Crime at the palace - Verbal interaction

Daniel Rodrigues Mendes

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisor(s): Prof. Isabel Maria Martins Trancoso
               Prof. Maria Luísa Torres Ribeiro Marques da Silva Coheur

Examination Committee
Chairperson: Prof. João Fernando Cardoso Silva Sequeira
Supervisor: Prof. Maria Luísa Torres Ribeiro Marques da Silva Coheur
Member of the Committee: Prof. Rui Filipe Fernandes Prada

November 2018
Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.
Although the shouldering of all responsibility is usually a social ritual, the admission that errors exist is not — it is often a sincere avowal of belief. But this appears to present a living and everyday example of a situation which philosophers have commonly dismissed as absurd; that it is sometimes rational to hold logically incompatible beliefs.

— **David C. Makinson** (1965)
Acknowledgments

I would like to express my sincere gratitude towards my thesis supervisors, Professor Luisa Coheur and Professor Isabel Trancoso who introduced me into the interesting research area that is natural language and speech processing. Their constant guidance and support throughout the thesis was invaluable. Inevitably, there will remain some mistakes which are of my sole responsibility. In addition, I would like to extend my sincerest thanks to Sérgio Curto and Pedro Fialho who, with great patience and no obligation, accompanied and advised me in most matters.
Resumo

Acrescentou-se uma camada ao já existente sistema de diálogo EDGAR (Fialho et al., 2013) de modo a que este agente virtual se comporte como suspeito num interrogatório policial. O sistema existente, por si, é um sistema de diálogo com vários plugins que ligam a pergunta do utilizador ao par pergunta/resposta mais próximo na sua corpora. Assim que a questão é encontrada na corpora, a nova camada calcula qual a resposta mais apropriada baseando-se em factores emocionais. As escolhas de quais factores emocionais e quão fortemente estes influenciam a decisão da resposta foram feitas de acordo com várias regras do modelo Entrevista Cognitiva usado pela Escola da Polícia Judiciária¹ tal como observações de entrevistas de treino e outras teorias de psicologia. Os factores emocionais do suspeito virtual partem de um estado default estático que se vai alterando à medida que vão passando interações.

Keywords: Sistemas de Lingua Natural, Sistemas Educativos, Jogos Sérios de Computador, Treino de Entrevista Policial.

¹https://www.policiajudiciaria.pt/
Abstract

A layer was developed above the existent dialog system EDGAR (Fialho et al., 2013) in such a way that the virtual agent acts as a suspect in a crime and is therefore in a police interview. EDGAR, by itself, is a chat bot with several plugins which picks user input and matches it with the closest question/answer pair in its corpora. Once the question is matched, the new layer calculates which is the most appropriate answer based on emotional factors from the virtual suspect. Which emotional factors and how strongly they impact the answer decision was set according to several rules from the model of the Cognitive Interview used by the Portuguese Escola da Policia Judiciária2 as well as observations of enacted interviews and other researched psychology theories. The emotional factors of the virtual suspect begin with a static default state which is modified throughout the interactions.

Keywords: Natural dialog systems, tutoring systems, serious computer games, police interview training.

2https://www.policiajudiciaria.pt/
# Contents

Declaration ................................................................. iii  
Acknowledgments ......................................................... v  
Resumo ................................................................. ix  
Abstract ................................................................. xi  
List of Tables .............................................................. xvii  
List of Figures .............................................................. xix  
Nomenclature .......................................................... 1  

## 1 Introduction 1  
1.1 Motivation .......................................................... 1  
1.2 Objectives .......................................................... 3  
1.3 Thesis Outline ....................................................... 4  

## 2 Background 5  
2.1 Edgar, the virtual butler ........................................ 5  
2.2 Interviewer's Intentions, Goals and Obligations ......... 8  
2.3 Rapport and Entrainment ........................................ 8  
2.4 Interviewing Techniques ......................................... 9  
  2.4.1 PEACE Model and Cognitive Interview ............... 10  
  2.4.2 Interviewing Strategies .................................. 13  
2.5 Interpersonal Stance Theory ..................................... 14  
2.6 Defining Personalities and Moods ............................ 15  
  2.6.1 Five Factor Model ......................................... 15  
  2.6.2 Pleasure, Arousal and Dominance emotional state model - PAD ............ 16  
2.7 Serious Game Lemniscate Model ............................... 17  
2.8 Turn Management or Impression Management? .......... 17  
2.9 Police Interview Analysis ....................................... 18  
2.10 Suspect's Mindset ................................................ 18  
2.11 Summary of what is good and bad for the interview applied to the game .. 19  

## 3 Related Work 23  
3.1 LOITER ............................................................. 23
B Character and animation

B.1 Character ................................................................. 79
B.2 Animations ............................................................. 79
  B.2.1 Creating the animations ........................................... 79
  B.2.2 Controlling the animations ....................................... 82
B.3 Lip-sync ................................................................. 85
B.4 Eyes and head movement ............................................. 86

C Script ........................................................................... 89

D Positive and negative instances ........................................ 93
List of Tables

4.1 Naive Bayes Classifier’s results for the IMDB’s English data set. . . . . . . . . . . . . . . 33
4.2 A detailed accuracy by class of the model tested with training data. . . . . . . . . . . . . . 33
4.3 Comparison of English and Portuguese samples. . . . . . . . . . . . . . . . . . . . . . . . 34
4.5 A detailed accuracy by class of the model tested with training data. . . . . . . . . . . . . . 35
4.4 Naive Bayes Classifier’s results for IMDB’s Portuguese translated data set. . . . . . . . . 35
4.6 Testing the Naive Bayes classifier trained with the IMDB’s translated data set with possible
data from an interview. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 35
4.7 Naive Bayes Classifier’s results for sentiCorpus-PT data set. . . . . . . . . . . . . . . . . . 36
4.8 Naive Bayes Classifier’s results for sentiCorpus-PT edited data set. . . . . . . . . . . . . . 37
4.9 Testing the Naive Bayes classifier trained with the sentiCorpus-PT edited data set with
possible data from an interview. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 37
4.10 List of discarded words. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 38

B.1 Phonemes and correspondent visemes (in ASCII). . . . . . . . . . . . . . . . . . . . . . . . 85

D.1 Positive and negative instances added by hand. . . . . . . . . . . . . . . . . . . . . . . . . 93
# List of Figures

2.1 EDGAR’s structure components. .................................................. 5
2.2 EDGAR’s XML corpus example. .................................................. 7
2.3 EDGAR’s architecture. ................................................................. 7
2.4 Leary’s Rose. ................................................................................ 15
2.5 Serious Game Lemniscate Model (Koops and Hoevenaar, 2012). .............. 17

3.1 Example of an interaction in Loiter (Linssen et al., 2015a) . ..................... 24
3.2 Example of an interaction with the player and the NPC. .............................. 25
3.3 Table relating the four motivators and emotions associated with them. .............. 25
3.4 Hassan’s interface components during a conversation. .............................. 26
3.5 FATIMA’s agent architecture (Dias et al., 2014). ........................................ 27
3.6 How mood influences emotions. ....................................................... 28
3.7 How new emotions influence mood. .................................................... 28
3.8 How new emotions influence mood. .................................................... 28

4.1 General Diagram of the application’s architecture. ....................................... 31
4.2 Post classification boundaries, negative, neutral and positive. ....................... 36
4.3 Question-Answer example in XML. .................................................... 39
4.4 Example of the new XML Knowledge Base structure with normal data. .......... 40
4.5 Important information tag examples for each value. ..................................... 43
4.6 Test set to show the use of “.SILENCE.”. .............................................. 44
4.7 Use of “.SILENCE.” as a “question”. .................................................... 44
4.8 Snippet taken out of the corpora showing the answers used to answer “.SILENCE.”’s when the application has none from the previous interaction. .................. 44
4.9 Suspect’s answer to “.SILENCE.”’s when the application has none no silence tagged stored from previous interactions. ..................................................... 45
4.10 Test set to how the application chooses by empathyRequired. ...................... 45
4.11 Setting the interpersonal states of empathy and pressure to 1 and 2 respectively. .. 46
4.12 How expression/mood of the suspect is set each interaction according to its interpersonal state. ............................................................... 46
4.13 Application selecting expression Sad for values empathy=1 and pressure=2. .... 47
4.14 Application selecting expression **Relaxed** for values **empathy**=53 and **pressure**=34.

4.15 Snippet taken out of the applications console via system prints. This shows **pressure** increasing by 10 after an instance of **truth**=-1.

4.16 Snippet taken out of the applications console via system prints. It shows a question with tag **importantInfo**=-1 and its impact on the interpersonal state.

4.17 Snippet taken out of the applications console via system prints. It shows an instance which was classified under the threshold and therefore had no impact on the interpersonal state.

4.18 Snippet taken out of the applications console via system prints. It shows two questions in the incorrect order. Note that the **empathy** stays the same as it also gains 5 points from **importantInfo**.

4.19 Amount of words repeated detected by questioning the suspect with his previous answers. Both answer and question are highlighted in yellow. Three words are detected (“Estava”, “Maria” and “quarto”) because it discards the pronouns and determiners.

4.20 Speech + Text Mode.

4.21 Video+Speech+Text Mode.

4.22 Virtual Reality Mode.

4.23 The suspect.

5.1 Example of one question/answer set in the knowledge base.

5.2 Setting suspect’s **pressure** and **empathy**.

5.3 Setting suspect’s **pressure** and **empathy** with values out of the range.

5.4 Setting suspect’s **empathy**=70 **pressure**=10 correctly identified as an happy mood.

5.5 Setting suspect’s **empathy**=10 **pressure**=10 correctly identified as sad mood.

5.6 Setting suspect’s **empathy**=80 **pressure**=80 correctly identified as nervous mood.

5.7 Setting suspect’s **empathy**=10 **pressure**=70 correctly identified as an angry mood.

5.8 Setting suspect’s **empathy**=50 **pressure**=50 correctly identified as relaxed mood.

5.9 Setting suspect’s to a boundary of **empathy**=50 **pressure**=0 and which is identified as an happy mood.

5.10 Corpora test set to validate the impact of **truth**=0.

5.11 Impact of a **truth**=-1 on the suspect’s **pressure** (increases in 10 points).

5.12 Impact of a **truth**=0 on the suspect’s **pressure** (increases in 5 points).

5.13 Questioning the suspect with his own answer (**Apesar de tudo, estou bem, obrigado.**) to which the application detects four words (**Apesar, tudo, bem** and **obrigado**). This correctly increases the suspect’s **empathy** by four points (one for each word).

5.14 Impact of a **importantInfo**=1 on the suspect’s **pressure** (increases in 5 points).

5.15 Impact of a **importantInfo**=0 on the suspect’s **empathy** (increases in 5 points).

5.16 Impact of a **importantInfo**=-1 on the suspect’s **empathy** (decreases by 5 points).

5.17 Impact of a **correctTiming**=0 on the suspect’s **empathy** (decreases by 10 points).
5.18 Impact of a $\text{correctTiming}=3$ followed by a $\text{correctTiming}=2$ (i.e incorrect order) on the suspect’s empathy (decreases in 5 points). ................................................. 57
5.19 Impact of a $\text{correctTiming}=1$ followed by a $\text{correctTiming}=2$ (i.e correct order) on the suspect’s empathy (increases in 5 points). ................................................. 57
5.20 Snapshot of one question/answer of the scripted interview. ................................. 58
A.1 Hierarchy for the Video + Speech + Text Mode. .................................................. 67
A.2 Main Menu with hover text effect. ................................................................. 68
A.3 Audio Menu. ................................................................................................. 68
A.4 Help Menu. ................................................................................................ 69
A.5 Options Menu. ............................................................................................... 70
A.6 Speech + Text. .............................................................................................. 70
A.7 Info example. .................................................................................................. 71
A.8 Question example. ......................................................................................... 71
A.9 Teach Model example. .................................................................................. 72
A.10 Video + Speech + Text Mode. ..................................................................... 72
A.11 The set viewed from the camera angle. ....................................................... 73
A.12 The set with all the components. ................................................................. 73
A.13 Clock from the scene. .................................................................................. 74
A.14 Question example. ....................................................................................... 74
A.15 Virtual Reality Mode. .................................................................................. 75
A.16 Virtual Reality Mode set. ............................................................................. 75
A.17 Rain on the window. .................................................................................... 75
A.18 Microphone level. ........................................................................................ 76
A.19 Microphone level. ........................................................................................ 77
A.20 Sound source sphere. .................................................................................. 77
B.1 Winston, fully rigged character from Reallusion. ......................................... 79
B.2 iClone's environment. .................................................................................. 80
B.3 Face Key tool. ................................................................................................ 80
B.4 Face Key’s muscle tab. .................................................................................. 81
B.5 Face Key’s Modifying tab. ............................................................................. 81
B.6 Examples of facial expressions imported. ..................................................... 82
B.7 Animator layers. ............................................................................................ 82
B.8 Right arm mask. ............................................................................................ 83
B.9 Different states of the animator. ................................................................. 83
B.10 Sub-State machine. ...................................................................................... 84
B.11 Relaxed, happy, nervous and angry facial groups. ..................................... 84
B.12 Addition of keyframes onto the audio file. .................................................. 85
B.13 Setting the “a” viseme animation. ............................................................... 86
B.14 Target sphere for head movement with a mesh rendered to be visible.
Chapter 1

Introduction

1.1 Motivation

The emergence of chatbots is large and clear. Nowadays, these are made popular at the consumer level and are aimed at several different domains: entertainment (MojiHunt\(^1\)), real estate (HOLMES\(^2\)), customer service (Twyla\(^3\)), health (Dr. A.I.\(^4\)), etc. Better and more human-like chatbots means an opportunity for serious games. Particularly, several serious games have been developed in order to train police social skills (Linssen et al., 2014). These have secured an increased importance in police training mainly due to the difficulty in recreating trustworthy scenes with real human interaction and real time feedback. Police training interviews are mainly done with the use of actors (Akker et al., 2013). This training delivers the reality components which machine interaction lacks but are usually arduous, non scalable and expensive to be arranged (Traum et al., 2005). This leads to a mostly theoretical training with only a few selected police officers being able to practice their skills. Serious games allow police officers to be immersed in a controlled universe while getting regular feedback on their behaviour. Moreover, having non-realistic scenarios provides more freedom to discover new ways to impact the suspect which would otherwise not been used. This because in real scenarios the interviewer cannot afford to make mistakes and try new approaches. The absence of duty to do everything correctly stirs the focus of these games from performing towards flexibility and experimentation, (Lucas et al., 2014). With more training, feedback and experimentation, the police academy should be able to better detect certain behaviours in real interactions and improve their ability to handle the situations. Realizing how the officer’s behaviour impacts the interaction will increase his social-awareness (Linssen et al., 2014). This is the essential motivation behind this project: to create a dialog system to train police officers on the best ways to achieve their interviewing goals. What distinguishes this project from most common dialog driven systems is the fact that these often assume cooperativeness between the user and the system. This does not happen here as suspects may have a different agenda other than that of the police officer. This project assumes a cooperative stance only if the correct actions are taken by the

\(^1\)https://mojihunt.com/
\(^3\)https://www.twylahelps.com/
\(^4\)https://medium.com/@HealthTap/dr-a-i-80b4cf06be30
interviewer hereby training the user on which actions to take.
1.2 Objectives

The main goal of this work is to create a serious game to train police officers on their interviewing skills. This includes a user interface where the interviewer can see and interact with a 3D virtual suspect. On each interaction the application should understand if the input is being compliant with the interviewing theories and reward or penalize the user. This compliance should have an effect on the displayed virtual suspect’s emotions. In order to create this application, several features are changed and added to the EDGAR dialogue system (Fialho et al., 2013). Edgar is the name of a virtual butler placed in the Palace of Monserrate\(^5\). It answers questions related to the palace such as questions about how many rooms there are in the palace or when was the palace built. This extra layer should gather important emotional information so that the most appropriate answer is selected. The emotional analysis of the interviewer’s sentences provide an heuristic emotional state of the police officer. Certain exceptions are to be added and modified to Edgar to turn it into an police interview. One example of this is the \_REPEAT\_ plugin which Edgar uses when the question is repeated or not understood. In this system the emotional state is changed when a question is repeated, unlike in Edgar, where it simply provides a different sentence with the same meaning or provides an alternative suggestion for a question to be made. Moreover, it is necessary that the suspect is characterized by a set of emotions, yielding it a simple personality/mood. Edgar has a simple question/answering corpus which must now include several emotional tags so that the system understands, weights the best answer and updates the current state of the virtual suspect. It is therefore necessary to enrich the corpus so that the system finds questions associated to that of the police officer with greater confidence and consequently giving a more realistic dialog. Having all the important features, it is crucial that the decision module makes the decision according to the Cognitive Interview Theory (Köhnen et al., 1999) used by the Portuguese Escola da Policia Judiciária. It is also necessary to understand which is the best interface. For instance, which variables could be shown to the police officer during the interview in order to enhance the learning process. One of these variables could be, for instance, a measure of how cooperative the suspect is at the moment. On a later stage it is needed to test the system and tune the emotional features added. Lastly, by the way the virtual suspect acts during the interviews, officers should pick the best approach to figure out important aspects of the investigation. To briefly sum the objectives:

- Create a personality for the suspect.
- Understand and create ways to retrieve emotional aspects from the dialogue.
- Manage how the latter impacts the suspect’s mood.
- Create a decision system to pick the correct answer given the suspect’s current mood.
- Build an interface.

All of which must be correlated with interviewing techniques such as the Cognitive Interview Theory.

\(^5\)https://en.wikipedia.org/wiki/Monserrate_Palace
1.3 Thesis Outline

This document will begin with a section about background. This is relevant as it involves description of psychology theories and interviewing techniques, as well as a short insight on EDGAR. Related Work follows in the second chapter where similar systems are discussed and what particularities can they bring to this project. Next, on the forth chapter the architecture of the system and its modules are detailed. More specifically:

- A general overview.

- Sentiment analysis and retrieval of important information according to the theories discussed in chapter two.

- A section regarding how the knowledge is structured.

- Which tags are being retrieved and why.

- A section regarding the scaling used in the tags.

- How the suspect picks the appropriate answer according to the interviewing techniques.

- The modeling of mood/personality of the suspect and how the tags interact with it.

- Small description of the user interface which include character animations.

On the fifth chapter there is the evaluation of the serious game. Lastly, on the sixth chapter, conclusions and which future work is, in my opinion, interesting to be developed regarding this subject and project.
Chapter 2

Background

2.1 Edgar, the virtual butler

As previously referred, the main goal of this work it to add features to the already existent EDGAR in order to make it suitable for a police interview. It is, therefore, important to describe it. EDGAR (Fialho et al., 2013) is a question-answer architecture. This embodied virtual agent answers questions about the palace of Monserrate\(^1\). EDGAR’s game environment is developed in Unity\(^2\). The speech is recognized by AUDIMUS (Meinedo et al., 2003a) which translates it to a sentence with a confidence value. This system selects one answer if it finds a match for the input, or uses string matching and string distance strategies to find the best answer. This process is done through the NLU (Natural Language Understanding) module. As seen in figure 2.1, the Manager (dialog manager) controls three main

---

\(^1\) Palace of the 19th century located in Sintra, Portugal

\(^2\) http://unity3d.com/
components:

• **STRATEGIES** Two different strategies are applied, mostly to measure string distances. One of these strategies is Jaccard which takes into account how many words are shared between the user’s interaction and the knowledge source. A second strategy is Bi-gram Overlap which gives preference to the shared sequences of words and Tf-idf statistic which contributes to the results attained by previous measures by giving weight to infrequent words (for instance if the word “MONSERRATE” occurs very often in the corpus it should have a lower weight because it provides little information whereas the word “owner” should have a higher value). Adding to the previous two strategies we have **Strategy modifiers** which allow the modification of the results from one strategy. For example, a modifier can filter answers for each strategy based on the input type. For instance, one may only accept questions which score between 0.3 and 0.9 on Jaccard. This added flexibility allows to answer certain questions or types of questions through specific strategies. All strategies are executed in parallel, with as many processing threads as the number of CPUs available, and the highest scoring reply is selected.

• **PLUGINS** This is the same component responsible for triggering a control tag (more specifically “_REPEAT._”) which tells the program that the question was not understood if the input was not correctly resolved by STRATEGIES or if the ASR\(^3\) returned a confidence level lower than a threshold respectively. The system will then substitute the control tag for an answer such as “I am sorry I could not understand, can you repeat please?”. There are pre-strategy (executed before the strategies) and post-strategy (executed after the strategies and with access to their result).

  - **Input plugins** are able to change or add information to the input received by the system, before the strategies take their decision.

  - **Independent plugins** are post-strategy plugins called in concurrency and before chained plugins. Here is where lays the plugin to choose the answer based on the repeat levels.

  - **Chained plugins** are also post-strategy plugins called in chain, one after the other, so they can see and take decisions based on previous plugins. They are always the last type of plugins being called and the order they are present in configuration file is taken in consideration.

• **HISTORY-TRACKER** This keeps knowledge of previous interactions until a default time without interactions is reached. History trackers are responsible to store interactions that may be relevant for the dialog system. The *IdleTimerHistoryTracker* tries to keep up to a fixed number of interactions are purged if the agent remains inactive for too long. For each interaction with a different user the Dialog Manager creates a new instance of the chosen history tracker so special care should be taken when modifying structures that share interaction knowledge between different user interactions.

  The question-answer pairs are stored in XML files which make it easier for corpus enrichment (Leuski et al., 2009).

\(^3\)Automatic Speech Recognition
Figure 2.2 shows an example of EDGAR’s corpora. The first block is for different ways of asking the same question and the second block has possible answers randomly chosen. Each set of question-answers contains several questions with the same meaning, together with a set of answers from which the system will pick one to display. The XML is enriched with an ID tag for each question-answer set. Both questions and answers have English and Spanish denoted by \textit{en} and \textit{es} respectively. The answers may, or may not have the tag \textit{emotion} and \textit{intensity} which lets the front-end know which expression and with what intensity Edgar has to produce (i.e the expression tag can take a value such as “happy” which tells the front-end that Edgar should be smiling).

To deal with slang, some small talk, questions about cinema and compliments EDGAR makes use not only of XML but also AIML\textsuperscript{4}. In AIML, knowledge is integrated as a set of rules which will match the user’s input associating it with templates. For instance, in XML “You are such an idiot” would only match that specific sentence whereas AIML “You are such an <\textit{slang}="slang"/>” would match an infinite number of sentences.

The chosen answer is selected and synthesized with the Text To Speech (TTS) system DIXI (Oliveira et al., 1991). The output is then heard through the speakers accompanied by displays of emotions based on movements and facial body language. An abstraction of the system is showed in 2.3

\textsuperscript{4}Artificial Intelligence Markup Language
2.2 Interviewer’s Intentions, Goals and Obligations

Most discourse models are based on an analysis of the intentions of the speakers (Cohen and Perrault, 1979)(Grosz and Sidner, 1986)(Allen and Perrault, 1980). In this work, however, the suspect has the intention of saying as little as possible to incriminate himself but must account for some obligations. A model proposed by (Traum and Allen, 1994) is based on both obligations and intentions. Obligations are actions which are either forbidden or obligatory. For instance, the simple act of answering a question, making promises or offerings are or incur in social obligatory actions and the act of swearing is or incurs in a social forbidden action. This being said, the agent’s behavior is resolved by factors which include the agent’s goals and a set of social-obligations. Goals and obligations are both considered before decision. If there is no conflict then the agent should pursue the goals. On the other hand, if there is conflict, the agent satisfies the obligations first. Notwithstanding, when the set of obligations and rules does not comply with the agent’s personal goals, it may choose to violate them (e.g., omit information, not answering or even lying). It is assumed that agents generally plan their actions to violate as few rules as possible. Broadly speaking, the model proposed by (Traum and Allen, 1994) is a looped discourse algorithm which checks if there are obligations, if so addresses them, checks for performable intentions and finally updates the obligations and intentions. In my work the concept of omitting information or lying is permanent therefore a similar but simpler model is applied. The suspect may omit information which incurs in an increase of his anxiety/pressure as it violates a social obligatory action to rightfully answer.

2.3 Rapport and Entrainment

Rapport in an interview is decisive for a successful cooperation from the suspect (Walsh and Bull, 2012). It is a psychological term which refers to the ability one has to create levels of trust and understanding towards another. This impacts the process of responsiveness at the unconscious level. This is why most interrogation techniques involve rapport. It creates a level of compliance valuable to obtain confessions. It must be built and maintained. A more detailed explanation describing the basic rules for establishing and maintaining rapport can be found in St-Yves (2013). Rapport in police interviews (Abbe and Brandon, 2014) is more easily established if the questions from the interviewer are never personal, for instance “Acha que uma pessoa normal faria uma coisa dessas?!” (“Do you think a normal person would do such a thing?!”). These building rapport questions should be as simple as verifying the suspect’s address, phone number, the spelling of a name or work history. More experienced interviewers can evaluate the behavioral and physiological responses to these questions. The interviewer can understand how to calibrate what is a normal response from the suspect and work from there. This eases the evaluation of truthful and deceptive responses later in the interview. A common way to tell if the level of rapport is high in a conversation is through mirroring. Mimicking or mirroring is when a person tends to suit his/her body language (posture, gestures, mannerisms, etc.), tone of voice and choice of words to the person he or she is talking to. What is interesting is that mimicking can be forced by on party to incur in rapport. When the pitch, loudness and speaking rate are similar in both speakers, there is what
is known as acoustic-prosodic entrainment. That is, a sort of mimicking at acoustic and prosodic levels. A way of understanding if rapport has been established is by making small movements like crossing or uncrossing legs. If the interviewee mirrors it means the rapport has been established. Further reading on rapport can be found in (Hoffman, 2005). In a personal note, rapport is such a good interviewing tool because it creates the illusion, in the suspect’s perspective, that he/she is talking to himself/herself. What would not one say to oneself? Rapport plays a major role in the interview and therefore must have impact on my virtual suspect decision making. In fact, the whole interview is based on creating as much rapport as possible. In this project there are several ways of detecting rapport. The more rapport there is, the more trustworthy information the interviewer will get.

2.4 Interviewing Techniques

Different techniques have been and are used to retrieve the most information in the most accurate way. One of this is the Reid Technique (Buckley, 2000) which was widely used. It uses nine steps:

1. Direct confrontation. Giving the suspect an early opportunity to explain.

2. Shifting the blame away from the suspect. This way putting the suspect at ease, psychologically justifying or excusing the crime.

3. Discouraging the suspect from denying his guilt.

4. Trying to use the reasons the suspect gives for how he or she could not commit the crime to move towards a confession.

5. Reinforce sincerity to ensure the suspect’s receptiveness.

6. At this stage the suspect will most likely become quieter and reticent. Move the theme towards offering alternatives.

7. Pose “alternative questions”. This type of question usually has two optional answers, one more socially acceptable than the other. The suspect is expected to choose the easier option but admitting guilt.

8. Lead the suspect to repeat the admission to establish the validity of the confession.

9. Documenting the suspect’s confession.

The technique is far more complex (for more information see Reid's official site\(^5\)). It relies on constant tests for reliability on known answers to control questions, rapport and body language reading to identify anxiety, which, according to Reid, is a signal for lying (Inbau et al., 2013). However, extensive research by Douglas Starr (Starr, 2013) and other critics (Gudjonsson and Pearse, 2011) have proven that the Reid technique relies on discredited psychology. The idea that anxiety connotes lying is a very biased issue and therefore should not be taken into account where justice is to be applied. Many liars can

\(^5\)www.reid.com
have the conviction of the universe whereas a honest person may show anxiety because it is a normal reaction to a stressful situation.

A second and widely known strategy is Good-cop/Bad-cop (Brodt and Tuchinsky, 2000). It is a psychological tactic involving two interrogators taking opposing approaches towards the suspect. Upon aggressiveness from the bad cop the suspect tends to be cooperative with the good cop who offers more support, trust and understanding. In a high stressful situation people usually seek comfort, sympathy and cooperation. This technique is usually disregarded nowadays for it alienates the suspect.

A third technique used is the Kinesic Interview Technique (Walters, 2016). It involves body language evaluation and connecting it to what the suspect is saying. It states that certain behaviours indicate guilt such has slumping in a chair or crying. It is a very aggressive approach on a more subconscious level. Interrogators use stress manipulation techniques to elicit confessions. Kinesic Interviewing involves letting the subject do all the talking without interruption allowing a baseline story to evolve. Confession rates increase with each additional hour of interrogation due to fatigue and increased stress. The police officer can later dissect what the suspect said to find flaws. In this project, both the PEACE Model and Cognitive Interview (see Section 2.4.1) are taken into consideration. These are partially similar approaches to that of the Reid’s technique but without its evasiveness. That is, the PEACE Model and Cognitive Interview are based on creating rapport and letting the suspect talk without interfering with the story whereas in the Reid’s technique the interviewer has a much more active role which might incur in not so trustful testimony. The model chosen on the project has impact mostly on what should be rewards or penalties based on what the interviewer chooses to ask.

2.4.1 PEACE Model and Cognitive Interview

The virtual agent has a story and information which is accessed by the officer if the rightful behaviour and course of the interview are met. This is done by theories from social psychology. There are various models and guidelines for police interviewing. One of these models was developed by the police forces of England and Wales (Griffiths et al., 2010) and has the acronym of PEACE (Paulo et al., 2014a) which stands for:

- **P**reparation and Planning
  - Define main goals
  - Analyse the available proof
  - Master the interrogation technique
  - Define the place of questioning, context where is going to take place and who is going to lead it.

- **E**ngage and Explain
  - Identify what are the legal requirements
  - Elaborate an explanation of the situation
- Develop a relationship with the suspect resourcing the process of communication (strategy)
- Explain the routines and proceedings of the interrogation context
- The police officer should retain that the main goal is information: reducing barriers to communication process; promote that the suspect tells “his side of the story”; and promote detailed answers.

- **Account, Clarify and Challenge**
  - Resort to communication strategies that promote information gathering
  - Questioning Form: Open Questions, Avoid Interruption
  - Identify the aspects related to lying (verbal and nonverbal cues)
  - Promote detailed information statements by the suspect

- ** Closure**
  - Explain what is going to happen in the suspect’s future
  - Allow time for the suspect to formulate questions
  - Appreciate the collaboration

- **Evaluation**
  - Analyze the information collected from the interrogation.
  - Evaluate the impact of the interrogation on the criminal investigation process.
  - The police officer should promote a process of self-evaluation and supervision (improving their interrogation skills).

These were taken from “Application Scenario Outlines: EPJ-MJ use case “Interview Skills for Police Officers (ISPO)” from Escola de Polícia Judiciária.

PEACE was developed in the early 80s and 90s to tackle the vast number of false confessions which resulted from an accusatory style of interviewing. The model is based upon the fact that if the interviewer manages to create rapport with the subject it will eventually lead to a more cooperative conversation in contrast to an atmosphere charged with aggression and intimidation.

Besides PEACE there is the Cognitive Interview developed by Geiselman e col. (1984). Its objective is to get the most of information without distorting it (McMahon, 2000). Cognitive Interview is recognized as one of the best models of interview (Fisher and Geiselman, 1992)(Prescott et al., 2011). The Cognitive Interview was developed into what is now called Enhanced Cognitive Interview (Fisher and Geiselman, 1992). This work is based on the Enhanced Cognitive Interview adaptation by (Paulo et al., 2014a) which is also influenced by PEACE and the outlines of the document “Application Scenario Outlines: EPJ-MJ use case “Interview Skills for Police Officers (ISPO)” from Escola de Polícia Judiciária.

Enhanced Cognitive Interview (ECI) has nine phases:
1. **Establishing Rapport:** The interviewer should shake hands, avoid having an stiff posture and introduce himself. Moreover, all the questions in this phase should be both open and neutral such as “How was your day?” and far from the topics related with the crime. Empathy, reducing anxiety and familiarization with the suspect's communication style are the major points in this phase. This phase should be as long as possible.

2. **Explaining the objectives of the interview:** In this phase the interviewer should communicate in a clear and precise way the main points of the whole interview. The interviewer should explain how detailed he wants the answers to be and that he will try not to interrupt the suspect. This allows the interviewer to transfer the control of the interview to the suspect.

3. **Free speech:** The objective of this phase is to get the best report possible out of the subject. The interviewer should not ask questions which distort the subject’s memory in any way possible. Questions such as “What about the crime weapon?” is a suggestive question which might twist the memory of the subject or discard information which would otherwise be told. The interviewer should try to question as little as possible during this phase and only make use of open questions if necessary to switch the direction of the interview. The police officer should promote free recall (e.g., “Is there anything you want to talk about?”) and make use of explanatory verbs such as “to say” or “to explain”. Avoiding the main crime related questions and not sharing the Police's information is encouraged.

4. **Questioning:** After having a detailed and extensive report, the interviewer may now question the subject about it. Open questions such as “could you describe the man that hit you?” should still be used to get more information out of the subject. Closed questions like “what was the offender wearing?” may be used but only to complete information or when open questions fail to examine confusing or omitted information.

5. **New information recovery strategies:** In this phase the interviewer should adapt the interview to the suspect. The interviewer may ask for him/her to tell a specific part of the story in a different time order or ask to tell the story from a different perspective. The interviewer may also ask the subject to report by changing the perceptive sense. In other words, people tend to remember events visually. Making use of other senses such as hearing or smell may increase the information retrieved.

6. **Important questions to the investigation:** Until this phase every question should be about what the subject said in order to avoid suggestive questioning. Sometimes it is fundamental to question about specific topics which were not told by the subject. Since the methods of questioning and free speech have been applied the interviewer may now question the subject about specific things such as “Did you see the knife?”. This allows to collect new information even if slightly distorted which may help in the investigation and will not change the information collected so far.

7. **Summary:** The interviewer can now sum up the major points of the subject’s report giving him opportunity to add or correct information.
8. **Closure**: The interviewer should now go back to neutral questions to soften and calm the subject down, appreciate the interviewee’s collaboration and allow questions from the interviewee’s part.

9. **Evaluation**: It is crucial to evaluate and reflect upon the information retrieved.

Throughout all the interview the use of silences and open questions are encouraged. Even during phase one the interviewee should be asked open questions so he/she gets more accustomed to a more elaborated and active speaking style of communication while not being intimidated (Paulo et al., 2014b). The PEACE model and Cognitive Interview are the guidelines used in this project to understand what must be reward or prejudice to the interviewer. The use silences, use of rapport, following of the stages of interview and the avoidance of leading questions impact the suspect's mood and its decisions on what answer to pick.

### 2.4.2 Interviewing Strategies

There are several strategies, techniques or small nuances used by the police to retrieve information or to understand, for instance, if the suspect is lying. I will enumerate some of the most important ones according to the police seminar held at Instituto Superior Técnico by Policia Judiciária.

- **Asking to reconstruct the circumstances of the event**
  The suspect or witness is asked to reconstruct all the circumstances surrounding the incident, from the beginning to the very end. Furthermore the subject is asked to think about the tiny details during the incident such as the weather, the noise and even the interviewee’s emotional state.

- **Instructing the subject to report everything**
  The interviewee is asked not to leave out any details regardless of how small they may seem (as explained in section 2.4.1).

- **Recall the events in a different order**
  A very often technique applied by the police is to ask to describe the incident backward, forward or even from a specific point in the middle of the story. This technique incurs in a way of improving the number of details the suspect remembers or finding flaws in the story for later confrontation. Research has proven that if a person is creating a story it is almost impossible to tell the story out of sequence.

- **Perspective Change**
  The interviewer asks the suspect to describe the incident as if it had happened in a different specific location or change roles with another person in the incident and consider what he or she might have seen and felt (Zulawski et al., 2001).

- **Building up**
  There is no particular name for this technique which was the most emphasized technique during the Policia Judiciária’s seminar and is very often referred in police interview literature (Vessel, 1998).
(Zulawski et al., 2001). This technique assumes that deceiver will gradually build up a series of false explanations. The more the subject lies, the more he has to juggle in his mind. Eventually, an inconsistent detail will break down the whole fabrication. In other words, the suspect should always be encouraged to do most of the talking and spurred to keep the story going. If the person is being honest, he or she is recollecting from memory and stress will not fail them to remember. In contrast, if the person is fabricating a lie, sooner or later he or she will commit flaws. Elevating and exhausting the suspects mental activity is what this technique is all about. Laurie Halse Anderson, an American writer ends up describing the power of this technique in a very unique way in one of her crime novels: “The trick to surviving an interrogation is patience. Don’t offer up anything. Don’t explain. Answer the question and only the question that is asked so you don’t accidentally put your head in a noose.” (Anderson, 2014).

- **80/20 Rule**

Not so much as a technique but a rule. Nevertheless it states that during the interview, the interviewer should talk for 20% and the suspect for the remaining 80% allowing not only for more information recovery but also for less footprint from the police on what was recovered (this was shown and explained during the seminar by Policia Judiciária).

These techniques are relevant to this work. Ideally we should be able to detect if these techniques and nuances are being applied so that the virtual suspect cooperates and leaks the correct amount of information. Correctly detecting them is a troublesome and arduous task. However, the mental building up of the suspect is such a crucial aspect of the interviews that I will explain the way I tried to tackle this issue later on in this document.

### 2.5 Interpersonal Stance Theory

Dutch police is trained using the theory of interpersonal stance. This is a communication model which represents the human interpersonal behavior (how a person feels towards the other) in a graph with affection (Opposed or Together) on the horizontal axis and power (Submissive or Dominant) on the vertical axis (Leary, 2004). This graph is often pictured as a circumflex divided into eight different areas which correspond to the interpersonal stances as seen in Figure 2.4.

Following Figure 2.4 a dominant behaviour will trigger a submissive one (and vice versa). An Together behaviour triggers another behaviour with the same level of affiliation (Leary, 1958). For instance, Compete behaviours are matched with Withdrawn behaviours whereas Helping behaviours are matched with Cooperative behaviours. Interpersonal Stance Theory is relevant to bring this serious game closer to real life in terms of emotion trading. This theory is behind the PEACE Model and Cognitive Interview as the ideal stances are the ones with positive Together behaviours (Leading, Helping, Cooperative and Depend behaviours).
2.6 Defining Personalities and Moods

Measuring and describing personalities as a construct has been challenging and object of much de-
viance throughout the years. One cannot completely define a person into certain blocks with values. However, to overcome this complexity, researchers have abstracted from the full definition and made
different models with the most important features.

2.6.1 Five Factor Model

The so called Five Factor model (McCrae and John, 1992) is a hierarchical organization of personality
traits in terms of five basic dimensions:

- **Extraversion** - Energy, positive emotions, surgency, assertiveness, sociability and the tendency to seek stimulation in the company of others, and talkativeness. For instance, outgoing/energetic vs. solitary/reserved.

- **Agreeableness** - A tendency to be compassionate and cooperative rather than suspicious and antagonistic towards others. It is also a measure of one’s trusting and helpful nature, and whether a person is generally well-tempered or not. For instance, friendly/compassionate vs. challenging/detached.

- **Conscientiousness** - A tendency to be organized and dependable, show self-discipline, act duti-
fully, aim for achievement, and prefer planned rather than spontaneous behavior. For instance, efficient/organized vs. easy-going/careless.

- **Neuroticism** - Identifies certain people who are more prone to psychological stress. The tendency
to experience unpleasant emotions easily, such as anger, anxiety, depression, and vulnerability. Neuroticism also refers to the degree of emotional stability and impulse control and is sometimes referred to by its low pole, emotional stability. For instance, sensitive/nervous vs. secure/confident.

- Openness to Experience - Appreciation for art, emotion, adventure, unusual ideas, curiosity, and variety of experience. For instance, inventive/curious vs. consistent/cautious.

As pointed out in (McCreery et al., 2012), despite being widely adopted it does have limitations:

1. The failure to provide causal explanations (McAdams, 1992).

2. The lack of account for situational (Mischel, 1968) or motivational influences (Dweck & Leggett, 1988).


Moreover (Mehrabian, 1996) describes with greater detail both FFM and its history as well as providing several tables with examples of adjectives and scales.

2.6.2 Pleasure, Arousal and Dominance emotional state model - PAD

This emotional state model was developed by Albert Mehrabian and James A. Russell (1974 and after) to measure emotional states (Mehrabian, 1996). It uses three dimensions:

- **Pleasure-Displeasure Scale** which measures how pleasant or unpleasant one feels. For instance, joy is a pleasant emotion whereas fear is an unpleasant emotion.

- **Arousal-Nonarousal Scale** measures how intense one feels. For instance, both boredom and rage are unpleasant emotions, but boredom is a nonarousal feeling, whereas rage is an intense arousal feeling.

- **Dominance-Submissiveness Scale** measures how dominant an emotion is. For instance, anger is a dominant feeling whereas fear is a submissive feeling.

Many more descriptions of the model and through the years it has been adapted in several ways as to simplify or complicate the number of possible states. There is a difference between mood and personality. Personality is how a person reacts to certain actions or events. These incur in different moods which are the current states of a person (for instance happy or sad). For example, a positive person is prone to be happiness. In this project, only the mood of the suspect is defined as having a personality would bring unprofitable complexity. The suspect’s mood is based upon the PAD’s state model. This is shown in Section 4.7 differing in the names chosen. The Pleasure-Displeasure scale is denoted as Empathy and the Arousal-Nonarousal scale is denoted as Pressure.
2.7 Serious Game Lemniscate Model

The main objective of the game is to develop police interviewing skills. From an educational point of view we must understand how to better affect the learning curve of the trainees. The Serious Game Lemniscate Model (SGLM) asserts that players need to be drawn out of the game world where they are intuitively acting to a learning state where they can rationally reflect on the game experience (Koops and Hoevenaar, 2012). In order to create functional knowledge there must be both experiential and reflective cognition (Kolb, 2014) and that is why the Serious Game Lemniscate Model is relevant. In Figure 2.5 the SGLM is shown in a more detailed manner.

![Figure 2.5: Serious Game Lemniscate Model (Koops and Hoevenaar, 2012).](image)

The interface provided in this project gives the option to show the suspect's current mood during the interview and giving the interviewer a chance to choose both the above mentioned experiential and reflective cognition.

2.8 Turn Management or Impression Management?

One factor which influences turn taking behavior is emotions (ter Maat and Heylen, 2009). Work by Mark ter Maat and Dirk Heylen has been done regarding the management of turns and how it influences the idea that people get from the virtual agent. The authors state that by observing turn taking behavior we can perceive the personality or current emotions of a person. By studying the detection of some interpersonal variables in conversations they managed to realize what impact turn taking behavior has on how the agent is perceived. In their study, ten people listened to the scripts of conversations and rated the speakers from zero to five according to the scales the following scales: unfriendly-friendly, distant-pleasant, passive-active, cold-warm, negligent-conscientious, disagreeable-agreeable, rude-respectful, unpredictable-stable, unattentive-attentive, submissive-dominant, undependable-responsible, negative-positive, not aroused-aroused. In other words, a subject in their studied rated the people in the script on scale from zero to five concerning their passiveness, attentiveness, etc. The results agreed greatly with what one would expect. For instance, if a person overlaps the other in a conversation it usually means that the person is more active and disagreeable. So turn-taking does seem to have an effect on...
the perception of the agent. One of the aspects relevant to my work is that this study implies the scales and their resolution rated by the participants in the study are able to define correctly the personality of the agent. In this project the suspect only answers and there is no detection of overlaps making the presence of a turn manager unfit. However, this shows that an ideal work would have a turn manager which accounts for emotions and overlaps in order to bring the serious game a step closer to real life interviews.

2.9 Police Interview Analysis

Several staged interviews were kindly provided by Escola da Policía Judiciária (EPJ). Analysis of these recorded enacted interviews yielded in several important points:

First interview, female suspect - In this interview there are several attempts of relaxation of the suspect and creation of empathy via generic questions about family and habits. From this, the interviewers take the suspect to the outer edge of the interview by explaining the reason of the interview. Officers keep posture even after confession so the suspect keeps on talking. It is noticeable the use of back-channeling (words such as “claro”, “está certo” or “pronto”). In sync with the former, there are perceptible mild laughs in order to express agreement.

Second interview, female suspect - The interviewers begin by contextualization - who are they and what are they doing. It is notable the repetition of the suspect's name many times along the interview in an attempt to establish rapport. Likewise, tone and vocabulary are adapted to the suspect. This includes the use of swear words. There is no use of incriminating sentences and the day of the incident is constantly asked to be reconstructed. Whenever the suspect displays signs of distress he is quickly calmed down. To do so, the interviewers express agreement, comprehension and say good things about him.

Third interview, male suspect - The last recorded staged interview is more of a bad example on what not to do. The interviewers are too aggressive. There is a failed attempt to establish rapport via general problems about life. The interviewers fail on the correct stages of the interview leaving the objective of the interview little addressed.

In the first two interviews it is noticeable the presence of the stages of the PEACE Model as well as the effort to create and maintain rapport. These serve as an example for what a normal interview should be. What is also taken out of these enacted interviews is that in reality these are compiled with a lot of “back and forth”, mumbling, audible fillers and overlaps which are difficult to produce in a virtual suspect.

2.10 Suspect’s Mindset

In a typical interview the suspect has a very specific goal inherited from his mindset. Usually the suspect is asking whether should he talk or keep information for himself as far as possible. Should he lie or tell the truth or wonder if his confession will help or complicate his defence. These are all questions one is
subjected to, even before the interview starts. (Leo, 1996) showed that by making use of positive and negative stimulus, 76% of the suspects incriminated themselves partially or totally.

**Negative stimulus:** Suggesting the suspect has no other other alternative but to confess. Confronting him with existent proofs and have their denials weaken. Identifying contradictions in their version and alibis.

**Positive stimulus:** Suggesting that the suspect profits from confessing or telling the truth. Trying to defend the interests of the suspect. Work on self-esteem. Making use of moral justifications or psychological excuses for what what made diminishing the guilt. Minimizing the gravity of the incident. Showing how important it is to cooperate with the authorities.

The negative and positive stimulus are present along the PEACE Model and Cognitive Interview. The latter excel in that the stimulus have a right place in time to come into action. For instances, the PEACE Model claims that the subject should be confronted with contradictions but only in the end of the interview. As this project follows the guidelines of the PEACE Model and Cognitive Interview, it also takes into consideration the positive and negative stimulus. However, this is done when constructing the knowledge base as explained later in Section 4.

### 2.11 Summary of what is good and bad for the interview applied to the game

It is important to make a summary of what are the general DO's and DON'T's in the different parts of the interview. Firstly, what should be done throughout the interview:

- Familiarization with the interviewee’s communication style (including tone, rhythm and choice of words)
- Reducing anxiety (information about the context)
- Neutral Posture
- Approach only goals of the interview
- Active listening
- Make use of open question's
- Interviewer is a guide in the communication process. This meaning the interviewer is a way to get information, not to create or distort it in anyway.
- Focus on information recovery
- Report everything that has been remembered
- Avoid having an stiff posture
- Non-threatening approach
- Appeal to the suspect's interests
- Reduce the suspect's guilt, make it seem reasonable.
- Show how useless it is to negate guilt
- Flatter the suspect.
- Reformulate the words that you heard.
- Silence management: Let the subject reflect about what was said correctly and reinforce it.
• Encourage: While the interviewee speaks show attention and agreement, verbally or non-verbally.
• Repeat what was said
• Availability to Listen

Secondly, what should not be done during the whole interview:
• Avoid sharing the Police’s information.
• Avoid suggestive questioning
• Interrupting the suspect.
• Forced-choice questions such as “Was it during the morning or afternoon?”
• Multiple questions such as “Did you see the offender? Was he sitting or standing up?”
• Leading questions such as “He was an aggressive man was he not?”

Thirdly, what one should be compliant with during each phase:

Establishing Rapport:
• Explain the interview goals
• The interviewer should tell how detailed he wants the answers.
• Approach neutral events and stay away from topics related to the incident
• Resist temptation to shorten this phase
• Shake hands.
• Introducer should introduce himself.

Free speech:
• Interviewer should ask questions as little as possible.

Questioning:
• Open and neutral questions such as “How was your day?”
• Creating empathy, reducing anxiety and familiarization with the suspect’s communication style are the major points in this phase.
• Talk slow and use short sentences
• Promote free recall: “Do you know what are you here for today?”
• Only make use of open questions necessary only if necessary to switch the direction of the interview.
• Open questions should still be used.
• Use of practical techniques such as change of perspective, story time line or remembering the story through other senses.

Important questions to the investigation:
• Here one might question about specific and suggestive topics which were not told by the subject such as “Did you see the knife?”. This allows to collect new information even if slightly distorted.

Summary:
• The interviewer should sum up the major points of the subject’s report.
• Give the suspect opportunity to add or correct information.
• Positive Reinforcement

Closure:
• Appreciate the interviewee’s collaboration
• Allow questions from the interviewee’s part.
• Ask for more detailed personal data
• Go back to the neutral topics

Evaluation:
• It is crucial to evaluate and reflect upon the information retrieved.

Lastly, the interview should follow the general phases of the Cognitive Interview:
• Establish Relationship
• Allow recovery/ Free narrative
• Place Questions
• Closing the Interview

Regarding this project, these DO’s and DON'Ts are taken into account when during the course of the interview. This is further discussed in Section 4.
Chapter 3

Related Work

Several Virtual Suspects have been developed throughout the years. It is important to understand which aspects contribute to this work.

3.1 LOITER

LOITER is a serious game in which players aspire to resolve a conflict with a loitering group of juveniles who are causing a nuisance by playing loud music and insulting passers-by, in a peaceful manner (Linssen and Theune, 2014). It develops the players social awareness. By analysing interactions between police officers and civilians, researchers came up with a human model behaviour so that the virtual characters respond believably to human users. The virtual juveniles use artificial intelligence to respond to the player according to theories from social psychology. This approach is unique by its ability of the characters to explain the reasoning behind their actions. LOITER makes use of two meta-techniques (information outside the enacted world setting). One of these techniques is the interruptions between acts of a live action role play during which players discuss the previous and following acts. In these breaks, information is exchanged about their motivations for certain actions and what the effects of these actions were (see 2.7). Consecutive acts have increasing levels of interaction complexity. The second meta-technique is providing feedback during the acts less intrusively through ‘thought bubbles’ giving the player insight on the suspect’s personal thoughts. The social psychology underlying the model is the concept of stance which relates to interpersonal behaviours by classifying attitudes people have toward each other in the interpersonal circumplex (see 2.5). This is based on the work of Bruijnes (Bruijnes et al., 2015).

Thought bubbles are not very useful to my project as these do not aid in the construction of mental models (Linssen et al., 2015b). The interesting part to my project is how the interviewee’s personality is defined and how it changes throughout the conversation. LOITER’s psychologic model is based on three major theories:

- Interpersonal stance (describes how people assume a certain stance toward the other when they interact) (Leary, 1958);
The face theory (related to the need of approval and autonomy a person has or “the reciprocal influence of individuals upon one another's actions when in one another's immediate physical presence” as defined by the Sociologist Erving Goffman in book The Presentation of Self in Everyday Life) (Goffman, 2005);

Rapport (the bond two people share) (Tickle-Degnen and Rosenthal, 1990) (see 2.3).

With these three theories we are able to describe the majority of interactions between police officers and suspects (Linssen et al., 2015b). While LOITER is a serious game to resolve street conflicts, some of its guidelines apply to this project as well. Mainly the creation of rapport. Note that in LOITER, the player can choose between four different interactions. This restriction allows the makers of the serious game to make use of the theories in a more stable and precise way compared to this project.

3.2 AGENT

AGENT (Awareness Game Environment for Natural Training) is a virtual environment in which serious games can be enacted. It combines research on interactive storytelling, game design, turn-taking and social signal processing (Linssen and de Groot, 2013).

This game’s environment relies on three major components:

- A communication module to regulate which modules are connected and used during a game;
- A storytelling engine which keeps track of the virtual world and controls the non-player characters (NPCs);
- A user interface (UI) which processes input and output.

This is made possible through the use of a message broker that serves as middleware between the different modules. During the design process the most important factors that are taken into account are:
NPC behavior, such as planning to take actions and turn taking in dialogs; the inclusion of gaming elements that give feedback about progress in the game; and social signal processing (SSP) to determine how the player behaves. The architecture of this project is less complex to that of AGENT. There is no storytelling engine as there are no NPC's and the communication module is replaced with a decision module which calls different strategies to answer the questions.

### 3.3 First Person Cultural Trainer

FPCT, short for First Person Cultural Trainer, is a serious 3D game developed for intelligence missions in Afghanistan and Iraq (Zielke and Linehan, 2009). Here the goal is to establish rapport and ultimately gather information about IEDs\(^1\). The user is expected to rate moods and reliability by looking at the virtual agent’s verbal and non-verbal behavior. The NPC has a personality or psychological model that includes personality, emotions and cultural attitudes which impact the its stance.

![Figure 3.2: Example of an interaction with the player and the NPC.](image)

The NPC’s psychological state is dynamic, for it defines and is defined by the environment, other NPC’s and even the player’s decisions. The design of the living-world architecture and the cultural models is ongoing and its cultural and environmental framework is still being developed to achieve a more realistic and believable synthetic training experience. This theory, as explained in 2.7, states that in an educational game a player is either in an intuitive gaming state or in a learning state rationally reflecting on the gaming experience. Work by Martijn Koops and Martijn Hoevenaar has showed that this theory not always works (Koops and Hoevenaar, 2012). FPCT uses four motivators to establish their psychological model. Table 3.3 shows how FPCT relates the emotions and the motivators.

<table>
<thead>
<tr>
<th>Psychological Motivator</th>
<th>Strong Negative</th>
<th>Strong Positive</th>
<th>Weak</th>
</tr>
</thead>
<tbody>
<tr>
<td>Survival</td>
<td>Fear</td>
<td>Devotion</td>
<td>Security</td>
</tr>
<tr>
<td>Ego</td>
<td>Anger</td>
<td>Satisfaction</td>
<td>Tranquility</td>
</tr>
<tr>
<td>Reason</td>
<td>Interest</td>
<td>Interest</td>
<td>Apathy</td>
</tr>
<tr>
<td>Morality</td>
<td>Sorrow</td>
<td>Joy</td>
<td>Neutrality</td>
</tr>
</tbody>
</table>

![Figure 3.3: Table relating the four motivators and emotions associated with them.](image)

\(^1\)Improvised Explosive Devices
The motivators are what drives the NPC to act. The emotions are how and to what degree those motivators are being felt. For instance, a strong positive survival instinct derives in a sentiment of devotion, whereas a negative survival instinct derives in a sentiment of fear. Another aspect about the First Person Cultural Trainer is that its emotional state can be modified in run-time by an instructor. This would negatively impact my project, as the police officers are not being trained to face every possible course of conversation and emotional states but rather a mental model defined by the Cognitive Interview and PEACE Model (see 2.4.1).

3.4 Hassan: A virtual Human for Tactical Questioning

Hassan is a virtual human for training Tactical Questioning dialogues (Traum et al., 2007). The police is trying to figure out why the local population in Iraq continues to use a broken-down market place instead of a recently built one. If the police trainee convinces Hassan to help him he will confirm that a tax has been levied on the new marketplace by Hassan’s employer. If the police trainee fails to convince Hassan to help, it may go as far as to lie about the tax. Figure 3.4 shows the interface used. A press and hold button to speak, several aspects about the interpersonal state of Hassan and his compliance level, a 3D scenario and the historic of the conversation.

The system uses an information-state based dialogue manager to track several states in order to decide how compliant the agent should be. Dialogue features and the best answer given a particular compliance level are picked by the use of several statistical classifiers from NPCEditor software (Leuski and Traum, 2010). These identify three features of the dialogue: a dialogue move, a main topic and a level of politeness. The dialogue moves are a way to characterize the actions/moves that the player or the NPC can do. The dialogue moves used are: opening (greetings, introduction), complimentary (flattery), general conversation (non-task-related talk), task conversation (task-related talk), threatening (threats), offering (offers to provide something) and closing (conclusion and farewells). The politeness feature returns polite, neutral or impolite. These features from the classifiers are used to update four
state variables: respect, bonding, fear and a compliance level (calculated from the previous three). These features are used by the dialogue manager to pick the most suitable answer. Hassan is similar to this work, as it tries to understand several elements of the question and determines the level of compliance on the suspect’s part. These elements, in the case of this project, are related to the PEACE and Cognitive Interview. In addition, Hassan has four state variables whereas this virtual suspect has only two (empathy and pressure). For instance, if a suspect has a low pressure and high empathy he will most likely answer with the truth. In the case of Hassan, the latter would be equal to high variables of respect and bonding, and a low variable of fear.

3.5 FAtiMA

FAtiMA (Dias et al., 2014) is a framework used to create and control the behaviour of virtual characters. Figure 3.5 shows its architecture.

![Figure 3.5: FAtiMA’s agent architecture (Dias et al., 2014).](image)

It was used in several applications such as:

- **Traveller** A game about raising intercultural awareness².
- **SUECA** A social robot a card game in augmented reality³.
- **FearNot!** A serious game that teaches children strategies to prevent bullying and social exclusion.

A set of tools and assets have been designed for this agent architecture which is called FAtiMA Toolkit which offers the following assets:

- **Emotional Appraisal** Manages the beliefs and the emotional state of the character according to how it is configured to judge the events that happen in the game world.
- **Emotional Decision Making** Decides how the character acts taking into account its emotions and beliefs about the state of the world;
- **Social Importance Dynamics** Adds the ability for the character to judge if an action is socially appropriate or not depending on how it perceives others from a relational standpoint.

²http://ecute.eu/traveller/
³https://vimeo.com/153148841
• **Role Play Character** Integrates a combination of the previous three assets in a simplified perception-action cycle.

• **Integrated Authoring Tool** Manages the authoring of a scenario including its characters and respective dialogues.

• **Real Time Emotion Recognition** Able to infer the player’s emotional state by combining multiple sources of affective inputs.

This architecture makes use of the OCC-based (Ortony, Clore and Collins (1988)) cognitive appraisal theory (Elliott, 1992). This states that emotions develop as a consequence of certain cognitions and interpretations which implies that one’s behaviour has both reactive and predictive levels (Smith and Lazarus, 1990). The OCC theory describes a person’s emotions as the result of that person’s appraisal of how the current situation fits with the person’s goals and preferences (Conati and Zhou, 2002). In addition to emotions, FAtiMA also takes mood into account. Positive moods favour positive emotions and negative moods favour negative emotions (Figure 3.6). New emotions increase or decrease the mood if these are positive or negative respectively (Figure 3.7).

\[
\text{Potential}(em) = \begin{cases} 
\text{BasePotential}(em) + \text{Mood} \times k_1, & \text{if Valence}(em) > 0 \\
\text{BasePotential}(em) - \text{Mood} \times k_1, & \text{if Valence}(em) < 0 
\end{cases}
\]

**Figure 3.6:** How mood influences emotions.

\[
\text{Mood} = \begin{cases} 
\text{Mood} + \text{Potential}(em) \times k_2, & \text{if Valence}(em) > 0 \\
\text{Mood} - \text{Potential}(em) \times k_2, & \text{if Valence}(em) < 0 
\end{cases}
\]

**Figure 3.7:** How new emotions influence mood.

Furthermore, the intensity of emotions is influenced by emotional threshold. Emotions are only taken into account if certain thresholds are met. This is a way to model differences in personalities. Overall, the emotional state of an agent is comprised of emotions and mood. Mood is valued between -10 and 10, slowly decaying with time towards 0. Emotions have a type, a valence (positive/negative) and an intensity which decays with time.

The Emotional Decision Making component (Figure 3.5) is rule based. Figure 3.8 shows an example of a rule to emulate the psychological effect of mirroring body language when one likes the person he/she is talking to.

```
* (Action : Express[e]), Conditions: [IsAgent[s]] = True, Like(SELF, [x]) > 0, Facial-Expression[x][e] = [e]
```

**Figure 3.8:** How new emotions influence mood.

By testing all conditions (rules), an activation tree is created and the instances of actions that satisfy all conditions are returned. Actions are sorted by priority and, if concurrent, are shuffled randomly.

The player chooses which question to make by choosing from the ones presented. Each question has an entailed set of answers creating a tree of possible states. Emotional Decision Making provides
the next possible states but rules are imposed trimming them down. The calculations, intensities and thresholds behind each interaction and decision making were found empirically.

Another important aspect about FAtiMA VR Demo is that the emotions and their scales are based on a PAD emotional state model (see Section 2.6.2). This framework differs from the one used on this project (based on EDGAR) mainly because the user only picks a question from several displayed. By selecting a certain answer, the user is accessing different “trees” of questions and answers possible. FAtiMA’s also decision making takes into account an autobiographical memory which is not present on this project. Like FAtiMA, the values and calculations behind each interaction are made empirically.
Chapter 4

Building the Virtual Suspect

4.1 General Overview

In Figure 4.1 one can have a general overview of what is the application’s architecture. Firstly, we have the audio or text input. If audio, it is converted to text via AUDIMUS (see Section 4.2). The sentiment underlining the text is analyzed (see Section 4.3) and this will impact the suspect’s mood. This text input (the question) is then linked to a knowledge base. How this knowledge base is structured is explained in detail in Section 4.5.1. If there is a question which resembles what the user inputs, the possible answers from that question are sent to be picked from. If the application is not able to find resembling questions, it answers saying it did not understand what was asked (this answer is not always the same). From the list of possible answers the application picks the one more suitable according to its tags and the suspect’s current mood (see Section 4.6). If there is no input for a number of seconds it is considered as a silence and the suspect tries to add information to the last given answer.

![Figure 4.1: General Diagram of the application’s architecture.](image)

This is the basic structure of the application, not including the user interface. It is now necessary to describe each of the features individually. This is made in the following fashion:
• **Speech Recognition**: How the speech is recognized and translated to text.

• **Sentiment retrieval**: How the sentiment underlining the interviewer’s question are retrieved sentiment.

• **Knowledge base**: How the knowledge is structured, which tags are used and their scales.

• **Choosing an appropriate answer**: How the application chooses the most suitable answer given the current situation.

• **Interpersonal State**: Which are the possible moods the interviewee can experience and how these occur.

• **User Interface**: A brief description of how the user interface is built.

The application on which this is based - Edgar - simplified the implementation as it provides the linking between an input question and an existent question in the knowledge base.

### 4.2 Speech Recognition

In order to capture audio from the user a third party software called AUDIMUS\(^1\) is used. This software is specifically made for the European Portuguese. AUDIMUS combines the temporal modeling capabilities of Hidden Markov Models (HMMs) with the pattern discriminative classification capabilities of multilayer perceptrons (Meinedo et al., 2003b). During runtime, the user presses a push button to record voice in the form of binary data. After release, the recording is sent to an external server which sends back the recognised text. These interactions make use of the WWW class\(^2\) from Unity’s API to be sent and received.

### 4.3 Sentiment Retrieval

There are several ways to go with when it comes to sentiment retrieval: speech, visual and textual. These three ways complement one another and therefore should be used to increase each others certainty. Another aspect and addition to this kind of analysis is to perceive the level of emotion rather than just deciding which one it is on a binary way. This way the application decides which answer could take into account, for instance, how angry the interviewer is. This is obviously closer to reality for we human are not just either anger or not anger. Obligation and evidence retrieval are two features difficult to extract because of their contextual nature. In this project only text sentiment retrieval is used. However, at the end of this section, we discuss how visual and speech sentiment retrieval could be added.

For the decision of which answer is the most appropriate given a situation, the application needs some input on what is the sentiment behind each interaction by the police officer. To extract this meta-

---

data, a sentiment analysis tool was created. This tool uses WEKA libraries (Hall et al., 2009) in order to
train a model, more specifically using a Naive Bayes Classifier.

To create a model it was necessary a suitable data set. Three data sets were tested and are de-
scribed in Sections 4.3.1, 4.3.2 and 4.3.3. Each model was validated using a 10 fold cross validation.
This means that the data is split into 10 parts in which 9 are used to train the algorithm and the 1 is used
to evaluate the algorithm. This process is repeated, allowing each of the 10 parts of the split data set a
chance to be the held-out test set. The validation results are combined for an estimate of the model's
predictive performance which are shown for the different models on their respective tables. The model
used in this project was created using the data set present in Section 4.3.3. Furthermore, rapport is the
main guideline of the PEACE model and the Cognitive Interview which should be rewarded if used by
the interviewer. How rapport is detected is explained in section 4.4.

4.3.1 IMDB data set

The first attempted model was trained by making use of IMDB's\(^3\) review data set (Maas et al., 2011)
which contains 50,000 highly polar movie reviews (25,000 positive and 25,000 negative). Half was used
to train, create the model and the other half for testing. The results in Table 4.1 and 4.2 represent the
output of the model, tested with a thousand instances of the same data the model was trained with. Note
that the ideal situation would be to train the model with a large Portuguese data set of classified police
interview questions. Such does not exist.

<table>
<thead>
<tr>
<th>Model Evaluation - General Overview</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correctly Classified Instances</td>
</tr>
<tr>
<td>Incorrectly Classified Instances</td>
</tr>
<tr>
<td>Kappa statistic</td>
</tr>
<tr>
<td>Mean absolute error</td>
</tr>
<tr>
<td>Root mean squared error</td>
</tr>
<tr>
<td>Relative absolute error</td>
</tr>
<tr>
<td>Root relative squared error</td>
</tr>
<tr>
<td>Total Number of Instances</td>
</tr>
</tbody>
</table>

Table 4.1: Naive Bayes Classifier’s results for the IMDB's English data set.

<table>
<thead>
<tr>
<th>Class</th>
<th>TP Rate</th>
<th>FP Rate</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Measure</th>
<th>MCC</th>
<th>ROC Area</th>
<th>PRC Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive Reviews</td>
<td>0.812</td>
<td>0.184</td>
<td>0.815</td>
<td>0.812</td>
<td>0.814</td>
<td>0.626</td>
<td>0.888</td>
<td>0.880</td>
</tr>
<tr>
<td>Negative Reviews</td>
<td>0.816</td>
<td>0.186</td>
<td>0.813</td>
<td>0.816</td>
<td>0.814</td>
<td>0.522</td>
<td>0.888</td>
<td>0.885</td>
</tr>
<tr>
<td>Weighted Avg.</td>
<td>0.814</td>
<td>0.186</td>
<td>0.814</td>
<td>0.814</td>
<td>0.814</td>
<td>0.628</td>
<td>0.888</td>
<td>0.884</td>
</tr>
</tbody>
</table>

Table 4.2: A detailed accuracy by class of the model tested with training data.

The obvious barrier here is the fact that the model was trained with a large data set in English rather
than in Portuguese. A data set containing reviews in Portuguese perfectly labeled is rare and the quest
to find one rather troublesome. An idea came to mind: using google translation tools to translate this
IMDB’s review data set from English to Portuguese. As the new Google Translate API\(^4\) is now a paid

---

\(^3\)http://www.imdb.com/

\(^4\)https://cloud.google.com/translate/docs/
The translation is very accurate for an automated one. Despite this, there are typo mistakes in the reviews, which are obviously not translated such as the word “priivate” found on the third sample. Moreover, some words are being translated to Brazilian Portuguese rather than to Portuguese such as in the first sample “silly dialogue” being translated to “diálogo bobo” which is typical Brazilian Portuguese. The translation was made to Portuguese from Portugal but it seems the translator fails to some extent on this matter. To approximate the data set to what is needed, a set of positive and negative adjectives, nouns, sentences and verbs (including swear words) was added. These were mainly picked from an open compilation of Portuguese slang (Almeida, 2014). Note that many of these words can be used in the exact opposite way. For instance, if the police officer uses irony the polarity of the words is kept

34
but the polarity of the sentence changes. The best aspect about adding this is that we can indirectly reward the use of words and sentences such as "reasonable", "I would probably do the same" which use is encouraged to create rapport. The table containing all these words is in appendix D.

The model with Portuguese data and added instances was then trained which yielded the results shown in Tables 4.4 and 4.7.

Table 4.5: A detailed accuracy by class of the model tested with training data.

<table>
<thead>
<tr>
<th>Class</th>
<th>TP Rate</th>
<th>FP Rate</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Measure</th>
<th>MCC</th>
<th>ROC Area</th>
<th>PRCArea</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive Reviews</td>
<td>0.793</td>
<td>0.206</td>
<td>0.793</td>
<td>0.793</td>
<td>0.793</td>
<td>0.587</td>
<td>0.873</td>
<td>0.871</td>
</tr>
<tr>
<td>Negative Reviews</td>
<td>0.794</td>
<td>0.207</td>
<td>0.794</td>
<td>0.794</td>
<td>0.794</td>
<td>0.587</td>
<td>0.873</td>
<td>0.871</td>
</tr>
</tbody>
</table>

Weighed Avg.
<table>
<thead>
<tr>
<th>Class</th>
<th>TP Rate</th>
<th>FP Rate</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Measure</th>
<th>MCC</th>
<th>ROC Area</th>
<th>PRCArea</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive Reviews</td>
<td>0.793</td>
<td>0.207</td>
<td>0.793</td>
<td>0.794</td>
<td>0.794</td>
<td>0.587</td>
<td>0.873</td>
<td>0.871</td>
</tr>
<tr>
<td>Negative Reviews</td>
<td>0.794</td>
<td>0.207</td>
<td>0.794</td>
<td>0.794</td>
<td>0.794</td>
<td>0.587</td>
<td>0.873</td>
<td>0.871</td>
</tr>
</tbody>
</table>

Table 4.4: Naive Bayes Classifier's results for IMDB's Portuguese translated data set.

The 79.3% is an acceptable result but it just proves the model works well for the data it was trained with. In order to understand if this model would be suitable for an interview type of question, it was tested with around 15 questions that a police officer is likely to ask rather than data from movie reviews. This is shown in Table 4.6.

Table 4.6: Testing the Naive Bayes classifier trained with the IMDB's translated data set with possible data from an interview.

<table>
<thead>
<tr>
<th>Text instances</th>
<th>Result</th>
<th>Appropriate Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>vais falar ou ficas calado</td>
<td>-0.63</td>
<td>Negative</td>
</tr>
<tr>
<td>fala ca***o</td>
<td>-0.69</td>
<td>Negative</td>
</tr>
<tr>
<td>provavelmente teria feito o mesmo</td>
<td>-0.71</td>
<td>Positive</td>
</tr>
<tr>
<td>o que você fez é completamente normal</td>
<td>-0.85</td>
<td>Positive</td>
</tr>
</tbody>
</table>

Table 4.7: A detailed accuracy by class of the model tested with training data.

The outcome of the class is made to be in the interval [-1,1] as shown in Figure 4.2. The $x$ value is a threshold for the analysis to have an impact at all. For instance consider the model has an high accuracy even with low values of confidence like 0.64 then it should impact the decision making. The threshold $x$ is picked by trial and error until a value is found that impacts mostly when the decision is right. Despite all of this, the model fails by identifying positive instances as highly negative (Table 4.6). This suggests the IMDB’s data set was not good to train the model.

A probable better alternative to IMDB’s data set would be to train the model resorting to Twitter messages (Tweets), as Tweets are more related to what a police officer might say. This because tweets resemble more conversational acts than film reviews, not only on their size but also on their nature.

---

6https://www.twitter.com
Film reviews are sometimes too complex and deep, unlike the short simple sentences in interviews and tweets.

### 4.3.2 sentiCorpus-PT data set

A second model was created resorting to a much different data set called *sentiCorpus-PT* (Carvalho et al., 2011). This is a compilation of 2,625 manually comments posted by the readers on the website of the Portuguese newspaper *Público*. The comments were made regarding a series of 10 news articles covering the TV debates on the 2009 election of the Portuguese Parliament. The data set is closer to what one would hear in a conversation and therefore better appropriated for police interviews. The results for the same type of evaluation as before are shown in Table 4.7.

<table>
<thead>
<tr>
<th>Model Evaluation - General Overview</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correctly Classified Instances</td>
</tr>
<tr>
<td>Incorrectly Classified Instances</td>
</tr>
<tr>
<td>Kappa statistic</td>
</tr>
<tr>
<td>Mean absolute error</td>
</tr>
<tr>
<td>Root mean squared error</td>
</tr>
<tr>
<td>Relative absolute error</td>
</tr>
<tr>
<td>Root relative squared error</td>
</tr>
<tr>
<td>Total Number of Instances</td>
</tr>
</tbody>
</table>

Table 4.7: Naive Bayes Classifier’s results for sentiCorpus-PT data set.

These results are not too good or too bad but they do not reflect what is wanted.

### 4.3.3 sentiCorpus-PT edited data set

The third model, which is used by the application, was trained using an edited version of the data set described in Section 4.3.2. There was a need to add, change and erase many instances. Some instances in the data set were either repeated, added nothing or made no sense altogether. Therefore, the whole data set was filtered manually for sentences such as “Á Sócrates!”, “Está decidido, votarei PS”, Spanish instances, personal names, shortened words (“pff” meaning “por favor”). Another problem regarding this data set is the amount of negative instances relative to the positive ones. An attempt was made to have the latter closer to a 50%-50% ratio. Overall, the data set was trimmed to roughly a third of its original 2,625 comments and around 100 new instances were added. The adding, erasing and changing of instances are all made so that the model correctly tags sentences according to the psychology behind the police interviews. For instance, sentences such as “would you care to explain” or “ask me anything you would like” must be positively tagged whereas sentences such as “you don’t
need to say anything else” (going against the idea of letting the suspect talk as much as he can) should be negatively tagged. Verbs such as “diga” (say), “falar” (talk), “explicar” (explain) are highly positive whereas leading questions are highly negative. Several examples of these were added. Moreover, in attempting to enrich the data set several instances were taken out of a Portuguese dictionary of slang (Almeida, 2017). These added instances were carefully chosen since the use of slang is, at times, useful to develop rapport with the suspect. Note that by adding these instances means this sentiment analysis serves a deeper purpose than telling the sentiment behind the sentence: it actually aims at telling whether the sentence is good or bad according to the Cognitive Interview and the PEACE Model. The filtered data set yielded the results shown in Table 4.8.

<table>
<thead>
<tr>
<th>Model Evaluation - General Overview</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correctly Classified Instances</td>
</tr>
<tr>
<td>Incorrectly Classified Instances</td>
</tr>
<tr>
<td>Kappa statistic</td>
</tr>
<tr>
<td>Mean absolute error</td>
</tr>
<tr>
<td>Root mean squared error</td>
</tr>
<tr>
<td>Relative absolute error</td>
</tr>
<tr>
<td>Root relative squared error</td>
</tr>
<tr>
<td>Total Number of Instances</td>
</tr>
</tbody>
</table>

Table 4.8: Naive Bayes Classifier’s results for sentiCorpus-PT edited data set.

Like all the models, there is a need to test whether it would or not rightfully rate common police interview questions. This is shown in Table 4.9

<table>
<thead>
<tr>
<th>Text instances</th>
<th>Result</th>
<th>Appropriate Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>vais falar ou ficas calado</td>
<td>-0.91</td>
<td>Negative</td>
</tr>
<tr>
<td>fala idiota</td>
<td>-0.93</td>
<td>Negative</td>
</tr>
<tr>
<td>provavelmente teria feito o mesmo</td>
<td>0.95</td>
<td>Positive</td>
</tr>
<tr>
<td>o que você fez é completamente normal</td>
<td>0.99</td>
<td>Positive</td>
</tr>
</tbody>
</table>

Table 4.9: Testing the Naive Bayes classifier trained with the sentiCorpus-PT edited data set with possible data from an interview.

The results surely seem great but a lot more instances should be added to the data from which the model is created thus having a more robust decision maker. This is such a strenuous task but such an important one that something was needed to tackle it. A new feature was added to make the system better with the more use it gets. One can type a sentence and include one of two special tags: _pos_, for positive instance and _neg_, for negative instance. By doing so, one is adding knowledge to the system which might have had wrongly rated a certain instance. After a predetermined number of tagged instances added the system retrains the model once it restarts. The default value was set to retrain every 20 new instances. Choosing the best classification threshold is a problem of maximizing the F1-score (harmonic average of the model’s precision and recall). However, in the sake of this project, false positives and false negatives have a far greater impact than getting the classification correctly. Thus, the classification threshold is set to an inflated value of 0.9. This means that only instances classified between [-1;-0.9] and [0.9;1] are considered. Another reason to set the threshold high is that the model is remade every 20 new instances as mentioned above. This means this value needed to be changed
everytime a new model comes into place.

4.4 Rapport detection

Rapport is a powerful tool in an interview and its establishment ought to be tried. Rapport attempts are troublesome to detect. However, one easy way to detect it partially is by analyzing the sharing of similar words in a conversation by both the subjects (see Section 2.3). A tool was created to test the similarity between what the user types and the suspect’s history. Every time there is an input, words are compared to the history of answers by the suspect. Words such as pronouns and determiners were discarded for these are used for language purposes rather than rapport itself. The set of discarded words is given in the table 4.10.

| ele, elas, ela, eles, vós, nós, tu, eu, de, das, dos, da, do, dele, deles, dela, delas, nele, nela, neles, nelas, meu, meus, minha, minhas, tua, teu, teus, tuas, nosso, nossa, nossos, nossas, vosso, vossos, vossa, vossas, seu, sua, seus, suas, esta, este, estas, estes, esse, essas, esses, essa, que, se, ao, à, a, os, o, é, são, e, ser, na, nás, no, nos, te, aquele, aqueles, aquela, aquelas, te, isto, isso, aquilo, em, um, uns, umas, por, para, não, sim, talvez, ok, com, como, menos, mais, mas, ou, já, até, vez, sem, sobre, lhe, lhes, lá, tão, neg, pos. |

Table 4.10: List of discarded words.

This is used every time there is an input by the user. Because rapport was detected, the interviewer is rewarded. This is further explained in Section 4.7.

4.5 Knowledge Base

4.5.1 Knowledge base structure

In Section 2.2 we have seen EDGAR’s knowledge structure. Each set of question-answer possesses a unique identifier number (ID). It also has various similar instances of the same questions with the corresponding translations and various answers with emotional tags to tell the visual representation which emotions to display.

The new knowledge base sits in distinctive XML documents. These are organized in the following way:

- **Small-talk**: This has sets of questions that apply to every suspect, independent of the crime or the suspect. For instance, a question such as “How are you?”.

- **Suspect’s personal information**: This XML has, like the name suggests, information regarding the suspect himself. Examples of questions are “How old are you?” or “What is your relationship towards...?”. 
• **Information about the crime:** All the information regarding the crime is put onto this document. Questions such as “Where were you when the crime happened?” or “Did you notice anything strange on that day?”.

• **Miscellaneous:** As the name suggests, all the questions that, for any reason, have no place otherwise, are put here.

Figure 4.3 shows how the data is structured for each set of question and answer.

![XML example](image)

**Figure 4.3: Question-Answer example in XML.**

The new XML keeps the ID to prevent duplicates and to further allow direct access from the application to that specific set. The XML tags <q> represent the set of questions. The multiple questions in Figure 4.3 are basically many different ways of asking the same thing. The tags <a> represent the set of answers. The application picks one of these as the answer from the interviewee. Each element may have the following added attributes:

- **ID:** Uniquely identifies the set of questions/answers which was already on Edgar.

- **correctTiming:** This states on which period of the interview should this question be asked. Its value ranges from 0 to 5.

- **importantInfo:** This tag states how relevant is the question. Its value can be -1, 0 or 1.

- **empathyRequired:** This represents the empathy required for the use of that answer. Its value ranges from 0 to 100.

- **truth:** This represents whether the answer is a lie (with value -1), truth (with value 1) or neither (with value 0).

- **silence:** Indicates that the new answer should be a continuation of a previously answered question on the same set. Its value can be either 0 or 1.

- **repeat:** If the question has been repeated more than once, the application looks to pick these repeats. Its value is either 0 or 1.

Note there is no need for an answer element to have all attributes. The attributes emotion and intensity were taken out. This because the emotion displayed by the interviewee no longer comes from the answer but rather from the calculated mood (see Section 4.7).

Figure 4.4 shows an example of a complete question/answer set.

On the above example one can understand that the way these sets of answers are built comply more or less with the following schema:
1. A cooperative, friendly and extensive answer which requires full empathy.

2. A not so cooperative yet polite answer requiring half empathy.

3. A very closed and uncooperative answer with no empathy required at all.

4. A complementary answer to the answers above which serves to complement any of them, if a pause from the inspector is detected.

This provides consistency and the assures that the application has a suitable answer for the possible states of the interaction.

4.5.2 Tags

An important aspect of the new features in Edgar’s structure is the knowledge-base tagging. Research was put into this with the endeavour of which theories and concepts better endorse our reality. Listening to police enacted interviews as well as police guidelines and other theories brought to surface several important aspects to take into account for the tagging. The interview starts with an attempt to establish rapport/empathy (app).

According to (Tickle-Degnen and Rosenthal, 1990):

- The tone and rhythm of voice applied by the police must be as close to the suspect as possible;
- The questions should be simple to answer usually about the suspect’s life without association to the crime whatsoever.

As the interview proceeds, like an inverted triangle, the questions are more open and finally closed for possible confrontation (Communication Process and Police Interrogation Technique - PEACE, Policia Judiciária). Four different tags were taken into account according to above.

- **Empathy Required** is the primary tag (Tickle-Degnen and Rosenthal, 1990). As mentioned in “Application Scenario Outlines”7, the PEACE model is based on rapport and empathy. This tag can be measured by the similarity with the victim’s communication style and the general way the interviewee asks the question (for instance in a pleasant tone of voice or otherwise in an aggressive way). In this system, through the interactions, an interpersonal state is developed which entails a value for empathy. According to that empathy/rapport the interviewee has to pick an answer. This is when empathyRequired comes into place. For example, the police officer asks “Há quanto tempo

---

7 Application Scenario Outlines: EPJ-MJ use case “Interview Skills for Police Officers (ISPO)” from Escola de Polícia Judiciária
trabalha para Dona Maria?” (“How long have you worked for Miss Maria?”). If the suspect carries a great deal of rapport with the police officer, the application picks the answer with the greatest empathyRequired. Usually this answer is a more cooperative and extensive one. Conversely, if the empathy on the conversation is low means the application will pick a low empathyRequired answer which usually translates into a short, bad humour response. See, infra, the values and scaling discussed.

- **Important information** tag is a way to understand how much evidence and important information is being asked at a certain point. The virtual agent needs to understand the importance of the question and how much is he being exposed by answering the question (if guilty). Moreover the enacted interviews provided by the Portuguese police have revealed that the suspect is usually angry when asked about irrelevant questions. This is also highly correlated with how aggressive a question is in terms of information. If the tag has a high value of Important Information it means the question is trying to retrieve information is really crucial and possibly incriminating. For instance, an high tag question would be something like “O senhor estava com Maria quando morreu?” (“Were you with Maria when she died?”). A low Important Information tag is, for instance, “Como está?” (“How are you doing?”).

- **Correct timing** should be included because, as said previously, the type of questions should be asked at the right timing of the interview. For instance open questions are usually best for information retrieval in the middle of the interview, after rapport has been established, whereas closed questions are used at the very end to confront the suspect with facts or things which made no sense. Thus, a correct timing tag is needed to detect if the question is used in the appropriate phase of the interview. The interview can be divided into four major parts: rapport establishment, allow recovery/free narrative, questioning and confrontation/closure. Each question on the database has one or more parts it should be in. This will affect the interpersonal state of the suspect and therefore the course of the interview (Application Scenario Outlines: EPJ-MJ use case “Interview Skills for Police Officers, Victims and offenders of violent crimes”, ISPO). Of course, to make use of this tag, there must be a way to detect on which stage of the interview one is in, which is very difficult. However, what is taken into account is the order of the tags. In other words, if the order of the questions’ correct timing tag is maintained, the application increments the interviewee’s empathy whereas if the order is violated it takes out empathy. For instance, if the application receives a stage one question followed by another stage one or higher it means the order is being maintained. If the application receives a stage two question followed by a stage one question it means the order is being violated and therefore decrements empathy. The way the rewarding and calculations occur are discussed in Section 4.6.

- **Truth/lie/omission** should be included in the answer’s tagging. This allows for the application to choose a lie or omit information. For instance if the interviewee is under a great deal of pressure and the empathy levels are not sufficient for the information to be given. In this case the application will pick an answer with the omission or lie tags. The amount of times the subject omits or lies
will have a linear impact on the **pressure**. It is known that the more a person lies the tougher it gets to think about all the lie structure rather than just remembering truths from memory. This was discussed in a seminar held by Escola da Polícia Judiciária (see Section ??). It will get more difficult to continue lying and omitting if the **pressure** is too high.

- **Silence** tag is also an important feature. These are used methodically by the police in order to force the suspect to talk. If the virtual agent does not detect voice for a specific number of seconds it should be perceived as a silence and therefore be engaged on a broader answer. This is a variable which enables a longer response or multiple responses altogether. To tackle the silence issue, a **silence** tag was added. This gives the possibility of having an answer in the XML Knowledge Base which is to be used in case there is a silence from the user.

More tags can be added at the cost of revisiting every question of the corpus and adding its impact on the application’s answer decision.

### 4.5.3 Tag Scales

In this section, we discuss what values should the tags discussed in Section 4.5.2 take. Since each tag differs on its nature, let us discuss one by one:

- **Empathy Required** makes use of a five point scale (0, 25, 50, 75 and 100). Previous studies by Ter Maat (ter Maat and Heylen, 2009) used the following tags: unfriendly-friendly, distant-pleasant, passive-active, cold-warm, negligent-conscientious, disagreeable-agreeable, rude-respectful, unpredictable-stable, unattentive-attentive, submissive-dominant, undependable-responsible, negative-positive, not aroused-aroused. These were evaluated on five point scales. A five point scale is especially useful for it allows us to have a neutral point (0), two extreme points (-1, 1) and two mild yet biased points (-0.5, 0.5). The scale is shifted towards 0-100 because it means little to have a negative **empathy** required, but rather no **empathy** required whatsoever. That being said, 100 means the answer needs full **empathy** from the suspect for him to answer (usually high valuable, extensive answers). On the other hand, 0 means the answer needs no empathy at all (for instance angry answers). The remaining values are the so called mild points.

- **Important information** can take values of -1, 0, and 1. These respectively mean that the information is really out of the scope of the interview, not important or important.

  Figure 4.5 shows questions the three possible values and how they differ.

- **Correct timing** has its values range from 0 to 4. These values represent the four different stages of the Cognitive Interview method plus the value of 0, which indicates that the question is not to be made whatsoever and incurs in punishment on the **empathy** of the interviewee. For instance, a question to establish rapport such as “Está tudo bem consigo?” (“Is everything alright with you?”) has the **Correct timing** tag set to 1. A question which allows recovery and free narrative such as “Sabe o que faz aqui hoje?” (“Do you know what you are here for today?”) has **Correct timing** tag
Figure 4.5: Important information tag examples for each value.

set to 2. An open topic question which invites the suspect to open its perceptive scope such as “O que pensa que se passou?” (“What do you think happened?”) has Correct timing tag set to 3. A closing statement with, for instance, positive reinforcement, such as “Fez o certo em nos ajudar neste caso. Obrigado” (“You did right helping us in this case. Thank you.”) has Correct timing tag set to 4. Finally, a question that should never be asked such as “Foi ela que a matou não foi?” (“She was the one who killed her right?”), which influences the suspect, has the tag set to 0.

• Truth/lie/omission can take values of -1, 0, and 1. These values mean respectively that the answer is a lie, omission or truth.

• Silence is a binary tag which, therefore, can take values of 0 or 1. 0 means it is not a silence and 1 otherwise. If set to 1 the answer is to be used to fill silences from the police officer.

4.6 Choosing an Appropriate Answer

In Section 4.7 it is discussed how the tags from answers and questions (Subsection 4.5.2) interact with the interpersonal state and makes it evolve. Here, we describe how the answers are chosen based on the interpersonal state. The module responsible for this matter is called Decision Core. Notice that the Interpersonal state both depends on the decisions made (for instance, which answers are chosen) and impacts the decision. The Decision Core is intrinsically connected with the Interpersonal State.

As shown in Section 4.5.2, each question has a empathyRequired tag, which represents, as the name suggests, the minimum empathy required for that answer to be chosen. The second tag taken into account in the decision making is the tag truth, which tells whether the answer is a lie, omission or truth. Lastly, the third tag used is the tag silence. To come up with a single answer, the application ensues the following steps in order:

• Silence: In the front end (Unity), when the interviewer passes a number of seconds without asking a new question Unity automatically asks the application with “.SILENCE..” indicating there was a silence/pause. The application keeps the first answer with silence tag set to 1 from the previous interaction. Upon receiving “.SILENCE..”, the application throws the answer tagged with a silence=1 from the last interaction (it is cleaned every interaction and updated according to the current interaction as it only makes sense that “.SILENCES..” makes the suspect talk solely about the last interaction).

For sake of an example, consider the question/answer set shown in Figure 4.6.
Figure 4.6: Test set to show the use of "SILENCE_".

1) Example Question 1
A: Example Answer 2

2) Example Question 2
A: Example Answer 2

3) Example Question 3
A: Example Answer 3

Figure 4.7: Use of "SILENCE_" as a "question".

Only “Example Answer 3” has silence=1. In Figure 4.7 is shown: Asking one of the questions from the test set in Figure 4.6 (1) to which the application chooses to answer with the set’s second answer (2). (3) shows the answer to “SILENCE_” which, as one can see in the test set, is the only answer with silence=1.

Figure 4.8: Snippet taken out of the corpora showing the answers used to answer “SILENCE_”s when the application has none from the previous interaction.

In case it does not find any, meaning the previous interaction had no silence, the application picks randomly from a set of answers shown in Figure 4.8. Figure 4.9 shows the answer to “SILENCE_” on the first interaction of the application altogether (i.e. the application has no silence tagged answer).

- **Empathy Required:** If not a silence, the application gets all the answers and picks the one with the maximum empathy required which is still less than the current Interpersonal State’s empathy. In case no answer’s empathyRequired satisfies the previous condition, the system picks the answer with the least empathyRequired even if above the current empathy. Note that usually there is always an answer with empathyRequired=0 and therefore satisfying any empathy possible state.

Figure 4.10 show a question/answer set which was tested for various values of Interpersonal State’s
Figure 4.9: Suspect's answer to "_SILENCE_" when the application has none no silence tagged stored from previous interactions.

Figure 4.10: Test set to how the application chooses by empathyRequired.

empathy. With empathyRequired=100 the application picks the first answer as it has the required empathy. With empathyRequired=74 the application picks the third answer as it does not meet the empathy required to answer the first or second answers. Finally, with empathyRequired=4 the application picks the forth answer because it has the lowest empathy required of all, excluding the one with silence=1.

4.7 Interpersonal State

This is a module which contains general information about the virtual suspect's current mood. This is being calculated during the interview. In order to make the character closer to reality it must have a personality. Some people are edgier, some are less prone to stress. The closer one wishes his virtual suspect to be closer to reality, the more variables one must define. Ideally, a PAD state model or Five Factor model (see Section 2.6) should be used, but it incurs in unnecessary complexity which eventually drives the project out of its scope. A much simpler approach was made. This is the current emotional state of the suspect which varies dynamically during the interview:

- **Empathy**: current value of the suspect's empathy;
- **Pressure**: current value of the suspect's pressure;
- **Expression**: what is the suspect's body language suppose to be (e.g. happy);

Both empathy and pressure are on a scale from 0 to 100 for the same reasons explained in Section 4.5.3, regarding the scale of Empathy Required. The player is given the opportunity to defined it before the conversation takes place or it is set to default. This is made by the tagging a question with _set_ followed by a value for empathy and a value for pressure as showed in Figure 4.11.

Note that there are small nuances for which the application must not classify the question or call strategies to answer it. For instance, in the case of a received _set_ command, the application should only set the interviewee's empathy and pressure levels. Classifying or calling strategies to answer a _set_ command makes no sense.
The Expression is calculated by taking into account empathy and pressure, which is the reason it is not possible to be set beforehand. This is a value only useful to Unity, so it knows exactly how to represent the suspect’s current mood. In the front end, the suspect is prepared to enact being **angry**, **sad**, **relaxed**, **nervous** or **happy** as shown in Appendix B.2.2. These are the possible values of the interpersonal state Expression. Figure 4.12 shows how each expression depends on pressure and empathy.

Representing mathematically the following conditions were applied:

- **Relaxed**: If \(70 \geq \text{empathy} \geq 30\) && \(70 \geq \text{pressure} \geq 30\)
- **Happy**: If \(\text{empathy} \geq 50\) && \(\text{pressure} \leq 50\)
- **Sad**: If \(\text{empathy} < 50\) && \(\text{pressure} \leq 50\)
- **Nervous**: If \(\text{empathy} \geq 50\) && \(\text{pressure} > 50\)
- **Angry**: If \(\text{empathy} < 50\) && \(\text{pressure} > 50\)

To avoid points where two or more states are possible, boundaries are attributed for different states as shown above by the equals sign (with no reason in particular).

On one hand, there are the negative emotions of sadness (**Sad**) and angeriness (**Angry**). These are negative which means it makes sense to match empathy below half (50/100). Furthermore, being angry requires pressure whereas being sad usually only entails lack of empathy. On the other hand, there are the positive emotions of happiness and nervousness. These are positive which is the reason they are matched above half of empathy. If the suspect has a great deal of empathy and feels no pressure he is in a state of happiness. On the other hand, if empathy is still great but pressure is high, the suspect is nervous (usually a submissive stance). As shown in Figure 4.12, if empathy and
pressure are of no extremes, the default state of Relaxed kicks in. This also assures that there are no sudden jumps between moods which in reality are always a gradient. For instance, a suspect can be angry and as pressure lowers and empathy gets higher, he gets relaxed and only then happy. The values that make the suspect enter the Relaxed state are as shown in Figure 4.12. Figures 4.13 and 4.14 show, with two examples, the correct setting of expression through the use of test empathy and pressure values.

Figure 4.13: Application selecting expression Sad for values empathy=1 and pressure=2.

Figure 4.14: Application selecting expression Relaxed for values empathy=53 and pressure=34.

Only using two values (empathy and pressure) to represent the personality is quite simplistic. Notwithstanding, they tackle the main objectives. Empathy represents rapport and pressure represents the subjects amount of brain activity. This goes along with the psychology behind Police interviewing: create rapport, let the suspect speak as much as possible and let him exhaust his brain capabilities (for instance by lying) which eventually leads to the production of controversial ideas and facts. Together with the tags from Section 4.5.2 they make sense. Let us now move onto how the tags from answers and questions (Subsection 4.5.2) interact with the interpersonal state.

If the suspect picks an answer with the tag truth set to -1 or 0 it means the suspect has chosen to lie or omit respectively. This incurs in greater pressure on the interpersonal state. Specifically, lying (value of -1) increases pressure in 10 points whereas omitting increases in 5 points. An example of the application acting on a truth=-1 tag is shown in Figure 4.15. Telling the truth is debatable as to whether simply doing nothing (value of 0) or rewarding by subtracting pressure. On one hand, telling the truth psychologically diminishes the suspects thinking effort of making things up because it acts as a solid pillar of facts to localize oneself and construct a new, more solid, web of lies. On the other hand, telling the truth at a certain point of the interview does not eradicate the thought process necessary to keep up all the omissions and lies been said so far. The introduction of truths into a set of lies might even provoke more effort of thinking as lies and truths are difficult to be tied together. For this very last reason, telling the truth is set not to increase or decrease pressure.

Figure 4.15: Snippet taken out of the applications console via system prints. This shows pressure increasing by 10 after an instance of truth=-1.

A question with importantInfo valued with 0 means this is a question more intended to make the subject talk and create rapport. No crucial information is being asked and the suspect is at ease,
relaxing. These reward the suspect’s interpersonal empathy by 5 points. Conversely, if importantInfo is valued with -1, meaning the questions is completely out of scope, the suspect’s interpersonal empathy drops by 5 points. Lastly, if importantInfo is valued with 1 it means the questions has important, delicate information that might incriminate the suspect. These increase the suspects interpersonal pressure by 5 points. An example of a question tagged with importantInfo=-1 is shown in Figure 4.16.

![Figure 4.16: Snippet taken out of the applications console via system prints. It shows a question with tag importantInfo=-1 and its impact on the interpersonal state.](image)

In Section 4.3, it is discussed the method to retrieve if a question is either positive or negative within an interviews scope. If a question is classified as positive, the suspect’s interpersonal empathy increases by 5 points. On the other hand, if the question is negatively classified, the suspect’s empathy drops by 5 points. In case the classification is below the classification threshold, there is no impact at all. An example of a classification under the threshold is shown in Figure 4.17

![Figure 4.17: Snippet taken out of the applications console via system prints. It shows an instance which was classified under the threshold and therefore had no impact on the interpersonal state.](image)

If the order of questions, regarding the tag correctTiming, is violated, the suspect’s empathy drops by 5 points. For instance, if the suspect is presented with a question with correctTiming valued at 2 followed by questions with correctTiming valued at 4, the suspect’s empathy drops. Conversely if the correct order is respected, the suspect’s empathy is rewarded with 5 points. As explained in Section 4.5.2, a great use of this tag is if the value is zero it means the question should never be made in the first place. If a correctTiming valued 0 is read, the suspect’s empathy drops by 10 points. An example of the application acting on a _truth_=1 tag is shown in Figure 4.18.

![Figure 4.18: Snippet taken out of the applications console via system prints. It shows two questions in the incorrect order. Note that the empathy stays the same as it also gains 5 points from importantInfo.](image)

Lastly, a method was created which slightly detects the attempt of rapport creation by using the same vocabulary as the suspect. The method returns the amount of repeated words and is more thoroughly

48
described in 4.3. To reward the attempt of rapport, 1 point of empathy is given for each word repeated detected. Figure 4.19 shows the output of the application when the interviewer asks what was answered by the suspect in a previous interaction. Using the exact same words is just to prove the good functioning of the detection.

Figure 4.19: Amount of words repeated detected by questioning the suspect with his previous answers. Both answer and question are highlighted in yellow. Three words are detected (“Estava”, “Maria” and “quarto”) because it discards the pronouns and determiners.

Note that it searches the whole conversation and not just the last interaction.

Bear in mind that the values are found empirically based on sense. These might need fine tuning. More important than their values is the differences between each other. For instance, inverting the order of a question (regarding correctTiming) should not be as impactful on the suspect's interpersonal state as, let's say, lying.

4.8 User Interface

The interface was developed within the cross-platform Unity3D Editor. This tool allowed to develop and manage objects in hierarchy (Figure A.1) which are controlled via C# scripts. The whole application starts with a main menu following an options menu and finally the game mode the user chooses. The latter are designated as “scenes” and are separately described in appendix A.

Three different modes were built:

Speech + Text Mode is the simplest mode. This mode is a basic text chat with the suspect with display of its current state of its personality and time elapsed. It enables the user to tell the system which questions are positive or negative if the Teaching Mode is on.

Video + Speech + Text Mode which is, as the name implies, the addition of video and speech to the previous mode. The camera is locked which means the user cannot look elsewhere. The user can choose to speak or type the questions. The feedback panel is brought up on the PC's display.

Virtual Reality Mode, unlike the previous mode, has no lock on the camera, only on the movement of the player (the camera is sitting on the chair). There are several differences on the set to make it more reality like, as this mode is more intended for the use of goggles (VR). In this mode the user cannot input text, only speech.

8www.unity3d.com
4.9 Character Animations

The character body is named Winston and was imported from Unity’s Store\(^9\) which was developed by Reallusion using their Character Creator 2 software\(^{10}\). Winston was an ideal character to work with due to its strong appearance, highly detailed textures but ultimately because a fully rigged body eases the animation process. Despite being free for Unity, it was used with Reallusion’s permission. Overall, around 40 different facial expressions were created for Winston. How these expressions were created is

\(^9\)https://www.assetstore.unity3d.com/en/#!/content/50694
\(^{10}\)https://www.reallusion.com/character-creator/
explained in more detail in the Appendix B. The expression are divided into 5 different groups: relaxed, happy, sad, nervous and angry. Each group is an overlay of different animations. Another aspect to the animations is the lip-sync which is made by creating a viseme to each phoneme that comes out of the text to speech engine on the server side. The head and eye movement are all randomized with certain constraints. The processes of creating, importing and controlling the animations as well as lip-sync are detailed in appendix B.

Figure 4.23: The suspect.
Chapter 5

Evaluation

The evaluation of the whole system is made in two experiments (Section 5.2.1 and 5.2.2). In experiment 1 it was analyzed if values are changing according to the expected, whereas in experiment 2 the results are discussed at a more fundamental level (i.e. if the values and the suspect's mood swings make sense according to the real world). Note that there was no evaluation with real users mainly due to linguistic variation and the sheer number of possible questions. In order to have a more realistic conversation there had to be many types of questions in the knowledge base multiplied by the many different ways one can say each one of these questions. A more real conversation would be tangible with a knowledge base in the thousands of sets of questions and answers. This would require massive amounts of tagged data driving the thesis out of its scope. Consequently, to evaluate the application, especially in a more fundamental level, it is necessary to come up with a scenario with a simpler knowledge base. This is described in the following Section 5.1.

5.1 Knowledge Enrichment

As aforementioned, to evaluate the application it is first necessary to create a story to be behind the interview. Most of the enacted interviews provided from the police are domestic violence related. A domestic violence story brings two problems:

- It would not be suitable for the underage for it brings upon a lot of sexual references.

- Domestic violence are a very closed topic. This is rather noticeable in Escola da Policia Judiciária's enacted interviews as these showed that the questions and answers did not fall far from each other. Overall is not a rich theme.

For these reasons, a different theme was chosen. A murderer crime story. Here is a short description:

* Maria was a rich widow living in a fancy villa. She lived with a spoiled 40 year son, the suspect being interviewed who played the roll of butler and a maid in her twenties. Police investigations understand that Maria died in her, room poisoned. Only the butler truly knows the full story. In fact, her maid was Maria’s daughter from an old affair for which she kept her working in the villa. Maria’s son is involved*
with the maid. The maid tries to poison Maria because she wants to marry her son and inherit part of the heritage. The butler finds out that the two are secretly involved and tells Maria who, by knowing the incest, dies with an heart attack on the spot and not from the poison.

As an example, one of the question/answer sets with its respective tags is shown in Figure 5.1.

![Figure 5.1: Example of one question/answer set in the knowledge base.](image)

The knowledge base was built following the structure (Section 4.5.1) and tags. The end result is a knowledge base which consists of around 30 sets of questions answers, performing a total of around 70 questions and 80 answers. In theory, the larger the knowledge base is, the better answer is given by the suspect. Such is not possible for it in a lot of time which is out of this thesis scope. Building knowledge is a tenuous work especially since the tags need to be carefully chosen according to the PEACE model and Cognitive Interview.

### 5.2 Verification

#### 5.2.1 Experiment 1

In this section, verification and validation are sought as far as the technical aspect of each component is concerned.

Firstly, setting both pressure and empathy is showed in Figure 5.2. Note that in case the user sets invalid pressure or empathy, the application picks the lowest or highest limit. This is showed in Figure 5.3.

![Figure 5.2: Setting suspect’s pressure and empathy.](image)

Regarding the suspects mood, there are five possible moods as shown in Figure 4.12.

The truth's tag can have impacts of 10 or 5 on the suspect's pressure for values of -1 or respectively. Tests are made on the question from corpora shown in Figure 5.10.
Figure 5.4: Setting suspect's **empathy**=70 **pressure**=10 correctly identified as an happy mood.

Figure 5.5: Setting suspect's **empathy**=10 **pressure**=10 correctly identified as sad mood.

Figure 5.6: Setting suspect's **empathy**=80 **pressure**=80 correctly identified as nervous mood.

Figure 5.7: Setting suspect's **empathy**=10 **pressure**=70 correctly identified as an angry mood.

Figure 5.8: Setting suspect's **empathy**=50 **pressure**=50 correctly identified as relaxed mood.

Figure 5.9: Setting suspect's to a boundary of **empathy**=50 **pressure**=0 and which is identified as an happy mood.

Figure 5.10: Corpora test set to validate the impact of **truth**=0.

The results are shown in Figures 5.11 and 5.12.

The detection of rapport via repetition of words is easily validated by repeating exactly the answer
Figure 5.11: Impact of a truth=-1 on the suspect’s pressure (increases in 10 points).

Figure 5.12: Impact of a truth=0 on the suspect’s pressure (increases in 5 points).

from the suspect in the form of a question. This is shown in Figure 5.13.

Figure 5.13: Questioning the suspect with his own answer (Apesar de tudo, estou bem, obrigado.) to which the application detects four words (Apesar, tudo, bem and obrigado). This correctly increases the suspect’s empathy by four points (one for each word).

The importantInfo field can take values of 1, 0 and -1. If importantInfo=1, pressure is increased by 5 points. If importantInfo=0, empathy is increased by 5 points. Lastly, if importantInfo=-1, empathy drops by 5 points. These are shown in Figures 5.14, 5.15 and 5.16.

Figure 5.14: Impact of a importantInfo=1 on the suspect’s pressure (increases in 5 points).

Figure 5.15: Impact of a importantInfo=0 on the suspect’s empathy (increases in 5 points).

The correctTiming tag can take values of 0, 1, 2, 3 and 4. If correctTiming=0 the suspect’s empathy drops by 10 points shown in figure 5.17. If the value of the tag decreases (correctTiming=3 followed by a correctTiming=2) or is violated altogether (correctTiming=3 followed by a correctTiming=1) the
suspect’s **empathy** drops by 5 points. If the order is respected, **empathy** increases in five points. These are shown in Figures 5.18 and 5.19.

Finally, regarding the detection of silences, when the question **SILENCE** is made, the application answers with the last interaction’s answer tagged with **silence**=1 (shown in Figures 4.6 and 4.7) or, if none exists, with a randomly answer from within a set of answers (4.8). This last case is shown in figure 4.9.

5.2.2 Experiment 2

In Section 5.2 the individual impact of each feature was verified. It is necessary, however, to validate the application as a whole. Meaning, to demonstrate and validate how, during a normal conversation, the suspect’s **interpersonal state** varies and its role on answer picking. To do so, a script, which follows the main procedures of a good police interview, was compiled. Note that this script had the sets of questions and answers built in the knowledge base beforehand as described in Section 5.1. This script is found in Appendix C. For sake of an example, a question and answer set is shown in Figure 5.20.
In the square box is the question, followed by the Suspect's mood and finally the answer beginning with “A:”.

Several conclusions are drawn from the performance of the application. Overall, it seems the suspect’s empathy is correctly updated and it only increases because only good things are happening (i.e. the script is made to comply with all that adds empathy). However, the suspect's pressure, with no reason not to decrease, always increases. There should be a way to decrease the pressure, for instance, with time elapsed. One other conclusion is that, despite being a positive script, the interpersonal state should vary up and down other than constantly increasing its values. This implies that there should be more balance between the number of points added and the number of points subtracted. This means it needs tuning as far as tags and their values are concerned, at a more fundamental level.
Chapter 6

Conclusions and Future Work

Overall, the demo proposed was built successfully. With a reasonable knowledge base, the application will answer the interviewer's question in a rewarding or punishing way if the correct stance towards the theories behind the Cognitive Interview and PEACE Model are taken. However, the way the application processes each interaction is made in a very simple way and the rewarding and punishing of some attributes is not accurate, as explained in Section 5.2.2. Furthermore, the application is still very reliable on the knowledge base. Which tags to use, their values and in which ways they impact the suspect's choices are three different aspects that amount to a great deal complexity in the non exact and very often biased world of psychology. Furthermore, the adding of tags makes the extension of knowledge extremely hard. This is due to the fact that each tag must be carefully thought of. Overall, the application correctly identifies which answer to pick, regardless on which assumptions the decision is made.

In the future several improvements could be made into this virtual suspect so as to make it closer to reality and the cognitive interview guidelines:

- **Knowledge**: As mentioned previously, the greater the number of questions and answers the suspect possesses the better is its ability to correctly answer a question.

- **Moods**: A way to improve the set up is by dissecting the suspect's personality into a more complex one. A suspect's mood cannot be said to be simply happy or sad. For instance, being envious is not the same as being angry or sad. This will obviously incur in a larger complexity while modeling the suspect's personality.

- **Sentiment analysis**: Video and speech analysis could be added to have a greater understanding of the interviewer's mood. The text sentiment analysis can also be improved by utilizing more robust machine learning models such as neural networks, thus increasing the sentiments accuracy.

- **Rapport Detection**: The rapport, which plays an important role in the interview, can be detected by the usage of similar acoustic and prosodic features between the interviewer and the interviewee (also known as acoustic-prosodic entrainment). As is, the application only searches for completely equal words in order to have a sort of rapport detection. This serves as a very basic detection of prosodic entrainment. Other techniques could be used to better detect entrainment both at the
prosodic and acoustic dimensions. One example is, at the prosodic level, the use of a Jaccard Coefficient (Niwattanakul et al., 2013) to detect a degree of the similarity between words. It is possible that the user does not use the exact same word but a slight variation of it which should still be accounted for.

- **New tags:** New tags, following the same schema of the currently implemented ones can be set to correctly affect the state of the suspect given a particular answer or question. Adding tags can lead to better punish or reward the interviewer for following the right approaches during the interview. For instance, according to the Cognitive Interview, the police should always avoid making leading questions. Tagging these questions with a different tag will allow for a specific impact on suspects mood or have other implications altogether. Another possible tag is one to distinguish closed and open questions. As explained in Section 2.4.1, police officers must ask these two kinds of question distinctively, throughout the interview. Open questions are usually best for information retrieval whereas closed questions are used at the very end to confront the suspect with facts or things which make no sense. Thus an open and closed tag would be great to detect if the correct form of question is used in the appropriate phase of the interview.

- **Answering:** The way the suspects picks an answer can be much more complex than what is supported in this application. So far, the suspect is primarily picking an answer based on its current empathy.

- **Strategies:** When the interview is silent for a determined time, the suspect will answer with a specific answer, complementing what he was talking previously. This partially tackles one of the most important aspects of the interview - letting the suspect talk as much as possible. Just like this, many more strategies can be overlaid not only to bring interview closure to reality but also to give the interview the possibility of being rewarded by being compliant with the Cognitive Interview.

- **Memory:** A big problem regarding this virtual suspect in particular is that it is basically memoryless. For instance if a person asks “Where was John and Mary?” followed by “What was she doing?” will incur in wrong answers as the suspect has no knowledge that the interviewer is asking about Mary specifically. To tackle this issue a sort of illusion of choice must be set onto the knowledge structure. What is meant by this is that by asking question A only a set up of answers and questions should be valid for the next interaction. This is sort of what happens in FAiMA’s structure, discussed in Section 3.6.

- **Interface:** The interface can obviously use some work regarding design and accessibility to make it easier and simpler to use. More features can be implemented such as voice activity detection (i.e. without the need to press a button to record voice), a way to pause the interview and adding more characters with different stories to enrich the experience.


Application Scenario Outlines: EPJ-MJ use case “Interview Skills for Police Officers (ISPO)/Victims and offenders of violent crimes”.


Appendix A

Interface

Unity3D Editor\(^1\) is the tool used to develop and manage objects in hierarchy (figure A.1) which are controlled via C# scripts. Note that explaining in detail every aspect, for instance, how the triggers are implemented, would drive this out of this thesis scope. Therefore, the descriptions are technically broad.

\(^1\)www.unity3d.com
A.1 Main Menu

The main menu is shown in figure A.2. There is an Exit button which, on click, calls a script that quits the application via the method Quit(). Moreover, a Help button which takes one to the Help Menu scene and likewise an Audio button. There are two buttons above the rest with the names of the suspects. The idea behind leaving two suspects is for replicability in the future. However, only one is available. When one has his mouse over this button, a figure and text appear generally describing the suspect and crime (see figure A.2).

![Main Menu with hover text effect.](image1)

This mouse over feature was made by using Unity’s inspector, an event trigger and a script (named hovertextze). Once the mouse is over, it calls two functions from the script which enable or disable the objects they are attached to. A click on the suspect's button will take us to the options menu. This is made by taking advantage of Unity’s scene management which loads another scene via script (named changeScene). The other buttons except Exit, do not load other scenes but simply disable everything on the Main Menu and display another set of objects (using GameObject.SetActive).

A.2 Audio Menu

![Audio Menu.](image2)

The reason behind audio menu is that the user should be able to change settings. If the application

---

2https://docs.unity3d.com/ScriptReference/Application.Quit.html
is ran resorting to different microphones there will be wrongly produced inputs. Here, one can change the volume from effects on the virtual reality scenes (sounds of clocks, rain, etc), control the master volume, as well as change the microphone input gain. For instance, one may want to disable the effects and hear the suspect louder. The audio menu is shown in figure A.3. The sliders are default at half the scale in order to allow the increase and decrease of the variables. Each slider is connected to a float variable which on change calls a method on a script (named `PlayerPrefsManager`). Its implementation was rather troublesome because the values needed to be stored through all scenes. A simple static variable would not work. The script needs to have setters and getters for the values. By making use of `PlayerPrefs`, we are storing and accessing player preferences between game sessions. The Back button brings back the primary Main Menu scene.

A.3 Help Menu

![Help Menu](https://docs.unity3d.com/ScriptReference/PlayerPrefs.html)

Figure A.4: Help Menu.

The Help menu (figure A.4) displays some information on contacts, how to use the different modes and a button to return to the main scene.

A.4 Game Options Menu

Once the user clicks the suspect's of his choice, the Options Menu pops up (figure A.5). Here the user chooses the game mode. Three modes are available each with a brief description alongside. There are also four different toggles for options:

- **Toggle Stereoscopic VR** Enables stereoscopic rendering on the VR mode in order for the user to use goggles.
- **Toggle Voice Activity Detection** Enables automatic voice activity detection.
- **Toggle Feedback** Toggles the display of emotions in real time.

---

[^3]: [https://docs.unity3d.com/ScriptReference/PlayerPrefs.html](https://docs.unity3d.com/ScriptReference/PlayerPrefs.html)
• **Toggle Teach Model** Enables the teaching mode.

These toggles, instantiated on the script `booleanFromOptions`, are attached to an object on the scene. If the toggle is pressed, the script changes the values of the static booleans.

### A.5 Modes

There are three different modes one can choose from in the **Options** menu. Each mode has unique `EtutorClient` and `SendScript` scripts. The `EtutorClient` deals mostly with sending and receiving information to places other than Unity such as recording and sending audio to the server. The `SendScript` deals with the game objects, how they interact and serves as a middle man to the `EtutorClient` script.

#### A.5.1 Speech + Text Mode

The **Speech + Text Mode** is the simplest of all. On the left, and if the feedback toggle is on, there is the feedback panel. An indication of the time the interview is taking is displayed on the right side, starting
on the first question asked. This is made possible using GameTimerTO script which starts and routinely updates the timer. The Info button displays a panel with information about the suspect. Details that an interviewer might get before the meeting with the suspect (see figure A.7). A scroll bar is available for extensive details (for instance, suspect’s prior crimes).

![Image](image_url)

Figure A.7: Info example.

The user may write his question and press the return key to send or simply click the Enter button. Unity’s editor provides an interface which makes it easy to trigger an action on click. However, to make it possible to send a question using a keyboard key, an update method had to be used on a script. This update checks every frame for keys pressed and refocuses the selection to the input field (using Unity’s Select() method). When the Enter button is pressed, it checks if the input field has a length superior to two characters and if so, calls a method called getAnswer() which sends it to Java. After getting the answer from Java, it logs the question and answer onto the object which displays the text with different colours (see figure A.8). In order to avoid conflicts, while the application is retrieving the answer, the question input field and buttons are disabled. This is made using the methods SetActive(boolean) or Interactable(boolean).

![Image](image_url)

Figure A.8: Question example.

Another way to input text onto the input field other than writing is through speech recognition. To use this feature, the user presses the Push to talk button while speaking. On press, the script EtutorClientTO triggers startrec method which starts recording and on release, calls stoprec stops the recording. While pressed, the other buttons and input field are disabled. The method startrec starts a coroutine which gets the audio into a float array and converts it into Byte format. On stop (button release), another coroutine starts sending the byte array and options to a server. In this mode, the only option is the language (Portuguese). The server sends back the recognized text which is then put onto the input field. This way the user may or may not choose to edit the recognized question. This extra step prevents the sending of
wrongly recognized questions. If the speech to text on the server was not able to recognize the speech altogether, a _REPEAT_ tag is sent by the server. In this case, the message "Unable to recognize speech, please try again..." is displayed. As the conversation grows, the TextLogControl script controls the handle on the scroll bar allowing the user to read anything previously said on the conversation.

If the Teach Model toggle is set true from the previous scene (Options menu), a panel is displayed as shown in figure A.9 asking to set the question’s nature (positive, negative, or neutral) before the question is sent.

![Teach Model example](image)

Figure A.9: Teach Model example.

By clicking, for instance, positive, the tag _pos_ is added to the question. For negative, the tag _neg_ and for neutral nothing is added. The processing of these key words is done on the back-end (Java).

### A.5.2 Video + Speech + Text Mode

![Video + Speech + Text Mode](image)

Figure A.10: Video + Speech + Text Mode.

The Video + Speech + Text Mode is, as the name implies, the addiction of video to the previous mode. The exception is that there is no option Teach Model here. The camera is locked which means the user cannot look elsewhere. The feedback panel is brought up on the PC’s display.
The Info, Quit, Send and Push to talk buttons have a close implementation to those on the previous mode with slight differences for the Update() method which would not work well otherwise. This is mainly due to the frame rate being different with video rendering.

![Figure A.11: The set viewed from the camera angle.](image)

In order to bring up the reality aspect of the scene, several assets were created or imported. The set, shown in figure A.11 was made above a preset called morgue_scene available on the Unity3D store\(^4\) from Rokay3D. The walls, blackboard, lights, shelves and jars from that scene were used. The chair is from Rakshi Games\(^5\). Other assets were added but they are all free on Unity3D asset store.

![Figure A.12: The set with all the components.](image)

The clock, see figure A.13, improves the reality nature of the scene but also serves as a measure of time. The clock materials and mesh are imported from Unity’s store. In order to animate it, Unity’s quaternion rotations\(^6\) were used on a coroutine, more specifically on the hour, minute and seconds pointers so that these would rotate according to the machine’s current time (using Unity’s DateTime). The clock also has an audio source which produces the tic-tac sound each time the second’s pointer moves.

---

\(^4\)https://www.assetstore.unity3d.com/en/#!/content/65817
\(^5\)https://www.assetstore.unity3d.com/en/#!/content/848
\(^6\)https://docs.unity3d.com/ScriptReference/Transform-rotation.html
When the user presses the **Send** button, the question is sent to the back-end in java, the answer which comes back is sent through *EtutorClient* script (more specifically the coroutine *SendTextToServer*) in the *WWW* format\(^7\). The server answers with the audio which is then played through the coroutine *playrawtts2()*.

The question and answer are displayed on the blackboard of the scene as shown in figure A.14. The lip syncing, animations and the character itself are explained in section B.

**A.5.3 Virtual Reality Mode**

Unlike the *Video + Speech + Text* Mode, this mode has no lock on the camera, only on the movement of the player (the camera is sitting on the chair). As one can see in figure A.16 there were changes to the set. Primarily the material used on the walls and the addition of a window to make the VR aspect more natural.

The window is a wall created on a reworked mesh grid with imported materials. The city view which appears to be the “outside” of the office is a skybox\(^8\) created from adding six different pictures taken from [https://opengameart.org/content/urbanskyboxes](https://opengameart.org/content/urbanskyboxes). These form up a cuboid around the scene thus giving the illusion that one is looking at the outside through the window. A directional light is set outside the

\(^7\)https://docs.unity3d.com/ScriptReference/WWW.html

\(^8\)https://en.wikipedia.org/wiki/Skybox_(video_games)
window as to fake the sun. To make it even more realistic, a particle system\(^9\) was added in order to make it appear that it is raining outside (see figure A.17). The rain itself is composed by three separate particle systems: one for the drips, another for the streaks and lastly one for the splashes. They are brought together with a mesh renderer. An audio source with rain sounds is looping in the window object.

\(^9\)https://docs.unity3d.com/ScriptReference/ParticleSystem.html
A volume bar is displayed on the computer which shows the microphone level (see figure A.18). This is made using a canvas scaler\textsuperscript{10} on the green bar, which is acted upon by the \textit{BarInteraction} and \textit{SpeechDetect} scripts. The latter uses Unity's API for microphone\textsuperscript{11}, sends it to \textit{BarInteraction} which uses transforms on the canvas scaler. The values chosen by the user in the audio menu are multiplied directly (a float for volume sensitivity).

![You may talk... Press Left Control](image)

\textbf{Figure A.18: Microphone level.}

The camera from which the user sees the scene depends if a stereoscopic view was toggled in the options menu. Without goggles, the camera is controlled by the script \textit{CameraScript} which acts upon the camera rotation through the cursor movement. This script keeps the cursor centered and invisible using Unity's Cursor API\textsuperscript{12} and keeps updating the quaternion rotation of the camera based on the current movement of the mouse multiplied by a sensitivity value for the $x$ and $y$ axis distinctively. This allows for a higher sensitivity on the $x$ axis rather than the $y$ axis which resembles more the nature of the human neck rotation (i.e. it is easier to rotate one's head from right to left rather than up and down). In addition, two buffers on both axis allow for a small delay on the rotation making it very smooth. The $y$ axis has its ranged clamped between 90 and -90 degrees and the $x$ axis has 360 degrees range. This prevents the user from losing his orientation on the scene and being, for instance, viewing scene upside down.

To get out of this mode, the user must press the \textbf{Escape} key which brings up the panel shown in figure A.19 controlled by \textit{exit\_script} script. The user is presented with the choice to hide the panel or get the user back to the \textbf{Options} scene.

The source of the suspect's voice is set using a sphere placed on the character's mouth (see figure A.20). This increases the 3D audio experience of the virtual reality because the sound is projected from the sphere just as it would in a real environment.

\textsuperscript{10}https://docs.unity3d.com/ScriptReference/UI.CanvasScaler.html
\textsuperscript{11}https://docs.unity3d.com/ScriptReference/Microphone.html
\textsuperscript{12}https://docs.unity3d.com/ScriptReference/Cursor.html
Figure A.19: Microphone level.

Figure A.20: Sound source sphere.
Appendix B

Character and animation

B.1 Character

Figure B.1: Winston, fully rigged character from Reallusion.

The character body is named Winston and was imported from Unity's Store\(^1\) which was developed by Reallusion using their Character Creator 2 software\(^2\). Winston was an ideal character to work with due to its strong appearance, highly detailed textures but ultimately because a fully rigged body eases the animation process. Despite being free for Unity, it was used with Reallusion's permission.

B.2 Animations

B.2.1 Creating the animations

The animations used for the body positions, hands and arm movements were imported from Unity store. Specifically, most came from *Huge FBX Mocap Library*\(^3\) asset which contains around three thousand

---

\(^1\)https://www.assetstore.unity3d.com/en/#/content/50694
\(^2\)https://www.reallusion.com/character-creator/
\(^3\)https://www.assetstore.unity3d.com/en/#/content/20282
body animations and the rest from Mixamo⁴.

Several bones on Winston’s structure had to be reset so that the animations would work. The facial animations however, were created from scratch. This was mainly due to two reasons. Firstly, the animations would not work correctly with this character in Unity Editor (bone structure is not entirely compatible with Unity). Secondly, it is from the face that most of the body language is perceived and therefore most of the individual’s state of mind. Hence, accurate expressions were necessary. The animations were created using iClone⁵ and imported to Unity via 3DXchange Pipeline⁶, all software from Reallusion. Figure B.2 shows an example of iClone’s environment, where one can create animations using its presets and afterwards fine tune them through the different body values for them to work with the specific character.

For the facial expressions, the Face Key tool was used. It provides preset facial expressions and their respective value modifiers (see figure B.3).

---

⁴www.mixamo.com  
⁵https://www.reallusion.com/iclone/  
The *Expressiveness* slide bar indicates how strongly to apply the preset. As said before, the presets needed some fine tuning for which the *Muscle* tab was used. Here one can drag the muscles to fine tune or create the facial expression altogether. Figure B.4 shows an example of dragging the chin to get a simple open mouth.

![Figure B.4: Face Key’s muscle tab.](image)

The *Modify* tab, shown in figure B.5, was what mostly helped create the expressions. This tab allows for easy and detailed modifications on the eyes, eyebrows and lips to accomplished the desired output.

![Figure B.5: Face Key’s Modifying tab.](image)

Afterwards, the Reallusion’s pipeline tool was used to import the animations to Unity. Around forty expressions were imported in total. Some are shown in figure B.6.
B.2.2 Controlling the animations

The character, as a gameObject, has an Animation Controller\(^7\) which is the baseline for managing all the animations. The animator was divided in eight layers as seen in figure B.7. Each targets a specific part of the body. This division is made so that different animations can interact on different body parts. For instance, an animation of being sit is set to the lower part of the body (legs, feet and torso) and at the same time a look up animation is set to the head. The reason behind this division is that the Unity’s store animations out of the box entail being standing up, moving arms and inconsistent head movement with what is being said when in fact we just wanted, for instance, a smile. Ideally, we would have a full

\(^7\)https://docs.unity3d.com/ScriptReference/Animations.AnimatorController.html
body animation for each stance (happy, sad, thoughtful, nervous, etc). To come up with this division it was necessary the make masks. In figure B.8, one can see an example of a mask for the right arm. Each layer, with the according mask, has an scheme of animations as shown in figure B.9. Each box represents a state, and similar to a petri net, it transits onto the next state if a certain boolean is set to true. These booleans are acted upon by the script Anim_control which is attached to the character gameObject. It updates on a regular basis and if a new state is ordered by the back end Java, it updates the parameters. The suspect will stay mostly in a default state which led to the creation of a Sub-State Machine to break monotony. For instance, from time to time, the suspect may close his hand. These states, shown in figure B.10, keep alternating between each other.

The parameters (booleans) which fire the transitions have the same names as the states for simplicity. For sake of an example, on the relaxed state, the Base, Torso and Head layers use a sitting idle motion.
The *Face_expressions, LeftArm* and *LeftHand* layers have no motion at all while the *RightHand* and *RightArm* layers are set with the *Tickling Fingers* motion. All animations have the speed set to 60% for appearing to be more human like. If the boolean *happy* is set to true the animation goes to the *Happy* state. Here all the layers except *Face_expressions* and the hand layers have no motion while the rest have *Happy* based motions.

The facial animations are controlled by the *Anim_control* script attached to the character’s gameObject. This script has five distinctive groups: *relaxed, happy, sad, nervous* and *angry*. Each group has several overlayed imported facial features (see previous section) to get the appropriate overall expressions shown in figure B.11.

For instance, the *angry* facial expression is, in fact, a result of overlaying the following imported expressions: raised outer left eyebrow, raised outer right eyebrow, lowered inner left eyebrow, lowered inner right eyebrow, tightly closed lips and tucked chin. This overlaying of expressions happen in the *happy, nervous* and *angry* groups. The state where the suspect is more likely to be, the *relaxed* group, has a more complex dynamic. In this group, twenty of the imported facial expression are randomly added and discarded to the suspect. The number of expressions, their duration, as well as their strength and blend speed (time it takes from the beginning of the animation to its normal state and the same for its end) are randomized (each with their own values). The range of values the random function can pick is different for blend, strength and duration which were handpicked. This gives a very human like appearance to the suspect and the combination of values and facial expressions is so sheer that no face is likely to be repeated twice.

Another aspect taken into account was the blink rate. Initially the blink rate was set higher to the *nervous* and *angry* states but later changed to the same value as the other facial expressions. The influence of anxiety on the blink rate has been investigated repeatedly, but the results obtained have been
uncertain or contradictory altogether. A study from the Defense Technical Information Center (Doehring, 1957) showed that in certain anxious verbal situations, individuals exhibit higher blink rates. (Fukuda and Matsunaga, 1983) showed that blink rates peaked right after auditory and visual stimuli. On the other hand, (Holland and Tarlow, 1972) show that when mental load is high (related to pressure and anxiety) the blinking rate actually decreases.

### B.3 Lip-sync

When a string of text is sent to the server, it answers back with the wave file (from text-to-speech) and an hashtable with phonemes with respective time of appearance on the wave file. The coroutine `parseJSONResponse`, within `EtutorClient` script, is responsible for receiving this information. When finished, `connectAnim` method, in `Anim_control` script rephrases each phoneme to the respective viseme using a table from the script `LangResources`. The table connects each phoneme to a certain viseme. The relationship between phonemes and visemes can be accessed at IPA\(^{11}\). In total there are 56 in contrast to the 13 used in the animation process. This means there is an aggregation of phonemes of other groups to those 13. This is made for simplicity sake. Ideally each phoneme should have its own viseme. The approximation used, shown in table B.1, is the same used in (Sousa, 2012) and slightly the same as Amazon Polly’s\(^{12}\).

| Phoneme | \( f \) | \( v \) | \( k \) | \( g \) | \( L \) | \( J \) | \( l \) | \(~ R \) | \( r \) | \( O \) | \( p \) | \( b \) | \( m \) | \( s \) | \( z \) | \( t \) | \( d \) | \( n \) | \( S \) |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| Viseme  | \( f \) | \( f \) | \( g \) | \( g \) | \( g \) | \( l \) | \( l \) | \( l \) | \( l \) | \( O \) | \( p \) | \( p \) | \( p \) | \( s \) | \( z \) | \( t \) | \( t \) | \( t \) | \( S \) |

<table>
<thead>
<tr>
<th>Phoneme</th>
<th>( a )</th>
<th>( 6 )</th>
<th>( 6 )~</th>
<th>( 6 )~</th>
<th>( 6 )~</th>
<th>( w )~</th>
<th>( e )</th>
<th>( E )</th>
<th>( e )~</th>
<th>( i )~</th>
<th>( j )~</th>
<th>( j )~</th>
<th>( o )~</th>
<th>( o )~</th>
<th>( o )~</th>
<th>( u )~</th>
<th>( w )~</th>
<th>( w )~</th>
<th>( u )~</th>
</tr>
</thead>
<tbody>
<tr>
<td>Viseme</td>
<td>( a )</td>
<td>( a )</td>
<td>( a )</td>
<td>( a )</td>
<td>( a )</td>
<td>( e )</td>
<td>( e )</td>
<td>( e )</td>
<td>( i )</td>
<td>( i )</td>
<td>( i )</td>
<td>( i )</td>
<td>( o )</td>
<td>( o )</td>
<td>( o )</td>
<td>( u )</td>
<td>( u )</td>
<td>( u )</td>
<td>( u )</td>
</tr>
</tbody>
</table>

Table B.1: Phonemes and correspondent visemes (in ASCII).

The next step was to make the thirteen viseme animations. This was made possible using iClone’s `Lips Editor` tool. A sample wave file was recorded using all the phonemes several times and loaded into iClone. Using the lip tool, key frames were added to the place where phonemes were heard on the audio file (see figure B.12).

![Figure B.12: Addition of keyframes onto the audio file.](807.2)

On each key frame (phoneme), the modifications to the lips, tongue and jaw muscles were applied to make them as close to the correspondent visemes as possible (see figure B.13) Afterwards, as men-

---

11 International Phonetic Alphabet
tioned for other iClone animations, the thirteen animations were pipelined and imported to Unity.

Once the suspect is ready to talk, the coroutine `etutor.playrawtts()` is started and the animations are played through the script using the animation hashmap at the same time as the audio from the wave file.

### B.4 Eyes and head movement

Both eyes and head are controlled by having applied transforms according to a target sphere. This sphere, shown in figure B.14, is controlled by `moveSphere` script. The latter has a base start point which is the point around which the sphere moves, a start point where the sphere starts and a destination point to where the sphere is going to move. These points are of type `Vector2`. Once the sphere reaches the destination point using transform position\(^1\), a new destination point is picked randomly within the circle centered on the base point using Unity’s `Random` method\(^2\). The speed with which the sphere moves is set accordingly to a float `progress`, resulting from the multiplication of two other floats: the defined speed and time unit `deltaTime` from Unity’s `Time` API. The `Update()` method constantly checks if the destination is reached or if it was an overshoot. Ideally, the sphere should have its speed vary randomly across time and smoothness should be applied once it gets closer to the target destination. Moreover, the destination points should not be totally random but set according to a probability distribution (more points closer to the center).

---

\(^1\)`https://docs.unity3d.com/ScriptReference/Transform-localPosition.html`  
\(^2\)`https://docs.unity3d.com/ScriptReference/Random-insideUnitCircle.html`  
\(^3\)`https://docs.unity3d.com/ScriptReference/Vector2.html`
The chosen values for radius and speed were respectively 0.3 and 0.4 which are small but were found by trial and error until it appeared like human movement. The headController script constantly gets the position of the sphere via LookAt\(^\text{16}\) method and applies the rotation to the head or eyes by multiplying the rotation towards the destination target with a Euler Quaternion\(^\text{17}\).

\(^{16}\)https://docs.unity3d.com/ScriptReference/Transform.LookAt.html

\(^{17}\)https://docs.unity3d.com/ScriptReference/Quaternion.Euler.html
Appendix C

Script

_Q: Olá Senhor Zé_
Empathy: 55.0 Pressure: 50.0 Mood: Relaxed
A: Olá Senhor Inspector.

_Q: Tudo bem consigo?_
Empathy: 65.0 Pressure: 50.0 Mood: Relaxed
A: Apesar de tudo, estou bem, obrigado.

_Q: Hoje vamos-lhe fazer algumas perguntas acerca do que se passou, ok?_
Empathy: 75.0 Pressure: 50.0 Mood: Happy
A: No que puder ajudar, estou à vossa disposição.

_Q: Como se sente com tudo isto?_
Empathy: 81.0 Pressure: 50.0 Mood: Happy
A: Conhecia a Dona Maria há muito tempo, tem sido complicado.

_Q: SILENCE._
Empathy: 86.0 Pressure: 50.0 Mood: Happy
A: ...foram muitos anos, gostava muito dela.

_Q: É complicado, tem continuado as suas funções?_
Empathy: 92.0 Pressure: 50.0 Mood: Happy
A: Continuo, um pouco para não deixar estragar a quinta em honra a Dona Maria.

_Q: O que é que faz na quinta?_
Empathy: 100.0 Pressure: 50.0 Mood: Happy
A: A minha função principal é assegurar que a quinta se mantém. Várias coisas advém daí, como por exemplo a limpeza e jardinagem.

_Q: Quantas pessoas vivem na quinta?_
Bem, sou eu, a empregada Defuntina e o filho de Dona Maria. 

Como era o seu relacionamento com Dona Maria? 

Maria do Carmo era uma pessoa muito assertiva embora um pouco mimada e antipática para algumas pessoas.

Eu e Maria do Carmo tínhamos uma relação de amizade. Nunca discutimos.

Dona Maria já sofria há alguns anos do coração.

De vez em quando ia ao hospital.

Dona Maria dava-se bem com toda a gente?

Dona Maria já tinha a sua idade mas sempre cultivou boas relações.

...chegou a ter umas discussões com o filho mas era normal.

Qual era a relação do filho com Dona Maria?

Ele tinha alguns problemas, por vezes era um pouco teimoso mas eles davam-se bem.

E o que acha, pessoalmente, do filho?

Como já disse, era teimoso. Também um pouco irresponsável e preguiçoso.

É uma situação normal com filhos abastados, ele dá-se com Defuntina?

Sim, por vezes até bem demais por vezes nem tanto assim.

Há quanto tempo trabalha Defuntina para Dona Maria?

Defuntina veio de um centro de acolhimento com os seus doze anos e trabalha na quinta desde então.

Um pouco estranho para uma pessoa tão nova, Dona Maria tratava-a assim tão bem?
A: Extremamente bem, tudo o que ela precisava era lhe dado.

Q: _SILENCE_

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: ...até tinha um quarto melhor que o do filho de Dona Maria.

Q: Qual pensa ser a razão em especial da relação?

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: A verdade é que Defuntina é filha de um caso que Dona Maria teve já depois do seu marido falecer.

Q: E a situação era conhecida pelo filho e Defuntina?

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: Tanto quanto sei nenhum deles sabia.

Q: Há pouco falou-me que o filho era um pouco mimado, ele tratava bem os empregados?

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: Nem sempre, costumava dar-se mais com a Defuntina que comigo e o jardineiro.

Q: Porquê Defuntina em especial? Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: Porque eles os dois tinham um caso amoroso.

Q: _SILENCE_

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: Já era costume eu vê-los aos beijos no jardim às escondidas.

Q: Isso quer dizer que houve uma situação de incesto?

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: Sim. Já com algum tempo

Q: Durante quanto tempo?

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: Sei que pelo menos uns cinco anos.

Q: _SILENCE_

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: ...no início pensei que não estava a ver bem.

Q: Nunca pensou contar a situação dos filhos a Dona Maria?

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: Sim, durante muito tempo até que finalmente ganhei coragem e contei.

Q: E quando é que isto foi?

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: Isto foi no dia em que Dona Maria faleceu.

Q: _SILENCE_

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: ...a verdade é que penso que Dona Maria não aguentou com as notícias e acabou por desfalecer.

Q: Tenha calma, fez bem em contar o sucedido. Chamou os paramédicos logo de seguida?

Empathy: 100.0 Pressure: 100.0 Mood: Nervous

A: Demorei um pouco porque não sabia bem o que fazer.
Q: SILENCE
Empathy: 100.0 Pressure: 100.0 Mood: Nervous
A: ...sei que reagi mal, talvez seja eu o culpado da morte de Dona Maria.

Q: É normal que as pessoas não saibam como reagir nestas situações. Tem alguma pergunta que me queira pôr?
Empathy: 100.0 Pressure: 100.0 Mood: Nervous
A: Não, é tudo o que precisa de mim?

Q: iremos entrar em contacto com o Senhor Zé o quanto antes. Até lá, tenha calma que tudo correrá bem.
Empathy: 100.0 Pressure: 100.0 Mood: Nervous
A: Obrigado pela compreensão.

Q: Obrigado Senhor Zé. Tenha um bom dia.
Empathy: 100.0 Pressure: 100.0 Mood: Nervous
A: Tenha um resto de um bom dia. Que tudo fique melhor.
## Appendix D

### Positive and negative instances

<table>
<thead>
<tr>
<th>Positive instances</th>
<th>Negative instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>admirável, adorável, agradável, alegre, normal, alucinante, apaixonante, muito bem, apaixonado, astuto, atraente, autêntico, belo, bem, bem-humorado, bom, bonito, brilhante, carinhoso, carismático, cativante, cativar, certo, claro, exatamente, classe, clássico, coerente, compreensível, comum, consistente, contagiente, contente, convincente, corajoso, decente, delicado, digno, dinâmico, direito, divertido, divino, doce, elegante, engraçado, esperto, espetacular, excelente, excepcional, excitante, exemplar, extraordinário, fácil, fantástico, feliz, fofo, forte, fundamental, genial, grato, humano, ideal, impecável, importante, incrível, inspirador, inteligente, intenso, interessante, íntimo, legítimo, magnífico, maravilha, maravilhoso, marcante, ótimo, pertinente, plausível, pontual, positivo, precioso, profundo, raro, razoável, realista, sedutor, seguro, sensível, sensual, sentimental, sério, sexy, significativo, simples, sincero, honesto, singular, tranquilo, inocente, válido, verdadeiro, vital, voraz, surpreendente, agradar, amadurecer, amar, animar, apaixonar, apreciar, aprender, atrair, conquistar, curtir, divertir, emocionar, inovar, simpatizar, valorizar, natural, naturalmente, brilhar, comover, razão, saborear, brilho, carisma, clímax, companheirismo, compreensão, diversão, doçura, emoção, empatia, entusiasmo, excitado, fã, facilidade, fascinado, inteligência, provavelmente, faria o mesmo, o mesmo magnetismo, mérito, paixão, perfeição, prazer, preferido, sensibilidade, talento, ternura, top, amor, chamar a atenção, dar certo, de corpo e alma, de fazer inveja, tudo de bom, razoável&quot;</td>
<td>burro, bárbaro, chato, chocante, confuso, decepcionado, defeituoso, deplorável, depressivo, deprimente, desagradável, desgastante, desinteressante, desnecessário, desprezível, difícil, dispensável, doentio, egoísta, enfadonho, enjooado, enjoativo, entediante, estranho, falso, fraco, frio, frustrante, fútil, horrível, idiota, idiótic, imaturo, impaciente, incompreensível, inconsistente, ingenuo, injustificável, insuportável, interminável, inútil, irritante, lamentável, maçante, machista, mal, mau, mero, monótono, morno, negativo, normal, obsesso, óbvio, oco, opressivo, paranóico, patético, péssimo, piorar, pobre, pomposo, preguiçoso, previsível, repetitivo, repulsivo, revoltante, ridículo, simpático, superficial, surreal, tédio, tedioso, tosco, triste, vazio, violento, lento, linear, óbvio, abandonar, aborrecer, arrepender, assustar, aterrorizar, atormentar, atrapalhar, cansar, chorar, complicar, decepcionar, deprimir, desanimar, desgastar, desis- tirm, desmerecer, destorcer, desesperar, dificultar, distorcer, empacar, enfaquecer, enganar, engolir, errar, estragar, estressar, exagerar, faltar, frustrar, incomodar, irritar, largar, limitar, odiar, pecar, perder, revoltar, decepção, defunto, demorar, desastre, desconforto, desgraça, desesperar, desesperar, engano, erro, exagerado, exastão, excesso, falhar, falho, falta, furo, imaturidade, incomodo, inexperiente, lixar, melodrama, melodramático, meloso, monótono, mundo, nojo, calado, porca, porcaria, problema, sacrifício, vergonha, caralho, fodasse, ótario, palhaço, fala, filho da puta, puta, rato, putice, paneleiro, panduca, filho da mãe, gozar comigo, bicha</td>
</tr>
</tbody>
</table>

Table D.1: Positive and negative instances added by hand.