condition can be described as “Location Equals Ice Cream Shop” or “Time Between 10:00:00 and 10:30:00”.



(a) Validation Question

(b) Information Gathering Question

**Figure 3.3:** Query Examples

In Figure 3.3 we can see two examples of questions represented as queries, with Figure 3.3(a) representing the question “Did John Smith buy a cake?”, and Figure 3.3(b) representing the question

“When did John Smith go to the park?”.

Answers to these questions are represented by a structure called a *query result*, which is defined by the tuple  $\langle Query, Result \rangle$ , where:

**Query** is the original query;

**Result** contains the answer to the query, either a Boolean in the case of Validation questions, or a list of values in the case of Information Gathering questions.

So, for the questions presented in Figure 3.3, the results would contain those same queries, and then a Boolean for the query in Figure 3.3(a) (in this case we can consider it would be False since John Smith bought a chocolate ice cream and not a cake - see Figure 3.2), and a value for the Time entity corresponding to when he was at the park (let us assume he was and it was on August 20th between four and five pm). Please note that what is returned in the query result is not the entity itself, merely its value.

### 3.1.3 Architecture

Now that we have learned how the agent’s memory is structured and how the interaction is defined through questions and answers, let us see how these elements are combined, and how the agent is capable of lying. The **Virtual Suspect Architecture** is composed by the **Knowledge Base**, the **Query Engine**, and three layers that allow the agent to lie: the **Theory of Mind Layer**, the **Strategy Selection Layer**, and the **Story Adjustment Layer**.

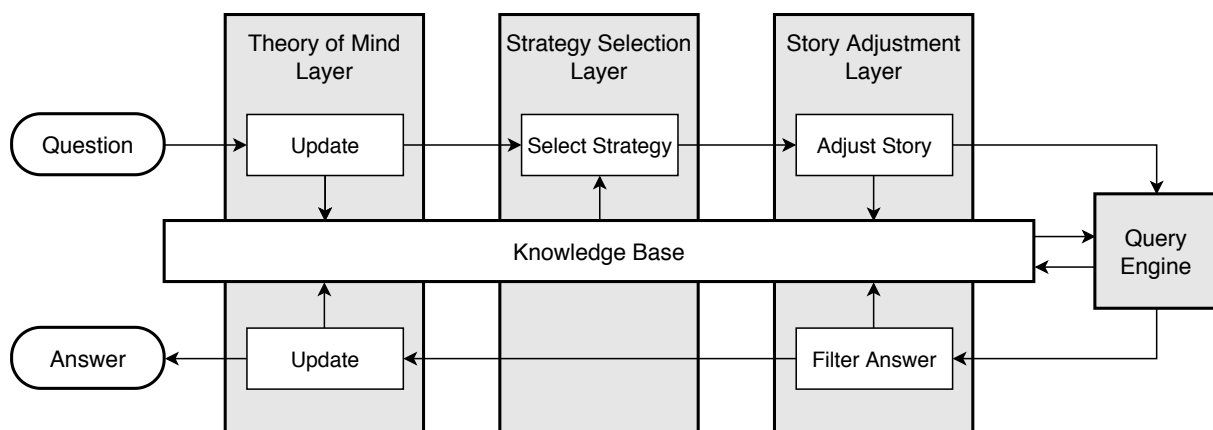


Figure 3.4: Virtual Suspect Architecture

The agent’s **Knowledge Base** contains all the entities and events, this is the agent’s memory. The questions and answers are handled by the **Query Engine**, which processes the query, accesses the Knowledge Base to find the events that match the query conditions, and computes the response. Just these two elements would be enough to have an interactive agent that would be capable of answering questions about its story, but the agent created by Rato et al. [1] is also capable of lying on top of that.

The way this works is through a three-layered two-pass control system. When a question is posed to the agent, it has to pass through each of the three layers before reaching the Query Engine, and then the answer passes through those same layers again before being returned to the user. Each of these layers serves a unique function that, when put together, enables the agent to lie about its story.

The first layer is the **Theory of Mind Layer**, which, on first pass, analyses the conditions included in the query to understand what information the user already knows about. This creates a representation of the user's knowledge in the agent's own memory (according to Theory of Mind theory), allowing the agent to know what pieces of information it cannot lie about because they are already known. On second pass, it updates the user's knowledge of events with the information about to be given in the response.

The second layer is the **Strategy Selection Layer** which looks both at the query and the Knowledge Base to decide the strategy that should be used in that circumstance. For example, if the answer to the query could contain incriminatory information that the user does not yet know (or has not shown to know), then the agent can decide to switch that incriminatory information to something more innocuous. This layer does not have a behaviour for the second pass.

The third layer is the **Story Adjustment Layer**, which makes the necessary changes to the Knowledge Base, according to the selected strategy. For example, this is where the agent creates the new fake events and stories that run parallel to the real events. It is important to note that the real events are never altered, as those are facts that cannot be changed. The agent is capable of creating new fake events and then alter the information in those events according to the selected strategy. The agent always keeps track of the real story and the story that it is currently telling the user, and is capable of switching back to the real events if he is found out. These adjustments can occur either on the first or second pass, depending on the strategy.

Each of these layers has access to the Knowledge Base, and after the query passes through each layer, it is processed by the Query Engine and the result passes through each of the layers again before being returned to the user. There are only three layers, but more could be added in the future depending on the functionality needed. Figure 3.4 shows a representation of all of these components and how the information flows between them.

### 3.1.4 Use Case: Peter Barker

The story example that was used in the implementation of the Virtual Suspect [1] was about an agent, Peter Barker, who stole a necklace from a jewelry shop on March 5th, and then travelled to Silvermoon City to sell it to John Frey, whose contact he obtained from a man called Jaime.

There are more details and events in the full story, but this is enough context to understand what was presented to the users.

### 3.1.5 Prototype

During the development of the Virtual Suspect, two prototypes were made [1]. Both used visual interfaces with buttons that allowed users to select different options.

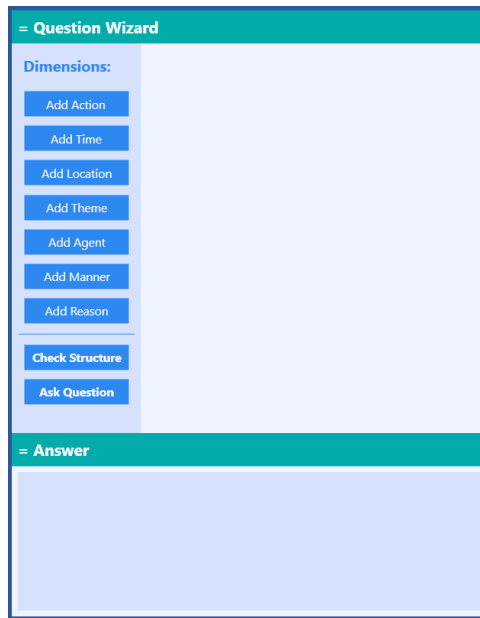


Figure 3.5: First Virtual Suspect Prototype [1]

The First Prototype (see Figure 3.5) allowed users to ask whatever questions they wanted by constructing the queries from scratch, with buttons to add conditions. According to feedback they gathered, users found the interface too confusing and difficult to learn, so another interface had to be developed.

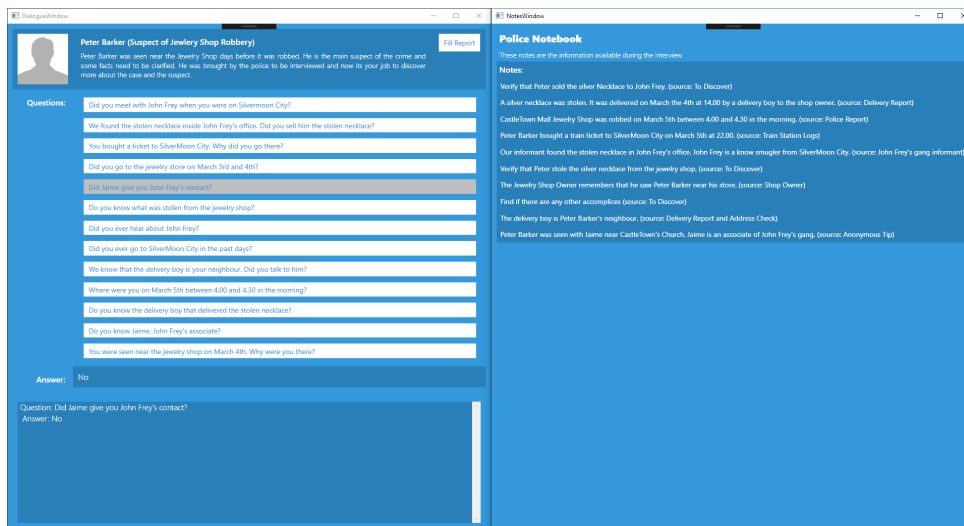


Figure 3.6: Second Virtual Suspect Prototype [1]

The Second Prototype (see Figure 3.6), instead of allowing users to ask whatever questions they

wanted, came with a preset number of questions that users could select. These were written in plain English and presented in a more natural way of speaking, with some being more direct than others. Users were limited to the number of different questions they could ask, but they could ask the same question more than once, which, depending on the order it was done in, could sometimes produce different answers and reveal the agent's lies. The interface also included information about the suspect and notes about the case (represented in the top left and right of Figure 3.6, respectively), to provide more context and immersion to the police interrogation scenario.

While the Second Prototype did produce better results than the First for the purposes of their work, it was not without flaws. The sacrifice of the freedom of being able to ask any question being the main one.

The questions available in the interface were, as written:

1. Did you meet with John Frey when you were on Silvermoon City?
2. We found the stolen necklace inside John Frey's office. Did you sell him the stolen necklace?
3. You bought a ticket to SilverMoon City. Why did you go there?
4. Did you go to the jewelry store on March 3rd and 4th?
5. Did Jaime give you John Frey's contact?
6. Do you know what was stolen from the jewelry shop?
7. Did you ever hear about John Frey?
8. Did you ever go to SilverMoon City in the past days?
9. We know that the delivery boy is your neighbour. Did you talk to him?
10. Where were you on March 5th between 4.00 and 4.30 in the morning?
11. Do you know the delivery boy that delivered the stolen necklace?
12. Do you know Jaime, John Frey's associate?
13. You were seen near the jewelry shop on March 4th. Why were you there?

Out of the thirteen questions, ten were Validation questions and three were Information Gathering, with two having a Reason focus and one a Location focus. This will be further explored in Chapter 5.

### 3.1.6 Implementation

We have analysed how the agent's memory is structured, how the interaction between the user and the agent takes place, and how the agent's architecture is organized to allow him to lie when answering the user's questions. We have also seen how this information is presented to the user. Let us take a moment to see how this was put together in practice.

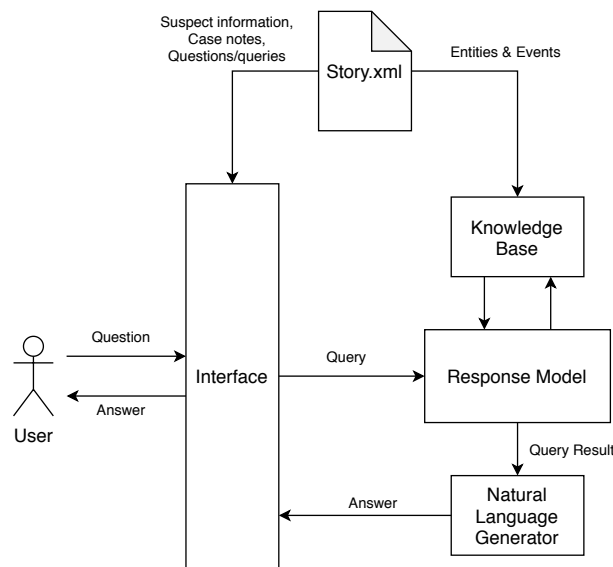


Figure 3.7: Virtual Suspect Implementation

First, there is a XML file containing all the static information of the scenario. This includes the suspect information, case notes, and questions that are presented in the visual interface (see Figure 3.6), and the entities and events that populate the agent's **Knowledge Base**. This story file not only contains the questions in English as they are presented to the user, but also the corresponding queries, so when the user selects the question from the interface, this sends the corresponding query to the agent's **Response Model**, already with the proper format. If the agent's **Architecture** was the theoretical definition of the agent's algorithm, then let us consider for the sake of analysis and future development that the **Response Model** is the practical implementation of that Architecture and encompasses the Query Engine and the three layers. The **Response Model** receives the query directly from the Interface and computes the appropriate query result, according to the method represented in Figure 3.4. That query result is then sent to a separate **Natural Language Generation** module, which was created for the purposes of transforming the query results into English. Finally, the answer computed by the **Natural Language Generator** is sent back to the Interface and displayed for the user to see, and the cycle begins anew, with the user now able to select a question. Figure 3.7 shows a representation of this process.

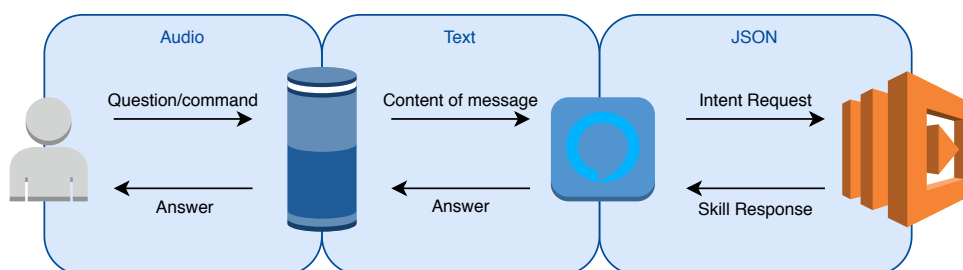


## 3.2 Amazon Alexa

As discussed in Chapter 1, this work will use Amazon Alexa as a way of improving the interaction with the Virtual Suspect. Amazon Alexa, more commonly known as just Alexa, is a virtual assistant AI developed by Amazon and first used on the Amazon Echo smart speakers. It is capable of voice interaction, making lists, playing music, podcasts, audio books, setting alarms, and provide news, weather, traffic, and sports updates. It is also capable of controlling other smart devices and becoming a home automation system. The presence of Alexa has become ubiquitous in many homes all around the world, and its functionalities have continued to evolve since its introduction in 2014.

There is one functionality that is of particular interest to this work, and that is the capability of users to extend the default Alexa capabilities by adding “skills”, which are essentially third-party application with a variety of functions. These can include extensions to already existing Alexa functions, like music or weather, or new features, like games. What makes this feature so interesting for this work is that the tools to develop one of these skills are free and available to everyone, which means we can take advantage of Alexa’s technology to develop our better Virtual Suspect interaction.

The **Alexa Skills Kit (ASK)** is the tool that allows us to build our own custom skill, and there are two essential components: the **Interaction Model**, which is essentially our skill’s front end, and the **Skill Service**, which is our skill’s back end. We can make both of these components by creating the required accounts in the Alexa Developer Console<sup>1</sup> and Amazon Web Services (AWS)<sup>2</sup>.



**Figure 3.8:** Workflow of an Alexa Skill

Figure 3.8 contains a simplified version of the workflow of an Alexa Skill, which will be further explained in the following subsections. The user asks a question or gives a command to their Alexa device, it parses the audio into text and send its content to the Alexa Skill, which in turns disambiguates the meaning behind the user’s message and sends an Intent Request to the Skill back-end. The Skill back-end receives the Intent Request, computes the response and sends a Skill Response, from which Alexa extracts the message to be given back to the user.

<sup>1</sup>developer.amazon.com

<sup>2</sup>aws.amazon.com

### 3.2.1 Interaction Model

The Interaction Model is made through the Amazon Developer Console (Figure 3.9). It acts as the skill's front end, and its function is to disambiguate and attribute meaning to what the user says. It does this by trying to perceive the intention behind the user's message/command.

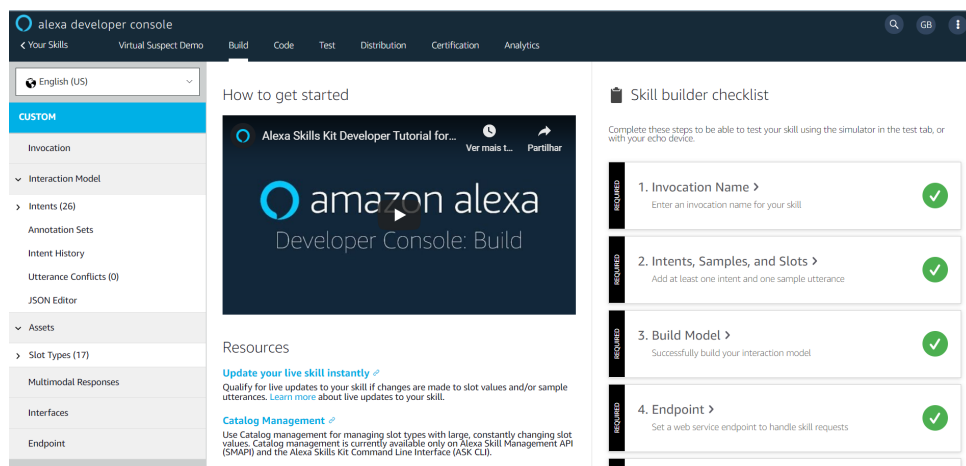
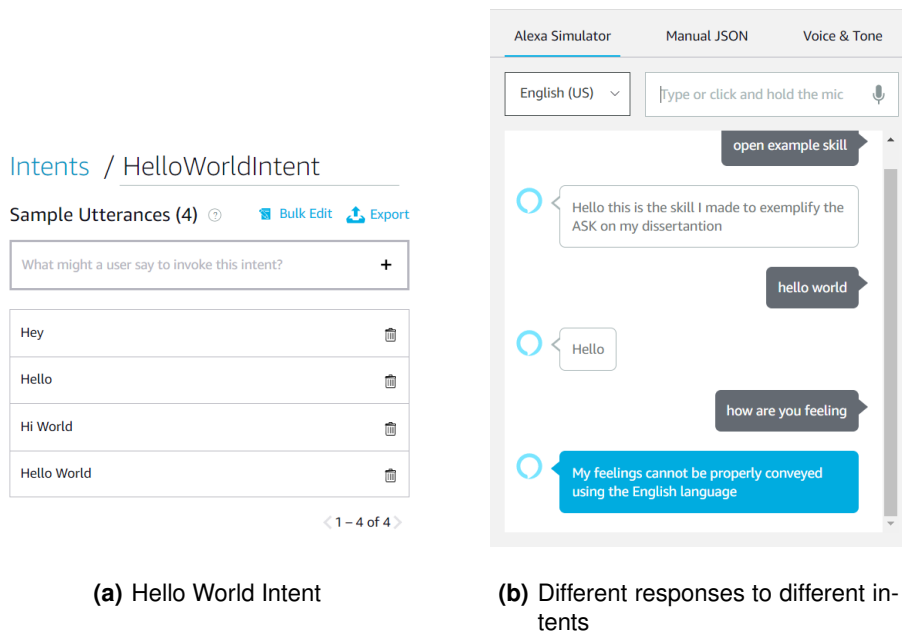


Figure 3.9: Alexa Developer Console

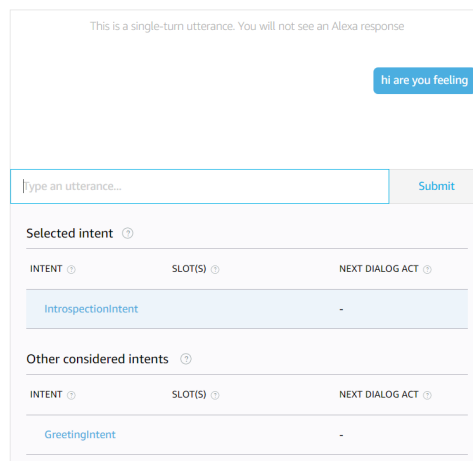
The Interaction Model is divided into several *intents*. These intents are defined by the author of the skill (although there are a few default built-in intents from Amazon) and contain sample phrases that represent what the user can say to “activate” that intent. For example, one can have a **HelloWorldIntent** that contains the sample phrases “*Hello World*”, “*Hi World*”, and “*Hello*” (Figure 3.10(a)); and a **HowAreYouWorldIntent** that contains the sample phrases “*How are you World*”, and “*I hope you are feeling well*”. Then, if the user says one of those phrases, or something similar, then their intent will be identified by the Skill, and a correct answer can be given. In this case, it would make sense to give different answers to the different intents, as the answer to the **HelloWorldIntent** could be a simple “*Hello*”, and the answer to the **HowAreYouWorldIntent** could be “*My feelings cannot be properly conveyed using the English language*” (Figure 3.10(b)).

Now, if all Alexa was capable of recognizing was the specific phrases in the intent samples, the technology would not be that impressive. What they do is take the Interaction Model as laid out by the skill's author and then they train a model of their own, with the knowledge in the Interaction Model and possibly their own knowledge base as well, to be able to identify the meaning of those and similar phrases according to the intents in the Interaction Model. It is not known exactly how they achieve this, as there is no documentation on the internal process of training the model and the decision process when identifying the intents. Using the Developer Console we can see that there is a degree of confidence when identifying an intent and other possible intents that it could have chosen (Figure 3.11), but as a developer there is no way of accessing this information. The only way to improve the accuracy of



**Figure 3.10:** Intents and Responses

your model is to add more samples and more intents so that they can better identify what users mean. Amazon does offer a **FallbackIntent** for when a sentence does not match any of the available intents, but it does not always work perfectly.



**Figure 3.11:** The Utterance Profiler in the Developer Console allows us to see the identified intent for a specific phrase, and other possible intents

Another feature of the Interaction Model is the concept of *slots*, which are essentially variables inside of intents. For example, we can have a **MyNameIntent** where the sample phrase can be “*My name is {name}*”, in which {name} represents a slot (Figure 3.12(a)). This way a user can say “*My name is Peter*” or “*My name is Isabelle*” and both phrases will equally satisfy the intent. Each slot must have

a *slot type*, which is essentially the list of acceptable values that slot can have (Figure 3.12(b)). These can be chosen from a wide variety of already available slot types from Amazon (for example, a list of names in English), or custom made according to the skill’s domain. For custom slot types, each slot value can also have synonyms. These slots cannot be iteratively defined, so there cannot be a slot inside of another slot, each slot just contains a list of possible values. The slot information is passed on to the back-end and their values can be incorporated into the response, allowing the skill to respond personally according to the name in the slot (Figure 3.12(c)).

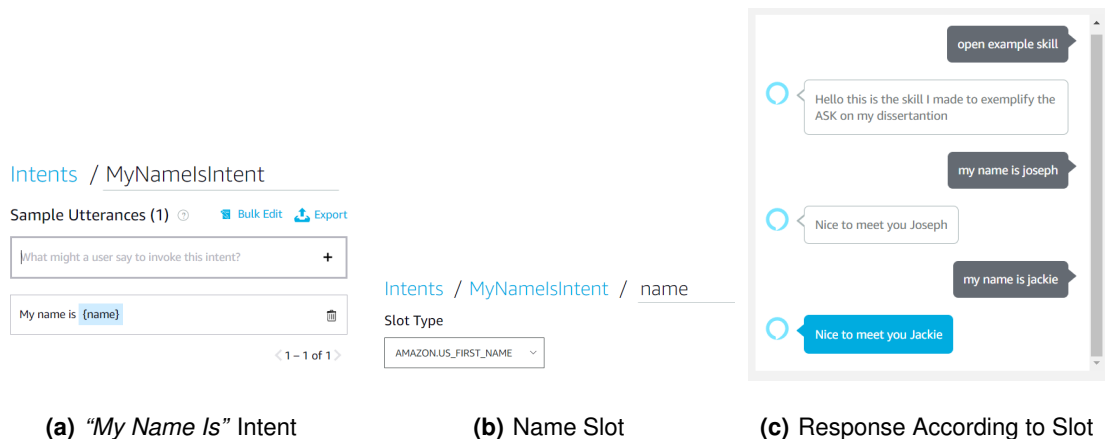


Figure 3.12: Slots

The way this information is processed and sent to the Skill Service (the back-end) is through a JSON file. This file contains the intent that the Interaction Model identified and the value of the slots present, if any. It does not contain more information beyond that, and therefore it does not contain what exactly the user said. Whether the user said *“Hello World”* or *“Hi”* does not matter much to figure out the response to the **HelloWorldIntent**. The only way to pass information other than intent to the Skill Service is through the use of slots. Listing 3.1 contains an example of the JSON sent by the Interaction Model to the Skill Service. Please note that it is not the complete file, but the rest of it is technical details pertaining to the devices and is not relevant to the skill or the response.

**Listing 3.1:** Excerpt of the JSON sent by the Skill

```
1  "request": {
2    "type": "IntentRequest",
3    "requestId": "amzn1.echo-api.request.a3716fc9-7a16-467d-b944-6b41c49f6459",
4    "locale": "en-US",
5    "timestamp": "2020-09-01T15:27:24Z",
6    "intent": {
7      "name": "MyNameIsIntent",
8      "confirmationStatus": "NONE",
9      "slots": {
10       "name": {
11         "name": "name",
12         "value": "peter",
13         "confirmationStatus": "NONE",
14         "source": "USER",
15         "slotValue": {
16           "type": "Simple",
17           "value": "peter"
18         }
19       }
20     }
21   }
22 }
```

### 3.2.2 Skill Service

The Skill Service is the skill's back-end. There is no one correct way to set up the Skill Service, it can be done using AWS, with HTTP, and in a variety of different coding languages. The recommended and more commonly used method is using AWS Lambda, which is an event-drive serverless computing platform. Using Lambda you can set up a function that will receive the JSON request sent by the ASK, process it, and send back the response.

Once the function receives the Skill Request, it can read the intent name and the slot values, and compute the appropriate response, sending it back in its own Skill Response JSON file (Listing 3.2). For the purposes of the example we have been using, the computing of the correct response is as simple as returning the right string according to the intent, and incorporating the slot value into the response.

**Listing 3.2:** Skill Response

```
1  {
2    "body": {
3      "version": "1.0",
4      "response": {
5        "outputSpeech": {
6          "type": "PlainText",
7          "text": "Nice to meet you peter"
8        },
9        "reprompt": {
10         "outputSpeech": {
11           "type": "PlainText",
12           "text": "Nice to meet you peter"
13         }
14       },
15       "shouldEndSession": false,
16       "type": "_DEFAULT_RESPONSE"
17     }
18   }
19 }
```

# 4

## Solution Architecture

### Contents

---

4.1 A Virtual Suspect Skill . . . . .	27
4.2 Implementation Plan . . . . .	29

---

Recapping all the concepts we have explored so far: In Chapter 1 we established the shortcomings of the previous Virtual Suspect interaction, and decided to use the tools provided by Amazon Alexa to improve upon it. In Chapter 2 we tried to look for similar works to draw knowledge from, with mixed results. In Chapter 3 we analysed how the Virtual Suspect was designed and built, and examined its previous interaction, and we explained what the Alexa is and how the ASK works. Now, we take the knowledge learned in the previous chapters and define our own solution to combine the Virtual Suspect with the Alexa to make a more engaging and more natural interaction.

## 4.1 A Virtual Suspect Skill

We will create an Alexa Skill with the functionality of the Virtual Suspect. That means the Interaction Model of our skill will handle the questions, parsing what the user asks to identify the question type, focus and conditions. This can be done through intents and slots as we saw in Section 3.2. Then our Skill Service can take the intent and slots, and transform that information into a query with the correct format, and this way we can send that query to the Virtual Suspect Response Model that we can keep exactly the same. The Response Model will compute the result and the Natural Language Generator will generate the proper answer, which we can then send back to the Skill so it can be relayed to the user.

### 4.1.1 Interaction Model

An Interaction Model is divided into intents and slots. A Virtual Suspect query is divided into query type, query focus, and conditions. Combining these two structures, we can build an Interaction Model where the intents can encompass the query type and focus, and the slots inside the intents can correspond to the query conditions. The alternative would be to have a separate intent for every possible question one could ask the Virtual Suspect, which would not scale well with the number of events and entities. Therefore, we can define the following intents:

- ValidationIntent
- TimeFocusIntent
- LocationFocusIntent
- AgentFocusIntent
- ThemeFocusIntent
- ReasonFocusIntent
- MannerFocusIntent

These seven intents cover all the different types of questions supported by the Virtual Suspect. They will have to have a wide variety of sample utterances with different combinations of slots to cover all the possible questions the user can ask the Virtual Suspect.

For example, let us look at the questions we used in Section 3.1: *“Did John Smith buy a cake?”* and *“When did John Smith go to the park?”*. If we include these questions as they are as possible utterances, then the values necessary for the query conditions would not be propagated to the Skill Service and the Response Model. We would have to do something like *“Did {agent} {action} a {theme}”* and *“When did {agent} go to the {location}”*, with the words in brackets representing slots (as seen in Section 3.2), so those slot values can be sent with the intent in the Skill Request and included in the query conditions. And we need to be careful with different variations and permutations of asking the same question. Moreover, slots require a slot type, so we need to define possible values for the different slots. The story file already contains all the entities in the story, so we can use that knowledge to populate our Interaction Model slot types with the values that the Virtual Suspect can recognize, so we can have a theme slot type, an agent slot type, etc.

### 4.1.2 Skill Service

Our Skill Service will receive the Skill Request object from the Interaction Model and will construct a query object from that information. The intent establishes the query type and focus, and the slots contain the query conditions. Once we have the query object, we can send it to the Virtual Suspect Response Model, which we will be able to access in the same environment as the Skill Service as sort of a code black box.

The original Virtual Suspect Prototype was developed in C# using Microsoft’s Visual Studio, which means we can take advantage of that and use that same code as a DLL package by building our Skill Service in the same environment, which is possible by using the [AWS Toolkit for Visual Studio](https://aws.amazon.com/visualstudio/)<sup>1</sup>.

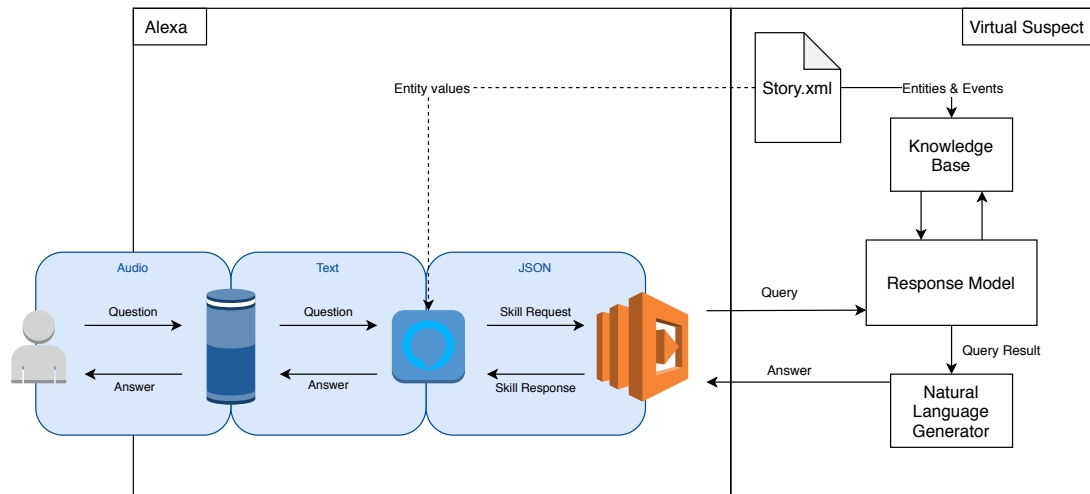
### 4.1.3 Architecture

Figure 4.1 shows what we have established so far, and is essentially a combination of the architectures presented in Section 3.1 in Figure 3.7 and in Section 3.2 with Figure 3.8. The user asks Alexa a question, which is interpreted by the Interaction Model, which sends a Skill Request to the Skill Service, which transforms it into a query and sends it to the Virtual Suspect Response Model, which computes the query results, sends it to the Natural Language Generator, which then sends the answer back to the Skill Service so it can be transmitted back to the user. We have studied how the interaction between the Interaction Model and Skill Service works, and how the Virtual Suspect Response Model processes the

---

<sup>1</sup><https://aws.amazon.com/visualstudio/>





**Figure 4.1:** Solution Architecture

queries, in Section 3.2.

As we saw in Section 3.1, the story file populates the Knowledge Base with events and entities, which the Response Model then accesses when processing the queries. We should note however, that the connection between the story file and the Interaction Model represented in Figure 4.1 is just a symbolic one, as there is no direct connection between the two entities. We will need to populate the Interaction Model with the entity values so the slots can be properly recognized, but this cannot be done by directly accessing the story file and automatically extracting the entity values, as the two are in completely different environments. This will have to be done by hand, but fortunately it only has to be done once when building the Interaction Model, unless changes are made to the story.

## 4.2 Implementation Plan

As we have established, we can take advantage of work that has already been done by using the Virtual Suspect Response Model as a code black box and accessing it from our Skill Service. We will start by building our Interaction Model, since it will need to have a wide variety of sample phrases in order for our Skill to provide an engaging and natural interaction. We will achieve this by conducting an iterative and user-centered development cycle.

Our goal is to improve the interaction with the Virtual Suspect. A lot of our work will be centered around expanding the comprehension capabilities of the agent, making sure that it is interpreting and answering questions correctly. With a lying agent, it would be difficult to perceive if this is happening or not, since the agent will attempt to obscure its answers. It would be difficult to tell if the agent was saying “No” because it understood the question and decided to lie, or because he did not understand the question in the first place. By altering the initialization parameters of the Virtual Suspect Response Model

we can turn off the lying component, which allows us to interact with an honest version of the Virtual Suspect. This will allow us to focus on making sure the agent is correctly interpreting what is being asked, instead of trying to decode each answer. This will not only be useful during the development, but also when interacting with users.

We will start development by implementing a version of our Skill with the same functionality as the final Virtual Suspect Prototype [1]. We can then expand that functionality past the original thirteen questions and have a basic coverage of all the agent's events and all of the Virtual Suspect capabilities. We will use that to conduct a User Study, allowing users to interact with the Virtual Suspect Skill in that early stage, to learn how people interact or want to interact with the agent, and to find possible shortcomings with the interaction. In order for users to have a better sense of the effect of their questions on the Virtual Suspect, and to make it easier to find any errors with the interaction, this User Study will be conducted without the lying component.

With the data from that First User Study, we can expand the capabilities of our Virtual Suspect Skill, by studying how users formulated their questions and how they interacted with the agent, to see if there are blind spots or shortcomings in the interaction, and by fixing any flaws or errors that might have arisen. Since the original Virtual Suspect [1] has a very limited interaction, we suspect that a lot of work will be needed in order for our new conversational interaction to feel natural.

After we iterate upon our Virtual Suspect Skill with all the improvements we can derive from the First User Study data, we will conduct a Second User Study, also without the lying component, to verify that the improvements led to a better interaction and performance by the agent. We will measure the User Experience (UX) of that version of the interaction.

Finally, since the goal of the Virtual Suspect is to have an interaction with a lying agent, we will turn that component back on at the end of the development and conduct a Third and final User Study to again measure the performance of the agent and the UX of the participants, and compare it to the no-lying version.

This whole process is all in service of our goal, have a more natural and engaging interaction with the Virtual Suspect, and it is expanded upon in Chapter 5 and Chapter 6.

# 5

## Development

### Contents

---

5.1 First Steps . . . . .	32
5.2 Improving the Interaction . . . . .	38
5.3 Last Adjustments . . . . .	45

---

In this chapter we will detail the process of developing the Virtual Suspect Skill and all the changes and deviations we had to make from the original in order to improve the interaction.

## 5.1 First Steps

In Chapter 4 we established the blueprint for our Virtual Suspect Skill: an Interaction Model that classifies the questions, and a Skill Service that redirects the queries to the Virtual Suspect Response Model. Our first objective is to achieve a working prototype with the same functionality as the original Virtual Suspect Prototype [1], so we can conduct a user study to evaluate its performance and identify what needs to change.

### 5.1.1 Recreating the Original Functionality

In order to recreate the same functionality as the original Prototype, we need to be able to ask the same questions and obtain the same answers. The Virtual Suspect Response Model will be the same, so we just need to make sure that our Interaction Model is properly identifying the questions and the same queries are being made. As we saw in Section 3.1, the original Prototype did not have a wide variety of questions available, and those questions did not conform to a consistent style, with some having statements before the actual question. Theoretically, we could include those questions with that same format in our Interaction Model, with just the relevant query condition information as slots, but that is not a solution that will scale well, as there are innumerable ways of asking the same question with different sentence structures (especially if we include the possibility of dividing the question into separate sentences like it happens in several of the examples). If we tried to model each specific way of asking each question, for all the different types of questions, we would never finish coming up with new questions, once we moved past the original thirteen. We need a strategy to create a more streamlined way of asking each of the original questions, with the same information that was included in the corresponding query (to preserve the same exact functionality), so we can then extend that same structure to other questions and have a consistent style that the agent can understand. The idea is that, the more streamlined the style of questions is, the less different utterances we will have to write in the Interaction Model, which will require a less extensive authoring effort and provide an easier comprehension for the agent.

In Table 5.1 we can see the questions presented in the Virtual Suspect Prototype [1] (and mentioned in Section 3.1), and their corresponding queries, which were already predefined in the story file and sent to the Response Model when the user selected the respective question (as illustrated in Figure 3.7). We have already mentioned the inconsistent styles of questions, but another thing that you can observe is the lack of consistency with the query conditions. In Table 5.1, the Agent\* represents the condition of

Original Question	Question Type	Conditions
Did you meet with John Frey when you were on Silvermoon City?	Validation	Agent*, Agent, Location
We found the stolen necklace inside John Frey's office. Did you sell him the stolen necklace?	Validation	Agent*, Agent, Action
You bought a ticket to SilverMoon City. Why did you go there?	Information Gathering (Reason)	Agent*, Location, Action
Did you go to the jewelry store on March 3rd and 4th?	Validation	Location, Time
Did Jaime give you John Frey's contact?	Validation	Agent*, Agent, Theme, Action
Do you know what was stolen from the jewelry shop?	Validation	Action, Location
Did you ever hear about John Frey?	Validation	Agent
Did you ever go to SilverMoon City in the past days?	Validation	Agent*, Location
We know that the delivery boy is your neighbour. Did you talk to him?	Validation	Agent, Agent*, Action
Where were you on March 5th between 4.00 and 4.30 in the morning?	Information Gathering (Location)	Time, Agent*
Do you know the delivery boy that delivered the stolen necklace?	Validation	Agent
Do you know Jaime, John Frey's associate?	Validation	Agent*, Agent
You were seen near the jewelry shop on March 4th. Why were you there?	Information Gathering (Reason)	Location, Time, Agent*

**Table 5.1:** Original Questions and corresponding Queries

“Agent Equals Peter Barker” (the titular Virtual Suspect), as opposed to other agents, like it happens in several questions. While all the questions are directly addressed to Peter, not all of them include that condition. In the end, it did not make a difference, as all the events in the Knowledge Base are told from Peter's perspective, so whether or not that condition is present did not alter the results of the questions. Since there does not seem to be a pattern to when that condition is included or not in the example queries, and since it does not make a difference whether or not it is there at the moment, this means we can establish, for the sake of consistency, that when the question directly involves the subject (Peter Barker) that the condition should be in the query. Not only does this make our queries consistent with one another, but if in the future there happen to be questions which are not directly addressed to our subject, then the presence, or lack thereof, of that condition will make a difference.

Apart from that inconsistency, there is also an inconsistency between the content of the question itself and the respective query, in some cases. For example, in the “We found the stolen necklace inside John Frey's office. Did you sell him the stolen necklace?”, even if we ignore the fact that John Frey's office is irrelevant to the query, the *stolen necklace* is not present in the query conditions, despite being clearly relevant to the question at hand. We should also note that in the question “Did Jaime give you John Frey's contact?”, while Jaime is acting as the subject of that sentence and Peter as the object, the actual query has the Action condition as “get” instead of give, essentially reversing the roles. This is because all the events are modeled from Peter's perspective, therefore there is not a corresponding event from Jaime's perspective where he *gives* the contact to Peter. We need to address these inconsistencies in

order to make sure our model will be coherent.

The changes we have to make in order to address these inconsistencies are:

1. Change the sentence structure to make it more consistent (by not separating questions into different sentences, removing irrelevant information and presenting the questions in their simplest form);
2. Consistently apply the “Agent Equals Peter Barker” condition when the question is addressed directly to the Suspect;
3. Make sure that relevant information that is included in the text of the question is also included in the query conditions.

Original Question	New Format
Did you meet with John Frey when you were on Silvermoon City?	Did you meet John Frey in Silvermoon City?
We found the stolen necklace inside John Frey's office. Did you sell him the stolen necklace?	Did you sell the stolen necklace to John Frey?
You bought a ticket to SilverMoon City. Why did you go there?	Why did you travel to Silvermoon City?
Did you go to the jewelry store on March 3rd and 4th?	Were you at the jewelry store between March 3rd and March 4th?
Did Jaime give you John Frey's contact?	Did you get John Frey's contact from Jaime?
Do you know what was stolen from the jewelry shop?	Did you steal something from the jewelry shop?
Did you ever hear about John Frey?	Do you know John Frey?
Did you ever go to SilverMoon City in the past days?	Have you been to Silvermoon City?
We know that the delivery boy is your neighbour. Did you talk to him?	Did you talk with the delivery boy?
Where were you on March 5th between 4.00 and 4.30 in the morning?	Where were you on March 5th between 4 am and 4:30 am?
Do you know the delivery boy that delivered the stolen necklace?	Do you know the delivery boy?
Do you know Jaime, John Frey's associate?	Do you know Jaime?
You were seen near the jewelry shop on March 4th. Why were you there?	Why were you at the jewelry shop on March 4th?

**Table 5.2:** Streamlined questions

In Table 5.2 we can see how we can change the questions to a more streamlined and consistent format, while still preserving the same information that was contained in the query. Of course, this is not the only way of rephrasing those questions, variations will always exist, but it is much easier for us to come up with those alternatives starting from this new consistent format. The questions as written in Table 5.2 are just one possibility of how they can be rewritten in the new style. For example, instead of “*Were you at the jewelry store (...)*” we can have “*Did you go to the jewelry store (...)*” for the same results. In fact, we can have both of those variations in the Interaction Model, the point of this

transformation is to get to the essential information of the question in its most basic form, which results in a more consistent and coherent framework for our Interaction Model.

Now we have to undergo one more transformation, where we abstract the information for the query conditions into slots so we can use those values to construct the query object in the Skill Service. For example, the question “*Did you meet John Frey in Silvermoon City?*” would be modeled like “*Did {agent} meet {agent} in {location}*”<sup>1</sup>, as the query conditions for this question are “*Agent Equals Peter Barker*”, “*Agent Equals John Frey*” and “*Location Equals Silvermoon City*” (see Table 5.1). This way, we can pass “*Peter Barker*”, “*John Frey*”, and “*Silvermoon City*” as slot values in the Skill Request JSON and add them as query conditions.

Technically, this is enough to create our query and have a functional Virtual Suspect Skill, but there are more steps we can take to make it more efficient. After all, asking “*Did you meet John Frey?*” is practically the same as asking “*Have you met John Frey?*”, but “*Did {agent} meet {agent}*” and “*Have {agent} met {agent}*” are two different utterances, which we would need to add to the Interaction Model in order for them to be recognized. But if instead of adding those two different utterances, we add something like “*{question\_verb} {agent} {filler\_verb} {agent}*”, where *{question\_verb}* can be a slot containing verbs that are used to start questions (like *did*, *have*, *were*, *do*, *etc.*) and *{filler\_verb}* can contain verbs that do not correspond to actions in the story and are therefore not relevant to be included as query conditions (like *meet*, *know*, *been*, *met*, *etc.*), then both of those variations would be recognized and accepted by the Interaction Model, as well as even other variants of the same question. By abstracting these non-essential words (that are not relevant for the query conditions) into slots, we can cover a wider variety of questions with fewer utterances.

While both “*Peter Barker*” and “*John Frey*” correspond to Agent conditions, Peter will always be referred as the subject of the sentence, since questions are addressed to him and his is the point of view of all the events. In order to avoid confusion, we can use a *{subject}* slot for the subject of the sentence, which will still be an Agent condition in the query.

Another thing that we should take note of is the Time conditions. In the original questions, we have three examples of a Time condition being present in a query, with two dates, a date and two times, and one date. If we just created a *{time}* slot for all the time conditions, we would have to populate it with every possible date and time combination within the agent’s domain, to ensure that every question would be understood. Given that the agent’s events cover three full days, from March 3rd to March 6th, having all those possible date and time combinations covered in the slot values would be very unlikely. Luckily, Amazon provides built-in slot types for dates and times, although they are separate. This means that we do not need to concern ourselves with predicting different ways of saying a possible date and time, but instead of just having one hypothetical *{time}* slot for all the Time conditions, we will need to unfold

---

<sup>1</sup> There is no punctuation in the Interaction Model

Question	Interaction Model
Did you meet John Frey in Silvermoon City?	{question_verb} {subject} {filler_verb} {agent} {preposition} {location}
Did you sell the stolen necklace to John Frey?	{question_verb} {subject} {action} {theme} {preposition} {agent}
Why did you travel to Silvermoon City?	Why {question_verb} {subject} {action} {preposition} {location}
Were you at the jewelry store between March 3rd and March 4th?	{question_verb} {subject} {preposition} {location} {preposition} {date_one} and {date_two}
Did you get John Frey's contact from Jaime?	{question_verb} {subject} {action} {theme} {preposition} {agent}
Did you steal something from the jewelry shop?	{question_verb} {subject} {action} {theme} {preposition} {location}
Do you know John Frey?	{question_verb} {subject} {filler_verb} {agent}
Have you been to Silvermoon City?	{question_verb} {subject} {filler_verb} {preposition} {location}
Did you talk with the delivery boy?	{question_verb} {subject} {action} {preposition} {agent}
Where were you on March 5th between 4 am and 4:30 am?	Where {question_verb} {subject} {preposition} {date_one} {preposition} {time_one} and {time_two}
Do you know the delivery boy?	{question_verb} {subject} {filler_verb} {agent}
Do you know Jaime?	{question_verb} {subject} {filler_verb} {agent}
Why were you at the jewelry shop on March 4th?	Why {question_verb} {subject} {preposition} {location} {preposition} {date_one}

**Table 5.3:** Translation to the Interaction Model

those utterances into several combination of date and time slots. Whether it be one date (like “on March 4th”), two dates (“between March 3rd and March 4th”), one date and one time (“on March 6th at 1:15 pm”), one date and two times (“on March 5th between 4 am and 4:30 am”), all these different conditions will have to be accounted for every time there could be a Time condition in a question.

Table 5.3 shows how the 13 original questions, using the new streamlined form (as seen in Table 5.2), look in their Interaction Model format, when taking all of the previous aspects into consideration. Each of those slots has their own slot type, with the time and date slots using the corresponding slot type provided by Amazon, and the remaining slot types being defined by us, either with the values defined in the story file (in the case of the query condition slots, like Agent, Theme, etc.), or with a list of words that could reasonably fit in that slots (like in the case of the question\_verb and filler\_verb slots, for example).

### 5.1.2 Preparing for the First User Study

With the thirteen questions from the original Prototype adapted into our Interaction Model, we have officially recreated the same exact functionality as the original implementation [1] - we can ask the same questions (with a new format) and obtain the same results. But these are still only thirteen questions, which do not even cover the whole timeline of the agent's story nor the full capacities of the query system (there were different types of questions that were not even showcased, Information Gathering questions with Time, Agent or Theme focuses). Without a visual interface to guide the user, or a script, it would



be very unlikely that they would be able to interact with the Virtual Suspect using just these utterances. Since we have already established a consistent style and structure for the questions in the Interaction Model, and we have already abstracted the non-essential words (that are not query conditions) into slots so that a wider range of questions can be covered, we can easily expand our current Interaction Model to allow for more questions, before allowing users to interact with the agent.

Looking at the structure of the agent's events (Figure 5.1 shows a table with all the events in the original story, and the distribution of entities per event), we can extrapolate what questions are possible to make based on those events, by seeing what combinations of slots make sense to include based on what entities the events have. By looking at Table 5.3, we can see that a question can be made with just a {question\_verb} slot, the {subject} slot, a verb slot, whether {action} or {filler\_verb}, and then different combinations of query conditions slots (like {location}, {theme}, {agent}, etc.), with some possible filler words in the middle (like {preposition} slots). A question with this basic structure can easily make a Validation question, and then one would only need to add a word like Where, When, Why, What or How in order to transform it into an Information Gathering question with the respective focus.

Event Id	Action	Real	Incriminary	Time	Location	Agent	Theme	Reason
1	Drink	t	0	1	1	1	1	0
2	Eavesdrop	t	95	1	1	1	1	0
3	Research	t	15	1	1	1	1	1
4	Buy	t	95	1	1	2	1	1
5	Watch	t	50	1	1	1	0	1
6	Rest	t	0	1	1	1	0	0
7	Watch	t	50	1	1	1	0	1
8	Follow	t	80	1	1	1	1	0
9	Rest	t	0	1	1	1	0	0
10	Prepare	t	60	1	1	1	1	0
11	Steal	t	95	1	1	1	1	1
12	Research	t	10	1	1	1	1	2
13	Rest	t	0	1	1	1	0	0
14	Get	t	50	1	1	2	1	1
15	Call	t	50	1	1	1	1	0
16	Buy	t	60	1	1	1	1	0
17	Wait	t	60	1	1	1	1	0
18	Travel	t	95	1	1	1	0	1
19	Sell	t	95	1	1	2	1	0
20	Travel	t	50	1	1	1	0	0
21	Rest	t	0	1	1	1	0	0
22	Rest	f	0	1	1	1	0	0
23	Watch	f	0	1	1	1	0	1

Figure 5.1: Entity Distribution in Events

By looking at the events, we can imagine examples of questions that can be asked about each event, with different combinations of query conditions, and then model those questions in our Interaction Model to expand its capabilities. This allows for a lot more than thirteen questions to be recognized by our Model, and will provide us with a better baseline from which to study user's interactions with the agent.

### 5.1.3 First User Study

We have recreated the original functionality of the Virtual Suspect Prototype [1]. We have expanded our Interaction Model to include more questions, and to cover not only the whole range of question types available (by adding questions of the missing Information Gathering focuses), and possibly the whole of the agent's story (depending on the query conditions). As mentioned in Chapter 4, the lying component is turned off, allowing us to better understand the interaction between user and agent. We have all the components needed to realize the First User Study. Details on the procedure and results can be found in Section 6.1.

## 5.2 Improving the Interaction

Section 6.1 contains a breakdown of the Question and Answer Problems encountered during the First User Study (see Table 6.2 and Table 6.3), organized into different categories. If we presented every single instance of each category of the Question and Answer Problems, it would probably be too long and hard to understand, as some of those problems can be very specific and irrelevant in the bigger picture. Instead, we will present a breakdown and analysis of the most prevalent and important problems, to give an idea to the kinds of changes we will have to make to the Virtual Suspect Skill.

These problems include:

**Missing intents:** questions that the users wanted to ask but the agent was not capable of answering. These include Greeting, Introspection, Thank, Get Action, and Contextual questions.

**Pronouns:** both direct pronouns (which reference a specific entity previously mentioned - it, him, there) and indirect pronouns (which hint at a potential unidentified entity - something, anyone).

**Context:** a context of the conversation is needed, to support both the Contextual questions and the direct pronoun functionality.

**Synonyms:** more synonyms in the Interaction Model to refer to the entities in the story.

**Missing information:** these can be considered to be 'gaps' in the story, information the users wanted to know but was not available. This includes more information about certain events, certain objects, or relationships between Agents (for example, *"How do you know John Frey?"*).

**Answer generation:** there were problems with the answer generation, not only with the Natural Language Generator, but also cases of no answers or too many answers being given.

**Time conditions:** there were cases where users asked questions with dates and times that were not recognized by the Interaction Model. More utterances need to be added and some code adjustments to how these are handled are also needed.

**More utterances:** besides the cases with the time slots, more utterances overall need to be added

to the Interaction Model, to reflect the kinds of questions that users asked.

**Filters in the Skill Service:** there were cases where queries with no conditions or incorrectly identified slots were going through. This can be prevented with other improvements, but we should add a filter to make sure it does not happen.

**Feedback:** a big thing that can go a long way with helping solve other problems is the feedback we give users, especially when a question does not work. By explaining what went wrong in an accessible way we can attenuate the impact of many of the other problems that we cannot solve so easily.

Some of these problems are quite specific and focused on just one component of the Virtual Suspect Skill, while others affect several components. We will now break down and explain everything we did to address these problems and improve the interaction with the Virtual Suspect Skill, separated by the different affected components: the Virtual Suspect Response Model, the Story, the Natural Language Generator, the Interaction Model, and the Skill Service.

### 5.2.1 Virtual Suspect Response Model

While we did not set out with intentions of altering the base Virtual Suspect Response Model when improving the interaction, there were nonetheless a few alterations we had to make to support the rest of the improvements. Some were minor, like adjusting the time span comparison formula in the Time Between condition predicate to more accurately identify intersections between two time intervals, or the addition of a few more condition predicates to support the indirect pronoun functionality (for example, in addition to an Agent Equals predicate there is now also an Agent Exists predicate). Some were more complex and altered more of the original Response Model.

One thing that was noticed in Agent Focus questions was that the agent was often returning himself as a value, and that ended up causing some errors. After all, in the original events, the Agent role was occupied both by Peter Barker and any other agent that was involved in the action. However, if one asked *“Who were you with on the night of March 5th?”* it is not very intuitive to reply *“I was with me and Jaime”*. Not only that, but as we discussed before, even though the two agents are equivalent in the same action, each action is clearly defined from the perspective of one of the agents. It is not the same thing to say that Peter bought something from Jaime and that Jaime bought something from Peter, those are two separate events. Therefore, to solve the problems being caused by the original Agent framework, and to safeguard for the future in case any actions from a different perspective could be added to the story, we decided to separate the roles of Subject and Agent. This way, the **Subject** is the agent who the action is centered around, and **Agent** contains any other agent that may have been involved. Now, each event requires exactly one entity in the Subject role, like it already had only one Time and one Location, and the Agent can still contain zero or more entities.

Due to user interest, we decided to add the possibility of asking questions with an Action focus. The question *“What were you doing on March 5th at 4am?”* seems perfectly reasonable, and possibly quite incriminating, to ask our suspect, but with the original framework the query focus could only retrieve entities, and an action is not an entity. In order to allow the creation of Action focused questions, we had to change the way that Actions and Entities were processed, and allow Query Results to use either Actions or Entities, instead of just Entities. This way, we could create a Focus that worked in a similar manner to the already existing focuses and retrieved the Action field from an event. Not only that, but since a lot of the Action verbs are transitive, it could also retrieve the Theme field, if it existed, since saying *“I stole the necklace”* makes more sense than just saying *“I stole”*.

Another feature we decided to add due to user interest was Knowledge questions. In this context, Knowledge can be understood to mean information regarding a specific entity rather than an event. For example, questions like *“Who is Jaime?”*, *“How do you know John Frey?”*, or *“What necklace?”* can be considered knowledge questions. Based on data from the user study, the types of knowledge we decided to add were:

**Details** - which can apply to any type of entity and contains additional information about that entity (for example, *“Who is John Frey?” “He is an art collector”*).

**Relationship** - which can apply to agent entities and refers how they relate to our Suspect (for example, *“How do you know Jaime?” “He is my friend”*).

**Value** - which only happened to apply to the Necklace entity, but can apply to any object to describe what they are worth (for example, *“How much was the necklace worth?” “It was worth three thousand euros”*).

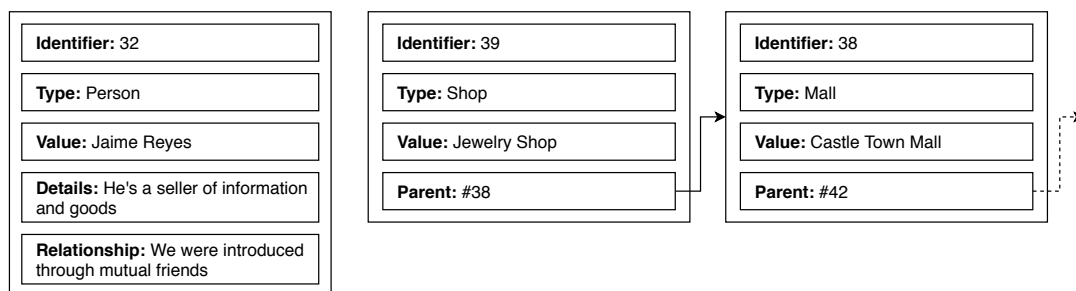
**Parent** - which applies only to location entities and describes what larger location they are contained in (for example, *“Where is the jewelry store?” “It’s in the Castle Town Mall”*). We also altered the Location condition to account for this relationship, so if the user asks *“When were you in Silvermoon City?”*, the answer would not only include events where the location is exactly Silvermoon City, but also locations inside Silvermoon City.

The way this information was added to the entities will be further explained in the Story section, but in the context of the Response Model we had to create a new Query Type (Knowledge questions) and four new Query Focuses (one for each knowledge type) to support these new questions.

### 5.2.2 Virtual Suspect Story

Like with the Response Model, we did not set out with the goal of making changes to the Story, but all the same some changes were necessary in order to improve the overall interaction. Some were more specific and addressed specific issues, like the already mentioned separation of the Subject and Agent roles, which required adjustments to be made in the definition of the events in the story. There was

also the inclusion of “knowledge” that was added to the entities in order to enable the new Knowledge questions (like “*Who is John Frey?*”, for example). These were defined in a similar manner to how entities occupy different roles in events, but instead of referencing another object, they just directly contain the information as text (similarly to how the type and value of each entity are just defined with text), with the exception of the Parent relationship, which directly references another location entity. Figure 5.2 shows a representation of how this knowledge was represented within the existing entities, with the Details and Relationship of the Jaime Reyes entity, and the Parent of the Jewelry Shop being the Castle Town Mall.



**Figure 5.2:** Knowledge Roles in Entities

Other changes were more general and not necessarily in response to specific issues. We have already mentioned how there were some details in the story that users wanted to know more about, but were unable to because they were not defined. While we could have just added those specific pieces of information, we felt that a restructuring of the story would be beneficial, in order to make the story more consistent. We kept to the original structure as much as we could, and merely added more details to flesh out parts of the story that were lacking.

Two specific examples of this were the Manner and Reason roles in events. In the original story, there was no entity that filled the role of Manner in any event, despite that being a feature of the Virtual Suspect Architecture, and despite there being a good opportunity for it in the story, given that the agent buys a gun that is then never used. So we added some entities as Manner where it made sense: he used the gun to rob the jewelry store, he used a phone to call John Frey, and he used a train to travel to Silvermoon City and Castle Town.

Another thing that was not very consistent were the Reason entities. These were entities like any other, but their Values consisted of small phrases with an infinitive verb to explain the Reason for the agent to do an Action. But they did not always make sense, for example, the Reason for the “*Steal the necklace*” event was “*To find a buyer*”, which does not feel very intuitive (“*Why did you steal the necklace?*” “*It was to find a buyer*”). Something that would make more sense would be something like “*Sell the necklace*”, which is a possible reason for one to steal something of value, and could then lead the user to ask if the agent sold the necklace, which he did, whereas there was no equivalent “*Find a buyer*” event.

These are some of the small issues that were not a problem by themselves, but when put together led us to rework the whole story in order to make it more consistent.

### 5.2.3 Natural Language Generator

The original Virtual Suspect Natural Language Generator had some flaws. Since only some question types were presented in the original Prototype, the Natural Language Generation for the remaining was not exactly calibrated to proper English, and that was an observed problem in the First User Study. But that was expected, and the study allowed us to gather more data on what was or was not working and what needed to change.

The Natural Language Generator was originally divided into different components for the different question types. One component handled the Location focus questions, another the Time, Agent, etc. Some of these only required small adjustments, others bigger overhauls, but in the end, the whole Natural Language Generator was updated to better reflect the quality of the interaction.

We also had to add new components to handle the new Action Focus and Knowledge Type questions.

### 5.2.4 Interaction Model

The Interaction Model suffered a lot of changes. Some were concentrated on just the Interaction Model, others were in service of other changes.

New Intents were added to the Interaction Model. Some were simple, like Greeting (“*Hello*”), Introspection (“*How are you*”), and Thanks (“*Thank you*”). The rest were intents in service of the new types of questions that were now possible: Action Focus questions, Knowledge questions and Manner Focus questions (which were already possible but did not make sense in the old story).

Among the old and new intents, a lot more utterances were added, to reflect the types of questions that users asked during the First User Study and all the questions that could be asked with these new possibilities. One particular aspect to note were the questions with Time slots, since we had to add a lot more different combinations of the different time slots in order to cover for more possibilities. The Interaction Model grew a lot, compared to its original state, and all the new intents and utterances covered a much wider range of possible questions.

In the slot values, we had to change some of them to accommodate the changes in the story, and we also added more synonyms to the different slot values. To each different type of slot, we added possible pronouns that could represent that slot, both direct and indirect, and including a new slot for time pronouns (seeing as that was not a single slot type that we could add new values to). This way, the {agent} slot could contain values like “*John Frey*”, “*Jaime Reyes*”, “*him*”, or “*anyone*”, and then those later values would be processed in the Skill Service.

## 5.2.5 Skill Service

As part of the changes to the Skill Service, we added a filter to check if the slot values correspond to a known slot in the story, to safeguard against incorrectly identified slots and to not lead users astray with questions that cannot be answered because they are not part of the word domain of the agent.

We added a filter to check for a minimum number of conditions in each query, depending on the query types and the context of the questions, since queries were being wrongly formed with zero conditions, which does not make any linguistic sense.

We added a filter to check if an answer did not contain any results and generate a simple answer in return (for example, *“Never”* if the question had a Time focus).

We added a filter to prevent the agent from giving too many answers to a question. This was done following the logic that a suspect should not be too eager to give too much information away, even if he was not able to lie, and that when presented with a very vague question it would be more natural to ask the user to specify what they meant rather than list every single thing. This was also done to add some challenge from the point of view of the user, since by asking a general enough question the agent could reveal almost the entire story in his answer and this way the user would have to be more thorough with their questions. For example, the question *“What did you do between March 3rd and March 6th?”* would yield *“Many things, you will have to be more specific”*, instead of recounting all of the events in the agent’s story.

We added functionality to allow users to use times of day (morning, afternoon, evening and night), instead of just specific hours, in their questions. These terms were already recognized by the default Time slot provided by Alexa, we just needed to add the logic to our Skill Service that would translate those values into appropriate time intervals. We defined morning as the period between 6am and 12pm, afternoon between 12pm and 5pm, evening between 5pm and 8pm, and night between 8pm and 12am.

We added support for the new Intents that were created. For the Greeting, Introspection and Thanks Intents we returned a simple pre-generated answer (*“Hello”*, *“I am fine. Ask your questions”* and *“You’re welcome”*, respectively). For the remaining new intents, we simply had to create those queries using the new functionality we added to the Response Model for those very purposes (in the case of Action and Knowledge questions).

We added Context and Pronoun functionality. A record of the context of the conversation was needed in order to support direct pronoun and contextual questions functionality (for example, asking *“Who did you sell it to?”* or *“Why?”* requires the agent to know what was previously said). We decided it would be sufficient for our purposes to simply keep the latest Query Result as a Context for the conversation, since it contained both the Question asked by the user (in query form) and the corresponding answer given by the agent. Therefore, if the user directly referenced a previously mentioned entity through the use of a direct pronoun (for example, *“him”* when talking about John Frey), then we simply needed to

look at the latest exchange and fetch the Agent entity that was contained in either the question or the answer. Conversely, if the user asked *“Did you sell the necklace?”* followed by *“Why?”*, then that second question could be understood to mean *“Why did you sell the necklace?”*. So, we just needed to look at the previous exchange and fetch the same conditions that were used in the previous query (and possibly include the previous response as a new condition), in order to create our new contextual query. In fact, the Contextual questions were not a new type of question at all, unlike the others we have mentioned, but simply a clever use of the Context functionality. The indirect pronoun functionality however did not use the previous context, but instead new condition predicates that used the operator *Exists* (instead of *Equals* like most already existing cases). If the user asked *“Did you meet anyone in the Castle Town Mall?”*, any event with an Agent and in Castle Town Mall would satisfy those conditions, without needing to specify which agent.

Since some entities could be used in more than one role (for example, the gun was a Theme in the Buy event, but a Manner in the Steal event), we had to account for these possible ambiguities when processing the pronouns. If the user asked *“Did you use the gun to rob the jewelry shop?”*, that gun would be acting as a Manner in that context, but if it was followed by *“Where did you buy it?”*, it would now be acting as a Theme. In those cases, instead of just fetching whatever Theme was used in the previous context, we also need to check if a gun was used as a Manner, and if no Theme was present, then this pronoun in the {theme} slot was referring to that.

Finally, we decided to give better feedback to the user in the cases where any of these steps could go wrong. By explaining that a word was not recognized, or that the question did not contain enough information, then the user could then readjust their question in an effort to be understood by the agent. Better feedback also safeguards against unforeseen circumstances, like questions we did not account for, because we can better indicate to the users what the agent can and cannot understand. For example, if the user triggered the Fallback Intent (which is the default response when an Intent cannot be identified), then the agent can respond *“I don’t think that has anything to do with what we’re talking about”*, as that indicates that what the user said is outside of the agent’s realm of understanding (ie, not identified by the Interaction Model). This type of in-character feedback could hopefully lead to longer and more successful interactions while still maintaining a fluid conversation.

## 5.2.6 Second User Study

With all these changes fully implemented, it was now time for us to conduct our Second User Test and verify if they made, in fact, for a better experience. Although we could not address every single problem that was reported in the First User Study, we could never prepare for every single thing that any user might say or ask, but with all these changes, and with better feedback, we were ready to once again present users with our Skill, test the new interaction and compare the results with our First Study.



Our hypothesis was that these changes and adjustments made the interaction better overall, and our procedure and findings can be found in Section 6.2.

### 5.3 Last Adjustments

After the Second User Study validated the improvements we made to the Virtual Suspect interaction, our next step was to turn the lying component back on and do our Third and final User Study. But before we did that, we made a few adjustments to the Virtual Suspect Skill based on some of the problems observed during the Second User Study:

- We improved the feedback given to the user by adding more detail to what was causing the question not to be understood. Previously, if a slot was not recognized, or if something went wrong when fetching the context, the message would be the same, which was not very helpful in understanding what had happened. We changed things so the agent could tell the user exactly what word he did not recognize, or why the context access failed.
- We reworked how the non-Action verbs were handled in the Interaction Model and Skill Service, allowing some verbs to have different meanings depending on the context (for example, the verb “*get*” could translate to the Action “*Buy*” if used in conjunction with the “*Gun*” theme, or the Action “*Steal*” if used with the “*Necklace*” theme).
- We added a new Intent to allow users to ask the agent what his name was, since that happened during the Second Study and the agent could not answer that simple question.
- We added a new type of Knowledge called **Residence** that represents in which of the two cities of our story the agents live and work in, since that was something that people wanted to know.
- We added a new slot to the Interaction Model called an {event} slot, used to refer to the “*robbery*” and the “*sale*”, as several users tried to do, in certain contexts.
- We did a minor rework of the story in which we mostly just changed the values of certain entities. This was a conjunction of several small reasons, the main one being in trying to avoid spelling and pronunciation errors (words like jewelry could cause issues sometimes), but also to avoid making references to other properties (Silvermoon, for example, is a city in World of Warcraft), avoid potential confusion between entities (both of the other agents in the story had names that started with the same letter), and to put our own spin on this version of the story for this version of the Virtual Suspect. We changed the jewelry store to a gallery, the necklace to a painting, we changed the names of the other agents in the story (from Jaime Reyes and John Frey to James Miller and David Turner), we changed the names of our fictional towns (from Castle Town and

Silvermoon City to Rose Town and Pacific City), and we changed the dates of the events from March to August.

### **5.3.1 Turning On the Lie**

After we made those last minute adjustments, we could finally turn the lying component of the agent back on, by reverting the initialization parameters back to their original values. Even after all the changes made to the Virtual Suspect, the lying algorithm was still working as originally implemented in the original Prototype by Rato et al. [1].

The original story as presented in the original Prototype included a few false events in the agent's story with the purpose of helping the agent lie. With our rework of the story, and since we did not have the lying component activated at the time, we did not preserve those fake events. So after we turned on the lying component and made sure it was working even with our new additions, we added a few fake events to the new story, for the purposes of helping the agent lie, same as the original story.

### **5.3.2 Third User Study**

With the final adjustments in place, and the lying component activated once more, we were ready to do our Third and last User Study. The goal of this work has been to improve the interaction with the Virtual Suspect, using the Alexa to create a new natural language conversational interface, but so far we have only worked and tested our Virtual Suspect Skill, in order to see how the interaction was working. The interaction with the lying agent has always been our ultimate goal, and now we can finally see if we have accomplished our objective.

The details and results of the Third User Study can be seen in Section 6.3.

# 6

## User Studies

### Contents

---

6.1 First User Study . . . . .	48
6.2 Second User Study . . . . .	56
6.3 Third User Study . . . . .	64

---

During the course of the development of this work, we carried out three user studies, following a user-centered approach. The first was at the beginning of the development cycle, and it served to identify shortcomings and areas of improvement in the initial prototype. The second was at the end of the main development cycle, and its purpose was to validate the choices and improvements made to the Virtual Suspect, and to verify that the User Experience was satisfactory in its base form (without the lying component). The third user study was conducted after the integration of the lying component and it served to test how the lying algorithm functioned in conjunction with the richer dialogue based interaction.

## **6.1 First User Study**

The First User Study was conducted at the beginning of the development cycle, after creating a version of the Virtual Suspect Skill that was capable of answering basic questions of every type (including those presented in the original Virtual Suspect Prototype [1]) - as described in Section 5.1. This study was conducted without the lying component active in order to get a better sense of the agent's comprehension abilities.

### **6.1.1 Objectives**

The main goal of this first user study was to gather information that would help us improve the quality of the interaction with the Virtual Suspect, and this was done in two different fronts:

1. Collect data on how people interact with the Virtual Suspect, what questions they ask, how they phrase those questions, in order to build a more robust Interaction Model.
2. Identify errors and shortcomings within the interaction, see what needs to be fixed and/or changed in order to adapt to this new natural language conversational paradigm.

The secondary goal of the study was to create a baseline evaluation of the agent's performance from which to compare and evaluate future iterations of the Skill, seeing as the only previously existing version was presented in a completely different format and was not comparable.

### **6.1.2 Scenario**

The Virtual Suspect Skill at this point was composed by an Interaction Model with some basic sample utterances for each question type supported by the Virtual Suspect Response Model, connected to a Skill Service that took the information from the intent and the slots and transformed it into a query object, which was then sent to the original Response Model to obtain the result. The story, Response Model and

Natural Language Generator were the same used in the original Virtual Suspect Prototype [1]. The lying component was turned off in order to better understand how the agent was interpreting and responding to the questions.

While our Interaction Model did allow for any query relating to the story to be formulated, there were still limitations as to what questions could be asked and understood by the agent. After all, one of the goals of this study was to study how the users interact with the agent, because there were bound to be possibilities that we had not thought of. Apart from the expectation of limitations related to the comprehension of the agent, we could also expect some limitations regarding the answer generation, seeing as the Natural Language Generator was the same as the original Virtual Suspect Prototype [1], and therefore not calibrated to all the types of questions equally.

Given these limitations, we came to three decisions regarding this first user study:

1. Regarding the information given to the user about the scenario, since the lying component was turned off, any information we would ask users to find would be immediately discovered with no effort. Because of this, and since we wanted to study how users construct questions, we decided to present users with a set of interaction goals, a list of prompts with information to discover that covered all the different question types. Users were asked to find all of the requested information, allowing us to study each of the different question types, and then invited to ask anything else they wanted of the agent, allowing us to see what other types of questions users wanted to ask. The full list of prompts is presented in the next subsection.
2. Regarding the limitations of the answer generation, we decided to conduct the study using a Wizard-of-Oz (WOZ) technique [12]. This technique is inspired by the classic movie *The Wizard of Oz*, where the magical Wizard was just an old man performing a trick behind a curtain. When applied to a human-computer interaction, it translates to having a human think they are talking directly with a machine, when instead they are talking with a human pretending to be the machine. For this study the technique would involve users communicating with a researcher via a text messaging service, in which the researcher would be posing as the agent, and the researcher would feed the questions posed by the users into the testing interface of the Amazon Developer Console, correcting any formatting errors that might cause problems, and then filtering the Skill responses to make them more grammatically consistent and readable to the user.
3. Regarding the limitations of the agent's comprehension capabilities, due to the limited domain of the Interaction Model, we decided to extend the WOZ technique into two different interactions:
  - (a) An interaction where users would indirectly interact with the Alexa Skill, with the aforementioned comprehension and answer generation limitations, which would allow us to get a fuller grasp of the existing limitations and how they affect the interaction;

- (b) And an interaction where users would interact with a human pretending to be the Virtual Suspect, with the researcher playing that role, which would mean better comprehension and answer generation capabilities and therefore allow us to analyse how users would interact with a more advanced version of the Virtual Suspect, what questions they would ask if they knew they would be understood.

This way, half the users would interact with the Alexa Skill interface and half with the human interface, while only being told they were talking with the Virtual Suspect, and we could collect more varied data.

At the end of each interaction, the conversation was logged and annotated, with a comparison between the two interfaces being made, by annotating what the Virtual Suspect Skill would have said in the Human interface conversations and vice-versa.

### **6.1.3 Materials**

The materials for this study were simple, requiring just the researcher's and user's own personal computers, an internet connection and a browser. A Facebook Messenger<sup>1</sup> account was created in the name of the Virtual Suspect (Peter Barker), and contact was established with each participant, presenting them with the set of prompts and allowing them to communicate directly with the Virtual Suspect. This connection allowed for the easy logging of the interaction, and the implementation of the WOZ technique, with Messenger serving as the curtain. After each conversation, the logs were recorded and annotated in Google Spreadsheets<sup>2</sup>.

#### **6.1.3.A Preamble**

At the beginning of each conversation, each user was presented with a set of goals for the interaction, a list of prompts with information to discover. These prompts were given in the form of "[Discover] when [the agent] did something", and they were the following:

- Where he was on March 3rd at 10am
- What he overheard in that place
- Who he met in the alley near the church
- What he bought from that person
- When he robbed the jewelry store

---

<sup>1</sup>messenger.com

<sup>2</sup>spreadsheet.google.com

- What he stole from the jewelry store
- Whether or not he sold what he stole
- If so, who did he sell it to, and where that sale took place

### 6.1.3.B Annotations

Each conversation was logged and annotated. We could not find a system to classify questions and answers according to our needs, so we had to come up with one on our own after the tests were concluded. Each interaction was annotated according to two systems: one for the **questions** and one for the **answers**.

The **question classification system** annotated the questions completely independent of the answer given by the Virtual Suspect. Each question was classified according to three fields: Intent, Focus and Domain. The Intent field sorted questions into more general categories, like Greeting, Introspection, Validation, and Information. The Focus field further classified Information questions, according to the information the user was after (similarly to query system described in Section 3.1), dividing them into categories like Location, Time, Action, Knowledge, etc. The Domain field indicated what the question was referring to, if it was events in the agent's story, information about a specific object, or just part of the conversation, like saying *"Hello"* or *"Goodbye"*. There were two additional fields that just indicated if a question was Indirect or Contextual. An Indirect question can be understood to be a question where the literal answer is different from the intended answer, like for example, asking the agent if he knows where the jewelry store is, instead of directly asking where it is. A Contextual question contains references to a previous question or answer, whether by pronouns or just follow-ups.

The goal of this question classification system was to provide a consistent framework with which to view and analyse all the questions asked by different users, and to allow us to group together similar ones. It was especially useful in identifying and classifying the types of questions that were not initially predicted by our Interaction Model. Figure 6.1 shows an example of this classification.

The **answer classification system** classified the answers given to the user in the context of the question that was asked. As we mentioned in the previous subsection, each conversation log included a comparison between the answer that was given to the user in that case (either through the Alexa or Human interface), and the answer that would have been given by the other interface to that same question. This comparison allows to evaluate each individual exchange (one question and one answer) by two characteristics:

- is the question **Answerable?** - which is the same as asking "Can this question be converted into a query with the same meaning that the Virtual Suspect is ideally capable of answering?" (in practice

Intent	Focus	Domain	Indirect?	Contextual?	Utterance
Greeting	--	Unrelated	No	No	Hello
Introspection	--	Unrelated	No	No	How are you?
Clarification	--	Unrelated	No	No	What do you mean?
Information	Location	Suspect's Events	No	No	So, where were you on March 3rd?
Information	Location	Suspect's Events	No	No	Where were you on March 3rd at 10am?
Validation	--	Suspect's Events	No	Yes	Were you with someone?
Validation	--	Suspect's Events	No	Pronoun	Did you hear something while you were there?
Information	Theme	Suspect's Events	No	No	What did you overhear at the mall's coffee shop?
Information	Theme	Other's Events	No	Pronoun	What did she say?
Information	Location	Suspect's Events	No	Pronoun	Where did you go after that?

**Figure 6.1:** Question Classification System

this can be seen by whether or not the Human interface was capable of giving an answer, as it operated by this criteria);

- was the question **Recognized?** - which means "Did the Virtual Suspect interpret this question in the same way that the user intended?" (in practice this can be seen by whether the answers of both interfaces are matching).

Apart from that classification, each exchange was logged with any problems that may have occurred. These problems could be anything that caused the user's question to not produce a proper answer from the agent, from technical and semantic errors, to gaps in the interaction. There were divided into to categories: **Question Problems** and **Answer Problems**.

**Question Problems** do not only include problems with the question itself, but also any reason why that question may not have been Answerable or Recognized, whether it be problems in the Interaction Model, or missing concepts from the story.

**Answer Problems** are anything that went wrong after the question was processed by the Interaction Model, whether a mistake in the Skill Service function, or a wrong answer being generated.

All the Question and Answer Problems were sorted into appropriate categories, and an additional Details field for each Problem field contained a specific description of each case. Even in the cases of the Human interface conversations, these Problems referred to the hypothetical answers given the by the Alexa interface, since the goal of this classification system was to identify what needed to be fixed and improved in the Virtual Suspect Skill. Figure 6.2 shows an example of this classification in an Alexa interface conversation.

The question classification system was helpful in coming up with consistent classifications for the



Utterance	Alexa Answer	Filtered Answer	My Answer	Answerable?	Recognized?	Question Problems	Details	Answer Problems	Details
Hello	I have no idea what you are talking about.	I have no idea what you are talking about	Hello	No		Missing Intent	Greeting	Nothing	
How are you?	Yes	Yes	I am fine	No		Missing Intent	Introspection	Back-end	No conditions
What do you mean?	I don't know.	I don't know	I don't know	No		Missing Intent	Clarification	Back-end	No slot filtering
So, where were you on March 3rd?	I was at the mall's coffee shop twice, at my house once, the alley near the church once, and on the jewelry shop once.	I was at the mall's coffee shop twice, at my house once, the alley near the church once, and on the jewelry shop once.	You'll have to be more specific	Yes	Yes	Nothing		Back-end	Too many answers
Where were you on March 3rd at 10am?	I was at the mall's coffee shop .	I was at the mall's coffee shop	I was at the mall's coffee shop	Yes	Yes	Nothing		Nothing	
Were you with someone?	No	No	I don't know	No		Pronoun	Someone	Back-end	No pronouns
Did you hear something while you were there?	No	No	Yes	Yes	No	Question structure Pronoun	While you were Something, there	Back-end	No slot filtering No pronouns
What did you overhear at the mall's coffee shop?	I eavesdropped the jewelry shop owner's conversation	I overheard the jewelry shop owner's conversation	I overheard the jewelry shop's owner conversation	Yes	Yes	Nothing		Synonym	Eavesdrop/Overhear
What did she say?	I don't know.	I don't know	I don't know	No		Question structure Pronoun Missing information	Indirect subject She No information about the conversation	Back-end	No slot filtering No pronouns
Where did you go after that?	I don't know.	I don't know	I don't know	No		Question structure	Event sequence	Back-end	No slot filtering

Figure 6.2: Answer Classification System

answer classification system, like the designations for questions that did not have an intent, or identifying questions that were not answerable or recognized because they were indirect or contextual.

### 6.1.4 Procedure

This was a more informal study, with twelve participants and no demographic information collection. The people asked to participate were fellow Computer Science students.

The interaction took place through the use of a Facebook Messenger text conversation, using an account in the name of Peter Barker. The researcher and participant were sitting in the same room, each on their own computer, with no access to the other's screen. The participant was sent the list of prompts with information they had to discover and encouraged to ask anything else they might want to. Following the WOZ technique [12], the participants were told that they were talking directly with the Virtual Suspect, and had no knowledge of which interface they were interacting with (the Peter Barker account helped with that immersion). For the sake of time consistency between the two interfaces, when the participants sent a question, the responses of both interfaces were calculated and immediately logged, with only the relevant one being sent back to the user. This process consisted of copying the question to the Alexa Developer Console, logging both the original Skill response and the edited, more grammatically consistent response, as well as logging the hypothetical ideal Virtual Suspect response.

The interaction ended when the participant had no more questions to ask, by their own admission and as they were instructed at the beginning of the conversation. After the conversation was finished, the participant was thanked for their help, and the interaction log was annotated by the researcher according to both classification systems.

### 6.1.5 Results

In total, 12 conversations were logged and annotated, 6 for each interface. In those 12 conversations, there were a total of 284 exchanges (each exchange is composed of one question and one answer), with an average of 23.67 exchanges per conversation. If we split these between the two interfaces, the Alexa interface had a total of 158 exchanges, with an average of 26.33, and the Human interface had a total of 126 exchanges, with an average of 21.

For the sake of simplicity, and for future reference, let us establish that an exchange that is both Answerable and Recognized is called a Positive exchange, one that is Answerable but Not Recognized is a Failed exchange, and one that is Not Answerable is a Negative exchange - see Table 6.1. It is not possible to have an exchange that is Not Answerable and Recognized, as if it is Not Answerable then by definition it cannot be Recognized.

	Answerable	Not Answerable
Recognized	Positive	
Not Recognized	Failed	Negative

**Table 6.1:** Exchange Classification

Out of the 284 total exchanges, 103 were Positive exchanges, 92 were Failed exchanges, and 89 were Negative exchanges. If we define a conversation's success rate as the percentage of Positive exchanges, and the Skill's success rate as the average of every conversation's success rate, then the overall success rate was 37.29%. If we follow that same logic, but split the two interfaces, then the Alexa interface success rate was 43.28%, and the Human interface success rate was 31.31%.

Nothing	102	35.92%
Pronoun	59	20.77%
Question structure	59	20.77%
Missing information	46	16.20%
Missing intent	41	14.44%
Synonym	13	4.58%

**Table 6.2:** Question Problems Breakdown

Back-end	164	57.75%
Nothing	64	22.54%
Answer generation	47	16.55%
Synonym	10	3.52%

**Table 6.3:** Answer Problems Breakdown

Table 6.2 shows the breakdown of Question Problems across the 12 conversations, and Table 6.3 shows the same for the Answer Problems. Since some exchanges can have more than one problem, the percentages do not add up to 100% in either case, but they show the percentage of exchanges out of the total 284 that contain each problem.

Table 6.2 shows that while the exchanges with no Question Problems (102 exchanges) managed to be the plurality, they are still in the minority compared to the exchanges that do present problems, with Pronoun and Question structure being the most common ones. Here Pronoun indicates that the question contained a pronoun that the Virtual Suspect was unable to process, and Question structure

is more of a catch-all term for problems with the question itself (which can include indirect questions, lack of prepositions, or subjects other than Suspect).

Table 6.3 shows that the majority of exchanges contained Answer Problems, with the most common one being Back-end, which is a catch-all term for problems with the Skill Service function logic (it can include the lack of contextual awareness, or the lack of pronoun functionality, for example).

### 6.1.6 Discussion

While the success rate for this version of the Skill was not very high (as we could expect given the already known limitations), we gathered a lot of insightful information about those limitations. Despite already expecting some of the Question and Answer Problems, we were able to have a more concrete basis of evidence with which to tackle these problems when improving the Virtual Suspect Skill. Beyond that, we managed to identify plenty of new Intents and potential Utterances that should be added to the Interaction Model. By doing this User Study we were able to have a much more thorough understanding of the limitations and shortcomings of this version of the Virtual Suspect Skill, which allowed us to systematically approach the most important issues during the next phase of development.

There is not much to say about the results of the conversations, as we do not have anything to compare them with. After all, the secondary goal of this study was to establish a baseline performance with which we can compare future versions of the Virtual Suspect Skill.

In order to discuss the differences in results between the Alexa and Human interfaces, let us remember that the Answerable/Recognized classification of the exchanges in the Human interface conversations were made regarding the Alexa answers to those same questions. When talking with the Alexa interface, users would often have to repeat their questions and rephrase them slightly in order to be understood, while with the Human interface those original questions would be more easily understood. For example, the question *“Where were you March 3rd at 10am?”* would not be immediately understood by the Alexa interface, due to a lack of prepositions, but it would be understood by the Human interface. So, in the case of the Alexa interface, the user would have to rephrase the question in order to be understood, but in the Human interface they would obtain their answer and move on. This leads to the larger number of exchanges in the Alexa interface, and since the success rate is measured based on the Alexa interface response in either interface, the lower success rate in the Human interface, as users would not have adjusted their manner of speaking in order to be understood.

In conclusion, this was a successful study, because it allowed us to identify both new and expected problems with the interaction (and get a better understanding of their effect on the interaction), and it provided us with a baseline with which to compare future iterations of the Skill.

## **6.2 Second User Study**

The Second User Study was conducted at the end of the main development cycle, after making all the changes and improvements to the Virtual Suspect Skill based on the problems found during the First User Study - as described in Section 5.2. Once again, this study was conducted without the lying component active, to replicate the conditions of the first study and to better measure the agent's abilities.

### **6.2.1 Objectives**

After all the changes made to the Virtual Suspect Skill as detailed in Section 5.2, and supported by the data collected in the First User Study (Section 6.1), the goal of this second study was to verify that those changes and improvements led to a better, more natural and more fluid interaction. We can measure that by comparing the agent performance metrics to what was recorded in the First User Study (number of exchanges, success rate, and number of problems). We will also formally measure the UX of interacting with the Virtual Suspect Skill, in this improved state.

Those two metrics, the agent's performance and the UX, will allow us to determine if the changes we made improved the interaction, and to measure how good the interaction is in its most basic form (just questions and answers about the agent's story, no lying).

### **6.2.2 Scenario**

The changes made to the Virtual Suspect Skill greatly improved its ability to understand and process a wider variety of questions, and lessened the impact of unanswerable questions by providing better feedback. Thus, the limitations that were present during the First User Study were no longer imposed. The agent was not limited to as few questions as before, and we no longer needed to focus on collecting data surrounding specific types of questions. This, combined with the lying component being turned off, meant that users could have a lot more freedom in the way they interacted with the Virtual Suspect.

With this state of interaction in mind, we decided to change the paradigm of how users interacted with the agent in this study. Instead of giving users specific prompts with information to discover, we would allow users to freely explore all of the agent's story, asking whatever questions they wanted to. This still presented the challenge of what information to give to the user, because if we gave no information at all then the users would not know where to start or what the scope of the conversation was. After all, the interaction was designed around enabling the types of questions supported by the Virtual Suspect Architecture, not around being prepared for anything any user might say and then guide them to the relevant information.

In order to take advantage of the Virtual Suspect capabilities we spent time developing and adapting into this new conversational interaction, it would be in our best interests to guide users into the scope of

that interaction, presenting them with the context of the conversation (interrogation of a crime suspect) and the confines of the story (it took place between March 3rd and March 6th, in the fictional towns of Castle Town and Silvermoon City, and centered around the stealing of a necklace). This way, we could give users enough information to situate themselves in the context of the conversation and start asking questions and exploring the rest of the story. For that purpose, a set of instructions was written, containing the previously mentioned information, and giving some very general indication of what the agent was capable of doing (the interaction was limited to questions about the story after all), with the goal of providing users with enough information to take full advantage of the Virtual Suspect Skill.

With the new interaction paradigm defined, we still needed to define the method of the interaction. Due to limitations enforced because of the global Covid-19 pandemic, we had to do our tests remotely. Because of this, we decided to use the same technique that we used for the First User Study, since it could easily be adapted into being done remotely, and it was already proven to work. We would use the same technique of using a text messaging service to establish communication between the user and the agent, with a human component in the middle relaying the messages to the Alexa Developer Console to obtain the agent's responses. However, since the limitations that were present during the First User Study were no longer an issue, we did not need to use the same WOZ technique [12] and the two interfaces to control the information presented to the user. This time, the researcher would not alter the responses given by the Virtual Suspect, and the only intervention would be possible corrections to formatting (like numerals or punctuation) that could sometimes interfere with the parsing of the question but did not change its meaning.

### **6.2.3 Materials**

The materials for this study were very similar to the first one, as it only required the researcher's and participant's own computers, an internet connection and a browser. The same Facebook Messenger account with the name of the suspect was used to establish a connection to the user, and the conversations were also logged in a Google Spreadsheet. A Google Document<sup>3</sup> containing the Interaction Instructions was presented to the user before the interaction, and a Google Form<sup>4</sup> containing a Questionnaire designed to evaluate the UX.

#### **6.2.3.A Instructions**

As described in the previous subsection, a set of instructions was written and presented to the users before beginning the interaction, with the intent of providing the necessary context for the conversation. These included a description of the setting (a conversation with an autonomous agent in the context of a

---

<sup>3</sup>[docs.google.com](https://docs.google.com)

<sup>4</sup>[forms.google.com](https://forms.google.com)

police interrogation, with the agent playing the role of suspect), the goals of the interaction (discover as much of the agent's story as possible), information about the story (when, where, what, etc.), as well as a few general guidelines to manage the user's expectations of the agent's capabilities (he can understand context but not the weather, for example). These instructions are included in full in Appendix A.

### 6.2.3.B Questionnaire

For the purposes of collecting data for this User Study, we constructed a questionnaire with four parts.

The first part was to be answered before talking with the agent, and contained general information collection (age, gender, fluency in English, experience with chatbots) in order to characterize the study sample. Fluency in English was measured through a self evaluation on a scale of 1 to 10, and for experience with chatbots participants were asked how many times they have interacted with a chatbot (Siri, Cortana, Google Assistant, Alexa, Cleverbot, etc.), with the options being Never, 1-2 times, 3-5 times, 6-10 times, and 11+ times.

The last three parts were answered after the conversation with the Suspect and measured the UX, the conversation flow, and general feedback.

The UX evaluation was heavily based on the User Experience Questionnaire (UEQ) developed by Schrepp et al. [2, 13]. The UEQ is a questionnaire designed to evaluate UX by presenting users with 26 items divided across 6 different scales - Attractiveness, Efficiency, Perspicuity, Dependability, Stimulation, and Novelty (see Figure 6.3).

**Attractiveness** measures the general appeal of the product,

**Efficiency** measures how easily users can solve tasks,

**Perspicuity** measures how easy to understand the product is,

**Dependability** measures if the user feels in control of the interaction,

**Stimulation** measures whether the product is exciting to use,

**Novelty** measures if the users consider the product to be creative.

Each of these items consists of a word pair representing opposing concepts (for example, Attractive and Unattractive) and users select on a scale of 1 to 7 how much they think those descriptors fit the product they are evaluating (in this case a 1 would mean totally Attractive, a 7 totally Unattractive, and a 4 would be equally Attractive and Unattractive), according to the order of the words. When utilizing the UEQ as part of a UX questionnaire, it is possible to omit certain scales entirely if they do not make sense in the context of the product [2], and we felt that the items in the Efficiency and Dependability scales (for example, slow/fast, cluttered/organized, not secure/secure, obstructive/supportive) did not necessarily make sense for our evaluation, so our questionnaire only included the items in the Attractiveness, Perspicuity, Stimulation and Novelty scales. As Google Forms does not allow binary evaluation grids (with one word at each end of the scale), we had to separate the word pairs. We kept the same 1 to 7 scale, so we

could then calculate the value of each item by combining the values of both words, and analyse the data as recommended by the authors of the questionnaire.

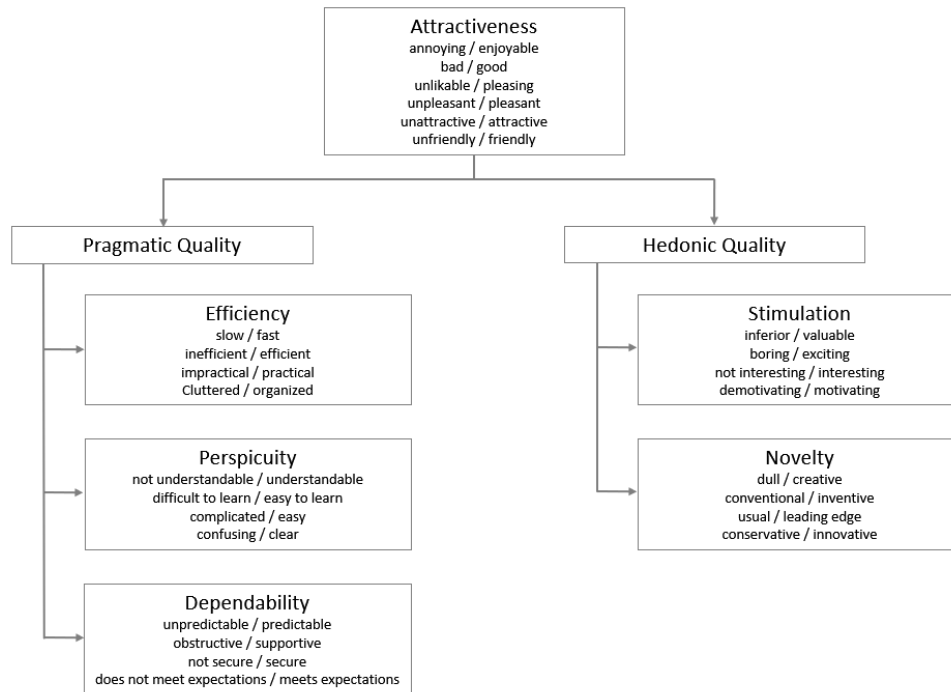


Figure 6.3: UEQ Scales and Items [2]

The conversation flow evaluation is composed of a series of statements written by us regarding the specific context of our Skill, that users can indicate their agreement with on a 1-7 scale. These statements are divided into the domains of: Agent Capabilities, Consistency, Experience and Limitations.

**Agent Capabilities** measures the user's perception of the agent's ability to understand natural language.

**Consistency** measures the consistency of the agent's answers in the context of the user's questions,

**Experience** measures the adequacy of the experience overall (for example, if users felt successful in uncovering the story),

**Limitations** measures the user's perception of the agent's limitations (and how much they impeded the interaction).

Table 6.4 shows the list of questions and their corresponding scale. Each question also had either a Positive or Negative Valence, meaning that in questions with a Positive Valence the best result would be a high number on the scale, and for those with a Negative Valence the best results would be a low number on the scale.

The general feedback was an open-answer collection of positive and negative aspects about the experience.

<b>Question</b>	<b>Scale</b>	<b>Valence</b>
The agent wasn't able to understand what I was asking	Agent capabilities	Negative
The agent changed subjects without me wanting to	Consistency	Negative
I established a fluid conversation with the agent	Consistency	Positive
There were a lot of unexpected interruptions	Limitations	Negative
The agent's answers corresponded to my questions	Consistency	Positive
The agent's answers weren't grammatically consistent with my questions	Consistency	Negative
The content of the agent's answers was expected considering the content of my questions	Consistency	Positive
The agent's capabilities met my expectations	Agent capabilities	Positive
The agent is able to understand natural language	Agent capabilities	Positive
I had to change my manner of speaking in order to be understood	Limitations	Negative
I was able to naturally communicate with the agent	Consistency	Positive
The agent is intelligent	Agent capabilities	Positive
I didn't feel like I was having a proper conversation with the agent	Consistency	Negative
I was able to uncover as much of the agent's story as I wanted to	Experience	Positive
The agent's limitations kept me from exploring the story as much as I wanted to	Limitations	Negative
The agent has too many limitations	Limitations	Negative
The agent's answers often broke my line of thought	Experience	Negative
I was often not sure what to ask or say	Experience	Negative
I was given enough information to question the agent	Experience	Positive
I managed to find more information than what I initially had by questioning the agent	Experience	Positive

**Table 6.4:** Conversation Flow Evaluation Questions



This questionnaire is not presented in Appendix A as it is very similar to the questionnaire used in the Third User Study. The differences are explained in Section 6.3 and that questionnaire is included in Appendix A.

### **6.2.3.C Annotations**

Like in the previous User Study, each conversation was logged and annotated, using the same system as before, allowing us to compare the performances. As the Skill has been changed and improved, the problems that occurred during the First User Study will not be exactly the same as the problems that occur now. In fact, we expect less problems to occur, but there will always be questions that the Virtual Suspect cannot answer for one reason or another. So, as the categories of the Question and Answer Problems during the First User Study represented the state of the Virtual Suspect Skill at the time, so will the categories in this study reflect the state of the interaction when it was realized, with some categories disappearing, some remaining, and a few new ones appearing. For example, the category of “Question structure” remained, the category of “Pronoun” was resolved because we added that functionality to the Virtual Suspect, and the category of “Context” appeared to denote errors specifically related to the Context functionality.

Apart from that, the only difference to the annotation system is the addition of a new Improvements field, as there were some occurrences during the interactions that no longer constituted problems but could still come to be improved in the future, so a note was made of those. The question classification system remained the same.

### **6.2.4 Procedure**

This study counted with the participation of fourteen people, some had already participated in the First User Study, others were newly invited to participate.

Like in the First User Study, the interaction between the users and the agent happened via Facebook Messenger. However, since now the participant and researcher could not be in the same room, a voice call between the two was maintained throughout the experience. After the voice call was initiated, the researcher sent the participant a link containing an Informed Consent Form, informing them that their participation would be anonymous and voluntary and no personal information would be collected. If the user agreed to participate, they were automatically redirected to the first part of the questionnaire with the demographic information collection. All fourteen participants agreed to the terms of participation.

When the participant finished the first part of the questionnaire, they were indicated by the questionnaire to tell the researcher they were ready to begin the interaction with the Virtual Suspect, and to wait before concluding that before proceeding with the questionnaire. The researcher sent them the document containing the interaction instructions, and after they had read and understood, the interaction

with the agent was initiated through the Peter Barker account. The interaction was completely user-driven, the participant had the freedom to ask whatever they wanted of the suspect, with the stated goal of discovering more information about the story, and the interaction lasted for as long as the user had questions to ask.

When users indicated they had no more questions they wanted to ask, they were directed to go back and finish answering the questionnaire. After the participant finished answering the questionnaire, the researcher thanked them for their participation and the call ended. Since the interaction with the agent was user-driven, these sessions could last anywhere between 20 to 80 minutes, at the extremes. With the interaction concluded, each conversation was logged and annotated.

## 6.2.5 Results

We had 14 people participate in the study. The age range was 20 to 26, with the median age being 22. 7 participants identified as Male, 6 identified as Female, and 1 identified as Other. The levels of self-evaluated fluency in English ranged between 5 and 10, on a scale of 1 to 10, with the median being 8. All of the participants had interacted with a chatbot (Siri, Cortana, Google Assistant, Alexa, Cleverbot, etc.) at least once, with the most common answer being 3-5 times, so the users were already familiar with this type of interaction.

In the 14 conversations that were logged and annotated, there were a total of 757 exchanges, with an average of 54.07 exchanges per conversation. Out of those 757 exchanges (and recalling Table 6.1), 474 were Positive, 138 were Failed, and 145 were Negative. The success rate, as we defined in Section 6.1 (being the average of each individual conversation's percentage of Positive exchanges) was 63.39%.

Nothing	493	65.13%
Missing information	142	18.76%
Question structure	94	12.42%
Event reference	18	2.38%
Relativeness	14	1.85%
Missing intent	11	1.45%
Numbers	1	0.13%

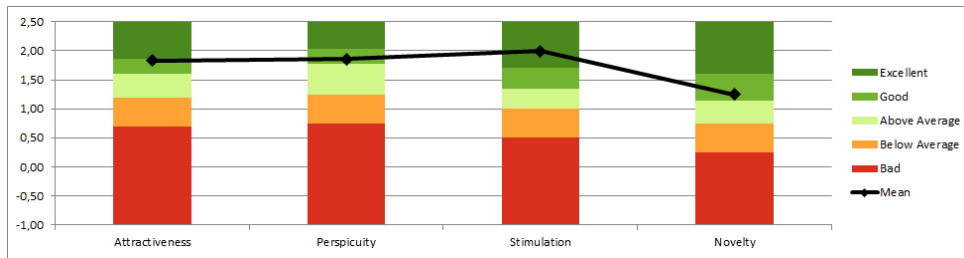
**Table 6.5:** Question Problems Breakdown

Nothing	680	89.83%
Reprompt	46	6.08%
Context	16	2.11%
Word ignore	10	1.32%
Midnight	2	0.26%
Value error	2	0.26%
Conditions	1	0.13%

**Table 6.6:** Answer Problems Breakdown

Table 6.5 and Table 6.6 show the breakdown of Question and Answer Problems according to category, respectively. As some exchanges may have more than one problem, the percentages may not add up to 100%, but show the percentage of exchanges with that problem, out of the total 757. In both cases, the majority of exchanges did not contain any problems, with 493 (65.13%) not having Question Problems, and 680 (89.83%) not having Answer Problems.

Figure 6.4 shows the result of the UEQ portion of our questionnaire, as calculated with the Excel tool



**Figure 6.4:** UEQ Benchmarks for Second Study

made available by the authors of the UEQ in their website<sup>5</sup>. The images with the results for each individual item and for the scale means can be seen in Appendix A, but Figure 6.4 shows our results when compared to the benchmark defined by the authors [2] for each of the scales. The results for the four scales we measured in our questionnaire are very good, with Attractiveness, Perspicuity, and Novelty falling into the Good category of the benchmark, and Stimulation falling into the Excellent category.

The full results of the Conversation Flow evaluation can be seen in Appendix A, with one image showing the mean value of each question in the conversation flow section of our questionnaire (after adjusting for the Valence), ordered from worst to best result, and another showing the mean values and confidence intervals for each scale. What these results tell us is that participants overwhelmingly managed to find more information than they started with, did not have the subject of the conversation change abruptly, and felt they were given enough information to question the agent. However, they did feel like they had to change their manner of speaking too much in order to be understood by the agent, they were often not sure what to ask the agent, and they felt the agent had too many limitations. These are just the top and bottom items in terms of results, but they can tell us which areas of the interaction are doing good, and which need more work.

With regards to the open-answer feedback questions, here is what some people said on the Positive Aspects:

- “The questions it understood were all answered in an **adequate and understandable way**”
- “He was very direct in the way he answered. Also the **story and context were very interesting and made me want to discover more** and keep inquiring the agent.”
- “**The agent was able to keep context** most of the time and gave the correct information when the questions were asked in an understandable way.”
- “I like that he doesn’t specify every time slot he spent at each place, it **adds to the experience** and makes us work harder for an answer. He speaks naturally enough and it’s not difficult to follow his thought process.”

<sup>5</sup>ueq-online.org/

And about the Negative Aspects:

- “When I asked for intention Peter was often confused, and **intention would help me understand the context of the story.**”
- “It would be really cool **if the agent lied**, and I had to find the inconsistencies in his lies.”
- “The agent was lacking some specific details about his actions **like why he did certain actions** or what information he acquired on the calls. He also has a limited amount of verbs he recognizes.”

Overall, most of the answers on the Negative Aspects were about more specific issues that users encountered during their conversations, or about technical limitations. The Positive Aspects mostly centered around how the agent’s capabilities and the scope of the story exceeded the expectations of the users, and how they enjoyed the interaction.

## 6.2.6 Discussion

As we can see in the higher success rate of the agent (65.13% as opposed to the previous 37.29%), the higher average number of interactions (54.07 versus 23.67), and the much lower rate of problems in most exchanges (comparing Table 6.2 and Table 6.3 with Table 6.5 and Table 6.6), the quality of the interaction went up tremendously when compared to the First User Study. This proves that the changes and improvements we have made to the Virtual Suspect Skill were successful, which validates our goal of showing a better interaction, and further validates the data we collected in the First User Study.

As for the UEQ results, they are very promising, as all the scales we tested scored in either the Good or Excellent categories. While the conversation flow results do not give a solid conclusion, we can see which areas of the interaction are working better or worse by looking at which items scored better or worse, and we can that no item scored below the median score, which is a positive.

Overall, this was a very successful study, as it validated the improvements we have made to the Virtual Suspect Skill and proved that our interaction has a good UX.

## 6.3 Third User Study

The Third User Study was conducted after making some final adjustments following the Second User Study, and after turning on the lying component - as described in Section 5.3. With the lying component active, this was how the interaction with the Virtual Suspect was originally envisioned by Rato et al. [1], and the changes and improvements we have made were all in service of easing that interaction.

### **6.3.1 Objectives**

After two user studies and a whole development cycle with the lying component of the Virtual Suspect turned off, the goal of this study was to study how the lying algorithm affected the interaction which has been proven to achieve good results on its own, and to measure the quality of the interaction itself and the UX with the lying agent, using the same metrics that we have used before in the Second User Study (see Section 6.2).

### **6.3.2 Scenario**

With the lying component being active, the way that users can interact with the agent changes, because the agent will lie and obfuscate his answers, and it will be much more difficult for users to extract information from the agent without already having information of their own with which to incriminate him. Due to the way the agent lies, the style of free exploration of the story that we adopted in the Second User Study would be very difficult to achieve, without providing the user with information about every single thing the agent has done. With this in mind, we decided to narrow the focus of this interaction, by providing the user with more details about the case, and directing them to focus their attention on making the agent confess the crimes he committed. So we rewrote the interaction instructions with this goal in mind.

Other than the changes to the interaction parameters brought on by the lying component, nothing else changed from the Second User Study. We were still limited to doing remote testing, so we once again adopted the same technique that we have been using in order to let participants interact with the Virtual Suspect and record their conversations, and other than the same formatting issues required by the Alexa Developer Console, the researcher did not interfere with the interaction.

### **6.3.3 Materials**

The materials for this study were exactly the same as the Second User Study. Facebook Messenger was used to establish the interaction between the user and the agent, the conversations were logged using Google Spreadsheets, and a Google Document containing the interaction instructions was presented to the participants, as well as a Google Form containing the questionnaire.

#### **6.3.3.A Instructions**

As explained in the previous subsection, the instructions suffered some changes brought on by the lying agent. We included more details about the case, and directed users to focus their efforts on making the agent confess to his crimes. The document once again contained details about the context of the

interaction (explaining what the Virtual Suspect is and how the interaction will work), and some general guidelines about the agent's capabilities. These instructions are presented in full in Appendix A.

### 6.3.3.B Questionnaire

The questionnaire included in this study was very similar to the one used in the Second User Study, and it was also divided into four parts.

The first part collected the same demographic information as the previous questionnaire (age, gender, self evaluated fluency in English, and experience with chatbots), with the addition of asking whether the participant was a native English speaker.

The second part was exactly the same as the previous questionnaire, using the UEQ to allow us to measure the UX of the new interaction and allowing us to compare with the results of the previous study.

The third part was very similar to the previous questionnaire, but we added some new questions and reworked others in order to reflect the new interaction. We added two new question domains in addition to the already existing ones, which were **Conversation** measuring aspects specifically about the quality of the conversation with the agent, and **Lying**, measuring the user's perception of the agent's lying capabilities. While the questions of the Lying domain were completely new due to the new state of the interaction, the questions in the Conversation domain were mostly already existing items that made more sense in their own category. Table 6.7 contains the new conversation flow evaluation questions, and the respective domains.

The fourth and final section was the one that suffered the most changes, including some multiple choice questions in which users were asked if the agent lied, if the agent told the truth, if the agent confessed to the crimes detailed in the instructions document, and which crimes he confessed to. On top of that, there were also two open-answer questions in which users were asked to recount what they thought had really happened in the story based on what they discovered, and to describe the limitations they encountered with the system. This questionnaire is presented in full in Appendix A.

### 6.3.3.C Annotations

Once again, each conversation was logged and annotated, using the exact same system as the Second User Study. While this time we cannot simply trust the agent's answers in order to figure out if the question was Recognized and if something went wrong, because of the lies, we can still make that classification just by looking at the questions and the application logs when necessary.

<b>Question</b>	<b>Scale</b>	<b>Valence</b>
The agent understands natural language	Agent capabilities	Positive
The agent didn't understand what I was asking	Agent capabilities	Negative
The agent has intelligence	Agent capabilities	Positive
The agent is less capable than I was expecting	Agent capabilities	Negative
The answers of the agent weren't grammatically consistent with my questions	Consistency	Negative
The answers of the agent corresponded to my questions	Consistency	Positive
The agent changed topics unexpectedly	Consistency	Negative
The agent answered the questions I asked	Consistency	Positive
I was able to establish a fluid conversation with the agent	Conversation	Positive
I was often not sure of what to say or ask	Conversation	Negative
I didn't feel like I was having a proper conversation with the agent	Conversation	Negative
I was able to naturally communicate with the agent	Conversation	Positive
I didn't have enough information to question the agent	Experience	Negative
I was able to uncover more information than I had initially	Experience	Positive
The information I was given before hand did not correspond to what I encountered	Experience	Negative
I felt like I was interrogating a crime suspect	Experience	Positive
There were many errors in the interaction	Limitations	Negative
I wasn't able to ask the questions I wanted because of the agent's limitations	Limitations	Negative
I didn't have to change my manner of speaking in order to be understood	Limitations	Positive
The limitations didn't keep me from interacting as I wanted with the agent	Limitations	Positive
The agent always told the truth	Lying	Negative
The agent contradicted himself	Lying	Negative
The agent's lies were plausible	Lying	Positive
I was able to easily tell truth from fiction in the agent's answers	Lying	Positive

**Table 6.7:** New Conversation Flow Evaluation Questions

### 6.3.4 Procedure

This study counted with the participation of sixteen people, some had already participated in both of the previous user studies, some had only participated in the second, and some had never interacted with the Virtual Suspect.

The procedure was exactly the same as the Second User Study. A voice call was initiated between the researcher and participant, and maintained throughout the whole session, the participants were informed of the terms of the study (everything was anonymous and voluntary, no personal information was collected) and all users agreed to participate. They were presented with the interaction instructions document, and after they had read and understood, the interaction with the Virtual Suspect was initiated through the Peter Barker account, as before. Participants could ask the Suspect whatever they wanted for as long as they wanted, and when they were satisfied they indicated to the researcher that they did not wish to continue the conversation and were directed to answer the questionnaire. This time around, users filled the first part of the questionnaire at the same time as the rest, after the conversation with the agent.

After the participants finished answering the questionnaire, they were thanked for their participation in the study and the call ended. Since the length of the interaction depended only on the user asking the questions, these sessions could last anywhere between 20 and 60 minutes. With the interaction concluded, each conversation was logged and annotated.

### 6.3.5 Results

16 people participated in this study. The age range was 20 to 27, with the median age being 22. 8 participants identified as Male and 8 participants identified as Female. All 16 participants were non-native English speakers. The levels of self evaluated English fluency ranged between 7 and 9, on a scale of 1 to 10, with the median being 8. Only one of the participants had never interacted with a chatbot (Siri, Cortana, Google Assistant, Alexa, Cleverbot, etc.), but the most common answer was 11+ times, indicating a greater familiarity overall with this type of interaction.

In the 16 conversations that were logged and annotated, there were a total of 738 exchanges, with an average of 46.13 exchanges per conversation. Out of those 738 exchanges (using the definitions in Table 6.1), 487 were Positive, 111 were Failed, and 140 were Negative. Using the same definition for success rate as the previous two studies, which is the average of each individual conversation's percentage of Positive exchanges, this time it was 65.01%.

Table 6.8 and Table 6.9 show the breakdown of Question and Answer Problems according to category, respectively. In this study, no exchange was marked with more than one category of problem, so the percentages add up to 100%. In both cases, the majority of exchanges did not contain any problems,



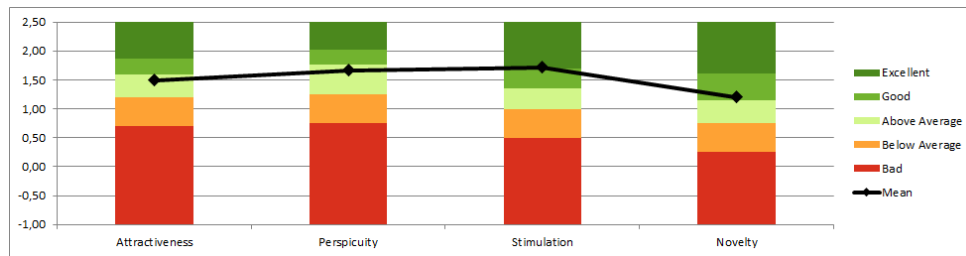
Nothing	493	66.80%
Question structure	105	14.23%
Missing information	96	13.01%
Missing intent	30	4.07%
Relative time	12	1.63%
Event reference	1	0.14%
Numbers	1	0.14%

**Table 6.8:** Question Problems Breakdown

Nothing	653	88.48%
Reprompt	78	10.57%
Context	4	0.54%
Conditions	2	0.27%
Word ignore	1	0.14%

**Table 6.9:** Answer Problems Breakdown

with 493 (66.80%) not having Question Problems, and 653 (88.48%) not having Answer Problems.



**Figure 6.5:** UEQ Benchmarks for Third Study

Figure 6.5 shows the result of the UEQ portion of our questionnaire, once again using the UEQ tools provided on the website. The images with the results for each individual item and for the scale means can be seen in Appendix A, but Figure 6.5 shows our results compared to the benchmarks defined by the authors [2] for each of the scales. In the scales we evaluated, Attractiveness and Perspicuity scored in the Above Average benchmark, Novelty scored Good, and Stimulation scored Excellent.

The full results of the Conversation Flow evaluation can be seen in Appendix A. What we can tell from the items that did the best is that users overwhelmingly felt that they had enough information to question the agent, were able to identify that the agent was lying, that despite the agent’s lies, the information in the instructions was what they encountered, and that they were able to find more information than they had initially. The items that did worse was the agent contradicting himself, the users having to change their manner of speaking in order to be understood, and the agent’s lies not being plausible. These are just the items that did better and worse, but they can tell us what areas of the interaction are working better and which ones are doing worse.

As for the last part of our questionnaire, 15 users were able to tell that Peter had lied during the conversation (with 1 not being sure), and all 16 were able to tell that he had also said the truth at some point. All 16 were able to make Peter confess to one or more of the crimes he was accused (between assaulting the delivery guy, stealing the painting and selling the painting), with 8 having managed to obtain a confession for all crimes and 8 only managing a partial confession. In total, 10 people were able to get him to confess to assaulting the delivery guy, 15 to stealing the painting, and 14 to selling the painting.

When recounting what they thought had happened, here is what some people said:

- “Peter followed the delivery guy and interrogated him to find out the best way to steal the painting, on the night of August fifth Peter broke into the gallery and stole the painting, he then proceeded to go to sleep and in the day after he went to the train station to sell the painting to a man named James, Peter needed money so he could return back home.”
- “He followed the delivery guy, intimidated him with a gun, interrogated him, stole the painting and sold him to David Turner, who he got in contact with through James Miller.”
- “I think Peter bought a gun from James Miller, intimidated and interrogated the delivery man, stole the painting and then sold it to James Miller for 3000 euros”

Only three people managed to correctly identify the man that Peter sold the painting to as David Turner, with most of the participants believing Peter’s lie about having sold the painting to James Miller instead. Only two people discovered that Peter had bought the gun he used to assault the delivery guy from James Miller.

As for the limitations, here is what some people said:

- “It’s not hard to see when his lie doesn’t make sense in a real world context (i.e. “robbing a coffee shop at 4am to steal a large cup of coffee which he sold to James Miller a day later) but he doesn’t understand why the lies don’t make sense. It’s hard to find logical ways around them that don’t rely on that type of real world logic.”
- “Peter didn’t understand “after”/“before”. He also didn’t understand questions in the negative form (e.g. “You did not sell the painting in Pacific City?”)”
- “Sometimes he didn’t understand things like “what did you do after this happened?”. There were a few words he didn’t understand, even some he had previously used himself.”

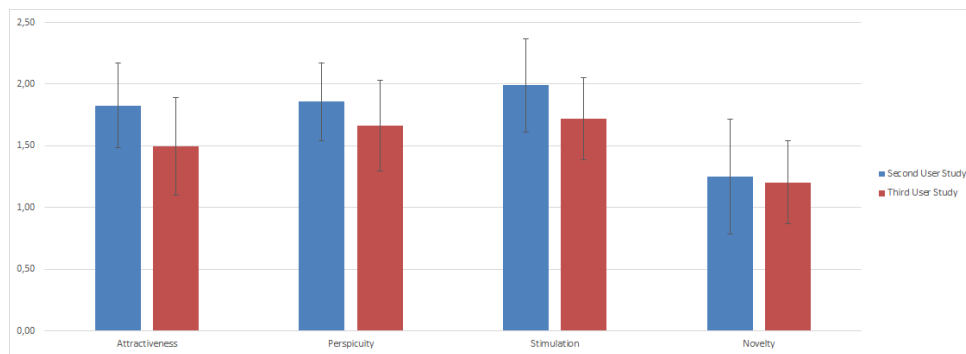
Overall, people commented about not being able to use real world logic to unveil the agent’s lies, as he was not following real world logic to lie, people commented about having to change the way they spoke, about Peter not understanding certain questions or expressions, and about the lies not being very good or believable.

### **6.3.6 Discussion**

If we start by analysing the agent performance by the conversation success rate and the error rate, we can see that it was about the same in both the Second and Third User Studies, with the success rate being 63.39% in the Second Study, and 65.01% in the Third Study, and 65.13% of exchanges not having any Question Problems in the Second Study, compared to 66.80% in the Third, and 89.83% of

exchanges in the Second Study not having any Answer Problems, compared to 88.48% in the Third. As we can see, those percentages are about the same in both studies, indicating the capabilities of the agent being able to understand what the user was asking remained about the same between studies, which makes sense considering the lying does not affect that. The major difference in terms of conversation data between the two studies is perhaps the average number of exchanges, which were 54.07 in the Second Study and 46.13 in the Third. This can be explained by the fact that in the Third Study the conversations were more focused around specific events, while in the Second Study they were more free form.

When comparing the UEQ results we start to have a better picture of the differences between the two studies. Just by looking at the benchmarks, both the Attractiveness and Perspicuity scales dropped into the lower category, with Stimulation and Novelty maintaining their classification.



**Figure 6.6:** Comparison of the UEQ Results

Figure 6.6 shows a comparison between the UEQ results of both studies, putting the mean value and confidence interval of each scale side by side. As we can see, all of the values are the lower in the Third Study, with Novelty being the most similar. What we can conclude from this is that the Virtual Suspect Skill became less appealing, less easy to understand, less exciting to use, and about the same level of innovative, with the introduction of the lying component.

Since there were no major difference in terms of agent performance of the two agents, and since the Third Study conversation flow evaluation seems to indicate that the agent's lies were not very plausible or consistent, we can conclude that the ability of the agent to understand what was being asked remained about the same, but the lying component made the interactions less engaging. Now, the UX results of the Third Study were still positive overall, which means that the lying is not a complete negative, just that it probably needs to be worked on and adjusted in order to better fit with the new interaction.

# 7

## Discussion

### Contents

---

7.1 Limitations . . . . .	73
7.2 Results . . . . .	75

---

In this chapter, we will analyse the limitations that we faced while trying to improve the interaction with the Virtual Suspect, talk about the results of our studies, and speculate on what could be done in the future.

## 7.1 Limitations

During the course of our work, we managed to make many changes and improvements to the interaction with the Virtual Suspect, but there were certain constraints that limited our ability to further improve the interaction. These constraints were centered around: the ASK and the structure of the Interaction Model, the organization of the agent's memory and the Virtual Suspect Architecture, and the implementation of the lying algorithm.

### 7.1.1 Interaction Model

The ASK is easy to use and contains a wide range of tools at our disposal, which was why we used it to create our new interaction, but we were also constrained by its limitations. It is very good at building simple Skills with specific functions and more limited domains, but since our interaction is more conversational, it required more nuance. A more non-deterministic, grammar-like approach could be beneficial for improving our interaction even further.

One constraint we felt was the fact that slots cannot contain other slots, which affected the way we had to structure our time slots. Since we had to use different slots for dates, times and time pronouns, every utterance that could have a time query condition had to include all possible combinations and permutations of those slots. Another example of this is the fact that certain slots could sometimes, but not always, be preceded by a {preposition} slot. If we had a grammar-like natural language model, we could define a {theme\_slot} "super-slot" that could contain both {preposition} {theme} and {theme} possibilities, and we could just reference that {theme\_slot} in our utterances. This problem compounds with the time slots situation, because we had to account for the cases where the time slots could or could not have prepositions before them, which led to a massive inflation in the amount of utterances in our Interaction Model.

Another example of this constraint relates to entity ambiguity, as there were some entities in our story that could serve different roles in different events. This means that Gun, for example, has to be present in both the {theme} and {manner} slot values, which could lead to the Interaction Model identifying the wrong slot sometimes. We were able to account for this ambiguity when dealing with the pronouns (as explained in Section 5.2), but since we cannot control the Interaction Model decision process (as mentioned in Section 3.2), we cannot have direct control over these resolutions. A solution where we could have a better understanding and control of the selection process would lead to a better interaction.

## 7.1.2 Virtual Suspect Architecture

The more natural an interaction with an interactive agent is, the more expectations the users will have about the realism of its behaviour, a phenomenon we experienced in our own User Studies. If the agent is able to understand certain questions very well, then users will expect it to be able to understand other questions that are just as complex in their eyes. But grammatical complexity is not the only constraint that keeps our agent from being able to understand certain questions, and he does not necessarily operate using real world logic. The way that the agent's memory is structured, and the kinds of questions it can answer, do not always correspond to what users expect the agent to be able to answer.

For example, users would often inquire as to the motives behind the agent's actions, which does not correspond to the Reasons associated with the events. He stole the painting in order to sell the painting, and he sold the painting in order to get money, but why did he need the money? These are the types of questions that users were often interested in, but that the agent could not answer, because there is no Motive defined in the agent's story, and the Reason entities do not connect one event to another.

Another example is the way that events are organized. All events have a Time entity, but these entities are not connected to one another. This makes it difficult to process something that a lot of users tried to ask - what happened before or after certain events. Events should be organized in the agent's memory in the way that they form a sequence, and it should be possible to make connections from one event to the next.

In the same way that events are not connected via a timeline, there is also no geographical connection between locations. We added the Parent relationship to the Location entities, which creates a location hierarchy, but there is still no relation between locations in terms of distance. When writing the story, we mostly decided to ignore that, otherwise the story would be full of Travel events, but this is something that should be relevant, and would especially be important when coming up with lies, which we will talk about in the next subsection.

All these limitations and the way that the agent's memory is structured makes it impossible to write a realistic story. If the memory were restructured in a way that events could be organized in a timeline, and reasons and motives could be reworked, that would go a long way in making the story more realistic, which would bring it in line with the quality of the interaction.

## 7.1.3 Lying Algorithm

The objective of our work was to improve the interaction with the Virtual Suspect developed by Rato et al. [1], so we did not change anything about the lying algorithm, as that was the focus of that work, but there are some aspects of it that can be improved upon in the future.

The way the lying algorithm currently works is by creating new fake events to substitute Incriminatory

events (which have a value greater than 0 in that field), keeping the same action and replacing all of the entities that have not already been marked as known by the user. The way it selects these new entities is by searching through all entities and then identifying the most similar ones. For example, if the user asks “*What did you do on August 5th at 4am?*”, the agent will almost always say something like “*I stole a large cup of coffee*”. The way that the agent forms these new events can definitely be improved. This constraint also compounds with the Reason constraint we mentioned previously. Since Reason entities do not make connections between events, the Reasons the agent selects for the new fake events often make no sense at all, and it is very difficult for users to be able to discover the truth. Also, since the agent focuses on duplicating incriminatory events, it will almost never use the fake events included in the story in order to help it lie.

The way the current lying algorithm works can be considered unrealistic and could use a lot of work, by changing the way the agent creates its fake events, and by changing the way it replaces entities.

## 7.2 Results

We can analyse the results of our three user studies with two different metrics: the agent performance, which we measured for all three, and the UX, which we measured in the last two user studies.

When we look at the agent performance (the conversations’ success rate and the error rate), we can see that it improved tremendously between the first and second user studies, indicating that the problems we identified during the first study and the changes we made to address those problems were successful. Between the second and third studies, the agent performance remained about the same - with the conversation success rate improving slightly and the error rate remaining the same. As we did not make a lot of changes between the second and third studies, this stability in results continues to support the improvements we have made since the start of this project.

When we look at the UX results, which we only have for the second and third studies, we can see that they became worse in the third study. This, combined with the agent performance remaining the same, indicates that it was the activation of the lying component that precipitated this change. As we saw in Figure 6.6, the Novelty component remained about the same, which makes sense considering it was still the same concept and the same technology, but all of the three other scales went down a lot more, which tells us that the version of the Virtual Suspect with the lying component activated had a worse UX than the honest version.

All of this seems to indicate that we succeeded in improving the interaction with the Virtual Suspect, but that the lying algorithm that was originally developed [1] was not appropriate for this new interaction, and requires more work in order to be better adjusted to this type of more natural interaction.

# 8

## Conclusion

### Contents

---

8.1 Future Work .....	78
-----------------------	----

---



Interactive Agents can be an incredibly versatile and useful tool, but a good interaction is necessary in order to be able to show off all the capabilities of the agent.

In Chapter 1 we presented the boundaries of our work and defined our objective. We presented the Virtual Suspect designed by Rato et al. [1] as a promising interactive agent with a limited interaction, and we presented Amazon Alexa as a tool that would help us create a new Natural Language conversational interaction with the Virtual Suspect.

In Chapter 2 we looked at other works with similarities to our own. We looked at how other Virtual Suspects have been defined, we looked at how certain conversational agents can work in contexts similar to ours, we analysed how video games handled interrogation scenarios, and we analysed how other, more complex, Alexa Skills could be developed.

In Chapter 3 we presented and explained the two main components of our work: the Virtual Suspect, and the ASK that allowed us to create our Natural Language interaction. We went over Rato et al.'s original definition for the Virtual Suspect, and how it was implemented and tested [1]. We learned how the ASK works and how we can use it to create our own Alexa Skill to incorporate with the Virtual Suspect.

In Chapter 4 we combined the knowledge we gathered in Chapter 3 and laid out the structure for our Virtual Suspect Skill, which would allow us to interact with the Virtual Suspect through Natural Language. We defined the development cycle of our Skill, starting out by recreating the original functionality of the Virtual Suspect Prototype [1], developing and testing with the lying component of the agent turned off to get a better understanding of how the agent was processing the interaction. Then we conducted a user study, to gather information about how users interacted with the Virtual Suspect, what problems the interaction had, and how the agent was performing. We used the data we collected in the User Study to make changes and improvements to the Virtual Suspect, and at the end of that cycle did another User Study to validate the improvements made to the Virtual Suspect Skill. After we validated that our interaction had improved a lot since the First User Study, we turned the lying component back on and did a final User Study to measure how the original lying algorithm performed when combined with the new interaction.

Chapter 5 contained all the changes and improvements we made to the Virtual Suspect Skill, starting out by recreating the original functionality of the original Prototype, making all the improvements after the First and Second User Studies, including changes to the Virtual Suspect Response Model and the Story, and everything we did with our Interaction Model and Skill Service. Chapter 6 contained the procedures and results of all three User Studies, how we established the connection between the users and the Virtual Suspect, how we logged and annotated their conversations, and how we evaluated the agent's performance and the UX.

In Chapter 7 we went over the constraints and limitations we faced while developing the Virtual Sus-

pect Skill, and how they could be addressed in the future to hopefully create an even better interaction.

In summary, this was a very experimental work that followed a heavy user-centered approach in pursuit of our objective of improving the interaction with the Virtual Suspect. Whether we were successful in achieving our goals comes down to whether we:

- were able to overcome the limitations of the original Virtual Suspect interaction [1];
- managed to create a Natural Language interaction that showcased the capabilities of the Virtual Suspect;
- created an interaction with good UX.

For each of those points our results were positive. We were able to create a natural and open interaction with the Virtual Suspect, using Natural Language, that showcased all of its capabilities (even adding new ones in the process), and we were able to vastly improve the quality of the interaction and achieve a good UX while doing it. On top of achieving our goal of improving the interaction with the Virtual Suspect, we were able to test whether the original lying algorithm [1] was suited to this type of interaction, and concluded that it needs further improvement.

## 8.1 Future Work

There are a wide variety of things that could be done in the future in order to further improve the interaction with the Virtual Suspect:

- Use a different tool to create the Natural Language interaction with the Virtual Suspect, preferably one that allows the creation of a non-deterministic grammar-like natural language model, as suggested previously. Since the Alexa did come with the advantage of handling a lot of the technical aspects of Natural Language Processing, this would something that would need to be accounted for.
- Reorganize the way the agent's memory works, by restructuring how the Time and Reason fields work, and by adding more information about the context around the story. The idea behind the organization of the agent's memory is sound, but the relationships between events and between entities needs to be improved upon in order to make the story more consistent and coherent.
- Change the lying algorithm. The way the Virtual Suspect Architecture was designed allows for the removal and addition of the lying layers without affecting the core query system. Create a new lying strategy that is more consistent with this new type of interaction, as a more natural interaction requires a more coherent strategy, as users have greater expectations of the agent's capabilities.

- Create different scenarios, stories and characters. The agent's story only covers three days and only focus on the agent's version of events. With other changes to the way the agent's memory is structured, bigger and more interesting stories could be created and explored.
- Create an authoring tool that allows for the creation of and easy adjustments to the story. Since the agent comes up with lies automatically, and the interaction is handled separately from its story and lying strategy, if those two things are working well, then the easy creation of new stories could easily create a variety of scenarios with which people could interact with and be introduced to the Virtual Suspect.
- Incorporating the Natural Language Interaction and the Virtual Suspect Response Model with a visual component could be interesting. Creating a visual representation of the agent and being able to interact with it could add another layer of immersion to the experience.
- Add more game elements and create a full game experience. If you add objectives, different characters and different possibilities to the story, one could easily take the work that has been done so far into the entertainment realm.
- Make the story and behaviour more realistic and treat it as a training tool. The basis of the work is already there, and this work could very well be taken in a more serious direction.

# Bibliography

- [1] D. Rato, R. Prada, and A. Paiva, “Virtual suspect,” Master’s thesis, Instituto Superior Técnico, October 2016.
- [2] M. Schrepp, “User experience questionnaire handbook,” *All you need to know to apply the UEQ successfully in your project*, 2015.
- [3] E. Sklar and D. Richards, “The use of agents in human learning systems,” in *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*. ACM, 2006, pp. 767–774.
- [4] M. Bruijnes, S. Wapperom, R. op den Akker, and D. Heylen, “A virtual suspect agent’s response model,” *Affective Agents*, p. 17, 2014.
- [5] M. Bitan, G. Nahari, Z. Nisin, A. Roth, and S. Kraus, “Psychologically based virtual-suspect for interrogative interview training,” in *Thirty-First AAI Conference on Artificial Intelligence*, 2017.
- [6] D. Rato, B. Ravenet, R. Prada, and A. Paiva, “Strategically misleading the user: Building a deceptive virtual suspect,” in *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2017, pp. 1711–1713.
- [7] T. W. Morris, “Conversational agents for game-like virtual environments,” in *AAAI 2002 spring symposium on artificial intelligence and interactive entertainment*, 2002, pp. 82–86.
- [8] I. Kenny and C. Huyck, “An embodied conversational agent for interactive videogame environments,” in *Proceedings of the AISB’05 Symposium on Conversational Informatics for Supporting Social Intelligence and Interaction*, 2005, pp. 58–63.
- [9] J. Falk, S. Poulakos, M. Kapadia, and R. W. Sumner, “Pica: Proactive intelligent conversational agent for interactive narratives,” in *Proceedings of the 18th International Conference on Intelligent Virtual Agents*. ACM, 2018, pp. 141–146.

- [10] A. Ram, R. Prasad, C. Khatri, A. Venkatesh, R. Gabriel, Q. Liu, J. Nunn, B. Hedayatnia, M. Cheng, A. Nagar *et al.*, “Conversational ai: The science behind the alexa prize,” *arXiv preprint arXiv:1801.03604*, 2018.
- [11] C.-Y. Chen, D. Yu, W. Wen, Y. M. Yang, J. Zhang, M. Zhou, K. Jesse, A. Chau, A. Bhowmick, S. Iyer *et al.*, “Gunrock: Building a human-like social bot by leveraging large scale real user data,” 2018.
- [12] M. Hajdinjak and F. Mihelic, “Conducting the wizard-of-oz experiment.” *Informatica (Slovenia)*, vol. 28, no. 4, pp. 425–429, 2004.
- [13] M. Schrepp, A. Hinderks, and J. Thomaschewski, “User experience questionnaire,” *Mensch und Computer 2017-Tagungsband: Spielend einfach interagieren*, vol. 17, p. 355, 2018.



# User Study Materials

This Appendix contains the supplementary materials used for the User Studies, the instructions and questionnaires given to the users in the Second and Third User Studies, as well as the full results of the User Experience Questionnaire (UEQ) and conversation flow evaluations.

## A.1 Instructions

### A.1.1 Second User Study

Hello!

For this test, you will be talking with **Peter**, an autonomous conversational agent, in the context of a police interrogation. Peter is a crime suspect, so it's your job to ask him questions about said crime to try to uncover more information about the events that took place. However, for this scenario Peter can only tell the truth, so he will answer all of your questions directly.

His story takes place between March third and March sixth in the fictional towns of Castle Town and Silvermoon City and is about a jewelry store robbery, from where Peter stole a necklace. Knowing this

information, your goal is to ask him questions to uncover as much information as you can about the robbery and the events leading up to and following it. Peter was an active participant in the events of the story, meaning that you should ask your questions directly to him and about him (ex: *Where were you? Did you do...?*) as opposed to indirect questions (ex: *Can you tell me about...? Do you know anything about...?*).

Now, since he is a virtual agent, Peter has limitations (as we all do). He may be able to understand and respond to certain questions very well, but have difficulty with others. There are certain things he can do very well, like understand context, and others he can't do at all, like talk about the weather. You should try to talk with him in the way that feels most natural to you, and if you feel like you're not being understood, try rephrasing your question or changing topics.

I will now let you talk directly with Peter. If you have any trouble or question don't be afraid to ask. Remember, your goal is to uncover as much information as you can with the information you were given by asking simple and direct questions.

If you're not sure how to start, try starting with Hello. Good luck!

### **A.1.2 Third User Study**

Hello!

This is a document containing useful information for the testing of the Virtual Suspect Skill. This skill is a prototype and is not meant to be a finalized commercial game or product. The Virtual Suspect is an agent that is capable of playing the role of the suspect in the context of a police interrogation. It is capable of answering questions about the events of its story, and like a real crime suspect, it is capable of lying about said events. The objective of this work was to create a natural language interaction with the Virtual Suspect, and the objective of this study is to evaluate that interaction.

When you open the Virtual Suspect skill, you will be placed right away in a dialogue with the suspect, whose name is **Peter**. His story takes place between August 3rd and August 6th, in the fictional towns of Rose Town and Pacific City. This is what we know about the case:

- A painting was stolen from the Gallery located at the Rose Town Mall between the hours of 4am and 4:30am on August 5th;
- That painting was delivered to the Gallery at 4pm of the previous day, August 4th, and the delivery guy claims that afterwards our suspect followed him, intimidated him with a gun, and interrogated him in order to find out if the painting had been delivered;
- We also managed to find out that Peter bought a train ticket to Pacific City on the morning of August 6th. We believe he may have travelled there in order to sell the painting.

Your goal is to make the suspect confess to robbing the gallery and stealing the painting, and to find out if he sold the painting and to whom, as well as any other information you manage to uncover in the process.

Before you begin, there are a few things you should keep in mind. First, the agent only responds to questions about his story, so it's no use making threats or statements. He also only views his story from his own point of view, so he doesn't know other people's actions. There are things that Peter understands well, like context, and things he doesn't understand at all, like the weather. You should try to talk with him in the way that feels most natural to you, and if you feel like you're not being understood, try rephrasing your question or changing topics.

## **A.2 UEQ Results**

### **A.2.1 Second User Study**

### **A.2.2 Third User Study**

## **A.3 Conversation Flow Results**

### **A.3.1 Second User Study**

### **A.3.2 Third User Study**

## **A.4 Questionnaire for the Third Study**

Since the questionnaires for both the Second and Third User Studies were very similar, we will only show the latter. The differences are detailed in Section 6.2 and Section 6.3.



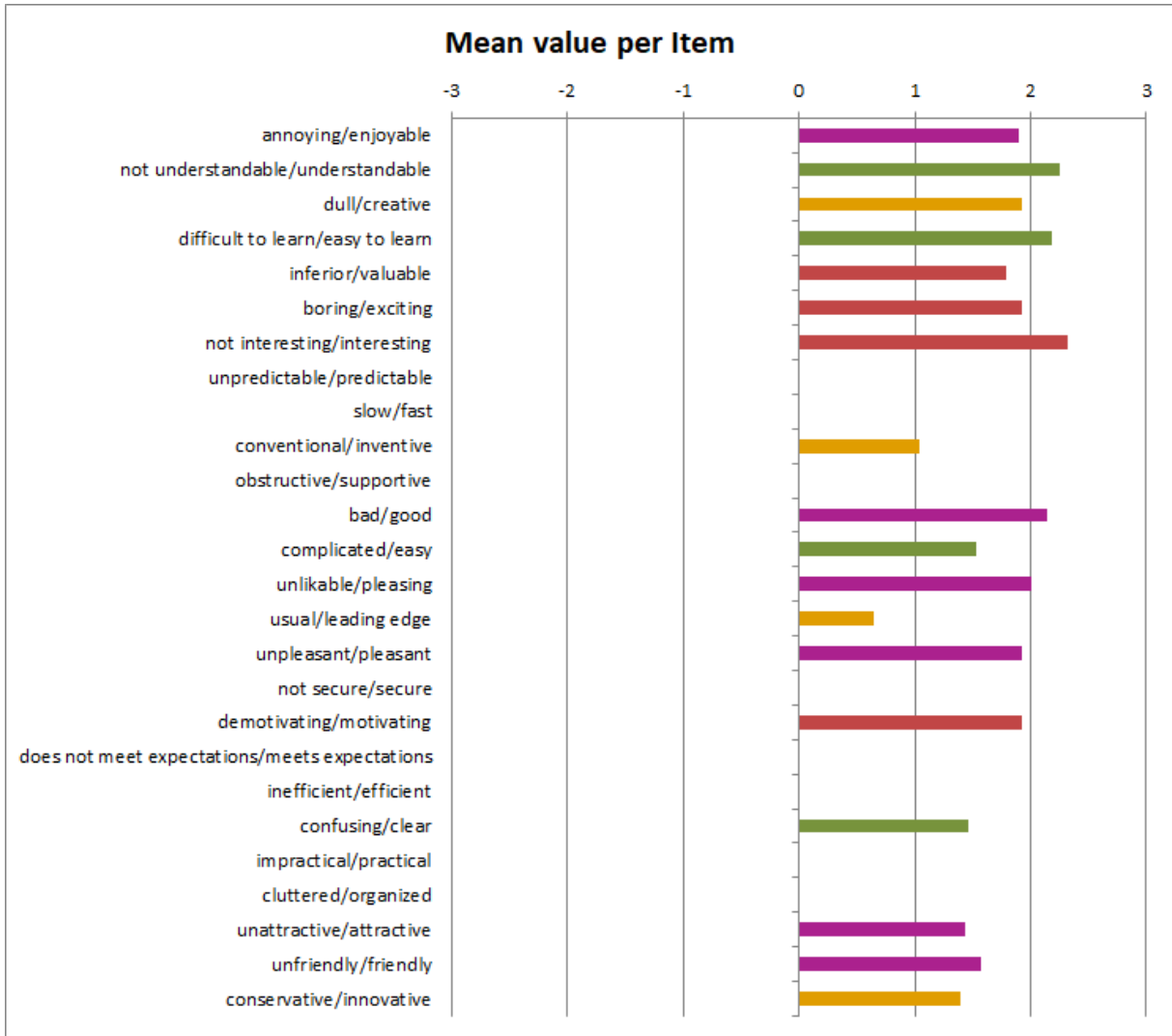


Figure A.1: Mean Value of each UEQ Item for the Second Study

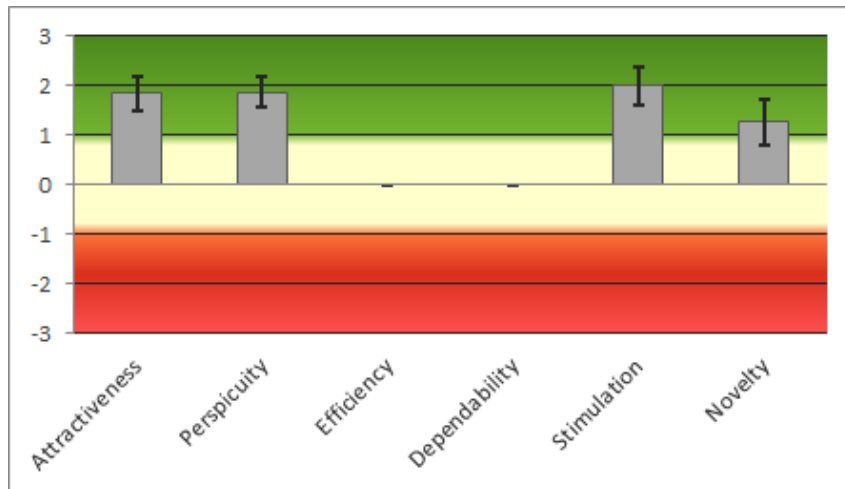


Figure A.2: Means and Confidence Intervals of UEQ Scales for the Second Study

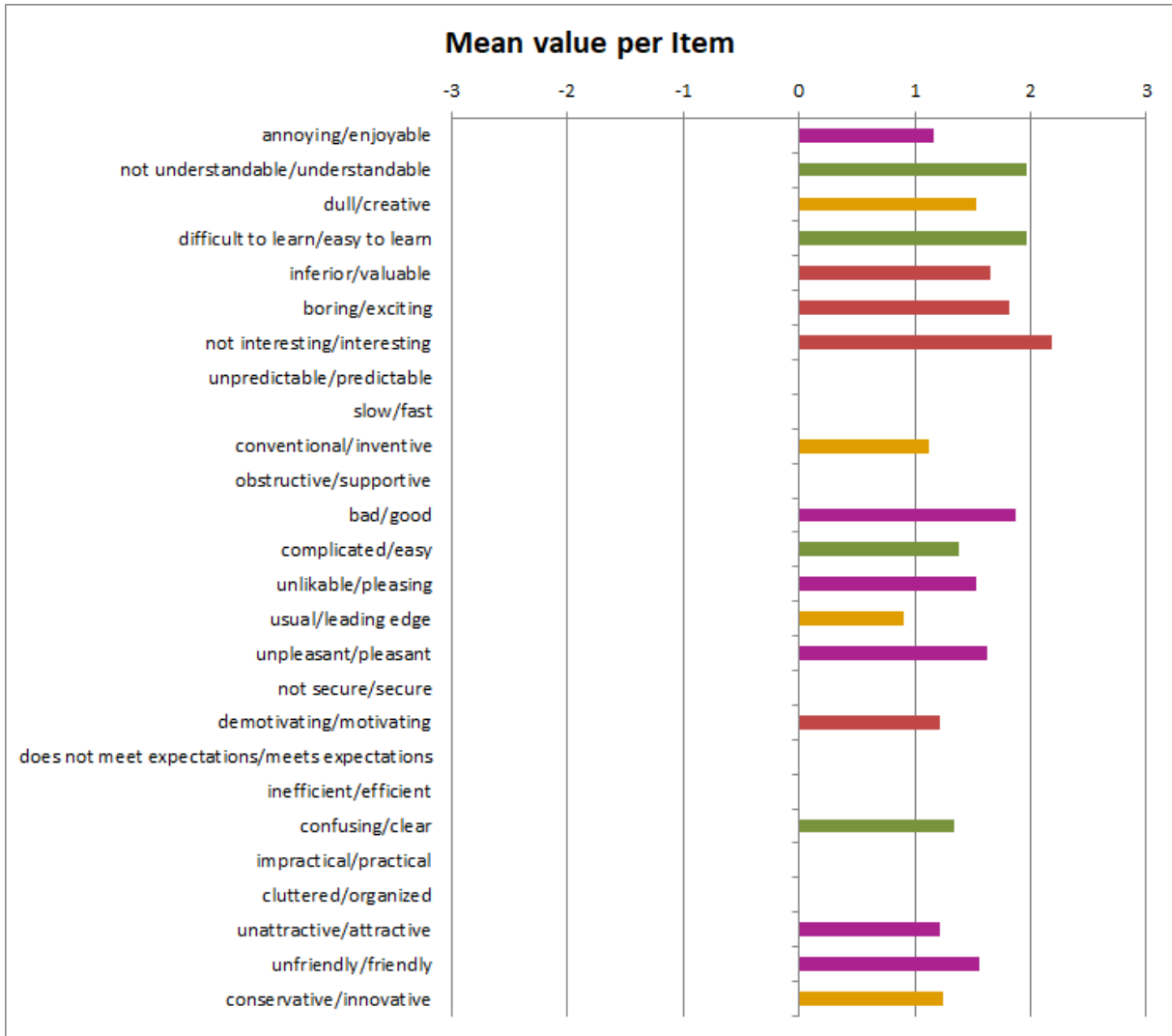


Figure A.3: Mean Value of each UEQ Item for the Third Study

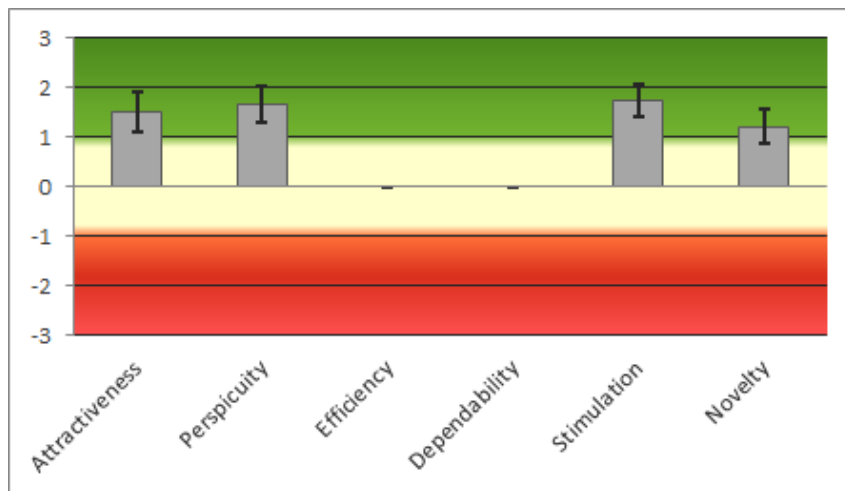


Figure A.4: Means and Confidence Intervals of UEQ Scales for the Third Study

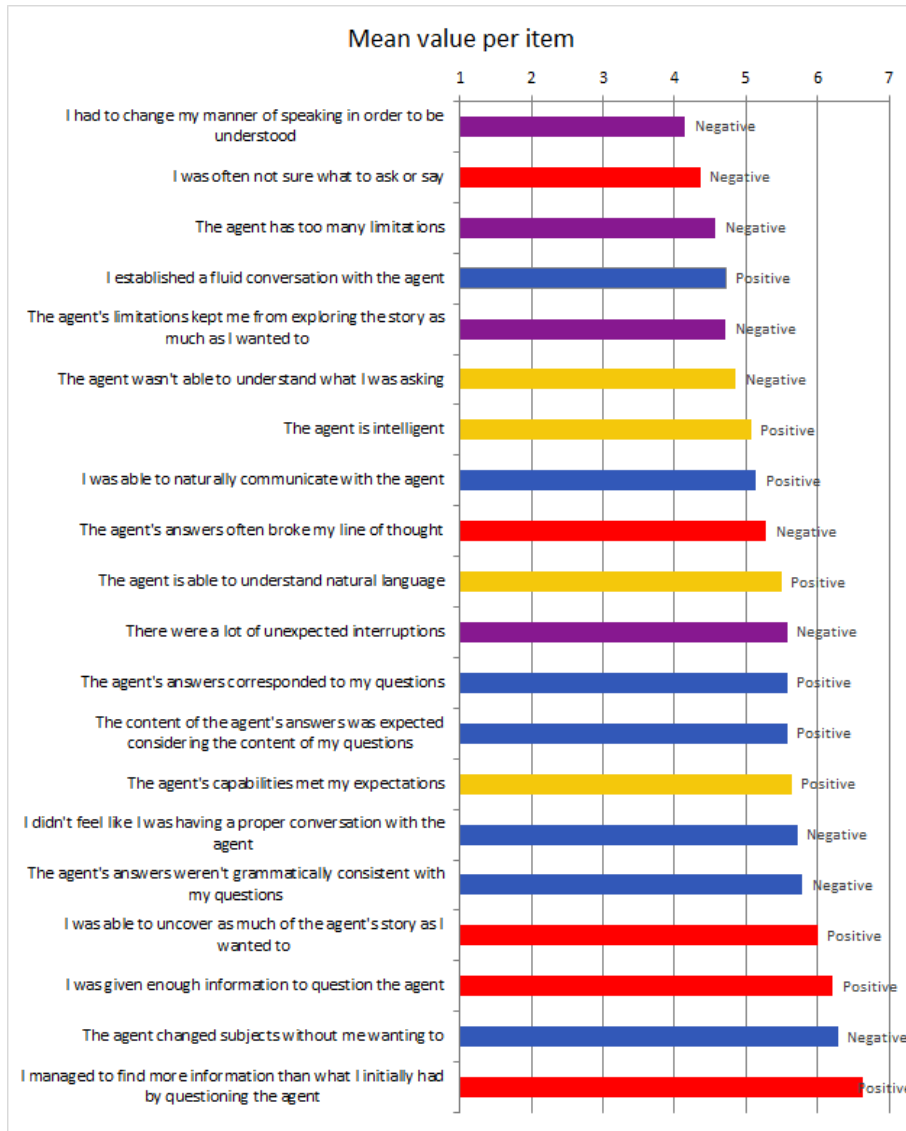


Figure A.5: Mean Value of each Conversation Flow Question for the Second Study

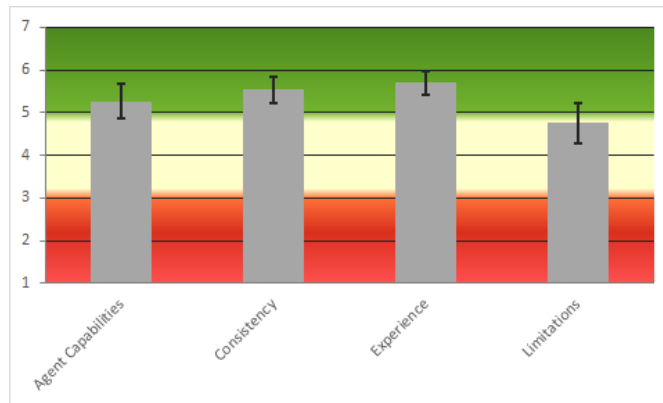


Figure A.6: Means and Confidence Intervals of Conversation Flow Scales for the Second Study

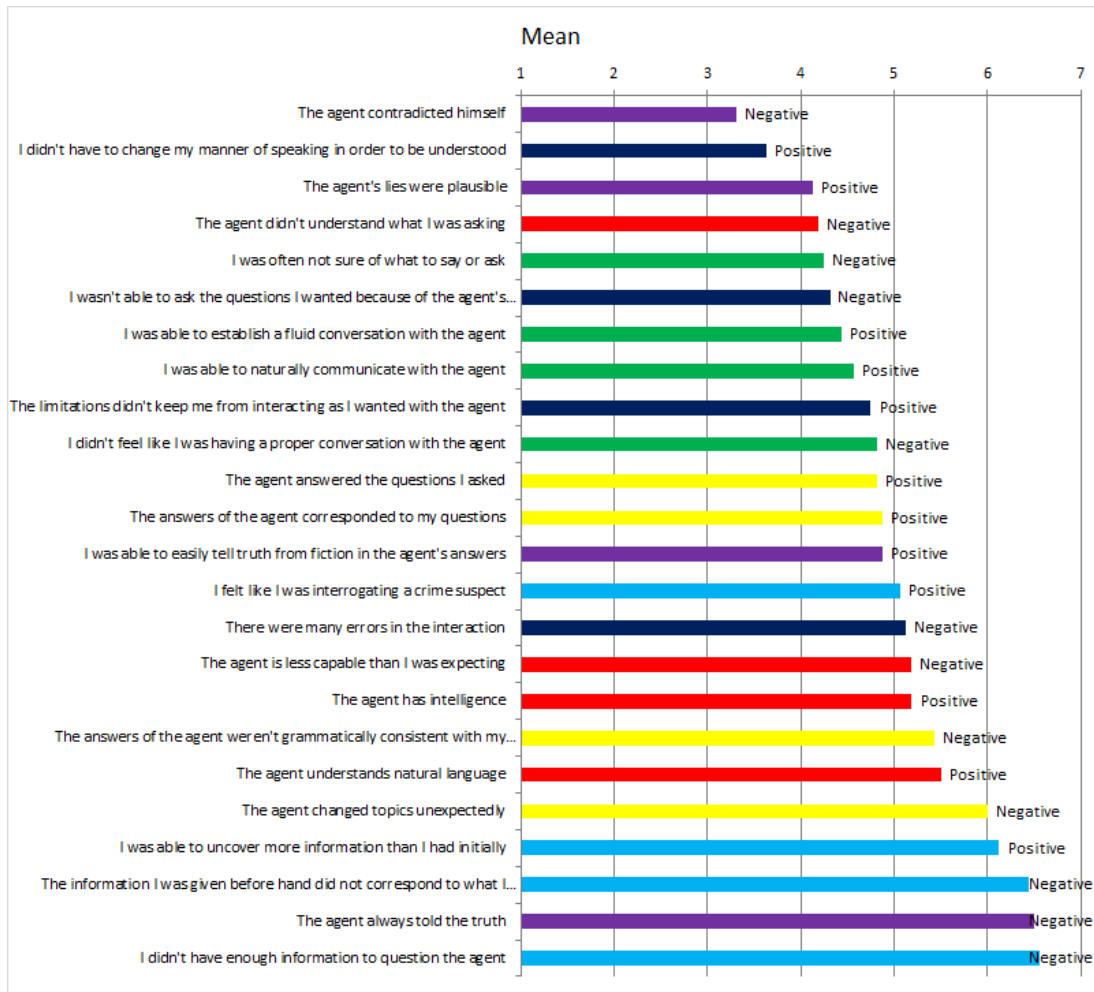


Figure A.7: Mean Value of each Conversation Flow Question for the Third Study

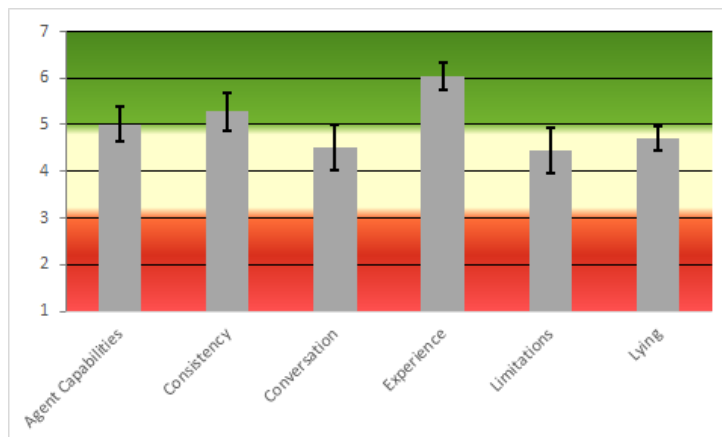


Figure A.8: Means and Confidence Intervals of Conversation Flow Scales for the Third Study

## Virtual Suspect User Tests

The purpose of this questionnaire is to evaluate the performance of the Virtual Suspect Skill. What you asked the Virtual Suspect and how much you managed to uncover is not being evaluated.

Participation in this study is completely anonymous and voluntary, none of the information collected by this questionnaire will be used to identify you. Your answers will only be collected at the end of the questionnaire, which should take no more than 5 minutes.

**\*Obrigatório**

The information collected here will not be used to identify you in any way, merely to characterize the study sample.

1. How old are you? \*

---

2. What is your gender? \*

*Marcar apenas uma oval.*

- Male  
 Female  
 Other  
 Prefer not to say

3. Are you a native English speaker? \*

*Marcar apenas uma oval.*

- Yes  
 No

**Figure A.9:** First Page of Questionnaire for the Third Study

4. Please rate your level of fluency in English, with 0 being not even being able to understand this sentence, to 10 being on the same level as a native speaker. \*

*Marcar apenas uma oval.*

0	1	2	3	4	5	6	7	8	9	10
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

5. How many times have you interacted with a virtual chatbot (ex: Siri, Cortana, Google Assistant, Alexa, Cleverbot, etc.) \*

*Marcar apenas uma oval.*

- Never
- 1-2 times
- 3-5 times
- 6-10 times
- 11+ times

Please evaluate your experience with the Virtual Suspect Skill by filling out the following grid. The grid contains several adjectives/descriptors and you can indicate how much you think they apply to the overall product with a value from 1 (Strongly Disagree) to 7 (Strongly Agree). There are no right or wrong answers, this is just your opinion. Try not to think too much about each of them, just answer spontaneously.

**Figure A.10:** Second Page of Questionnaire for the Third Study

6. I found the Virtual Suspect Skill... \*

Marcar apenas uma oval por linha.

	1 (Strongly Disagree)	2	3	4	5	6	7 (Strongly Agree)
annoying	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
enjoyable	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
not understandable	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
understandable	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
creative	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
dull	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
easy to learn	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
difficult to learn	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
valuable	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
inferior	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
boring	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
exciting	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
not interesting	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
interesting	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
inventive	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
conventional	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
good	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
bad	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
complicated	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
easy	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
unlikable	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Figure A.11: Third Page of Questionnaire for the Third Study

pleasing	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
usual	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
leading edge	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
unpleasant	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
pleasant	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
motivating	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
demotivating	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
clear	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
confusing	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
attractive	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
unattractive	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
friendly	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
unfriendly	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
conservative	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
innovative	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Now you will be asked to evaluate a series of statements about the conversation itself. You can once again evaluate them from 1 (Strongly Disagree) to 7 (Strongly Agree). There are no right or wrong answers.

**Figure A.12:** Fourth Page of Questionnaire for the Third Study



7. \*

Marcar apenas uma oval por linha.

	1 (Strongly Disagree)	2	3	4	5	6	7 (Strongly Agree)
The agent understands natural language	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The agent didn't understand what I was asking	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The agent has intelligence	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The agent is less capable than I was expecting	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The answers of the agent weren't grammatically consistent with my questions	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The answers of the agent corresponded to my questions	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The agent changed topics unexpectedly	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The agent answered the questions I asked	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I was able to	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Figure A.13: Fifth Page of Questionnaire for the Third Study

establish a fluid conversation with the agent

---

I was often not sure of what to say or ask

---

I didn't feel like I was having a proper conversation with the agent

---

I was able to naturally communicate with the agent

---

I didn't have enough information to question the agent

---

I was able to uncover more information than I had initially

---

The information I was given before hand did not correspond to what I encountered

---

I felt like I was interrogating a crime suspect

---

There were many errors in

**Figure A.14:** Sixth Page of Questionnaire for the Third Study

the interaction

---

I wasn't able to ask the questions I wanted because of the agent's limitations

---

I didn't have to change my manner of speaking in order to be understood

---

The limitations didn't keep me from interacting as I wanted with the agent

---

The agent always told the truth

---

The agent contradicted himself

---

The agent's lies were plausible

---

I was able to easily tell truth from fiction in the agent's answers

---

Secção sem título

**Figure A.15:** Seventh Page of Questionnaire for the Third Study

Now I would like to ask you for some feedback regarding your conversation with Peter.

8. Did Peter lie? \*

*Marcar apenas uma oval.*

- Yes  
 No  
 I'm not sure

9. Did Peter ever tell the truth? \*

*Marcar apenas uma oval.*

- Yes  
 No  
 I'm not sure

10. Did Peter confess to the crimes he was accused of? \*

*Marcar apenas uma oval.*

- Yes, to all  
 Yes, to some  
 No

11. What crimes did he confess to? \*

*Marcar tudo o que for aplicável.*

- Following/attacking the delivery guy  
 Stealing the painting from the gallery  
 Selling the painting  
 None of the above

**Figure A.16:** Eighth Page of Questionnaire for the Third Study

12. What do you think really happened? \*

---

---

---

---

---

13. What limitations did you encounter in your conversation with Peter? \*

---

---

---

---

---

---

Este conteúdo não foi criado nem aprovado pela Google.

Google Formulários

**Figure A.17:** Ninth Page of Questionnaire for the Third Study