Mentor System Customisation

Andrew Marriott
Curtin University of Technology
raytrace@cs.curtin.edu.au

Abstract

*This paper details the personalising facilities of the **Mentor System** - a Java-based client-server Dialogue Management system that functions as a mentor to help students in their university studies. The system uses Natural Language Parsing (NLP) along with a state-based Dialogue Manager (DM) and a Knowledge Base marked up using the Virtual Human Markup Language (VHML). The paper outlines the advantages and dis-advantages of this personalisation, especially in a Web-based environment, as well as problems that arose from students' use of the system in a recent case study. The paper also investigates "Personality" personalisations, and reports on integrating a DM with a Talking Head (TH) to create an Embodied Conversational Agent (ECA) - a Virtual Information Provider.*

1. Introduction

An early goal of AI was to emulate human dialogue (Churcher et al., 1997). Seminal work by Turing (1963) stated that if we could not tell the difference between the responses from a computer vs. a human, the computer could be said to be thinking (Turing test). Weizenbaum (1966) developed an effective but naïve system – Eliza – but it did not really model the dialogue. Chatterbots are an improvement because they use more tricks and have better memories of past events and conversations (Maes, 1995). Intelligent Tutoring Systems (ITS) (Mark and Greer, 1993) (CIRCSIM, 1996) can use the dialogue to help students in their studies.

2. The *Mentor System* Architecture

The ***Mentor System*** (Figure 1) embodies a:

- client-server system and protocol,
- DM plus ITS,
- Client interface: text as well as GUI based,
- VHML compliant Knowledge Base (KB)
- suitable pedagogies that help the students learn.

Each user connects to the central server through the network via a ***Mentor*** client. The ***Mentor*** server - a Java based mini operating system in its own right - is about 70,000 lines of code in 170 classes in 50 packages.

The typical ***Mentor*** client uses the Unix password file to classify the user as an undergraduate or postgraduate student, academic staff, technical staff or support staff. Depending on this classification, a different Java *MentorClient* class will be instantiated to customise the look and feel and/or functionality of the interface to the server(Figure 3). For example, academics can have more functionality on their GUI menubar to help manage the students in their classes, technical staff can have an extra "status" textarea for the reporting of system problems such as machines going down, or printer problems. The classification could be used to instantiate an ECA, a TH or Virtual Human (VH) instead of a GUI.

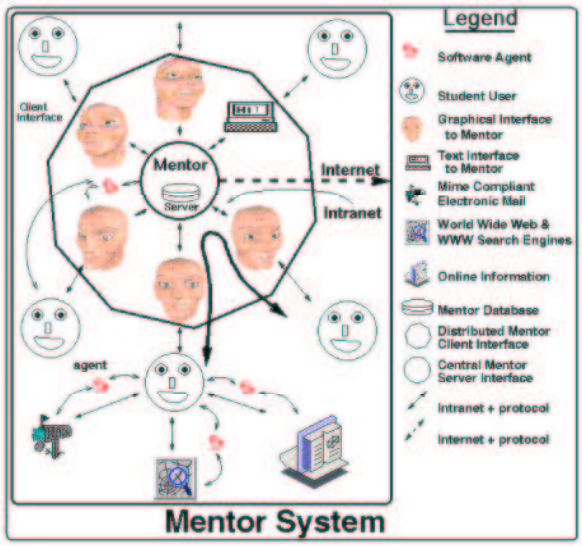

Figure 1 The *Mentor System*

In the reported case study, the ***Mentor*** client is **not** an ECA or TH but simply a text-based GUI similar to the Unix Talk program. The system has used a TH client (Gustavsson et al., 2002) but the extra audio/video output is not always appropriate in a laboratory environment. The results of this study though, are applicable to the TH or Virtual Human (VH) scenario.

A user can also specify a classification when they start the client; this is always restricted so that students cannot get more "privileges" or get a GUI that is beyond their current experience level. The classification, along with other environmental information, is sent to the server to aid further DM customisation (e.g. client type).

On the server side, each user gets a separate DM interface to ***Mentor,*** also customised via their Unix username or via a unique username/password if they connect via the web. This user **Context** is automatically created and develops over time as the user interacts with the DM. Because of this, the interface environment they

```
<first_name/>, <welcome/>. I am <mentorName/>. <smile>I was developed by
<mentorMaster/>.</smile><mentorDescription/> <mentorPurpose/>
<pause length="long"/>You can find out about me from <mentorHomeURL/>.
```

Figure 2 Segment of marked up text showing the use of VHML tags

experience is unique, dynamic and temporally continuous. Trivially, for example, the system knows about their full name as well as what they like to be called. Similarly, it knows the Unix usernames (and hence names) of their friends. This customisation makes the HCI experience more humane to users.

When a user connects, the system greets them in a non-deterministic fashion, occasionally using various parts of the user's full name. An actual but contrived dialogue (Figure 3) shows some of the system functionality including an initial pro-active question about jokes. The top lines are user input and the ***Mentor*** responses are shown in the lower part, which can display HTML information.

The heart of the DM is a loop that checks what each connected user has typed, and what they have typed in the past, against a list of ***Mentor*** topics. These topics hold information about such things as the time, weather, who is using the system and their names, politeness and rudeness, as well as learning topics for specific units to help students in their studies. These topics are Java classes that match the user input via Perl5 Regular Expression's and return ranked responses marked up using VHML (Marriott, 2002). VHML is an XML/XSL based language developed to facilitate the realistic and natural interaction of a VH with a user. It can add emotional and gestural effects to spoken dialogue through markup tags, and these increase the believability of the interaction (JRPIT, 2001b).

3. Personalised Responses

The use of tags can aid in customising the response appropriateness. For example, the text of Figure 2 may be the VHML response to the user enquiry "What are you?". The name of the user who is making the enquiry is represented by the VHML tag `<first_name/>`, `<mentorName/>` is the name of the DM, `<welcome/>` is a language dependant greeting that has been set using the user's home country or domain name, etc. It is also possible for users to alter this language (stored in their **Context)** by telling ***Mentor*** in plain English "I am Swedish" or "I come from Italy". This does not change the normal dialogue language, only various aspects of it such as "hellos", "good evenings", "I do not understand", etc (see http://www.elite.net/~runner/jennifers/).

In this example, extra VHML tags (`<smile>` and `<pause>`) have also been included. So the response may be transformed into the final plain text:

Freda, Guten Tag. I am Mentor. I was developed by Andrew Marriott. ….

It should be noted that at a more abstract level, the use of VHML even aids in personalising the interface. The response markup should stay the same even if the way in which it is displayed changes:

- a GUI text interface as shown,
- an interactive Web based display,
- voice TTS only - lets the user hear the answer,
- a Talking Head with complex facial gestures, voice, personality and emotion,
- a Virtual Human with body language.

The same response output to a VH system may cause it to smile, point, open up a Web browser, load the `<mentorHomeURL/>`, etc. Future ***Mentor***-based systems (e.g. Virtual Information Provider) can use the existing knowledge base to create believable THs.

As shown in Figure 3, the system is normally used by students to ask questions along the lines of "help me with xxx", where xxx is something to do with the unit, and most typically to do with an assignment problem. Also, the user is "looked after" – the system pro-actively asks the user if they want to hear a joke, have started the assignment, want any help with the assignment, etc. For extra user-customisation, the system can also load user-developed and additional specialist topics. So, for example, a user environment that has specific domain knowledge about MPEG-4 or Italy could be developed and used dynamically.

4. Personalised Response Problems

Many of the student users did not have English as their first language (47%) and since the system tries to match against correct English, if users do not use this, then it poses a serious problem that even the generality of Regular Expression pattern matching often cannot overcome. The system can cope with standard abbreviations like the first entry below but has trouble understanding the intent of the second due to the poor English:

```
how r u
yap! waht to do u idoit?
```

Although outside the scope of ***Mentor,*** machine learning may be able to remedy this problem.

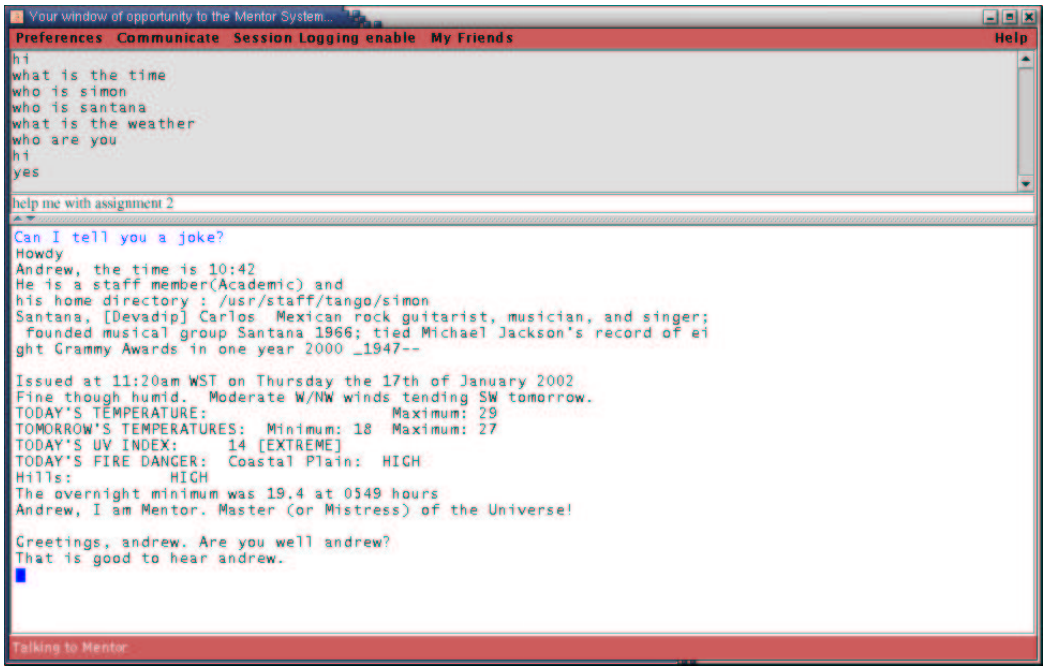

Figure 3 The *Mentor* Client Interface

Personalised parsing of user input could be catered for based on the nationality of the user by ignoring words with no semantic meaning:

aaargh! it's not working
erh! i'm not miss **leh!**
hheeheee! dun wan to tell u **leh!**
yeas! i really luv u **leh!**
"nothing **la**" means i have nothing to say
makan u lah! day!

The first example shows a typical English word that can be discarded, the next 3 show the use of "la" language common in Singapore and Malaysia where the "la" or "leh" is just added, often at the end of sentences, with no semantic meaning. The fifth example shows how a student is trying to help the system by indicating what "la" means. Unfortunately, as an AI system becomes more effective in understanding, users expect even more from it. The last example shows multi-language user input!

This usage highlights a significant problem that emerged in this study: many students, in talking to ***Mentor***, would indicate their nationality and hence store their preferred greeting language in their **Context**. The next time they were greeted, students believed that the system understood their entire language! This was compounded by a system design error that used greetings like "ciao" and "ca va?". So, many users asked "can you speak XXX?" where XXX was French, Italian, Chinese, Japanese and Indonesian. Worse still, users also simply used foreign language phrases expecting the system to understand. In total, users entered 70 foreign language phrases, some incorrect, some mixed language.

Some typical input being:

apa kabar?	apa tuh?
selamat pagi	loe suka apa?
kasar loe	nasi lemak
ni how man?	ni ke yi jiang hua yi, man
wo ai ni	wo bu ming bai yi wei
wo he hao	jiang hua yu
Sayonara	sayonara means bye
au revoir	bona petite
cava?	comment appellez vous?
quelle heure est-il?	quelle temps est-il?
vous parlez francais	
tu italiano	capisco
come stai?	ciao, im good!
come se llama, est	puebloe?
dev dave	dev hei
dev hoho	dev oi oioi
man mpage	

For the author, the first 3 double column entries are recognisably Indonesian, although `nasi lemak` is in fact just a popular food dish. The next 3 are Mandarin and/or Cantonese (many students are from Singapore, Malaysia or Hong Kong). The next is "English" Japanese and the following 4 are French (or nearly so). Both French and Italian (the next 3 lines) were very popular because they were featured as part of the greetings. The author is not certain about the phrase concerning `llamas`, it even has a Spanish ring to it!! The next two are unclassifiable by the author although they could be Italian, pidgin Italian or just plain garbage. The last, given the preceding user input context could have been a foreign phrase, but was in fact just a request for the man page for a Unix utility!

The personalisation of the responses to cater for nationality raised user expectations about the system's ability to understand and these could not be met. This was a short term disappointment for students but had a long

term benefit in that students started to realise that the system was not "all powerful" and grounded their expectations accordingly. It also became apparent from user input that the students were thinking about the problems of Natural Language Parsing and Dialogue Management:

```
do you really have to comb through all this stuff?
  oh to be a mentor....
```

This inner reflection was seen as a positive outcome of the system research. Also, when asked what they found wrong with the system, formal qualitative evaluation responses indicated,:

```
…It didn't know everything, but thats impossible
  anyway.
Small knowledge base? But can't really fault it much
  since it's not human.
```

5. Personalisation through Names

Over 50% of the students took advantage of the system's ability to change the user's preferred output name. That is, customising the contents of tags like `<first_name/>`. Many names were just "nicknames", but some were corrections to the names data-mined from password files (for example Mary Jane Smith may have preferred to be called Jane rather than Mary):

```
my preffered name is tim, not timothy
```

Of importance in a multi-cultural society like Australia where there are many Asian students, the personalising of an interface through the use of names becomes a problem for two reasons: the gender of names and also the placement of the "family name" in the entire name.

To categorise the gender of a user, the system currently uses a compiled list of probabilities of typical names vs gender. This fails on several common Anglo-Saxon names but is quite accurate overall. It does **not** cater for non-Anglo-Saxon names and with 30-50% of our users being non-Anglo-Saxon, this can cause embarrassing mistakes:

```
don't call me miss
i am a male!
```

Users can say "I am a mr" or "I'm female", etc to change the system's categorisation of their gender.

Many cultures swap the first-name, family-name order in their names and this is especially true on many Asian cultures. To further complicate this issue, many students from Asian cultures who are aware of this or who have a strong Anglo-Saxon background (such as Singapore, Hong Kong), will automatically adjust their formal enrolment name to conform to Australian custom. This means that the password file will contain name information in both formats and hence a family-name may be used as a first-name. Students can tell the system their first-name, middle-name, family-name, etc to clarify this.

The ***Mentor System*** attempts to provide an informal, relaxed environment conducive to learning and the use of first or preferred names when conversing with users helps in this. The incorrect use of a last name in place of a first name is often seen as rebuking, or as a sign of aggression or anger especially if used by a Talking Head and hence should be avoided. Inspection of the 31000+ inputs to the system for this study indicated that very few Asian students used the name clarification facility. So this may not be a big personalisation problem for educated users.

6. Personalisation through Personality

The formal evaluation done at the end of the case study about aspects of using ***Mentor*** elicited the following response from one individual:

Comments on the most annoying aspect :

```
Not that courteous. ie it will be good to say
"see you around" or "nice talking to you"
instead of "bye" <- I find bye offensive
```

The current "programmed" personality of the DM, that is, the tone of the language used in the responses of the topics, tries to be informal and friendly. For some reason, the above user has not seen the language as friendly but has found it to be most annoying. However, other feedback has suggested that some users prefer a **more** formal response from the system, more in keeping with that associated with an academic authority. Therefore, the overall personality of the DM must dynamically match the expectation of the user if the system is to be effective.

The current version of VHML (www.vhml.org) specifies that the `<person>` tag may indicate various optional aspects of the DM (or the TH that is rendering the DM output) such as the age or gender as well as which emotion it is supposed to use by default for the rest of the element whenever there is no other emotion tag specified.

It has been argued (Marriott and Stallo, 2002) that another attribute – "*personality*" – must also be included. This is seen as necessary for proper dynamic personalisation of ECAs or the Virtual Human interface (Ruttkay et al., 2002) (Poggi et al., 2001; Brna and Cooper, 2001).

Personality Theory (Pervin, 1989) is a major field of study in psychology used to describe individual attributes of people. The Big-Five Model (Norman, 1963) is trusted in the field of psychology, and is adaptable to VHML. The five factors are extraversion, neuroticism, conscientiousness, agreeableness and openness, with each trait representing a range of values, for example "agreeableness" represents a range of values from agreeable to disagreeable. The five factors have been found to give stable and valid results with different observers, instruments and across different adult ages (McCrae, 1988). This is a strong argument for use of the Big Five model, particularly in combination with the ease in which emotions can be applied to the five personality traits. These traits can be described using natural language, making it simpler for a user to define and select a personality.

The specification of the long-term temporal dynamics of a personality (what happens when the VH gets bored, is constantly harassed, is manic-depressive, etc) also needs to be addressed.

No formal research has yet been done on **how** a user may specify the preferred DM personality although a basic "friendly, helpful, consistent" one is assumed to be the most effective in a learning environment. Informal analysis has indicated that many Asian students come from a learning culture which is more "directive" than "explorative" - they prefer to be told **exactly** what to do rather than be shown possibilities and left to think about possible solutions. This broad categorisation could be used initially to specify a base personality.

Some initial research has been done at Curtin University in analysing how to specify personalities, as well as the more important procedure of dynamically modifying the personality to best suit the user.

7. The Virtual Weather Woman

Dam (2002) describes the implementation of a TH interface application developed at Curtin: a Virtual Weather Woman (VWW). The VWW is able to reply to text input via a dialogue management tool using speech combined with emotions. The delivery of the weather information is improved by being personalised to suit each individual's profile.

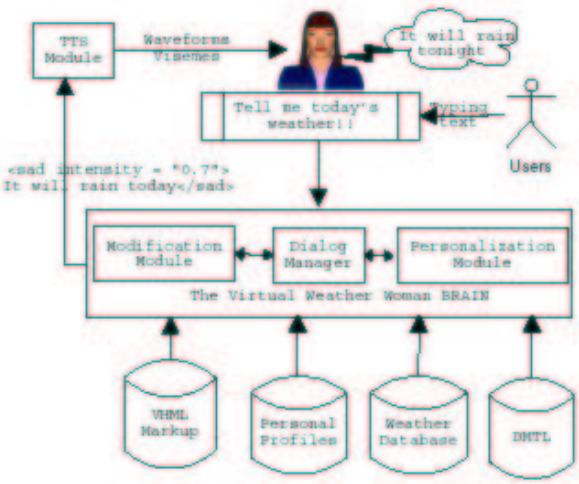

Figure 4 Virtual Weather Woman Architecture (from Dam 2002)

The VWW is Web-based with a VHML-based DM (see Figure 4), and functions as a weather presenter. Users can interact with her using plain text and she can respond to their weather requests or give suitable advice on what a user could do given certain weather conditions.

The data-mined weather information is marked up dynamically using emotions appropriate for the user. While interacting with a user, the VWW gathers information : i.e. it learns the user's emotions and attitudes towards a given weather condition. This information is then used in future dialogues with that user when expounding upon the same weather conditions.

The VWW starts out with default emotions for the initial user profile; the personality of the VWW is bland. Through dialogue, the VWW learns the user's attitude by giving weather information and then asking questions about that type of weather. For example:

- *Do you like rain?*
- *You don't like rain, do you?*
- *Do you like hot weather?*
- *Would you enjoy it being sunny?*, etc.

Based on this interaction, the VWW will attempt to predict the user's attitudes towards various weather conditions and change her/his profile. A simplified profile is shown in Figure 5 - obviously, the reality of emotions and weather is more complicated. If the VWW observed that the user dislikes rain, then the sad emotion is added to the profile. At the same time, the VWW adjusts the level of sadness in order to assign an accurate intensity value for the VHML **<sad>** tag. The degree of emotion is justified by observing adverbs of the user's answer such as *very much, a lot, extremely, fairly, a bit,* etc.

The user profile is continuously modified to reflect the most updated attitude of the users. For example, a user may not like rainy weather in summer, but love it in winter. The user profile also records advice on what to do under certain weather conditions.

```
<person name="Helen" country="Australia"
                     city="Perth">
  <pref>
    <temp>C</temp>
  </pref>
  <records
    <weather type="rain">
      <emotion intensity="0.6">
        sad
  .   </emotion
      <advice>Listen to the music </advice>
      <advice>Do homework </advice>
    </weather?
  </records>
</person
```

Figure 5 Person Profile.

This initial work is guilty of still equating emotion with personality. Emotions are short-term responses to an event, whereas a personality is a consistent description of the sorts of reactions a person may have over time. The personality of an individual will affect what emotions they perceive and display.

The use of emotions is naïve but works for users at a superficial level and is effective in personalising the interface. However, it denies the instantiation of an ECA that has its **own** personality that may be modified when

interacting with users who also have personalities. A mature ECA should exhibit its own personality traits, and through interaction modify them, not just reflect or model those personalities around it.

8. Future Work

A Virtual Information Provider (VIP) TH has also been implemented that enables an interactive Web-based dialogue on Talking Head Technology, VHML, face and body animation, etc. It is planned to include mature personality features into this VIP.

Future plans also include a Virtual Lecturer (***Mentor*** + TH) that is capable of managing user interactions and providing spoken multi and hyper media answers to their requests for learning help. The system will also aim to test the effectiveness and believability of a TH as a teacher and as a humane interface to information.

9. Conclusions

The users of the ***Mentor System*** saw its one-on-one help as being effective and beneficial. The personalisations implicit in its design enabled a more informal and friendly interface conducive to learning but these also caused unforeseen problems as user expectations rose.

Problems also arose with personality conflict between the system and some users and it is believed that these can be reduced or eliminated through the use of a dynamically adjusting personality module layered on top of each user's interface to the DM.

The ***Mentor System*** has met its design aims and students eagerly await its use in other units. The system does not provide a perfect solution but is one more step towards helping students to learn more effectively.

10. References

Bloom, B. S. (1984) The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring, *Educational Researcher,* Vol. **13,** pp. 4-16.

Brna, P. and Cooper, B. (2001) Marching to the Wrong Distant Drum: Pedagogical Agents, Emotion and Student Modelling., *Second Workshop on Attitude, Personality and Emotions in User-Adapted Interaction*, Sonthofen, Germany.

Churcher, G. E., Atwell, E. S. and Souter, C. (1997) School of Computer studies, University of Leeds, Leeds.XXXX.

CIRCSIM (1996), *CIRCSIM-Tutor Project* Illinois Institute of Technology, Dept. Of Computer Science and Rush College of Medicine, Dept. of Physiology, http://www.cs.iit.edu/~circsim/.

Dam, H. and Souza, S. d. (2002) Applying Talking Head Technology to a Web based Weather Service, *Virtual Conversational Characters Workshop, Human Factors 2002*, Melbourne. http://www.vhml.org/workshops/HF2002/papers/dam.

Fletcher, J. D. (2003) Does this stuff work? A review of technology Used to Teach, *TechKnowLogia,* Vol. **5(1),** pp. 10-14. www.TechKnowLogia.org.

Gustavsson, C., Strindlund, L. and Wiknertz, E. (2002) Verification, Validation and Evaluation of the Virtual Human Markup Language (VHML), *Institutionen för Systemteknik*. http://www.ep.liu.se/exjobb/isy/2002/3188/.

JRPIT (2001b), *Experiment details for "The Face of the Future"*. Online at http://www.interface.computing.edu.au/papers/jrpit-hci/html/ and http://www.interface.computing.edu.au/papers/jrpit-hci/word-docs/.

Maes, P. (1995) Artificial Life meets Entertainment: Interacting with Lifelike Autonomous Agents., *Special Issue on New Horizons of Commercial and Industrial AI,* Vol. **38 (11),** pp. http://pattie.www.media.mit.edu/people/pattie/CACM-95/alife-cacm95.html.

Mark, M. A. and Greer, J. E. (1993) Evaluation Methodologies for Intelligent Tutoring Systems, *Journal of Artificial Intelligence and Education,* Vol. **4,** pp. 129-153. ftp://ftp.cs.usask.ca/pub/aries/papers/mark-jaied.ps.

Marriott, A. (2002) A Facial Animation case study for HCI: the VHML-based Mentor System, In *MPEG-4 Facial Animation - The standard, implementations and applications*, (Ed, Forchheimer, R.) John Wiley, London.

Marriott, A. and Stallo, J. (2002) VHML- Uncertainties and Problems... A discussion, *AAMAS workshop on Embodied conversational agents - let's specify and evaluate them!*, AAMAS 2002, Bologna, Italy.

McCrae, R. R. (1988) Why I Advocate the Five-Factor Model: Joint Factor Analyses of the NEO-PI with Other Instruments, In *Personality Psychology Recent Trends and Emerging Directions*, (Eds, Buss, D. M. and Cantor, N.) Springer-Verlag, New York, pp. 237-245.

Norman, W. T. (1963) Toward an adequate taxonomy of personality attributes, *Journal of Abnormal and Social Psychology,* Vol. **66,** pp. 574-583.

Pervin, L. A. (1989) *Personality Theory and Research, Fifth Edition,* John Wiley & Sons, New York.

Poggi, I., Pelachaud, C. and Carolis, B. D. (2001) To Display or Not To Display? Towards the Architecture of a Reflexive Agent, *2nd Workshop on Attitude, Personality and Emotions in User-Adapted Interaction in conjunction with User Modeling 2001*, Sonthofen, Germany. http://aos2.uniba.it:8080/list-ws-um01.html.

Ruttkay, Z., Dormann, C. and Noot, H. (2002) Evaluating ECAs – What and how?, *AAMAS workshop on Embodied conversational agents - let's specify and evaluate them!*, AAMAS 2002, Bologna, Italy.

Turing, A. (1963) Computing Machinery and Intelligence, In *Computers and Though*, (Eds, Feigenbaum, E. and Feldman, J.) McGraw-Hill, NewYork, pp. 1-35.

Weizenbaum, J. (1966) Eliza, *Communications of the ACM,* Vol. **9,** pp. 36-45.