Constructing Personality Model from Observed Communication

Ivan Kopeček
Faculty of Informatics,
Masaryk University Brno
Botanická 68a, 602 00 Brno, Czech Republic
E-mail: kopecek@fi.muni.cz

Abstract

In this paper we deal with deriving a personality model from a corpus of observed dialogues. We use finite state analysis based on a slightly generalized notion of personality model. The approach is illustrated by a simple example of agents playing the iterative prisoner's dilemma game.

1. Introduction

In what follows, we analyse the following problem: having a corpus of dialogues, can we find an algorithm constructing a personality model simulating the behaviour of the agent? The analysis is based on a finite state automata (FSA) model of personality, which slightly generalizes the models presented in (Gmytrasiewicz & Lisetti, 2001) and (Kopeček, 2001).

We use standard terms and notation of the algebraic theory of formal languages and automata.

M^* denotes the free monoid over the set M, i.e. the set of all strings consisting of the elements of the set M (including the empty string).

An *alphabet* is a finite nonempty set. If M is an alphabet, any subset of M^* is said to be a *language*. By *corpus*, we understand a finite language. If U and V are sets, $U \times V$ denotes the Cartesian product of U and V.

2. FSA Models – Pros and Cons

Finite state analysis uses algebraic models that can be applied to a wide variety of the problems related to human-computer interaction, user modelling, dialogue systems, and many applications. Generality of the approach and the possibility of exploiting many disciplines of mathematics and artificial intelligence makes it applicable even for complex problems like modelling and investigating emotions and personalities. Nevertheless, the application of the simple structure of finite automata to modelling extremely complex structure of human personality emotions may evoke a scepticism regarding the plausibility of this approach. Let us mention a few reasons why and when we consider this approach advantageous.

First, HCI needs technological solutions that involve modelling personality. This implies the necessity of using formal models. Between the possible formal models, FSA approach, both deterministic and stochastic, plays a central role being enough simple to be well mentally mastered and simultaneously having the potential to express and model very complex instances of the real world.

Second, FSA are closely related to the VoiceXML standard (see VoiceXML Forum), which is suitable for direct applications of the FSA models to programming dialogue systems.

Third, FSA are extensively investigated in other relevant fields, particularly in AI and in the game theory (see, e.g., Abreu & Rubinstein, 1988, Piccione & Rubinstein, 1993). This is giving further potential to the FSA approach in the future research.

On the other hand, it is clear that FSA analysis will not cover all the problems related to personality and emotion research. Presumably, in many cases it will depend on our ability to restrict general complex problems to more simple or domain-oriented instances, plausible to the FSA structure.

Second, we may meet some typical problems that brings FSA approach, like the complexity problems related to the state explosion effect, or need of too many states to get an adequate model, etc.

3. Modeling Personality

A *personality model* is an ordered quadruple $P = (Q, X, \lambda, s_0)$, where

- Q is a finite nonempty set of emotional states,

- X is a finite nonempty set of input and output symbols (modelling environmental inputs and output reactions),

- $\lambda: Q \times X \times X \times Q \rightarrow [0, 1]$ is a probabilistic transition function determining the probability of the transition from a state into another one when accepting an input symbol and outputting an output one,

- s_0 is an initial emotional state.

In the deterministic version of personality model the next state and output symbol are uniquely determined by the current state and the current input symbol (i.e., all probabilities are equal to 1).

Here we slightly generalize the personality model presented in (Gmytrasiewicz & Lisetti, 2001) and in (Kopeček, 2001). Let us mention, that learning algorithms (Carmel & Markovitch, 1996, Sandholm & Crites, 1995) can be applied to this model.

Example 1.

This example presents a personality model of a Tit-for-Two-Tats agent, for a communication in iterative prisoner's dilemma game (see, e.g., Sandholm & Crites, 1995, Osborne & Rubinstein, 1999). The example is based on (Gmytrasiewicz & Lisetti, 2001). Here, the model is generalized in the sense that we take into account also output reactions of the user. For the sake of simplicity, we consider the deterministic model.

In our example, the set of emotional states Q , the set of input and output symbols X and the initial state s_0 are defined as follows:

Q = {COOPERATIVE, SLIGHTLY-ANNOYED, ANGRY}

X = {cooperative, uncooperative}

s_0 = *COOPERATIVE*

λ	*C*	*S*	*A*
c	*(C, c)*	*(C ,c)*	*(C, c)*
n	*(S, c)*	*(A, n)*	*(A, n)*

Table: Definition *of* λ

(we use the following notation in the table: *C = COOPERATIVE, S = SLIGHTLY-ANNOYED, c = cooperative, …* etc.)

Example 2. Let us have the following sequence:

(c, c), (n, c), (n, n), (c, c), (n, c), (c, c)

which can be seen a dialogue between two agents *A* and *B; A* is related to the first position in a pair and *B* to the second one. Here, "*c*" stands for "cooperative" and "*n*" for "uncooperative". This dialogue can be seen as an iterative prisoner's dilemma game.

We can see that that agent *B* behaves in accordance with the personality model presented in Example 1 The model presented in the example represents one of the possible personality models that are consistent with the communication presented above. However, this model is not the only one with this property. What model should we choose and how can we construct it? Can we assign, in a reasonable way, a personality model to a dialogue or to a set of dialogues?

Generally we get an infinite number consistent personality models to any given finite set of dialogues. To specify the right model, we must add some additional conditions. The first condition is minimality of the states, because redundant states uselessly increase the complexity of the model. The second condition expresses that the behaviour that is not indicated in the data, should not be, wherever possible, allowed. Otherwise we could simply take the trivial one-state model that allows everything.

4. Deriving Personality Models

Instead of analysing personality models directly in the form of Mealy automata, we can use a more convenient form of deterministic finite automata. This type of automata differ from (deterministic) Mealy automata in that they do not produce any output. Some of their states are called final and serve for recognizing (accepting) strings. The set of accepted words is called the language generated (recognized) by the automaton. For details see e.g. (Kozen, 1997, Gecseg & Peak, 1972). Partial deterministic finite automata with all states final are suitable for a formal simplification of our problem.

Prefix automaton is a partial deterministic finite automaton with all states final. Simple analysis shows that our problem can be now reformulated as follows: Having a dialogue corpus *C* find a minimal prefix automaton generating a language, which is *prefix-similar* to *C*. (Prefix similarity, see Kopeček & Škarvada, 2002, for exact definition, says that if *L* is prefix similar to *C* then *L* must contain all prefixes of all elements of *C* and it simultaneously must contain all postfix extensions of all elements from *C*.)

Finally, all that remains to be done is a backward transformation to Mealy automata and assigning consistent probabilities to the (nondeterministic) transition function.

The algorithm that derives a personality model from a corpus of dialogues proceeds in the following steps.

1. Construct a deterministic finite automaton generating exactly *C*. The most straightforward construction is to take a *trie automaton T* (trie automaton is a partial deterministic finite

automaton generating precisely the corpus *C*, whose diagram is a tree).

2. Define relation σ on the set of states of *T* with the following property: any equivalence relation $\sigma \subseteq p$ determines the corresponding quotient automaton T/σ whose language $L(T/\sigma)$ is a postfix extension of *C*.

3 Modifying the quotient automaton by making all its states final gives us a prefix automaton that generates a language prefix-similar to *C*. This can be done analogously to the deterministic case (see Kopeček & Škarvada, 2002, for details).

4. We transform this prefix automaton to (non-deterministic) Mealy automaton, i.e. we get the non-deterministic version of the personality model.

5. We assign probabilities to the (non-deterministic) transition function, obtaining the formal structure of the personality model. This can be done by evaluating the probabilities directly from the given corpus *C*.

Let us mention, that if we apply our algorithm to Example 2, we get a personality model with only two states. In the first state, the agent always responds cooperatively. In the second state, the agent responds cooperatively only to cooperative behaviour and switches to the first state, otherwise it responds uncooperatively and stays in the second state.

We can assign the meaning "cooperative" to the first state and "conditionally cooperative" or "annoyed but ready to cooperate" to the second one. This model, in comparison to the model presented in Example 1, shows the same external behaviour, but its internal behaviour differs – it uses the same strategy but it "stays calm".

5. Conclusions

In the paper, we have outlined a method for getting a personality model directly from the observed data. An interesting problem is how to interpret the given formal structure, i.e. how to assign some semantics to the formally obtained emotional states. It is probably not possible, at least not in current stage of emotion research, to develop a universal method, and the use of appropriate heuristics can be expected.

On the other hand, we can try to get maximum information from the obtained model. For the probabilistic model we can see an emotional state as a random variable and compute its entropy, or other probabilistic characteristics. Provided we know these characteristics for "real" emotions, we could try to assign them for the abstract ones Applying this within dialogue systems and further theoretical research are future goals.

6. Acknowledgement

The research has been partially supported by Ministry of Education of the Czech Republic under the Grant LI200027.

7. References

Carmel, D., Markovitch, S. (1996). Learning Models of Intelligent Agents. *Proceedings of the National Conference on AI*, Portland, pp. 62-67.

Abreu, D., Rubinstein, A. (1988). The Structure of Nash Equilibrium in Repeated Games with Finite *Automata. Econometrica* 60, pp. 1259-1281.

Gecseg, F., Peak, I. (1972). *Algebraic Theory of Automata.* Akademiai Kiado, Budapest .

Gmytrasiewicz, P.J., Lisetti, Ch.L. (2001). Emotions and Personality in Agent Design and Modeling", in *Proceedings of UM 2001 Workshop on Attidudes, Personality and Emotions in User-Adapted Interaction*, Sonthofen.

Greenberg, J. (1997). Situation Approach to Cooperation". in *Hart. S, Mass-Colell, A. (Eds.): Cooperation: Game-Theoretic Approaches*. NATO ASI Series, Springer Verlag.

Kopeček, I. (2000). Emotions and Prosody in Dialogues: An Algebraic Approach Based on User Modelling. In *Proceedings of the ISCA Workshop on Speech and Emotions*. Belfast, pp. 184-189.

Kopeček, I. (2001). Personality and Emotions – Finite State Modelling by Dialogue Automata. In *Proceedings of UM 2001 Workshop on Attitudes, Personality and Emotions in User-Adapted Interaction*, Sonthofen.

Kopeček, I., Škarvada, L. (2002). Modeling Dialogue Systems by Finite Automata. In *Proceedings of ICON 2002 Conference*, Vikas Publishing, pp. 241-249.

Kozen, D.C. (1997). *Automata and Computability*, New York, Springer.

Osborne, M.J., Rubinstein, A. (1999). *A Course in Game Theory*. MIT Press, London.

Piccione, M., Rubinstein, A. (1993). Finite Automata Play a Repeated Extensive Game. *Journal of Economic Theory* 61, pp. 160-168.

Rapaport, A. (1987). *Prisoner's Dilemma.* In The New Palgrave, Vol. 3, Macmillan, pp. 973-976.

Sandholm, T., Crites, R.H.(1995). Multiagent Reinforcement Learning and Iterated Prisoner's Dilemma. *Biosystems Journal*, 37, pp. 147-166.

Shubik, M. (1982). *Game Theory in the Social Sciences*. Cambridge, Mass. MIT Press.

VoiceXML Forum. Voice Extensible Markup Language VoiceXML. http://www.voicexml.org/