A Generic Platform for Training Social Skills with Adaptative Virtual Agents

Demonstration

Mathieu Chollet
USC Institute for Creative Technologies
Playa Vista, CA
mchollet@ict.usc.edu

Pranav Ghate
USC Institute for Creative Technologies
Playa Vista, CA
pghate@ict.usc.edu

Stefan Scherer
USC Institute for Creative Technologies
Playa Vista, CA
scherer@ict.usc.edu

ABSTRACT

We present a versatile system for training social skills with interactive virtual agents reacting to the user's automatically assessed performance.

KEYWORDS

Multimodal Interaction; Social Skills Training; Virtual Agents

ACM Reference Format:
Mathieu Chollet, Pranav Ghate, and Stefan Scherer. 2018. A Generic Platform for Training Social Skills with Adaptative Virtual Agents. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 3 pages.

1 INTRODUCTION

One promising application domain for virtual agents is the training of "soft" (social or interpersonal) skills, such as the ability to conduct meetings, speak in public speaking, or excel in job interviews, through virtual simulations of relevant social situations. Here, virtual agents are used to enact a social situation that requires to utilize the social skill that is to be trained; this experimental training mode is similar to the role-playing experiments that are used to traditionally train social skills [5]. Such virtual simulations hold a number of potential benefits that could make them particularly suitable for training social skills [7]: (1) they are likely to become cheaper and more available than training with human experts; (2) they are fully standardizable highly customizable to resemble every type of social situation; (3) they lower certain resistances to learning, such as social anxieties related to the fear of being judged [6]; (4) feedback is typically delayed in traditional training: combining virtual agents with automatic behavior assessment also enables real-time, implicit feedback to trainees through virtual humans' behaviors [3].

Several research projects have created virtual agent applications for the training of various social skills. The TARDIS [1] and MACH [4] systems simulate job interviews: both contain an after-action review module where trainees can review their performance along with feedback on their behavior; additionally, the TARDIS virtual recruiters adjust their attitude during the simulations depending on the trainees' automatically assessed performance. The CICERO public speaking training system automatically assesses trainees' performances and provides real-time feedback by adjusting the behavior of an interactive virtual audience [3]. The STRESS project trains aggression de-escalation skills by having users take on the role of virtual tram drivers and confronting them with a intimidating passenger trying to get a free ride [2]. The scenario is influenced by user dialogue choices and emotional states, and textual feedback is presented at the end of the interaction.Creating and integrating such systems for training social skills requires significant amounts of development and integration work. However, as we can see from the above examples, different systems training different social skills involve very similar modules: a behavior perception module (real-time or post-interaction); a performance modelling and assessment module using the user behavior as inputs; a post-training visualization tool displaying behaviors and assessments.

Driven by separate social skills training projects at our institute, we decided to create a common platform that would be shared between different projects. With that architecture, different projects share the same software layers for behavior perception, automatic assessment and after-action visualization tasks, requiring only domain-specific configuration files to be adapted between projects. Only the virtual environments and scenarios remain specific to each project. In this demonstration paper, we describe the architecture of this module and present the configuration steps required to adapt it to a new project. In the conference demonstration session, we will present this module in action with two different projects, a public speaking training application and a medical student training application.

2 SYSTEM ARCHITECTURE

The general workflow for social skills training interactions that is implemented by our platform is the following. First, a variety of sensors track the user's face, gestures, voice, and output raw audio-visual signals. One or several perception modules extracts low level features from these raw signals at a high frequency (*e.g.* facial action units for each video frame; vocal fundamental frequency for audio chunks of 10ms). These features are buffered, and at an opportune time (*e.g.* dialogue turn, or every *n* seconds) a model computes a domain-specific performance output based on these features. This performance is saved for later review in an after-action report, or is used directly to display a feedback message or alter the state of the simulation's virtual agents.

We propose that a number of modules of social skills training applications could be generic, and that reusing them for new applications could only require configuration changes. Our platform consists of several such modules, as presented in figure 1. Specifically,

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

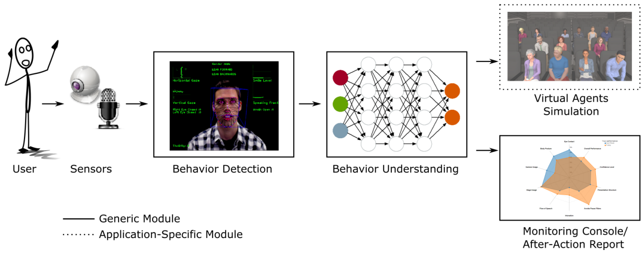

Figure 1: Architecture of the proposed generic platform for social skills training.

those are (a) the *Behavior Detection* module, whose task is to process raw audio-visual signals into detected behavioral signals; (b) the *Behavior Undersanding* module, which produces domain-specific performance outputs depending on the detected user behaviors, and (c) the *Visualization Tool*, which is used for after-action reports or to monitor the trainee's performance in real-time. In order to adapt this general workflow to a specific application, a number of aspects need to be tailored to the particular needs of the social situation to be simulated and to the skill to be assessed.

- *Behavior Detection:* this module can be configured in order to plug sensors in and out, and to disable or enable the processing of behavioral features that are relevant or irrelevant depending on the particular application. We use the publicly available Multisense perception software[1]
- *Behavior Understanding:* this component regularly received behavioral features from the *Behavior Detection* software and produces performance outputs based on them. It requires two crucial pieces of configuration to operate:
  - The time interval between performance output updates, using the behavioral features buffered during the interval. In our platform, these windows can be defined either as sliding windows with a fixed duration, or as variable-length windows, for which we rely on application-specific messages to define window boundaries.
  - Definitions of the models used to process performance score from behavioral features. These can either take the form of hand-crafted rules or of previously trained machine learning models. For the former, we have defined a representation schema with which users can define simple operations on behavioral features. For the latter, the configuration consists of a reference to a previously saved machine learning model, along with a mapping of the model's inputs to the *Behavior Detection* module's output features.
- *Visualization Tool:* this component receives and stores both the behavioral features and performance scores from the above modules. It can be used to monitor the status of the application in real-time, as well as to produce an after-action review tool, including the video from the user enriched with graphs and textual feedback. The module is configured with a file defining the graphs to display and which data source they use, as well as feedback messages to display according to conditions on the user's performance scores.

[1] https://vhtoolkit.ict.usc.edu/

3 EXAMPLE APPLICATIONS

The CICERO system is a public speaking training tool, which includes an interactive virtual audience which adapts its behavior according to the behavior of the user. In this case, assessment windows consist of sliding fixed-duration windows: every 5 seconds, the previous 10 seconds of behavior from the user are assessed and the resulting performance score is used to modify how positive or negative the audience behaves. After the interaction, a report showing the mean performance score of the user along with modality-specific feedback messages.

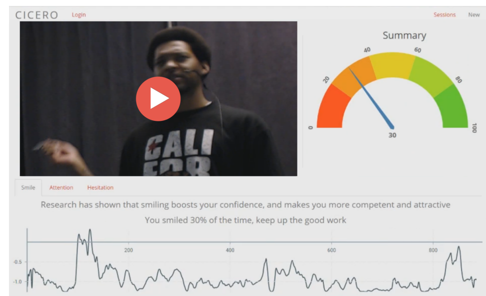

Figure 2: Screen capture of Cicero after-action report.

The PerceptivePatient system is aimed at training medical doctors to foster positive relationships with their patients. The interaction follows a simple question-answer structure: here, the assessment windows are variable-length and are aligned to the user-virtual patient dialogue turns. Different behavior assessment models are defined for the listening phases and the speaking phase, as well as for special "opportunity" instants where emotional information is delivered by the virtual patient, and we expect the user to react adequately (*e.g.* if the patient describes a personal tragedy, show adequate compassion, do not react with joy or contempt).

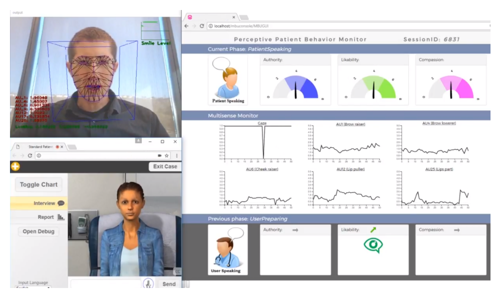

Figure 3: Screen capture of a PerceptivePatient interaction (left) and the behavior monitoring tool (right).

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under grant No. IIS-1421330 and U.S. Army Research Laboratory under contract number W911NF-14-D-0005. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or the Government, and no official endorsement should be inferred.

REFERENCES

[1] Keith Anderson and et al. 2013. The TARDIS Framework: Intelligent Virtual Agents for Social Coaching in Job Interviews. In *Proceedings of International Conference on Advances in Computer Entertainment*. 476–491.

[2] Tibor Bosse and Simon Provoost. 2015. Integrating conversation trees and cognitive models within an eca for aggression de-escalation training. In *International Conference on Principles and Practice of Multi-Agent Systems*. Springer, 650–659.

[3] Mathieu Chollet, Nithin Chandrashekhar, Ari Shapiro, Louis-Philippe Morency, and Stefan Scherer. 2016. Manipulating the Perception of Virtual Audiences Using Crowdsourced Behaviors. In *International Conference on Intelligent Virtual Agents*. Springer, 164–174.

[4] Mohammed Ehsan Hoque, Matthieu Courgeon, Jean-Claude Martin, Bilge Mutlu, and Rosalind W Picard. 2013. Mach: My automated conversation coach. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*. ACM, 697–706.

[5] Dennis R Laker and Jimmy L Powell. 2011. The differences between hard and soft skills and their relative impact on training transfer. *Human Resource Development Quarterly* 22, 1 (2011), 111–122.

[6] Gale M Lucas, Jonathan Gratch, Aisha King, and Louis-Philippe Morency. 2014. It's only a computer: Virtual humans increase willingness to disclose. *Computers in Human Behavior* 37 (2014), 94–100.

[7] Marianne Schmid Mast, Emmanuelle P. Kleinlogel, Benjamin Tur, and Manuel Bachmann. 2018. The future of interpersonal skills development: Immersive virtual reality training with virtual humans. *Human Resource Development Quarterly* (2018).