Swiss Cottage – a game to train speech recognition for an affective computing treatment of ADHD patients

Martin Porcheron Kyle G. Arch Steven D. Luland
Peter Blanchfield Michel F. Valstar Andry Chowanda

School of Computer Science, University of Nottingham

14th June 2013

1 Introduction

Both Attention Deficit Disorder (ADD) and Attention Deficit Hyperactivity Disorder (ADHD) are being looked at for treatment using a games-based approach (Carr & Blanchfield 2009, Lis et al. 2013). The former of these papers deliberately looks at the behaviour of players by attempting to analyse their expressions while playing the game. The purpose of this is that through facial expression it is potentially possible to identify the emotional state of the players. As stated by Carr & Blanchfield (2009), when dealing with adolescents and pre-adolescents with ADHD it is vital that the game engages them from the beginning. This can be through the use of an adventure-based story line for example and must use lifelike graphics. Another way to engage players, and one which we adopt here, is to use interaction with life-like virtual humans that employ speech recognition, social signal processing, and affective computing to interact with the player.

One problem that arises when trying to use such games is obtaining adequate recognition rates in e.g. speech and facial expression recognition that is high enough to facilitate smooth gameplay. To attain this, we use an initial training stage that will allow a game to adapt to the individuals involved. In order to train the speech recognition of a system it is necessary for the user to engage with the program and produce a large amount of spoken dialogue, embracing the vocabulary that the game is likely to need to use. As pointed out by Carr & Blanchfield (2009) the reading ability of younger people with ADHD is often low and so other means have to be gained to get them to talk. Commercial speech recognisers such as Dragon Naturally Speaking^{TM} use the reading of large amounts of text in training. This can be tailored to suit the user but generally is artificial and will lead to a lack of emotion in the training voice used.

The aim of the current game is to produce an attractive introduction to a larger affective game that will be used in the treatment of ADHD patients. The idea of the game was based on the concept behind the spoof game of "Mornington Crescent" as played on the BBC Radio 4 programme "I'm Sorry I Haven't a Clue". The original game has no real rules but ostensibly involves travelling around the London Underground from station to station and attempting to reach Mornington

Crescent or get your opponent to be forced to go there first. The challenge for the user playing the game is to beat a computer opponent to a target station. The approach taken by the non-player character (NPC) is to attempt to get the player to visit enough stations to get a large number of phonemes spoken by the player. At the same time as learning the phonemes the game is also providing challenge and amusement or frustration to the player. Thus it is expected that the phonemes will be spoken in a voice that involves the emotions that will be met in later parts of the game. The name given to the current game is "Swiss Cottage" but in the current game the names of stations used are artificially generated. The names are generated from the set of names available on the London Underground but these have initially been analysed for phoneme content so that the player can be made to say as many of the required phonemes as possible.

The game makes use of a number of components. The user sits in front of a monitor that will display the virtual human as the opponent and a game board is displayed on a tablet computer placed between the player and the computer monitor. The virtual human will introduce the game and allow the user to sign on to a session. The game play proceeds as follows:

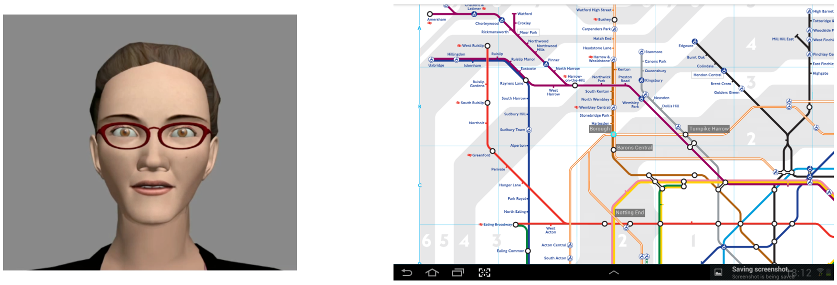
- The virtual human announces the game is ready to play
- An underground map is displayed on the tablet computer
- The player is instructed to go
- The player asks to move to a chosen station
- The game responds:
  - If the request has been understood the game "train" will move to the chosen station
  - If the request is not understood the NPC will request the user to re-enter their choice
  - If the request is illegal the NPC will refuse the request and ask the player for a new choice
- The NPC's turn has been programmed to make moves that will try to win but more importantly try and get the player to have to go somewhere that requires use of previously unused phonemes. The NPC's response involves audio feedback that is meant to make the character seem like it is making choices in order to increase the reality of the process as an interaction with another human player.
- If the player takes too long to make their response the NPC will interact in ways that are designed to provoke an emotional response from the user.

The game records the user speech recognition data in combination with data that relates to the game state that should allow the process of treatment and diagnosis to proceed. The speech can be analysed for emotional response as well as for speech recognition. In addition visual data is being collected that records facial expressions of the player together with dynamic expression data that will be combined with the speech data to allow further diagnosis of player emotion.

Central to the interpretation of the recorded data is the SEMAINE system (*Semaine Project*) which "*aims to build a SAL, a Sensitive Artificial Listener, a multimodal dialogue system which can:*

- *interact with humans with a virtual character*
- *sustain an interaction with a user for some time*
- *react appropriately to the user's non-verbal behaviour*"

A screenshot of the game's virtual human and game board is given in figure 1 below. The virtual human runs on a standard Windows-powered PC and the screesnhot of the game board is from an application developed for the Android operating system.



(a) Virtual human

(b) Game board

Figure 1: Screenshots of the current version of the game

2 Game Development

The current version of the game uses a map based on the actual London Underground map to allow the user immediately find the experience familiar and understandable. Initially it was decided to create the map using a randomly generated game graph that would provide a unique experience each time the game was played. The graph, which would be generated in a mathematical approach, would then be given an attractive and appealing visual identity. This generation of graphs was implemented by making use of an external library (*JGraphT* 2012) although it required extensive configuration to be suitable for graphs in the proposed game. There were a number of constraints on the generated graphs to allow the game to function as intended. The first, *hard*, constraint was that all vertices in the graph must be connected by two or more edges — a "dead end" would mean the player would be unable to make a move as one rule within the game is that players cannot move back to the immediate previous station and thus an unconnected vertex would represent an impossible state. A second, *soft*, constraint was that edges should not allow a game to finish too quickly and starting/destination stations must be sufficiently far away from each other as a game must be long enough to gather useful voice data from the user. Both this restriction and the former, hard constraint were implemented to provide entirely randomly generated graphs. The next stage was to be the "decoration" of the graph so that it looked realistic. This step will be undertaken in future developments. An alternative design implemented in the game involves the allocation of selected station names to a fixed graph adopted directly from the layout of the London Underground but modified for game play requirements; game variation is then achieved by making the start and end stations different.

To generate station names we used a list of London Underground station names as a starting point such that the names wound be familiar and thus believable. However, using the unaltered list of stations would limit the number of stations to the 270 different stations in the Underground network meaning that if the user played two or more games there would be a high chance of them having the same station name more than once. To address this, analysis of the list of station names was performed by breaking down the complete list of station names into three separate lists:

- a list of words from existing Underground stations that can be used as the first word in a station name
- a list of words from existing Underground stations that can be used as the second word in a station name (such as "-on-the-Hill" and "quay")
- a list of Underground station names that can be used, verbatim, without a prefix or suffix (such as "Embankment").

These three lists are then dynamically combined to produce a theoretical limit of nearly 20,000 station names. At the moment the selection of possible words is mostly random with restrictions in place such that there are no two stations with the exactly the same name and, secondly, that no prefix is used more than once in a single game.

The final challenge in creating the station name generator was to implement support for the phonemes in the words. To solve this considerable time was devoted working through the three separate word lists and converting all phrases into their constituent phonemes.

Speech recognition was initially implemented using an external library *Sphinx-4* (2011) rather than the system used in SEMAINE. The latter system has two modes (a high accuracy slow mode and a low accuracy quick more) neither of which would give a good play experience. Sphinx-4 required some modification to be suitable for use in the game, however, as its standard dictionaries did not include all the words used in station names in the game. Sphinx-4 was configured such that it did not rely on a language model but instead was given a sequence of words, called a grammar, that the user was expected to say. These words corresponded to neighbouring stations and other phrases around station names (such as "move to", "go to", "please" and so on).

Integration of the game within the SEMAINE framework required a number of modifications. As already stated a different speech to text programme was used. Other components including rendering and controlling the game avatar as well as the text-to-speech conversion needed for the interaction used modifications of the SEMAINE system. The resulting software integrated feedback from the game with the SEMAINE messaging system to create XML-based '*SwissData*' file of useful information. SEMAINE already used this approach for communication; for example: the results of user's emotion analysis components are represented using Emotional MultiModal Annotation (EMMA) markup language (Baggia et al. 2009) and communication to the speech synthesis and virtual human co-ordination components is done using FML-APML (Mancini & Pelachaud 2008); both of which are XML-compliant. The information that the system needed to collect from the game was chosen to be:

- The performance of the player's last move, which may give an indication of the player's overall mood
- The amount of time the AI takes to respond, because this may be a cause of frustration for the player
- The amount of time that the player takes to make a move, which could help infer the amount of time the player takes to think

The game as presented works well. Limitations occur with communication in noisy environments but this is not thought to be a problem as the expectation is that it will be used in clinical or otherwise controlled environments. Further work has now begun on continuing stages of the game that will begin to look at aspects of diagnosis and of treatment including potential refinement of the game.

References

Baggia, P., Burnett, D. C., Carter, J., Dahl, D. A., McCobb, G. & Raggett, D. (2009), 'Emotional multimodal annotation markup language', `http://www.w3.org/TR/2009/REC-emma-20090210/`. Retrieved on 13th June 2013.

Carr, J. & Blanchfield, P. (2009), A game to aid behavioural education, *in* 'Proceedings of the 3rd European Conference on Game Based Learning', Academic Conferences Limited, p. 78.

JGraphT (2012), `http://jgrapht.org/`. Retrieved on 13th June 2013.

Lis, S., Baer, N., Franzen, N., Hagenhoff, M., Gerlach, M., Koppe, G., Samer, G., Gallhofer, B. & Kirsch, P. (2013), 'Social interaction behavior in adhd in adults in a virtual trust game', *Journal of attention disorders* .

Mancini, M. & Pelachaud, C. (2008), 'The fml-apml language', *Why Conversational Agents do what they do. Functional Representations for Generating Conversational Agent Behavior. AAMAS* .

Semaine Project (), `http://www.semaine-project.eu/`. Retrieved on 10th June 2013.

Sphinx-4 (2011), `http://cmusphinx.sourceforge.net/sphinx4/`. Retrieved on 10th June 2013.