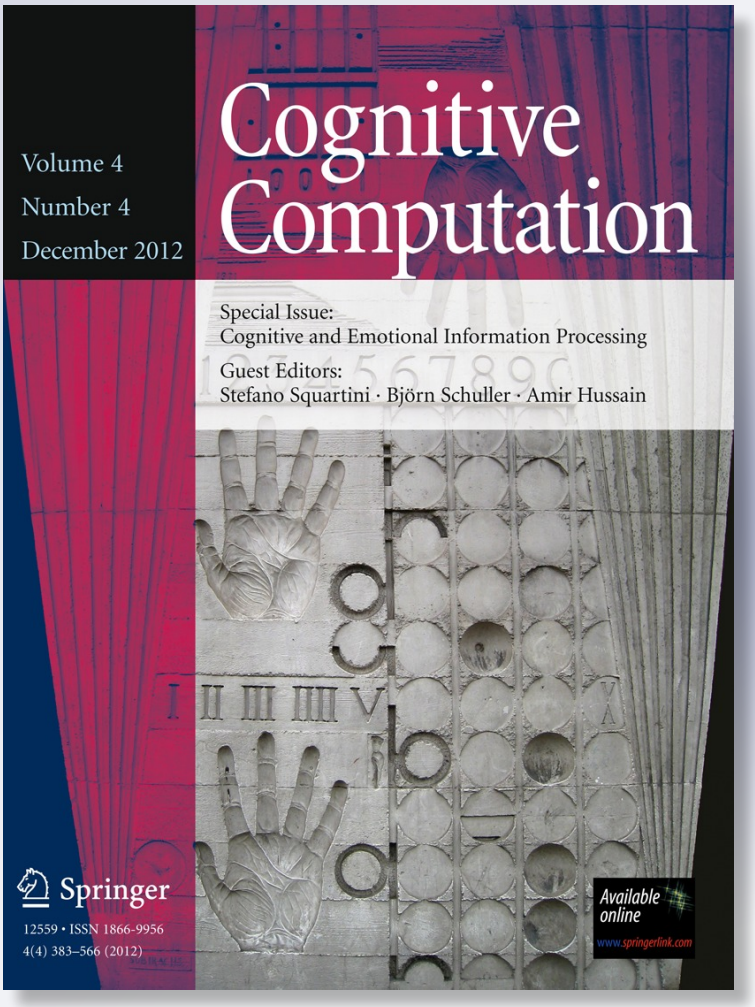
Affect Sensing Using Linguistic, Semantic and Cognitive Cues in Multi-threaded Improvisational Dialogue

Li Zhang & John Barnden

Cognitive Computation

ISSN 1866-9956
Volume 4
Number 4

Cogn Comput (2012) 4:436-459
DOI 10.1007/s12559-012-9170-3



Springer

Cogn Comput (2012) 4:436–459
DOI 10.1007/s12559-012-9170-3

Affect Sensing Using Linguistic, Semantic and Cognitive Cues in Multi-threaded Improvisational Dialogue

Li Zhang · John Barnden

Received: 1 August 2011 / Accepted: 10 July 2012 / Published online: 19 July 2012

Abstract In this paper, a context-based affect detection component embedded in an improvisational virtual platform is implemented. The software allows up to five human characters and one intelligent agent to be engaged in one session to conduct creative improvisation within loose scenarios. The transcripts produced showed several conversations being conducted in parallel. Some of these conversations reveal personal subjective opinions or feelings about situations, while others are caused by social interactions and show opinions and emotional responses to other participant characters. These two types of conversations serve to inform the descriptions of the *personal* and the *social* contexts, respectively. In order to detect affect from such contexts, first of all a naïve Bayes classifier is used to categorize these two types of conversations based on linguistic cues. A semantic-based analysis is also used to further derive the discussion themes and identify the target audiences for the social interaction inputs. Then, two statistical approaches have been developed to provide affect detection, respectively, in the social and personal emotion contexts. The emotional history of each individual character is used in interpreting affect relating to the personal contexts, while the social context affect detection takes account of interpersonal relationships, sentence types, emotions implied by the potential target audiences in their most recent interactions and discussion themes. The new development of context-based affect detection is integrated with the intelligent agent. The work addresses one challenging cognitive topic in the affective computing field, the detection and revealing of the relevant "context" to inform affect detection. The work addresses the journal's themes on human emotion behavior analysis and understanding.

Keywords Affect detection · Emotion modeling · Multi-threaded improvisation · Hidden Markov model · Neural network

L. Zhang (✉)
School of Computing, Engineering and Information Sciences, Northumbria University, Newcastle NE2 1XE, UK
e-mail: l.zhang@cs.bham.ac.uk

J. Barnden
School of Computer Science, University of Birmingham, Birmingham B15 2TT, UK
e-mail: j.a.barnden@cs.bham.ac.uk

Introduction

Previous work has developed online multi-user role-play software that could be used for education or entertainment [1]. Up to five human actors and one AI actor were engaged in one improvisational session. Emotionally charged loosely defined scenarios were provided to allow creative improvisation. A human director was also needed to monitor the drama improvisation. The human director had a number of roles. He or she had to constantly monitor the unfolding drama and the actors' interactions, or lack of them, to intervene if they were not keeping to the general spirit of the scenario. For example, a director could intervene when the emotions expressed or discussed were not as expected (or were not leading usefully in a new interesting direction). The director could also intervene if, for example, one character was not getting involved or was unduly dominating the improvisation.

In order to reduce the human director's burden, an affect detection component was previously developed and embedded in the AI agent, which detected affect from human characters' turn-taking input (input contributed by

an individual character at one time). The detected affect allowed the intelligent agent to make responses that, it was hoped, would stimulate the improvisation, thus leading to less need for intervention by the human director.

Several loose scenarios for the drama improvisation are employed. Human actors are allowed to be creative in their role-play. The intelligent agent normally will play a minor role in the improvisation, such as a best friend of the sick or bullied leading character. The scenarios used tend to be emotionally charged. In general, such a framework setting enables young people to discuss these emotionally intensive scenarios anonymously, which normally are difficult to talk about in a face-to-face situation. It also provides the platform for the development of the intelligent agent, which makes attempts to detect emotion and feelings from users' creative inputs and makes appropriate responses. The transcripts created by the test subjects provide valuable resources for dialogue and affect study and evaluation.

The previously developed affect detection model was able to detect a wide range of emotions including basic and complex emotions and value judgments. The complex emotions in this application domain refer to any emotion included in the global structure of emotion types suggested by Ortony, Clore and Collins (also known as the OCC model) [2] but not belonging to the 6 basic emotions (anger, disgust, fear, happiness, sadness and surprise) defined by Ekman [3]. Thus, the complex emotions include the following: embarrassment, disapproval, etc.; *meta-emotions* (emotions about emotions) such as desiring to overcome anxiety; and *moods* such as hostility, etc. [1].

Moreover, the previous affect detection processing was mainly based on textual pattern-matching rules that looked for simple grammatical patterns or templates partially involving specific words or sets of specific alternative words. It did not take any context into consideration. A rule-based Java framework called Jess [4] was used to implement the pattern or template-matching rules in the AI agent allowing the system to cope with more general wording and ungrammatical fragmented sentences. The rules conjectured the character's emotions, evaluation dimension (negative or positive), politeness (rude or polite) and what response the AI actor should make. In order to deal with complex grammatical structures, sentence type information obtained from a syntactical parser, Rasp [5], was also adopted in the rule sets. Such information helped the agent not only to detect affective states from the input (such as the detection of imperatives), but also to decide whether the detected affective states should be counted (e.g., affects detected in conditional sentences would not be valued) [6, 7]. An example of the system interface is shown in Fig. 1.

From the analysis of the previously collected transcripts, the previous affect interpretation based on the analysis of

Fig. 1 An example user interface with three human characters (the first three characters counting from the *left hand side*) and one AI actor (the last character)

individual turn-taking input itself without any contextual inference proved to be effective enough for some users' inputs. These inputs usually contain strong clear emotional indictors such as emotional words, for example, "hate/like," and emotionally charged short phrases, for example, "shut up" and "get lost." Metaphorical expressions such as "you are a rat" and "you are a big bully" were also detected and analyzed by the previous processing [7, 8]. One testing transcript was annotated by the previous affect detection and two human judges, respectively, using three labels, positive, negative and neutral, for evaluation. The inter-annotator agreement between human judges was 0.65, while the inter-annotator agreements between the previous system and the human judges were 0.55 and 0.42, respectively.

However, there are also situations that users' inputs do not have any obvious emotional indicators or contain very weak affect signals that challenge the previous affect detection processing; thus, contextual inference is needed to further derive the affect conveyed in such user inputs. Inspection of the previously collected transcripts also indicates that the improvisational dialogues are multi-threaded. The conversations include not only descriptions of personal situations (e.g., worrying or embarrassment about personal situations) but also comments and responses aroused by social communication (e.g., arguing for different opinions). The context-based affect detection presented here mainly targets affect interpretation from such personal and social interaction contexts without strong affect indicators.

Moreover, context is also very broad and plays very important roles in revealing the social goals that hide behind each social interaction. The cognitive research of kappas [9] discussed the diversity of affect embedded in "smile" facial expressions during social interaction and the importance of the understanding and employment of the

related social contexts for the accurate interpretation of the implied affect in such facial expressions. For example, people tend to use smile facial expressions in order to show happiness and politeness, and to hide desperation and embarrassment. A broad social interaction context may include semantic interpretation of the conversation, discussion themes, tone of voice and body language. Such a context will significantly help to interpret the most probable emotions and feelings implied in one smile facial expression. This is also one of the most challenging research topics in the affective computing field and the long-term research goal of the work presented here. But at the current stage, the context discussed in this paper indicates ones most recent personal or semantically most related social inputs during the drama improvisation. Such a context normally contains more than one user input and may provide a social communication or personal mood background to inform affect detection especially for those without strong affect indicators.

In order to produce an effective context-based affect sensing model, first of all, the two different types of contexts, that is, the personal and social contexts, are identified. Linguistic "signals" are also collected, which indicate whether the discussion is oriented more toward personal experience or more toward responding to the social interaction contexts, from the previously collected transcripts of the Crohn's disease scenario.[1] Personal emotional articles collected by the Experience project (www.experienceproject.com) have also been used to extract grammatical "signals" for personal context descriptions. These grammatical and linguistic "signals" are then used to train a naïve Bayes classifier. It then identifies whether a user input belongs to a personal context or a social interaction context.

Both a hidden Markov model (HMM) and a neural network are developed to detect emotion, respectively, for the personal mood and social interaction contexts. For the development of the HMM-based affect detection in the personal contexts, the work statistically models how emotions evolve for individual characters throughout the improvisation. Personal contexts extracted from the previously collected transcripts of the Crohn's disease scenario and personal articles from the Experience project are used to train the HMM. If the naïve Bayes classifier determines that one user input is more related to opinions or feelings about personal situations, then the HMM with the consideration of each individual character's emotions expressed throughout the improvisation is used to detect the affect implied in this input.

If user inputs are classified as social contexts, then latent semantic analysis is used to detect their discussion themes and identify potential target audiences. Interpersonal relationships, emotions implied by target audiences in their most recent inputs and sentence type information (such as imperatives or rhetorical questions) are considered in the neural network-based reasoning to detect the most likely affect implied in these social inputs. Such reasoning also interprets potential emotional influences of discussion topics and other participant characters' previous inputs toward the speaking character.

The paper is arranged in the following way. Related work is discussed in Sect. "Related Work," and the classification of personal and social contexts using the naïve Bayes classifier is presented in Sect. "Classification of Personal and Social Contexts." Section "Semantic Topic Theme Detection" presents the development of latent semantic analysis using a semantic vector package to derive the discussion themes, etc., for social interaction inputs. The affect interpretation in the personal and social contexts using both the HMM and the neural network-based reasoning is discussed in Sect. "Affect Detection in Personal and Social Contexts." Various evaluation results of context-based affect detection are presented in Sect. "Evaluations." Conclusions and future work are discussed in Sect. "Conclusions."

Related Work

Emotion theories, particularly that of Ortony et al. [2] (OCC), have been used widely in affective computing research. The OCC model discussed appraisal theories of 22 emotions. However, the OCC model did not provide discussions about changes between different emotional states. The work of Hareli and Rafaeli [10] discussed psychological studies on emotion cycles and how emotions of an individual influence other people's emotions and behaviors in social organizational settings, and how this may affect their future interactions. Such context-based emotional models are especially relevant to the research presented here although their models are limited to organizational settings and represent simple interaction cases.

Much research has been done on creating affective virtual characters in interactive systems. Picard's work [11] made great contributions to building affective virtual characters overall. Prendinger and Ishizuka [12] used the

[1] The Crohn's disease scenario is mainly about Peter who has had Crohn's disease since the age of 15. Crohn's disease attacks the wall of the intestines and makes it very difficult to digest food properly. Peter has the option to undergo surgery (ileostomy), which will have a major impact on his life. The task of the role-play is to discuss the pros and cons with friends and family and decide whether he should have the operation. The other characters are the following: Janet (mom) who wants Peter to have the operation, Matthew (older brother) who is against the operation, Arnold (Dad) who is not able to face the situation and David (the best friend) who mediates the discussion.

OCC model in part to reason about emotions and to produce believable emotional expressions. Mehdi et al. [13] combined a widely accepted five-factor model of personality, mood and OCC in their approach for the generation of emotional behavior for a fireman training application. Gratch and Marsella [14] presented an integrated model of appraisal and coping to reason about emotions and to provide emotional responses, facial expressions and potential social intelligence for virtual agents. Aylett et al. [15] also focused on the agent development of affective behavior planning. Endrass et al. [16] carried out a study on the culture-related differences in the domain of small-talk behavior. Their agents were equipped to generate culture-specific dialogues.

Recently, textual affect sensing has also attracted researchers' interest. However, there has been only a limited amount of work directly comparable to the research presented here, especially given the concentration on improvisation, open-ended language and the employment of contexts for affect interpretation in the chosen application domain. ConceptNet [17] includes a toolkit to provide practical textual reasoning for affect sensing for six basic emotions, text summarization and topic extraction. Neviarouskaya et al. [18] provided sentence-level textual affect sensing, the @AM system, to recognize judgments, appreciation and different affective states. They adopted a rule-based domain-independent approach with extensive semantic analysis of verbs. Although some linguistic contexts introduced by conjunctions such as "but" were considered, the detection task setup was still limited to the analysis of individual inputs. Their system was evaluated with 1,000 sentences describing personal experiences. It achieved an average accuracy rate of 62 % with 14 labels (9 emotion labels + (positive or negative) judgment + (positive or negative) appreciation + neutral) for annotation, an average accuracy rate of 71 % with seven labels ((positive or negative) affect + (positive or negative) judgment + (positive or negative) appreciation + neutral) and an 88 % accuracy rate with three labels (positive, negative and neutral). More discussion is provided for the comparison between their @AM system and the HMM-based affect detection in personal contexts presented in this paper in Sect. "Evaluations."

Although *Façade*, an AI-driven character-centric interactive drama game [19], included shallow natural language processing for characters' open-ended utterances, the detection of major emotions, rudeness and value judgments is not mentioned. They also did not report any specific accuracy rates or precisions of their affect detection analysis. However, they clearly indicated that the open-ended natural language input posed a big challenge to the system and the system suffered from three types of failures regarding natural language input: non-understood utterances, false positives and an asymmetrical range of expression.

Zhe and Boucouvalas [20] demonstrated an emotion extraction module embedded in an Internet chatting environment. It used a part-of-speech tagger and a syntactic chunker to detect the emotional words and to analyze emotion intensity. The detection focused only on emotional adjectives and first-person emotions and did not address deep issues such as figurative expression of emotion. The evaluation of their system was also conducted in a static way. That is, questions had been pre-defined by the authors, and users were asked to write down their emotional responses to these prescribed questions. Then, the responses were subsequently entered manually into the system to test their system's performance. Four hundred and fifty sentences generated by the test subjects were returned with the pre-defined questionnaires, and the system achieved a 90 % accuracy rate for the affect classification of these test sentences. However, comparing with real-time online chatting settings, the conversation inputs in their experimental setting were much more predictable due to the pre-defined questions. Their experimental settings also removed any potential involvement of small-talk behaviors (e.g., to get to know each other's names, hobbies and academic background) between scenario-related task talk, which happen very often in real-life or online chatting.

Also, Ptaszynski et al. [21] developed a context-based affect detection component with the integration of a web-mining technique to detect the affect from users' input and verify the contextual appropriateness of the detected emotions. Their system achieved 75 % accuracy in determining both valence and the specific emotion types for the annotations of 20 short conversations. The system achieved 45 and 50 % accuracy of determining contextual appropriateness, respectively, for the detected specific emotion types and valence. However, their system targeted conversations only between an AI agent and one human user in non-role-playing situations, which greatly reduced the complexity of the modeling of the interaction contexts. There is also work on general linguistic cues useful for affect detection (e.g., Craggs and Wood [22]).

Comparing with the above related work, the work presented in this paper focuses on the following aspects: (1) real-time affect sensing for basic and complex emotions in improvisational role-play situations from individual turn-taking inputs; (2) affect interpretation from personal contexts with the consideration of each individual character's previous emotion profile; and (3) affect detection in social interaction contexts with the consideration of interpersonal relationships between characters, target audiences' most recent emotional implications and special sentence types.

Classification of Personal and Social Contexts

In the previous testing, the language used during improvisation is complex and idiosyncratic, that is, often ungrammatical and full of abbreviations, misspellings, etc. It borrows heavily from the language of text-messaging (textese) and chat rooms [23]. Compared to the language normally analyzed in computational linguistics, it provides significant additional challenges.

Several pre-processing components were produced previously to recover the standard user inputs. Moreover, as discussed earlier, the improvisation conversation is also often multi-threaded. For example, Peter sometimes showed anxiety and stress about his ill health and revealed his feelings about his personal situations. He sometimes also responded emotionally to other characters' improvisation (e.g., he was "angry" at the Dad character who often avoided talking about his disease). Other characters also showed emotions about personal situations and were emotionally affected by interaction contexts. This makes the affect detection task difficult. Since the previous original detection model, which only analyzes the input itself without taking any context into consideration, is capable of detecting affect from the inputs with strong clear affect indicators (e.g., "thanks," "sorry," "I like it"), it does not seem necessary to activate contextual affect analysis for such inputs. However, if the input itself does not contain any strong clear emotion indicators and the original model fails to detect any affective implication, then contextual affect detection in the personal emotional contexts (i.e., the modeling of improvisational mood of individual characters) and affect detection from the social contexts (i.e., the modeling of emotional influence between characters) are used to further interpret affect. First of all, the naïve Bayes classifier is used to find out the nature of the context (caused by personal situations or aroused by other characters). If the type of context is identified by the classifier, the corresponding analysis is used to predict affect in such inputs.

In order to produce a classifier to recognize the two types of contexts, linguistic indicators signaling the personal and social contexts are extracted from five representative transcripts of the Crohn's disease scenario. The Experience project Web site also publishes personal stories about life experience. The Web site includes 12 categories of personal articles including Education, Family & Friends, Health & Wellness, Lifestyle & Style, Pets & Animals, Arts & Entertainment, Culture & Religion, Current Events, Food & Drink, Jobs & Finance, Recreation & Sports and Relationships & Romance. Twenty stories across five different categories (Education, Family & Friends, Health & Wellness, Lifestyle & Style and Pets & Animals) are used to extract grammatical signals for personal situation descriptions. On the basis of the above study, the following grammatical structures signaling personal emotion contexts are collected and demonstrated in Table 1. For example, people tend to use "first-person subject + verb phrases" to describe personal situations or feelings. Five categories of such personal emotion contexts are provided in Table 1.

The following signals indicating the social contexts are also collected from the inspection of the selected example transcripts and presented in Table 2.

The naïve Bayes classifier is used to identify the two different types of contexts. The typical signals for the personal and social context descriptions presented in both tables are taken into consideration, and 200 unique instances for the training of each type of the context are collected. Four hundred instances have also been employed to evaluate the performances of the categorization of these two types of the contexts. Detailed evaluation results are presented in the evaluation section.

The following example interaction produced by the test subjects is provided to illustrate the classification of the personal and social contexts. As mentioned earlier, if strong affect indicators are detected from user inputs, the original affect detection component without the consideration of any contextual inference is used to provide affect interpretation. If any input does not contain any clear affect signals, then the contextual affect detection is activated. The naïve Bayes classifier is first used to identify whether the input describes personal experiences or is the response caused by the social interaction context. In the following example, the original affect detection processing is able to annotate inputs from conversation 1 to conversation 8 and the input of conversation 10, which contain strong affect indicators (see italics). Since conversations 9, 11 and 12 contain very weak affect indicators, the classifier is activated to sense the types of the three inputs, so that the personal or social context-based affect detection using either HMM or the neural network will be employed in the next step to predict affect for them. As mentioned earlier, the syntactical parser, Rasp, is used in the pre-processing stage to analyze the grammatical structure of each input. Conversation 9 mentions a family role and has social linguistic features mentioned in Table 2 part (ii). Although it also contains linguistic features in personal contexts mentioned in Table 1 part (i), the classifier trained with typical examples from both personal and social contexts identifies it as a social context. Conversation 11 is recognized to contain both the social context indicator, "verb phrases + second-person object," and the social signal, "I know," discussed, respectively, in Table 2 (iv) and (viii). Conversation 12 contains the social context signal mentioned in Table 2 (v) as well: "question sentences with subjects 'you'." Thus, both of the conversations 11 and 12 are also recognized as responses or comments caused by social interaction. Then, the neural network-based social context

Table 1 Grammatical signals for personal situation descriptions

Linguistic signals	Examples
(i) The first-person singular tends to be used extensively in such expressions, such as in: "I + copular form" and "I + {quantifier or adverb} + verb phrase (except for 'I know')."	"I don't want it to ruin our lives," "I have Crohn's disease," "I now have to keep it simple," "I never thought losing my best friend would come so soon," "I'd give anything to hear Roxie's little claws on the hardwood floor," "I am so lost/confused right now."
(ii) Inputs have singular first-person expressions as objects, such as in "third-person subject + (verb phrase or copular form) + me."	"It hasn't really hit me yet," "It crushed me," "It's hard to look after me," "She would just give it to me."
(iii) Inputs contain first-person possessive pronouns such as in "third-person subject + verb + my or mine."	"He is my family," "It was my choice," "The operation is to remove the last part of my small intestine."
(iv) Expressions are mainly regarding third-party subjects such as in "third-person pronoun + verb phrase."	"She didn't make it," "Well it is a struggle," "It is so different without him."
(v) Inputs include prepositions followed by singular first-person expression such as in "to me."	"It was far more than a pet to me," "She never was there for me," "To me, this is the end."

Table 2 Grammatical signals for the social contexts

Linguistic signals	Examples
(i) A second-person subject + {quantifier or adverb} + verb phrase or copular form.	"You just said that dinner is going to be really hard for me," "You just think about yourself," "You shouldn't have done it," "You won't be made a fool in public," "You were there with us," "You are so late."
(ii) Names or family roles are mentioned in the inputs. Or inputs contain "everyone, anyone, mate, buddy," etc.	"Mum, I wouldn't say dying," "u happy Dave," "Son, you take it easy," "Sweetheart, I have a stressful day at work," "Please can everyone calm down," "Whatever Peter wants," "Peter, we are here for you."
(iii) Inputs contain plural first-person subjects such as in "we + verb phrase."	"We are family," "We have to ask him how he feels," "For a quite meal so we could talk about things," "We'll manage somehow," "We are here to help."
(iv) Inputs contain "verb phrases + first or second-person objects (us or you)."	"It really let us down," "We'll carry you through."
(v) Question sentences with subjects such as "we or you," or "our or your + nouns."	"How do you think Peter feels," "How can your body cope with it," "Why are we discussing it in public," "How are you feeling about the doctor's word," "Don't you think it will make it worse," "Are you feeling well," "Are you happy with it," "Are we ready to face the enemy."
(vi) Statement inputs contain possessive pronouns such as "our or your."	"This is our decision," "It is not good to your blood pressure," "The operation will change your life."
(vii) Prepositional phrases such as "to us," "for you," etc.	"You are everything to us," "The treatment would be good for you," "Your mother and I are here for you."
(viii) Idioms, proverbs and others.	"You scratch my back and I'll scratch yours," "You never know what you can do until you try," "Be all ears," "Here you are/go," "You can say that again," "I know," "Excuse me," "Pardon me," "You know" (used by itself).

emotion detection will be employed to annotate these three inputs in the next step.

Conversation 1 Matthew: *I don't think* you should have the treatment peter. [disapproval]
Conversation 2 Arnold: *lets not* talk about it now. Can we just get our food and discuss this later. [disapproval]
Conversation 3 Peter: *why not* hav it matt [neutral]
Conversation 4 Matthew: you will *get bullied* cuz of it. [sad]
Conversation 5 Janet: nobody will bully u. *shut it* matt [angry]
Conversation 6 Dave: *stop arguing* every1. [disapproval]
Conversation 7 Matthew: dad, *stop talking* about food. [disapproval]
Conversation 8 Janet: stand up to da bullies and *do not be* afraid. [disapproval]
Conversation 9 Peter: mum, I don't think u really understand what it is. Its not like a carrier bag. I will use it when I go loo [*A social context: disapproval; Target audience: Janet*]
Conversation 10 Arnold: *excuse me*, *can we not*? Im eating [disapproval]
Conversation 11 Janet: I know what it is, but it will still help you [*A social context: disapproval; Target audience: Peter*]

Conversation 12 Matthew: how do you feel about having a bag attached to you? [*A social context: angry; Target audience: Peter and Janet*]

The following presents another example interaction extracted from the collected testing transcripts. The first three inputs with clear emotional implications (see italics) are annotated by the original affect sensing component. The conversations 4 and 5 contain weak affect indicators; thus, the classifier is activated to recognize whether they are regarding personal situation descriptions or responses to the social context. Conversation 4 contains the grammatical structure of "third-person subject + copular form + my" shown in Table 1 part (iii), and conversation 5 shows the signal of personal situation descriptions shown in Table 1 (i) as well: "I + quantifier or adverb + verb phrase." Thus, the classifier identifies both of them as the personal contexts. The original affect detection component also finds conversation 6 with clear indication of emotion, while the last two inputs are again with weak affect indicators. Thus, the classifier is used to categorize conversation 7 as a social context and conversation 8 as a personal situation description according to the social context signals in Table 2 part (ii) and the personal context signals in Table 1 (ii).

Conversation 1 Janet: peter its ur choice and *i will support u*. [caring]
Conversation 2 Arnold: yeah, *we'll be there for you*. [caring]
Conversation 3 Peter: yes, *thnx* mum and dad 4 da support [grateful]
Conversation 4 Peter: at da end of da day, it is my choice. [*A personal context: neutral*]
Conversation 5 Peter: and I've decided to go throu wid it. [*A personal context: neutral*]
Conversation 6 Janet: *good for you* if that's wat u want to do [approval]
Conversation 7 Peter: but the cost, mum…the aftereffects…[*A social context: disapproval*]
Conversation 8 Peter: its gonna be really hard to look after me!! [*A personal context: disapproval*]

The API for the naïve Bayes classifier embedded in a data mining package, Weka [24], is used for the implementation of the classifier discussed above. After the classification of these two types of contexts using the classifier, in the following section, the identified social interaction inputs will be further analyzed using latent semantic analysis to detect their most likely discussion themes and potential target audiences. Relationships between characters will also be retrieved. Such sources of information will then be used as inputs to the social context affect detection processing.

Semantic Topic Theme Detection

The above personal and social context classification relies heavily on the linguistic signals. Thus, when other linguistic features are used in the description of such contexts, the performance of context identification will be affected. It is also noticed that because of the multi-speaker nature of the system, conversations with different themes are embedded together. Some of the inputs also do not clearly indicate their target audiences. In order to improve the robustness of the context processing and detect affect accurately from the improvisational inputs without strong affect indicators and clear target audiences, semantic interpretation of the social interaction contexts is integrated with the affect detection processing. In this section, developments of using latent semantic analysis (LSA) [25] and its related packages for terms and documents comparison to recover the most related discussion themes and potential target audiences are discussed to benefit affect analysis in social contexts.

The previous rule-based affect detection implementation mainly relied on keywords and partial phrases, pattern matching with simple semantic analysis using WordNet, etc. However, it is noticed that many terms, concepts and emotional expressions can be described in various ways. Especially if the inputs contain no strong affect indicators, other approaches focusing on underlying semantic structures in the data should be considered. Thus, latent semantic analysis is employed to calculate the semantic similarity between sentences to derive the discussion themes of such inputs.

Latent semantic analysis generally identifies relationships between a set of documents and the terms they contain by producing a set of concepts related to the documents and terms. In order to compare the meanings or concepts behind the words, LSA maps both words and documents into a "concept" space and performs comparison in this space.

In detail, LSA assumes that there are some underlying latent semantic structures in the data that are partially obscured by the randomness of the word choice. This random choice of words also introduces noise into the word–concept relationship. LSA aims to find the smallest set of concepts that spans all the documents. It uses a statistical technique, called singular value decomposition, to estimate the hidden concept space and to remove the noise. This concept space associates syntactically different but semantically similar terms and documents. These transformed terms and documents in the concept space are used for retrieval rather than the original terms and documents.

In this work, the semantic vector package [26] is employed to perform LSA, analyze underlying relationships

between documents and calculate their similarities. This package provides APIs for concept space creation. It applies concept mapping algorithms to term–document matrices using Apache Lucene, a high-performance, full-featured text search engine library implemented in Java. This package is integrated with the intelligent agent's affect detection component to detect discussion themes by calculating the semantic distances between identified social improvisational inputs without strong affect signals and the training documents with clear discussion themes.

In order to perform such semantic comparison, some sample documents with strong topic themes have to be collected. Personal emotional articles from the Experience project are used for this purpose. As mentioned earlier, the articles from the Experience Web site belong to 12 discussion categories. In order to improve the robustness of the system, sample articles close enough to the scenario are extracted including articles of Crohn's disease (five articles), school bullying (five articles), family care for children (five articles), food choice (three articles), school life including school uniform (10 short articles) and school lunch (10 short articles). Phrase- and sentence-level expressions implying "disagreement" and "suggestion" have also been gathered from several other articles published on the Experience Web site. Thus, training documents with eight discussion themes are gathered, including "Crohn's disease," "bullying," "family care," "food choice," "school lunch," "school uniform," "suggestions" and "disagreement." The first six themes are sensitive and crucial discussion topics to the chosen scenarios, while the last two themes are intended to capture arguments expressed in multiple ways.

Affect detection from metaphorical expressions often poses great challenges to automatic linguistic processing systems.

During the improvisation of the chosen scenarios, metaphorical expressions were sometimes used to express emotions, such as "The disease turns my life upside down," "I've been trying to put it out of my mind all day," "My plate is already too full…there aren't other options" and "You are an angel/dog." Statistical-based approaches relying on semantic preference violations were employed to detect food, animal and size metaphors in our previous work [27]. However, there are also metaphorical expressions that do not indicate semantic preference violations. Thus, in this research, we aim to employ a semantic-based approach to focus on the underlying semantic structures in such language phenomena to detect metaphors.

Therefore, in order to detect a few frequently used basic metaphorical phenomena, four types of metaphorical examples published on the Web site http://knowgramming.com are included in the training corpus. These include cooking ("It was a half-baked idea"), family ("The universe is full of cousins"), weather ("You are the sunshine of my life") and farm metaphors ("His roots ran deep in the region"). A group of "Ideas as External Entities" metaphor examples ("The whole of my childhood rushed through my head like an electric train") is also borrowed from our ATT-Meta project databank [28] to enrich the metaphor categories. Individual files are used to store each type of the metaphorical expressions, such as cooking_metaphor.txt and ideas_metaphor.txt. All the sample documents of the above 13 categories are regarded as training files and have been put under one directory for further analysis.

Overall, 38 full articles are used in total from the Experience Web site for the construction of the training corpus with an average length of 585 words per article. There are another 20 articles from the same Web site used as sources to extract phrases and sentences indicating "disagreement" and "suggestions" in order to gather the diversity of such expressions. The average length of these two training files is 350 words. Each metaphor sample training file contains sentence-level metaphor examples belonging to one specific type. They are extracted from dedicated metaphor resources as mentioned above (the ATT-Meta databank and the Web site: http://knowgramming.com). The average length of the metaphor training files is 1,223 words per file. For the LSA-based processing, command line parameters are used to remove stop words from the corpus while processing. With default settings, the semantic vector API generates a LSA concept space with 200 dimensions.

The first example interaction of the Crohn's disease scenario mentioned in the above section is used to demonstrate how the discussion themes are detected for those social inputs with weak or no affect indicators and ambiguous target audiences.

Since the previous affect detection processing focuses on affect interpretation from inputs with strong emotion signals, it provides an affect label for such inputs in the above example. Those inputs without an affect label attached directly are those with weak or no strong affect indicators. Therefore, further processing is needed to recover their most related discussion themes and identify their most likely audiences in order to identify implied emotions more accurately in these social contexts. The general idea for the detection of discussion themes is presented in Fig. 2.

The conversational theme detection processing now starts with conversation 9 from Peter to demonstrate the topic detection. This input is stored as a separate individual test file (test_corpus1.txt) in the same folder containing all the training sample documents of the 13 categories.

The corresponding semantic vector APIs are activated to create a Lucene index for all the training samples and the test file. This generated index is also used to create term

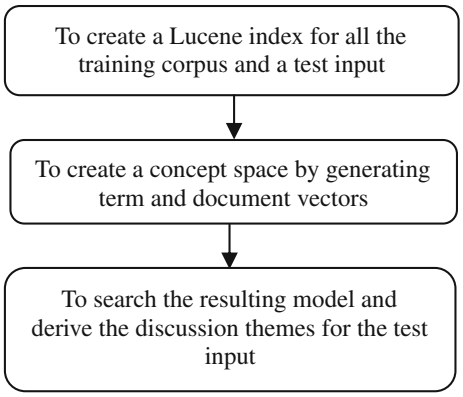


Fig. 2 Procedures for semantic analysis and discussion theme detection

and document vectors. In this application context, there are 45 vectors of dimension 200 generated as document vectors and 5,383 vectors of dimension 200 generated as term vectors. Since LSA aims to map a set of documents and the terms they contain into a concept space and perform comparison in this generated space, in this application, the concept space with document and term vectors is re-calculated each time when a new test file arrives in order to cover the test input in the concept space and perform comparison directly.

Various search options could be used to test the generated concept model. In order to find out the most effective approach to extract the topic themes of the test inputs, several experiments were conducted. First of all, rankings for all the training documents and the test sentence, based on their semantic distances to a topic theme, are provided. The system achieves this by searching for document vectors closest to the vector for a specific term (e.g., "disease"). An example *partial* output is listed in Fig. 3.

In the above outputs, except for the test file test_corpus1.txt (containing conversation 9 from Peter), other listed files are all from the training corpus taken from the articles published on the Experience Web site. The values shown in the first column are the semantic distance values between each document and the chosen topic theme, "disease." The system also intends to rank all the files based on their semantic closeness to the other topic themes (such as "bullying," "disagreement," etc.), but as mentioned earlier, there are multiple ways to describe a topic theme such as "disagreement" and "suggestion." It affects the file ranking results more or less if different terms indicating such themes are used for enquiry. Thus, other more effective search mechanisms still need to be used to accompany the above file ranking findings to detect topic themes.

There is another search algorithm that is able to find terms most closely related to each document based on the concept space built earlier. This algorithm has been first applied to the training corpus to test its efficiency and findings. According to the first step processing presented in the above to find the semantic distances to the topic term "disease," the training document "crohn4.txt" was listed on the top of the ranking list. However, when this file is used by this search algorithm to find the terms most closely related to it, the following outputs presented in Fig. 4 are returned.

In the above outputs, the most useful disease-related theme term (such as "disease") has not returned with a top ranking on the term list, but is listed in the middle. Thus, it indicates that most related terms are not reliable enough for automatic processing. Another approach especially suitable for this application domain is to find the semantic similarity between documents. All the training sample documents are taken either from articles under clear discussion themes within the 12 categories of the Experience project or from dedicated metaphor sources. The file titles used indicate the corresponding discussion themes. If the semantic distances between files, especially between training files and the test file, can be calculated, then it provides another source of information for the discussion theme detection. Therefore, the CompareTerms semantic vector API is used to find out

```
Found vector for 'disease'
Search output follows ...
0.8112944014737917:F:\test_data\test_lsa6\crohn4.txt
0.7954427523946758:F:\test_data\test_lsa6\crohn3.txt
0.6244268984149078:F:\test_data\test_lsa6\test_corpus1.txt
0.5990879215052521:F:\test_data\test_lsa6\family_care2.txt
0.5779413288260994:F:\test_data\test_lsa6\family_care5.txt
0.5200265398697822:F:\test_data\test_lsa6\crohn1.txt
0.5184941042931127:F:\test_data\test_lsa6\crohn5.txt
0.5094701704973981:F:\test_data\test_lsa6\crohn2.txt
0.4962319424068785:F:\test_data\test_lsa6\family_care3.txt
0.4606811844883898:F:\test_data\test_lsa6\family_care4.txt
0.4583252892724773:F:\test_data\test_lsa6\family_care1.txt
```

Fig. 3 Example partial output for searching for document vectors closest to the vector for a topic theme, "disease"

```
Found vector for 'F:\test_data\test_lsa7\crohn4.txt'
Search output follows ...
0.9351587400907257:have
0.8887467992451593:else
0.8508699187263247:times
0.8069219978282258:hard
0.8027186536514817:told
0.79851066842271:think
0.7976727548113798:i
0.7947333669061222:living
0.7857429057396778:remember
0.7829770373672106:when
0.779169627545226:disease
0.777977648288496:my
0.7730323460332773:you
0.771804897514864:over
0.7682103498791507:labor
0.765358656841035:know
0.761497898373525:rest
0.7574792496840674:understand
0.7386694834219586:own
0.73511471731328:lot
```

Fig. 4 Example output for finding terms most closely related to a training document "crohn4.txt"

```
similarity of "crohn4.txt" with "test_corpus1.txt": 0.8520039156044099
similarity of "disagree1.txt" with "test_corpus1.txt": 0.7601959945997482
similarity of "crohn3.txt" with "test_corpus1.txt": 0.7054680619761999
similarity of "crohn2.txt" with "test_corpus1.txt": 0.6827992954523574
similarity of "bullied3.txt" with "test_corpus1.txt": 0.6593339908493666
similarity of "bullied2.txt" with "test_corpus1.txt": 0.6457772244890972
similarity of "crohn1.txt" with "test_corpus1.txt": 0.6039827766746925
similarity of "bullied1.txt" with "test_corpus1.txt": 0.5775331845539662
similarity of "family_care1.txt" with "test_corpus1.txt": 0.5740221209564 89
```

Fig. 5 Part of the output of the semantic similarities between training documents and the test file (conversation 9)

semantic similarities between all the training corpuses and the test documents. Part of the example output is presented in Fig. 5.

The results showed in Fig. 3 indicate the file rankings based on their semantic similarities to one topic theme ("disease"). That is, they indicate how each training file and the test file are semantically close to a topic term, such as "disease." The results presented in Fig. 5 further complement or justify the above outputs in Fig. 3 by calculating semantic distances between the training documents and the test user input. That is, the results indicate how the test file is semantically close to a specific training file. Thus, if the test user input shows high semantic similarities to any training documents with clear topic themes (indicated by the training file names), then the system concludes that the test user input shares the same topic themes as those of the semantically closely related training documents. Overall, the results in Figs. 3 and 5 complement each other to draw a stronger conclusion for topic detection.

For example, the similarity results in Fig. 5 show there are two training files (crohn4.txt and disagree1.txt) semantically most similar to the test file (test_corpus1.txt, i.e., conversation 9). These two training files, respectively, recommend the following two most related discussion themes: "disease" and "disagreement." In the first step processing mentioned earlier to find document vectors closest to that of a topic theme, the test sentence achieves the best ranking for the "disease" topic theme shown in Fig. 3 and the second best ranking for the "bullying" topic theme (results are not presented here). With the integration of the semantic similarity results between the training document vectors and the test input presented in Fig. 5, the processing concludes that conversation 9 from Peter relates most closely to the following negative topics: "disease," "bullying" and "disagreement." Also, due to its semantic closeness to the topic theme "disagreement," it also most probably indicates "disapproval."

In a similar way, the conversation theme processing has identified the following two semantically most similar training documents to conversation 11 from Janet:

```
similarity of "crohn2.txt" with "test_corpus2.txt":  0.8875302851886464
similarity of "bullied3.txt" with "test_corpus2.txt": 0.8311841420279548
similarity of "crohn3.txt" with "test_corpus2.txt":  0.8031293680512144
similarity of "crohn1.txt" with "test_corpus2.txt":  0.6701409905059565
```

Fig. 6 Part of the results showing the semantic similarities between training document vectors and conversation 11

"crohn2.txt" and "bullied3.txt." These two training files, respectively, recommend the same two discussion themes: "disease" and "bullying" as those for conversation 9 from Peter. The partial results showing the semantic similarities between training corpus vectors and conversation 11 are listed in Fig. 6. Conversation 11 also achieves a top 4 ranking for the enquiry of search for document vectors closest to the vector for "disease."

Conversation 10 from Arnold contains strong affect indicators (see italics). The previous affect detection algorithm labeled it with "disapproval." Since conversation 11 did not mention clear target audiences, the topic themes of conversation 10 from Arnold have to be recovered. The result shows that the most similar training document vectors to that of conversation 10 are "suggestion" and "food related," which are different from the recovered topic themes of conversation 11. Therefore, as mentioned above, because of the similar discussion themes between conversations 9 and 11, it is assumed that Peter is the most likely target audience of conversation 11 from Janet.

By searching for document vectors closest to the vector for the discussion theme "disease," the last input (conversation 12) from Matthew shows high semantic closeness to the topic with a value over 0.65 and a top 4 ranking. The similarity processing indicates that it is most similar to "crohn4.txt" and "bullied3.txt" in the semantic domain. Thus, this input is most likely to be a further piece of discussion aroused by conversations 9 and 11, respectively, contributed by Peter and Janet, and its most probable target audiences are Peter and Janet.

In this application domain, the conversation theme detection using semantic vectors analysis is able to help the AI agent to detect the most related discussion themes and therefore to identify the most likely target audiences. It is believed that these are very important aspects for the accurate interpretation of emotions in social contexts. It is also envisaged that the above processing would be really helpful to distinguish small-talk (task-unrelated discussion) behaviors and task-driven talk during human agent interaction. Thus, it may enable the AI agent to respond more appropriately during the social interaction. The following section will focus on the discussion of how affect is concluded for conversations 11 and 12 in the social contexts using a neural network and affect detection in personal contexts using a HMM.

Affect Detection in Personal and Social Contexts

The research of Hareli and Rafaeli [10] has mentioned that "human emotion is typically studied as a within-person, one-direction, non-repetitive phenomenon." They also stated that research has traditionally focused on evolution of emotional states within one individual given various stimuli. In this research context, first of all it is also especially interesting to find out how emotions evolve within each individual character throughout the improvisation. A hidden Markov model is built based on the learning of personal emotion evolution of the five human-controlled characters, respectively, during improvisation. The HMM is then used to predict the affective implication for a specific character's current input if the input is identified as a personal situation description by the classifier. Furthermore, Hareli and Rafaeli also discussed emotion cycle research on how people react emotionally to the emotional expressions of other individuals in working situations. For instance, an angry team leader makes his or her teammates scared, and the leader feels embarrassed later on. In the research presented here, such social emotional influence between characters is also modeled, and a neural network-based approach is produced to predict the affective implication of a specific character's current input if this input is aroused by the social communication context recognized by the classifier. Discussions of the building of the HMM-based affect detection in the personal contexts and neural network-based affect detection in the social interaction contexts are presented in detail below.

Affect Detection in the Personal Emotion Contexts

HMMs are widely used for speech recognition and pattern matching. A HMM has both a set of hidden states and a set of observation states. It is like a finite-state machine but with the inclusion of not only transition but also observation (output) probabilities. The hidden states in HMMs are also interconnected, so that any state can be reached from any other state. One of the well-known problems HMM targets is the determination of a best sequence of model states given an observation sequence. This HMM application is thus used to detect affect in personal contexts. In detail, text inputs contributed by human characters and observations of the turn taking of the characters are available. That is, the sequence of character names in the order of their contribution is observed. However, without further processing, the emotional implications embedded in these inputs are unknown. That is, these emotional states are hidden. The emotional states in the personal emotion context are also transferable between one another based on the guidance of personal mood. Therefore, the HMM for the personal context affect detection is built to find the most likely hidden emotion state of a particular character given this character's previously expressed emotions.

The previously selected five transcripts of the Crohn's disease scenario were annotated in order to train the HMM. Two human judges (not involved in any technical development) were employed to mark up the example transcripts. Six hundred inputs with agreed annotations were used to build the HMM-based affect detection component. Annotations where they disagreed were discarded. In each improvisational session, there are altogether five human characters, that is, the number of distinct observation symbols (O) per state is five. It indicates that there are five characters who are able to contribute to an emotional expression. The number of distinct emotions used for the annotation of the five example transcripts is regarded as the number of states in the model. Thus, the 10 distinctive emotional states used for the annotation are considered as hidden states (S) in the model. These emotions are the following: "neutral," "approval," "disapproval," "angry," "grateful," "regretful," "happy," "sad," "worried" and "caring." In summary, the *states* in the HMM affect detection model refer to the *emotions* expressed by the characters. Since there are 10 distinctive emotions expressed by the characters during the improvisation, the number of states is 10 in the HMM. Moreover, the *observation symbols* correspond to the *five characters* involved in each improvisation. The number of distinct observation symbols per state is thus five.

The initial state probabilities are the probabilities of a state s_i being the first state of a state sequence. In this research context, the prior probability for each emotional state is set to 1/10, assuming that each emotional state has an equal opportunity to be used as the initial state of a state sequence. A special symbol, "π," is used to represent initial state distribution, that is, the prior probabilities. Transition probabilities, represented as A, ($a_{\mathrm{ij}} = P[q_{t+1} = j|q_t = i]$), are the likelihood to transfer from one emotional state i to another j. The emotion states are thus interconnected with one another with the theoretical assumption that one emotional state is able to evolve to any emotional category but with different transition probabilities. The observation probabilities, represented as B, ($b_j(k) = P[o_t = v_k|q_t = j]$), characterize the likelihood of a certain observation v_k in state j. In this application domain, given a specific emotion j, observation probabilities are the likelihood of an individual character v_k to imply this emotional state. In the following discussion, the production of transition and observation probabilities based on the training data is presented. The overall HMM is presented in Fig. 7 with S representing states, that is, the 10 emotional states, O representing possible observations, that is, the five characters, A indicating state transition probabilities and B indicating observation probabilities.

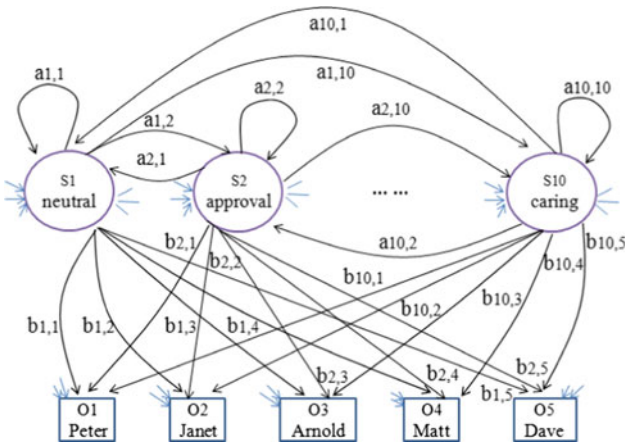


Fig. 7 The HMM for emotion detection in personal contexts

In general, the model built for affect detection in personal contexts has the following features:

- The number of states in the model is 10. The states refer to a set of frequently used distinctive emotions. Thus, the states of the model include "neutral," "approval," "disapproval," "angry," "grateful," "regretful," "happy," "sad," "worried" and "caring."
- The number of distinct observation symbols per state is 5. The individual symbols refer to the five characters involved in one session. Thus, we denote the individual symbols as $V = \{v_1, v_2, v_3, v_4, v_5\} = \{\text{Peter, Janet, Arnold, Matt, Dave}\}$.
- The state transition probability distribution is represented as $A = \{a_{ij}\}$, where $a_{ij} = P[q_{t+1} = j | q_t = i]$. The probability, a_{ij}, indicates the likelihood to transfer from one emotional state i to another j.
- The observation probability distribution is represented as $B = \{b_j(k)\}$, where $b_j(k) = P[o_t = v_k | q_t = j])$. The probability, $b_j(k)$, indicates the symbol distribution in state j. In this application, it indicates the likelihood for an individual character v_k to imply a specific emotional state j, that is, $P[v_k = \text{character} | j = \text{emotion}]$.
- The initial state distribution is represented as $\pi = \{\pi_i\}$, where $\pi_i = P[q_1 = \text{i}]$. In this application, the start probability for each emotional state is set to 1/10, assuming that each emotional state has an equal opportunity to be used as the initial state of a state sequence.

In the following, we discuss the specification of the transition and observation probability measures, A and B, using the 600 training inputs.

Although the human characters are allowed to be creative in their improvisation, the emotion changes within a specific role show similar inclination to some degree. For example, the sick leading character, Peter, may constantly feel stressful about his life-changing operation. The Dad character, Arnold, may frequently feel embarrassed and avoid the discussion about Peter's illness in public. The mom character, Janet, often seems very protective and wants what is best for Peter. Matthew and Dave provide their constant support to Peter's decision, but Matthew is more against the operation idea. In order to build an effective emotion evolution model, all the human characters' personal emotion changes in the selected five example transcripts are analyzed. It is also believed that the improvisation reveals only partial aspects of one normal human-like character designated by his or her role-play. In other words, different role-play represents how one normal character behaves or responds under different circumstances. Therefore, in order to make the HMM capable enough to predict affect in the personal emotional contexts for each character, the emotion changes for all the human characters are taken into consideration for the emotion transition probability production. The 600 example inputs with agreed annotations were used for the training of the HMM to gain transition and observation probabilities.

Transition occurrences between nine emotional states and neutral and between the nine emotions themselves are collected for each individual character. For example, in the first example interaction in Sect. "Classification of Personal and Social Contexts," the change of Matthew's personal emotion is borrowed to give an example. From Matthew's first input to his second input (conversation 4), he implies "disapproval" and "sad," respectively; thus, the training processing increments the transition occurrence counter from "disapproval" to "sad." Similarly, the processing increments the counter of the frequency of emotion transition from "sad" to "disapproval" based on his second (conversation 4) and third (conversation 7) inputs. In this way, emotion transition occurrences between each pair of emotions are collected from each personal context.

The occurrences are then used to produce emotion transition probabilities between each pair of emotions for each individual character. In detail, a transition probability for one specific character is produced using the transition occurrence from one emotion to another divided by the sum of emotion transition occurrences between this emotion and all emotional states for this specific character. Then, averaged transition probabilities between each pair of emotions stored in a 100-element matrix are produced as the transition probabilities, A, for the HMM of the personal emotion evolution. Some example transition probabilities are listed in Table 3.

As discussed earlier, observation probabilities, B, are the likelihood of an individual character v_k to imply a specific emotional state j, that is, $P[o_t = \text{character } v_k | q_t = \text{emotion } j]$. As mentioned above, each role-play has also revealed inclinations toward particular emotional expressions although the improvisation is creative. In other words, a

Table 3 Transition probabilities, A, of the HMM for personal emotion evolution (10 × 10)

	Neutral	Approval	Disapproval	…	Caring
Neutral	0.276316	0.144737	0.236842	…	0.118421
Approval	0.234043	0.234043	0.276596	…	0.0638298
Disapproval	0.188406	0.130435	0.362319	…	0.101449
…	…	…	…	…	…
Caring	0.16129	0.129032	0.354839	…	0.193548

particular character tends to experience several types of emotions more than other categories of emotions determined by this character's specific role. For example, Peter is constantly "sad" and "worried" about his ill health and the stressful decision making about his treatment. The sick character shows "happiness" only occasionally. Arnold tends to feel "embarrassed" about Peter's disease and is constantly in "disapproval" of the discussion topics, while Janet often reveals her "anger" toward Arnold and "caring" about Peter's condition. Such phenomena determine differences in the observation probabilities of each character showing different categories of emotions. For example, given the "sad" emotional state, Peter has higher observation probabilities to show this emotion than other characters, while given the "embarrassed" emotional state, the observation probability of Arnold is higher than other's. The observation probabilities are produced in the following way using the 600 example inputs with agreed annotations. For a specific character, the training processing gathers the occurrences of each of the ten emotions expressed by this character across all the selected training transcripts. Then, the above occurrences of each emotion for each character are divided by the frequencies of each emotion expressed overall, that is, across all characters, to produce the observation probabilities. In this way, given the ten emotional states, observation probabilities of each character showing these emotions are produced. Table 4 shows some obtained observation probabilities for each character.

After the determination of transition and observation probabilities, the training algorithm provides the overall model $\lambda = (A, B, \pi)$. As discussed earlier, if given an observation sequence and the built model λ, the test stage needs to find the most likely state sequence. In other words, if an observation sequence of recently speaking characters' names is provided, the task is to derive the optimal state sequence of emotions. However, as mentioned earlier, emotional implications of the inputs with strong emotional indication have been firstly detected by the original affect detection module. When the original affect sensing component fails to interpret the inputs without strong clear affect indictors, the naïve Bayes classifier is used to determine whether the input is caused by social interaction or limited to the description of the personal situation. If it is a personal context input, then this HMM-based affect detection is activated to predict affect. Thus, in this application context, the state sequence of emotions is not totally hidden but partially derived by the original affect interpretation component with the last state of emotion unrevealed. The model λ needs to find the most likely last state of emotion given an observation sequence of recently speaking characters and the corresponding derived partial emotional state sequence. On the basis of the Viterbi algorithm [29], to find the most likely state sequence of emotion for the given observation sequence, it is assumed that the best score along a single path at time t, $\delta_t(i)$, is achieved, which accounts for the first t observations and ends in state i. In order to reveal the optimal state sequence, the next emotional state j that maximizes the following Eq. 1 needs to be found.

$$\delta_{t+1}(j) = \left[\max_i \delta_t(i) a_{ij}\right] b_j(o_{t+1}) \quad (1)$$

The second example interaction in Sect. "Classification of Personal and Social Contexts" is used to illustrate how the model λ is used to predict affect in the personal emotion contexts. Previously, conversations 4 and 5 were classified as Peter's personal situation descriptions. Then in this example, the HMM focuses on the emotion changes of the Peter character. The Peter character also had another input beforehand, conversation 3 with emotion indication "grateful." In order to predict affect for conversation 4, the observation sequence is [Peter (conversation 3), Peter

Table 4 Observation probabilities, B, for each character showing the ten emotional states (10 × 5)

	Peter	Janet	Arnold	Matt	Dave
Neutral	0.32857	0.08571	0.12857	0.24286	0.21429
Approval	0.45652	0.26087	0.04348	0.08696	0.15217
Disapproval	0.30137	0.13699	0.23288	0.13699	0.19178
…	…	…	…	…	…
Caring	0.06667	0.33333	0.23333	0.2	0.16667

(conversation 4)], while the partial state sequence is [grateful (conversation 3)]. On the basis of the Viterbi algorithm, to find the most likely state sequence of emotion for the given observation sequence, it is assumed that the best score $\delta_t(i)$ ([grateful (conversation 3)]) along the path that accounts for the first observation of Peter's conversation 3 is already achieved. In order to reveal the optimal state sequence, the model needs to find the next emotional state that maximizes the output of the Eq. 1.

Since the revealed last best state of emotion is "grateful," according to the transition and observation probabilities in Tables 3 and 4, the model retrieves the transition probability from "grateful" to each emotion a_{ij} and multiplies it with the observation probability for Peter to show a specific emotional state $b_j(O_{t+1} = \text{Peter})$. The emotional state, which results in the maximum value of $a_{ij} \times b_j(O_{t+1} = \text{Peter})$, is regarded as the optimal state. On the basis of such a perspective, "neutral" is revealed as the optimal next state which produces the best state sequence following the preceding "grateful" emotion. The model thus annotates conversation 4 from Peter as "neutral."

Similarly for the affect prediction for conversation 5, given the observation sequence: [Peter (conversation 3), Peter (conversation 4), Peter (conversation 5)], the partial state sequence, [grateful (conversation 3), neutral (conversation 4)], has achieved the best score so far. The model is again used to find the next best state that maximizes the results of $a_{ij} \times b_j(O_{t+1})$. With the preceding emotion as "neutral," the next most likely emotion is also predicted as "neutral." Therefore, conversation 5 shows "neutral" implication.

The implementation of the above HMM-based affect detection follows tutorials of example applications, such as http://www.run.montefiore.ulg.ac.be/~francois/software/jahmm/ and http://webdocs.cs.ualberta.ca/~lindek/hmm.htm to ensure its accuracy. The transition and observation probabilities are stored in two individual files, and their values are read in dynamically at run time. Any test set with one observation sequence and a partial state sequence is also stored in an individual file and used as another input to the HMM. The model then calculates the most optimal emotion state using the Viterbi algorithm as the output.

In order to test the performance of a machine learning approach, normally it would be ideal if a large representative sample is used to train the model and another independent large representative sample is employed to test the model. However, when the amount of (labeled) data is limited such as the case in this application domain, hold-out validation is one of the suitable methodologies that can be used to evaluate the generalization performance of the chosen machine learning approach. Generally, it reserves a certain amount of data for testing and uses the remainder for training, that is, the test data are held out and not used during training. In the HMM-based affect detection in personal contexts, the annotated data available for each chosen scenario were also classified into two sets: the training and test sets. For each scenario, there was no overlapping between the training and test data for the development and evaluation of the HMM-based affect detection. That is, no data used at the training stage were also employed in the test set in one running cycle. Thus, hold-out validation was used to evaluate the generalization performance of the HMM-based affect detection.

The drawback of the hold-out method is that it does not employ all the data available when training cannot afford to set aside a portion of the dataset for testing. Also, the evaluation results rely heavily on the training and test data split. Such problems can be partially solved by the repetitive running of the hold-out method multiple times. The results obtained from the multiple applications of the hold-out method could be then averaged to produce the final evaluation result. This evaluation process is also known as the repeated hold-out method, which makes the estimated results more reliable.

Moreover, standard deviation in statistics has been used to indicate variability of a population, that is, how much variation exists from the average or an expected value. A low deviation indicates that the data elements tend to be close to the average, while a high deviation implies that the elements are spread out over a large range of values. Thus, the standard deviation of the performance results such as precision scores obtained from the iterative runs of the hold-out method can also indicate the stability of the system. This also gives a good indication of the possible results of k-fold cross-validation. Therefore, based on the above discussions, a simple repeated hold-out validation is employed to evaluate the performance of the model using the annotated transcripts of the Crohn's disease scenario. Moreover, in future work, k-fold cross-validation will also be used in order to further evaluate the performance of the HMM-based reasoning.

Generally, the HMM-based affect detection in personal contexts used a basic supervised learning HMM. The model used 600 inputs from the Crohn's disease scenario at the training stage in order to gain transition and observation probabilities and employed 170 examples from three different transcripts of the same scenario to measure the performance of the HMM at the initial testing stage. Then in order to perform another two iterations of the hold-out validation, first the test set with 170 examples was exchanged with 170 inputs embedded in the previous 600 training examples. Thus, the second test was conducted using the updated training (430 inputs from the previous training set + 170 examples from the previous test set) and test sets (the 170 inputs from the previous training set). In a

similar way, another new test set of 170 inputs from the original training data was also used to conduct the third hold-out validation to identify the HMM's stability.

In order to further evaluate the robustness of the system, another 250 examples extracted from another testing scenario, the school bullying,[2] were used to provide further training of the HMM-based affect detection gained from the initial training using the original 600 inputs of the Crohn's disease scenario in order to obtain updated transition and new observation probabilities. A further test with 130 new testing samples with 83 inputs from the school bullying scenario and 47 inputs from articles borrowed from the Experience Web site was done to evaluate the systems' robustness.

The HMM generally achieves promising affect detection results (see Sect. Evaluations for details) and reveals affective inclination of individual characters using their personal emotion history. It is also crucial to find out how emotions of characters are affected by social communication contexts and to model such effects. Thus, the next section will discuss the development of a neural network-based approach with the consideration of interpersonal relationships, emotions implied by potential target audiences in their most recent inputs and sentence types in order to interpret affect in social interaction contexts.

Affect Detection in the Social Interaction Contexts

The cognitive emotion research of Hareli and Rafaeli [10] pointed out that "one person's emotion is a factor that can shape the behaviors, thoughts and emotions of other people." They also believed that "emotions may affect not only the person at whom the emotion was directed but also third parties who observe an agent's emotion." In this application context, one character's manifestations of emotion can also thus influence others (e.g., see conversations 11 and 12 in the first example interaction in Sect. "Classification of Personal and Social Contexts"). Research of Wang et al. [30] also discussed that feedback of artificial listeners can be influenced by interpersonal relationships, personalities and culture. For example, if two characters share positive relationships and one of them experiences "sad" emotion, then it is more likely the other character responses with an empathic response of sadness. Otherwise if they have a negative relationship, then the other character is more inclined to show a gloating response of happiness. Thus, such interpersonal relationships (such as positive (friendly) or negative (hostile or tense) relationships) are employed to advise the affect detection in the social contexts.

In the first example interaction in Sect. "Classification of Personal and Social Contexts," the topic theme processing has identified that the most likely audience of conversation 11 from Janet is Peter. Especially in Peter's previous input (conversation 9), the family role "mom" used also indicated Peter started the conversation with Janet in the first place. The input of conversation 11 is aroused by such social interaction. The above topic theme detection also noticed that in conversation 10, Arnold "suggested" a topic change. Instead of following on the previous discussion theme, Arnold switched to the food-related topic. This also potentially shows less interest or indifference to the discussion of the previous "disease"-related topic suggested by Peter. Thus, conversation 10 may indicate a "negative" emotion by avoiding or showing less interest in the previous discussion theme. The original version of the affect detection without any contextual analysis has also interpreted conversation 10 showing "disapproval."

The topic theme detection also reveals that conversation 11 from Janet is mainly related to negative topics such as "disease" and "bullying." Therefore, the target audience of conversation 11 from Janet is not Arnold but Peter. Peter showed "disapproval" in the most recent input (conversation 9). This indicates that the most related context to conversation 11 is a "negative" context. Moreover, as mentioned in the Introduction, Rasp was used to obtain sentence type information for each user input in the preprocessing. Thus, the Rasp parser outputted a "conjunction" sentence type for conversation 11 and "but" is the conjunction word. Such a conjunctive phrase is often used to express a contradictory opinion or another point of view.

Moreover, the mom character, Janet, wants what is best for Peter and has a positive relationship with the sick leading son character, Peter. Thus, Janet's emotion can be detected purely based on the linguistic feature of conversation 11 without any emotion influences caused by other factors. That is, Janet is more likely to provide another point of view under the above "negative" discussion theme by showing "disapproval" to Peter's previous point of view. Thus, conversation 11 is more likely to indicate "disapproval." If Peter and Janet shared a negative relationship and Peter showed a negative emotion in the most recent input, then Janet either may behave with a gloating response of "happiness" or may respond with an "outrageous" emotion. A neural network implementation is used to perform such reasoning with the consideration of relationships, emotions implied by target audiences and sentence types to detect affect in the social interaction contexts.

For the input of conversation 12 from Matthew, the Rasp parser also implied that its sentence type is a question sentence. In English, the expression of question sentences

[2] The bully, Mayid, is picking on a new schoolmate, Lisa. Elise and Dave (Lisa's friends), and Mrs Parton (the school teacher) are trying to stop the bullying.

is very diverse. Most of them will require confirmation or replies from other characters, while there is a small group of question sentences that do not really require any replies, that is, rhetorical questions. Such questions (e.g., "What the hell are you thinking?" "Who do you think you are?" "How many times do I need to tell you?" "Are you crazy?") encourage the listener to think about what the (often obvious) answer to the question must be. They tend to be used to express dissatisfaction. In the original rule-based affect detection component, phrases and idioms stored in the knowledge base were used for the detection of rhetorical questions. In this application domain, in order to go beyond pattern matching and keyword spotting, the affect detection component especially detects such rhetorical questions using latent semantic analysis based on Rasp's initial analysis of the sentence type information. Two training documents for question sentences are constructed: one with normal question sentences and the other with rhetorical questions. The semantic vector API is used to perform semantic similarity comparison between the two training document vectors and conversation 12 from Matthew. The output is shown in Fig. 8.

The above results indicate that conversation 12 is regarded as a rhetorical question, which implies dissatisfaction. The topic theme detection previously reveals that the discussion themes of conversation 12 are also "disease" and "bullying" related and its most likely target audiences are "Peter" and "Janet." Both Peter's and Janet's most recent inputs are regarded as the most related social context to the last input from Matthew. Since Peter and Janet both implied negative emotions in their most recent inputs, conversation 12 is embedded in a negative interaction context. According to the scenario, Matthew also has a positive relationship with Peter and he believes that Peter will be bullied because of the side effect of the operation. Thus, he is against the operation idea. On the contrary, Janet wants Peter to have the operation. Therefore, Matthew and Janet have a tense relationship. Moreover, as mentioned earlier, conversation 12 is a rhetorical question reflecting dissatisfaction by itself. In such a negative interaction context and with a comparatively tense relationship with one of the target audience characters, Matthew is more likely to express "anger" in conversation 12.

The above interpretation of emotional influence between characters with the consideration of their interpersonal relationships, recent emotions of target audiences and sentence types has been implemented by Backpropagation, a supervised neural network algorithm. Neural networks are generally well known for classification tasks and pattern recognition. Backpropagation is also one of the most classic supervised neural network algorithms. It is chosen due to its promising performances and robustness of the modeling of the problem domain. The affect detection in social contexts intends to use this neural network implementation to accept the sentence type of the current input, most recent emotions of the current input's potential target audiences, and averaged relationship values between the target audiences and the speaking character as inputs. The number of target audience members could range from one to four for one social input in one drama improvisation session with altogether five characters. The output will be the most probable emotion implied in the current input expressed by the speaking character. Since in this application context, multiple layer perceptions are trained by Backpropagation and such an application does not have much to do with neurons, AI experts in the field also call such systems "connectionist systems." But in this paper, neural networks will be used throughout for the sake of the general audiences to avoid confusion.

```
similarity of "rhetorical_ques.txt" with "test_corpus.txt": 0.7690450440578355
similarity of "normal_ques.txt" with "test_corpus.txt": 0.7092013666709898
```

Fig. 8 Semantic similarities between the training question documents and the test document

Moreover, a single hidden layer can approximate any continuous functions. Therefore, a model with one single hidden layer is chosen for this application. The three-layer topology of the neural network includes one input, one hidden and one output layer, with six nodes in the input layer and 10 nodes, respectively, in the hidden and output layers. The six nodes in the input layer indicate the most recent emotional implications expressed by up to four target audiences, an averaged interpersonal relationship and sentence type information. Three values are used to define relationships: 1 for a positive relationship, 0 for a neutral relationship and -1 for a negative relationship. An average relationship value will be calculated and used as one input to the neural network if the user input has more than one potential target audience.

Although in English, there are only four main sentence types (declarative, interrogative, imperative and conditional sentences), the neural network implementation considers eight types of sentences including declarative, declarative with conjunctions "and" or "but," exclamatory, imperative, conditional, normal question and rhetorical question sentences. The sentence type information is used as one input to the neural network. Thus, values ranging from 0.1 to 0.8 are used to, respectively, represent each of the above eight sentence types, that is, declarative (0.1), declarative with conjunctions "and" (0.2) or "but" (0.3), exclamatory (0.4), imperative (0.5), conditional (0.6), normal question (0.7) and rhetorical question (0.8) sentences. Up to four emotions implied by the intended audiences are also used as inputs to the network. We assign values ranging from 0 to 1 to represent each of the 10

emotions. According to their distances to "neutral," happy = 0.1, grateful = 0.2, caring = 0.3, approval = 0.4, neutral = 0.5, regretful = 0.6, disapproval = 0.7, sad = 0.8, worried = 0.85 and angry = 0.9. Other ways of assigning values for emotion inputs (e.g., all values for emotions are distributed between −1 and 1 with positive values assigned for positive emotions and negative values assigned for negative emotions) were also attempted, which produced exactly the same results, that is, recommending the same emotional indication. If there are less than four target audiences available for a conversational input, the value for the neural emotion (0.5) is used to represent the emotion of an unintended audience with the intention of not providing any emotional influence to the speaking character. Finally, the 10 nodes in the output layer represent the 10 output emotions.

The 600 example inputs with agreed annotations extracted from the selected five example transcripts of the Crohn's disease scenario are also used for the training of the neural network. The training data are generated in the following way. Potential target audiences of each input have been identified by two human judges. The most recent emotions implied by the identified audiences have been collected as input values for the neural network. Scores for interpersonal relationships between characters are predefined. For example, Arnold has a tense relationship (−1) with Peter but a medium relationship (0) with Matthew and a positive relationship (1) with Janet, and so on and so forth for other characters. Then, an average relationship score is produced. Sentence type information is obtained using Rasp for each input. The subsequent emotion experienced by the speaking character is used as the expected output. A sequence consisting of up to four emotion values, a score for relationship interpretation and a sentence type score is regarded as one element of training data. In this way, 553 training data elements are used to train the Backpropagation algorithm. Standard error functions of Backpropagation are used to calculate errors in the output and hidden layers. Then they are, respectively, used to adjust the weights from the hidden to output layer and the weights from the input to hidden layer.

In order to maintain the algorithm's generalization capabilities, the training algorithm minimizes the changes made to the network at each step. This can be achieved by reducing the learning rate. Thus, by reducing the changes over time, the training algorithm reduces the possibility that the network will become over-trained and too focused on the training data. After the neural network has been trained to reach a reasonable average error rate (less than 0.05), it is used for testing to predict emotional influence of other participant characters toward the speaking character in the test interaction contexts.

In the above example interaction discussed at the beginning of this section, for the emotion detection of the input of conversation 11, the following sequence is used as the inputs to the Backpropagation algorithm:

1. The most related emotion context: "Disapproval (0.7) (implied in conversation 9 from the target audience, Peter), null (0.5), null (0.5) and null (0.5)." "0.5" is used to represent emotion values of other non-target audiences.
2. Relationship: "1"—Peter and Janet share a positive relationship.
3. Sentence type: "conjunction_but (0.3)"—a conjunction type with a "but" phrase.

The neural network uses the above as inputs and outputs "disapproval" as the implied emotion in conversation 11 from Janet.

Similarly, for the input of conversation 12 from Matthew, the following sequence is used as the inputs to the Backpropagation algorithm:

1. The most related emotion context: "Disapproval (0.7) (implied in conversation 9 from one target audience member, Peter) and disapproval (0.7) (implied in conversation 11 from another target audience member, Janet), null (0.5) and null (0.5)."
2. Relationship: "1": Peter and Matthew have a positive relationship; "−1": Janet and Matthew have a tense or negative relationship. The average value is: (1–1)/2 = 0.
3. Sentence type: "rhetorical_que (0.8)": a rhetorical question type.

Then it outputs that Matthew is most likely to be "angry."

Another three transcripts of the Crohn's disease scenario have also been used for the testing of the neutral network-based reasoning. Two human judges are also used to provide affect annotation of the test example inputs. One hundred and ninety emotional contexts with agreed affect annotation are extracted in a similar way to evaluate the performance of the Backpropagation algorithm. Each emotional context consists of the affective states expressed by target audience human characters in their most recent inputs. Character relationships and sentence type scores are also appended after these emotional contexts. They will be used as the inputs to the neural network to predict their influence to the emotion of the subsequent speaker. The output of the network is then used to compare with the human judges' annotation of the current speaker's input. One hundred and sixty example inputs from the school bullying scenario are also used to evaluate the robustness of the neural network-based affect detection. Evaluation details are presented in Sect. "Evaluations."

Evaluations

The architecture for the context-based affect analysis has been illustrated in Fig. 9.

The overall system employs a client and server architecture and is implemented in Java. Human actors, the intelligent agent and the human director work through software clients connecting with the server. Clients communicate using XML stream messages via the server, which is usually remote from the terminals, which may themselves be remote from each other. Terminal server communication is over the Internet using standard browsers. The intelligent agent is implemented as a Java client. Figure 10 gives an overview of the control of the expressive characters. Users' text input is analyzed by the AI agent in order to detect affect in the text. The output is an emotion label with intensity derived from the text. This is then used in two ways. Firstly, it is used by the minor character played by the AI agent to generate a response. Secondly, the label and the intensity are sent to the emotional animation system (via an XML stream) where they are used to generate animation.

The context-based affect detection will only be activated if the user inputs contain weak or no obvious strong affect indicators. The component is implemented in the following way. As mentioned previously, Weka's API for the Bayes classifier has been called by the AI agent to classify personal and social situation descriptions. If inputs are recognized as social discussion contexts, then semantic vector APIs are activated to detect their topic themes and identify their potential target audiences. Finally, the neural network

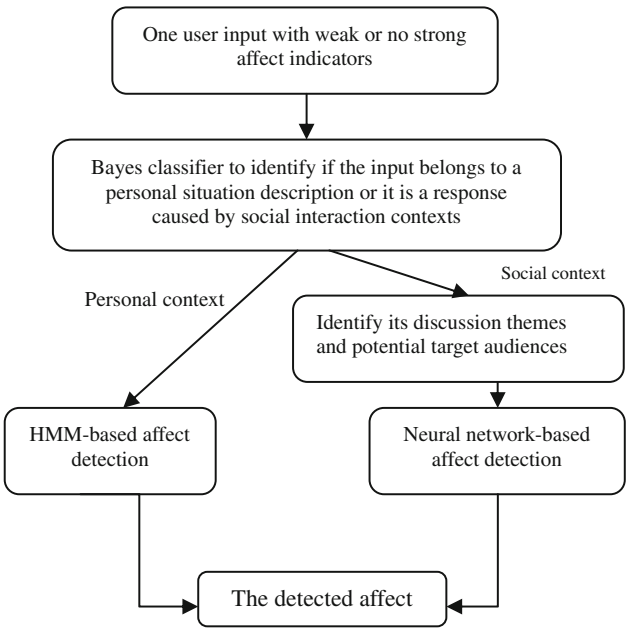


Fig. 9 The context-based affect detection processing

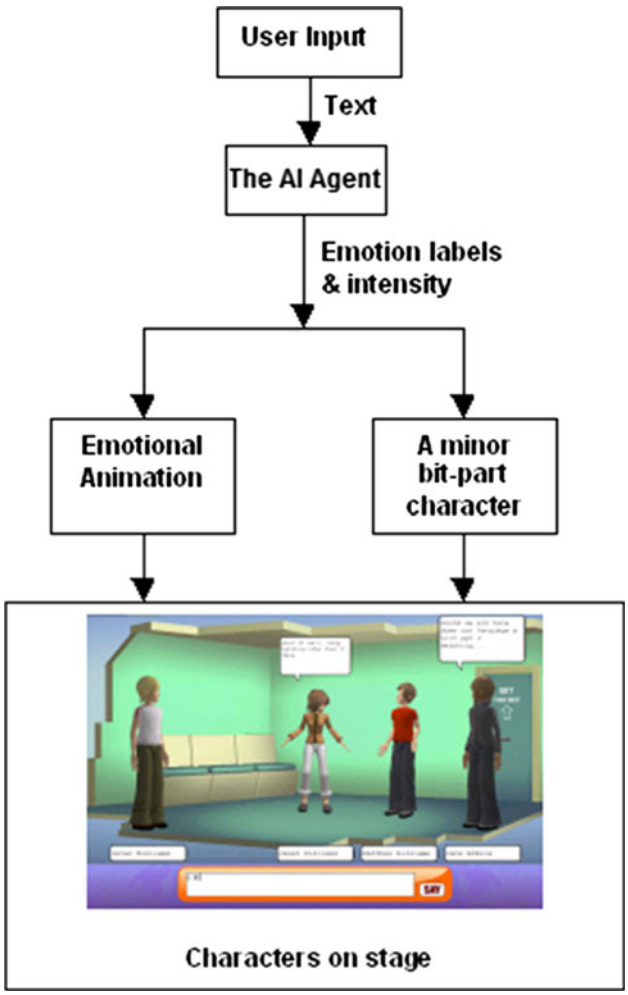


Fig. 10 Affect detection and the control of characters

Java class or HMM Java class is used to detect affect for inputs representing either social communication contexts or personal mood descriptions.

The detected affective states in the user's text input and the AI agent's responses to other characters have been encoded in an XML stream, which is sent to the server by the AI agent. Then, the server broadcasts the XML stream to all the clients, so that the detected affective states can be picked up by the animation engine to contribute to the production of 3D gestures and postures for the user-controlled avatars. The overall affect detection component works in real-time applications. Although the animation sometimes showed a little delay, the overall affect detection processing achieved reasonable performance and provided the detected affect and the AI agent's responses within 300–610 ms or in approximately 500 ms on average with the following type of processor: Intel(R) Core(TM)2 Duo CPU T9500 @2.60 GHz 2.60 GHz. The detailed average computational time for each component is presented in Table 5 for 30 random example runs. Generally

Table 5 Computational time for affect detection processing

Processing	Average processing time based on 30 example runs (ms)
Pre-processing (abbreviation, interjection, spelling checking)	252
Jess rule-based reasoning	86
Weka API for naïve Bayes classifier	15
Discussion theme detection	45
HMM-based affect detection	203
Neural network-based affect detection	133

in order to recover standard user inputs, the pre-processing, which uses lexicon and slang online dictionary and other algorithms (such as Levenshtein distance algorithm) to deal with acronyms, abbreviations and misspellings, took the longest processing time of the components. It is followed by the HMM-based affect detection processing taking 203 ms on average for the 30 example runs and the neural network-based affect reasoning with an average computational time of 133 ms.

User testing was conducted previously with 200 British secondary school students to evaluate the affect detection component and the AI agent's performance. Subjects were 14–16-year-old students at Birmingham and Darlington UK schools. Forty students were chosen by each school for the testing. There was no control of gender. Four-two-hour sessions took place at each school, each session involving a different set of ten students.

Briefly, the methodology of the testing is that we had each test subject having an experience of both scenarios, one including the AI minor character only and the other including the human-controlled minor character only. After the testing sessions, users' feedback via questionnaires and group debriefings was obtained. Improvisational transcripts were automatically recorded to allow further evaluation. The fact that the AI agent was involved in some sessions was concealed in order to have a fair test of the difference that was made. The scenarios used for the testing were Crohn's disease and school bullying. The AI agent normally played a close friend (called "Dave") to the bullied victim or a sick leading character. Surprisingly, good results were obtained from the test subjects regarding the AI agent's performance. Having a minor bit-part character called "Dave" played by the AI agent as opposed to a person made no statistically significant difference to measures of user engagement and enjoyment, or indeed to user perceptions of the worth of the contributions made by the character "Dave." Users did comment in debriefing sessions on some utterances of Dave's, so it was not that there was a lack of effect simply because users did not notice Dave at all. Furthermore, it surprised us that few users appeared to realize that sometimes Dave was computer-controlled. It is stressed, however, that it is not an aim of the work to ensure that human actors do not realize this.

Evaluation of the Classification of Personal and Social Contexts and General Affect Detection Performance

The previously recorded transcripts were taken to evaluate the efficiency of the updated affect detection component with contextual inference. In order to evaluate the performances of the naïve Bayes classifier for the categorization of the personal situation descriptions and the social interaction contexts, another three transcripts of the Crohn's disease scenario were used. Two human annotators were employed to classify the two types of contexts for the testing transcripts. Three hundred and fifty inputs with agreed categorization were used for the evaluation of the naïve Bayes classifier with 150 inputs for personal statements and 200 for the social context descriptions. Three example articles about personal emotional experience were also taken from the category of "Family & Friends" in the Experience project Web site. Each article was regarded as a sentence-by-sentence description of a personal emotional experience. Annotators selected another 50 example sentences of personal situation descriptions. Thus, in the test set, there were 200 examples provided for each type of the context. These agreed categorizations were also used as gold standards in the experiment. A keyword spotting rule-based baseline system was also chosen to perform the same classification task. Table 6 shows the detailed measurement of the classifier and the baseline system.

Generally, the baseline system has 73 % examples correctly classified, while the naïve Bayes classifier correctly identifies 92 % of examples. Briefly, precision indicates the exactness of a classifier. A higher precision implies less false positives. Recall measures the completeness with a higher recall, indicating less false negatives. In Table 6, the naïve Bayes classifier generally achieves higher values in both precision and recall. Thus,

Table 6 Evaluation results of personal and social context categorization

	Precision	Recall	F-measure
Baseline			
The personal context	0.707	0.784	0.744
The social context	0.765	0.684	0.722
Naïve Bayes classifier			
The personal context	0.943	0.892	0.917
The social context	0.9	0.947	0.923

the classifier achieves very promising results for the categorization of the two types of contexts. Classification errors occurred for the processing of the inputs with third-person subjects such as "that's it" and "it is for the good." Some of them are responses or convey judgment in social communication. However, the classifier recognized them as personal situation descriptions. We noticed that such type of inputs challenged the classifier the most since there were expressions from both personal and social contexts employing such third-person pronouns. Future work needs to focus on such third-person expressions in order to improve the modeling of both contexts.

In order to provide some initial evaluation for the HMM-based affect detection for the personal context descriptions and the neural network-based affect detection for the social interaction contexts, two human judges are employed to provide affect annotation of the test 400 examples. The inputs with agreed annotations have also been used as gold standards to measure the performance of the HMM-based affect detection in personal contexts and the neural network-based affect detection in social contexts. A third human judge will also be employed in future work to further justify the annotation of the test set and remove any ambiguity, so that more inputs from the test set can be used to measure the system's performance.

First of all, in order to verify the efficiency of the new developments, the annotations of the 400 example interactions provided by the two human judges were used to produce Cohen's kappa inter-annotator agreement for the overall affect detection performance with the new developments. That is, inter-annotator agreements will be calculated between human judges and the updated affect detection processing and between human judges themselves to indicate the system's performance. Moreover, Cohen's kappa is a statistical measurement of inter-annotator agreement. It provides robust measurement by taking the agreement occurring by chance into consideration. It is also generally considered as a conservative measure of agreement and more robust than simple percent agreement calculation. It is also widely used in the linguistics field to measure the inter-annotator agreement. Thus, it is used as an effective channel to measure the performance of the system presented here.

Since 10 emotions were used for annotation and the annotators may not experience exactly the same emotions as the test subjects did, it led to low inter-annotator agreement between human judges. The inter-annotator agreement between human judge A/B is 0.53. While the old version of the affect detection without any contextual inference achieves 0.36 in good cases, the new version achieves inter-annotator agreements with human judge A/B, respectively, 0.46 and 0.48. The updated affect detection component achieves inter-annotator agreements generally fairly close to the agreement level between human annotators themselves. A detailed inspection of the annotated test transcripts by the new version of the AI agent also indicates that many expressions regarded as "neutral" by the previous version have been annotated appropriately as emotional expressions.

Evaluation of the HMM-based and Neural Network-based Affect Detection

Moreover, in order to provide initial evaluation results for the HMM-based affect detection, the human judges' previous annotations have also been converted into three emotion labels: positive, negative and neutral. Cohen's kappa is also produced to measure the human annotators' inter-annotator agreement: 0.83. Then, 360 inputs with agreed annotations are used as gold standards. Among the 360 examples, 170 examples are used to measure the performance of the HMM for the emotion prediction in the personal contexts with 190 used for the evaluation of the neural network to predict emotion in the social contexts.

Generally, there were three iterations of the hold-out validation conducted in order to evaluate the HMM's performance and robustness. The first test used these 170 personal context inputs mentioned above. These 170 examples have 50 % negative inputs, 32 % neutral and 18 % positive ones. Also, as mentioned earlier, the HMM-based affect detection has the following topology. It regards the 10 distinctive emotions used in the annotation as its 10 states and uses five characters involved in each session as its five distinct observation symbols per state. The affective annotations achieved by the HMM-based affect detection are also converted into solely positive and negative. In order to perform a second hold-out validation, 170 inputs in the original training set were also exchanged with the 170 inputs in the above test set and used as the new test inputs. Thus, the above test set together with the remaining of the training inputs was employed as the updated training data. Subsequently, a third hold-out validation was also performed in a similar way with another new test set of 170 inputs from the original training data for testing and the rest of the training data together with the original test set used for training. Then, the results obtained from these three iterations are averaged to produce overall results for the HMM-based affect detection in personal contexts shown in Table 7.

A baseline system has been built using simple Bayesian networks in order to further measure the HMM-based affect detection. The Bayesian networks have the following topology: The two most recent experienced emotions of a speaking character are used as inputs, and the output will be the predicted affect of the current input of the same

Table 7 Emotion detection results for both of the HMM (averaged results from the three iterations) and the baseline system

	Precision	Recall	F-measure
The HMM-based affect detection			
Positive	0.949	0.938	0.943
Negative	0.939	0.921	0.93
Neutral	0.862	0.905	0.883
The baseline system			
Positive	0.587	0.711	0.643
Negative	0.845	0.652	0.736
Neutral	0.395	0.536	0.455

speaker. Training was also conducted for the baseline system with 250 training examples. Table 7 shows the final evaluation results of the HMM-based affect detection in comparison with the baseline system.

The overall averaged accuracy rate for personal context affect prediction using the HMM is 91 %, while the Bayesian network-based baseline system has an overall accuracy rate: 65 %. Generally, negative inputs are well identified in both of the models. The HMM-based affect detection considers emotions of individual characters throughout the improvisation and generally performs better than the baseline system on the detection of positive, negative and neutral expressions. However, both the HMM-based approach and the baseline system did not cope well with the sudden change of emotions in the personal contexts due to unexpected topic changes. In future work, the topic theme detection is also intended to be employed for affect detection in personal contexts to improve its performance. Also, in Sect. "Related Work," an affect detection system, @AM, developed by Neviarouskaya et al. [18] was mentioned. The @AM system focused on affect detection from individual sentences that were extracted from the Experience Web site and described personal experiences. Their development mainly used linguistic features of individual sentences to detect affect and did not take any context into consideration, while the HMM-based approach discussed here uses personal emotional profiles for affect interpretation with the support of the naïve Bayes classifier. Although there are differences on technical aspects between the HMM-based affect detection and the @AM system, the two systems are still compared to get some indication of our system's performance. Their @AM system was used to annotate 1,000 sentences using three labels (positive, negative and neutral). Their system achieved 92 % precision scores for the annotation of positive inputs, 91 % precision for negative inputs and 47 % precision for neutral ones. Neutral sentences still challenged their system's performance greatly. Tested using a simple repeated hold-out validation, the HMM-based affect detection from the personal contexts generally performs stably on the detection of each category of emotional and neutral expressions comparing with the @AM system. However, more testing is needed to further prove the HMM-based system's robustness.

Table 8 Standard deviations for the precision and recall scores obtained from the three runs of the hold-out validation

	Standard deviations for precision	Standard deviations for recall
Positive	0.0483	0.0557
Negative	0.0237	0.0122
Neutral	0.032	0.0423

As discussed earlier, since the standard deviation can also give some indications of the system's stability, we have provided standard deviation results of the precision and recall scores obtained from these three runs of the hold-out validation in Table 8.

The standard deviations obtained from the three hold-out validations shown in Table 8 are comparatively low and not statistically significant. This also means that the data points tend to be very close to the mean precision and recall scores shown in Table 7 for the performance of HMM. Thus, the deviation results indicate that the system performed reasonably stably, although more iterations of the hold-out validation are needed in order to draw a stronger conclusion.

Moreover, accuracy rates for the performance of the affect sensing in social interaction contexts using neural networks are also provided. The social context testing set has 190 examples with 38 % negative examples, 36 % neutral and 26 % positive ones. One human judge has also provided topic theme annotations for these 190 social context inputs. The latent semantic analysis implemented using the semantic vector package achieves an accuracy rate of 88 % for the topic theme detection. The detected affective states by the Backpropagation algorithm are also converted into binary evaluation values, and it obtained a 75 % accuracy rate for positive emotions, 73 % for neutral expressions and 83 % for negative emotions by comparing with the annotation of one human judge. The results indicate that other characters' emotional influence to the speaking character embedded in the interaction contexts is well recovered in this application context using neural net inference. The future work will also verify whether the results of evaluation are also significant using other statistical testing approaches.

Evaluation of the System's Robustness

The linguistic and grammatical signals for personal and social situation descriptions presented in Tables 1 and 2 are

gathered from the improvisational transcripts and articles published on the Experience Web site. Most of these linguistic phenomena represent typical personal expressions and social interaction descriptions in general interaction situations and show generalization features across different application domains. For the topic theme (including metaphor) detection, articles representing eight topic themes extracted from the Experience Web site and metaphorical examples taken from two different metaphor sources have been included in the training documents. These examples are gathered outside of the chosen testing scenarios (Crohn's disease and school bullying) in order to gain system's robustness. Thus, both of the naïve Bayes classifier and semantic-based topic theme detection show generalization features and are able to allow the system to deal with general interaction contexts outside of the chosen scenarios to some degree.

Also, both the HMM-based reasoning and the neural network-based affect detection are built using supervised learning algorithms and are capable of learning new emotional inclinations and adapting to newly unseen emotional situations based on further learning of any test interaction contexts. Thus, with appropriate further training, they are capable of improving their generalization abilities and robustness to fulfill demanding affect detection requests across scenarios. Therefore, in order to evaluate the robustness of the affect detection processing at different stages, four transcripts of the school bullying scenario are employed. Other example articles from the Experience Web site and metaphorical examples gathered across scenarios and from the dedicated metaphor resources are also used to further evaluate the generalization ability of the personal and social context classification and topic theme detection. The details are presented below.

Three sample improvisational transcripts of the school bullying scenario and 10 articles related to Education and Family & Friends from the Experience Web site are selected. Two human annotators are also employed to classify personal and social contexts for the testing transcripts. Agreed annotations for the 270 inputs from three sample improvisational transcripts and 60 inputs from the 10 articles are used as gold standards to evaluate the personal and social context classification using the naïve Bayes classifier. The results are presented in Table 9.

Table 9 Further evaluation results of personal and social context classification

	Precision	Recall	F-measure
Naïve Bayes classifier			
The personal context	0.88	0.793	0.834
The social context	0.816	0.895	0.854

Comparing with the previous performance of the personal and social context classification in the Crohn's disease scenario, although the new set of results shows worsened precision scores, the classification results of personal and social contexts using the school bullying scenario and articles from the Experience Web site are reasonable. Affect annotation of these 330 inputs is also provided by both of the human annotators using the positive/negative/neutral labels in order to provide initial evaluation of affect detection from personal and social contexts. The inter-annotator agreement between human judges is 0.806 with 290 inputs showing agreed annotations. They are used to evaluate the performance of HMM-based emotion detection in personal contexts and neural network-based affect detection in social contexts in these new application domains. Among the 290 inputs, there are 130 inputs belonging to personal contexts and 160 inputs indicating social contexts. For the HMM-based affect detection, different training sample transcripts with 250 agreed annotations of the school bullying scenarios are used to provide further training of the existing model, so that it will provide updated transition and new observation probabilities for each character in this new test scenario (e.g., the observation probability for the bullied character, Lisa, to show the "angry" emotion).

Moreover, in order to improve the generalization abilities of both the HMM- and neural network-based affect detection, each test data set is also recorded and added to the corresponding training set. Thus, the training data are enriched gradually with the running of the test emotional contexts. These testing emotional sequences may also help the system to cope with any new emotional inclination because of each character's creative improvisation.

The 130 test personal context inputs with 39 % negative, 25 % positive and 36 % neutral expressions are used to further evaluate the HMM-based affect detection in personal contexts, and the detected affective states are also converted into positive, negative and neutral values. The updated HMM obtains a new set of precision and recall scores presented in Table 10. Generally, negative and positive expressions are reasonably recognized. However, in some cases, neutral expressions are regarded as emotional, which poses challenges to the current affect

Table 10 Evaluation results for the HMM-based affect detection using the new test data set

	Precision	Recall	F-measure
The HMM-based affect detection			
Positive	0.816	0.895	0.854
Negative	0.88	0.75	0.81
Neutral	0.788	0.901	0.841

detection processing due to sudden topic changes. Since the 130 inputs contain 83 inputs extracted from the school bullying scenario and 47 inputs taken from the articles from the Experience Web site, the evaluation results shown in Table 10 also show some initial indications of the robustness of the HMM-based emotion detection in personal contexts.

The 160 social context inputs are also used to further evaluate the generalization ability and robustness of the topic theme detection and the neural network-based affect detection. Human judges also provide topic theme annotations using the 13 categories of topic themes. Comparing with the annotations provided by one human judge, the semantic-based topic theme detection achieves a 76 % accuracy rate. The latent semantic analysis implemented using the semantic vector package proves to perform well for the topic theme detection using sample inputs from another scenario. Another 26 articles taken, respectively, from the Experience Web site (16 articles) and the metaphor online resources mentioned earlier (10 files) are also used to further evaluate the performance of LSA-based topic theme detection. The test articles borrowed from the Experience Web site belong to the following themes: "Crohn's disease," "bullying," "family care," "food choice," "school lunch," "school uniform," "suggestions" and "disagreement." Two articles with six sentences on average per article for each of the above themes are included in the test documents. Metaphorical examples borrowed from the ATT-Meta databank and the previously mentioned metaphor Web site include the following phenomena: cooking, family, weather, farm and ideas metaphors. Each metaphorical input consists of one test document. Two metaphorical documents are included in the test files. The metaphor examples used include the following: "I couldn't bear to touch the memories (ideas metaphor)," "He began to market himself as an author (farm metaphor)," "His ideas began to bear fruit (ideas and farm metaphor)," "The boss thundered into the room (weather metaphor)," "Children have an enormous appetite for learning (cooking metaphor)" and "The work is in its infancy (family metaphor)."

Thus, 26 files in total are used to further evaluate the robustness and generalization abilities of the topic theme detection. The LSA-based topic detection achieves an 81 % accuracy rate for the topic classification for the 16 test articles borrowed from the Experience Web site and an accuracy rate of 80 % for the recognition of the 10 metaphorical expressions. In future work, a bigger sample size will be used to further evaluate the system's efficiency and robustness.

As mentioned earlier, the 160 example inputs from the bullying scenario are also annotated by the human judges using three labels of positive, negative and neutral. The Backpropagation algorithm with the trained weights using the previous example transcripts of the Crohn's disease scenario is applied directly to the new test data set of the school bullying scenario. Moreover, each test input from the new scenario is also appended to the training samples, so that the algorithm is able to gradually pick up new emotional situations in this new test scenario. The 160 inputs contain 36 % negative, 29 % positive and 35 % neutral expressions. Relationships between characters in the bullying scenario are also predefined, for example, Mayid (the big bully) having tense relationships (−1) with Lisa (the bullied victim) and Elise (Lisa's best friend), but a medium relationship (0) with Dave (played by the AI agent) and the school teacher, and so on and so forth for other characters. The outputs of the neural network are also converted into binary values. Comparing with the annotation provided by one human annotator, the Backpropagation algorithm achieves a 73 % accuracy rate for the negative emotions, a 69 % accuracy rate for the positive emotions and 65 % for the neutral expressions. The current neural network implementation can also be easily extended to a much more sophisticated topology in order to improve its learning capabilities. For example, in future work, the input layer will employ a node for each different sentence type, and for each of the four characters, a node for each different emotion.

Although the further robustness testing based on the example transcripts from the school bullying scenario and articles borrowed from the Experience Web site and the metaphor resources has shown worsened affect detection results at different stages, the system (especially the HMM-based and neural network-based affect detection) shows reasonable generalization capabilities in handling inputs from other comparatively unfamiliar scenario and sources for affect interpretation. The above evaluation results also give some indications of the generalization abilities of the personal and social context classification and the semantic-based topic theme detection. Other strategies (e.g., the use of semi-supervised and unsupervised learning) will also be considered in order to improve the robustness of the system in future work.

Conclusions

The overall context-based affect detection model integrated with the original affect sensing component is embedded in the AI agent. It has generally made the AI agent perform better for the revealing of emotions in the personal and social contexts. In future work, more example transcripts from different scenarios (such as bullying and career training) and articles from the Experience project will be used to further improve the context-based affect detection

performance. Moreover, future development will also extend the emotion modeling with the consideration of personality and culture. Topic extraction to support affect interpretation directly will also be considered, for example, the suggestion of a topic change indicating potential indifference to the current discussion theme. It will also ease the interaction and make human characters comfortable if the AI agent is equipped with culturally related small-talk behavior. It is also noticed that the training and testing transcripts contain more negative inputs than positive ones due to the nature of the chosen scenarios (in this case Crohn's disease and bullying). The future development also intends to employ partially supervised learning to deal with imbalanced affect classification.

Acknowledgments We thank Dr. Meurig Beynon for his help with proofreading and detailed feedback on the technical aspects and manner of writing. We also thank Dr. Zach Pardos and Dr. Tristan Nixon, the organizers of a tutorial, Parameter Fitting for Learner Models, associated with 2012 International Conference on Intelligent Tutoring Systems.

References

1. Zhang L. Exploitation of contextual affect-sensing and dynamic relationship interpretation. ACM Comput Entertaint. 2010; 8(3) (article no.: 18).
2. Ortony A, Clore GL, Collins A. The cognitive structure of emotions. Cambridge: Cambridge University Press; 1998.
3. Ekman P. An argument for basic emotions. Cogn Emot. 1992;6: 169–200.
4. Jess, the rule engine for the Java platform. http://www.jessrules.com/. Accessed in Feb 2012.
5. Briscoe E, Carroll J. Robust accurate statistical annotation of general text. In: Proceedings of the 3rd international conference on language resources and evaluation, Las Palmas, Gran Canaria; 2002. p. 1499–1504.
6. Zhang L, Barnden JA, Hendley RJ, Wallington AM. Exploitation in affect detection in improvisational E-drama. In: Proceedings of the 6th international conference on intelligent virtual agents. California, USA. Lecture notes in computer science. 2006;4133: 68–79. Springer.
7. Zhang L, Barnden JA, Hendley RJ, Lee MG, Wallington AM, Wen Z. Affect detection and metaphor in E-drama. Cont Eng Educ Life-Long Learn. 2008;18(2):234–52.
8. Zhang L, Gillies M, Dhaliwal K, Gower A, Robertson D, Crabtree B. E-drama: facilitating online role-play using an AI actor and emotionally expressive characters. Int J Artif Intellig Educ. 2009;19(1):5–38.
9. Kappas A. Smile when you read this, whether you like it or not: Conceptual challenges to affect detection. IEEE Transact Affect Comput. 2010;1(1):38–41. doi:10.1109/T-AFFC.2010.6.
10. Hareli S, Rafaeli A. Emotion cycles: on the social influence of emotion in organizations. Res Organ Behav. 2008;28:35–59.
11. Picard RW. Affective computing. Cambridge MA: The MIT Press; 2000.
12. Prendinger H, Ishizuka M. Simulating affective communication with animated agents. In: Proceedings of Eighth IFIP TC.13 conference on human–computer interaction. Tokyo, Japan. 2001, p. 182–189.
13. Mehdi EJ, Nico P, Julie D, Bernard P. Modeling character emotion in an interactive virtual environment. In: Proceedings of AISB 2004 symposium: motion, emotion and cognition. Leeds, UK; 2004.
14. Gratch J, Marsella S. A domain-independent framework for modeling emotion. J Cogn Syst Res. 2004;5:269–306.
15. Aylett A, Louchart S, Dias J, Paiva A, Vala M, Woods S et al. Unscripted narrative for affectively driven characters. IEEE Comput Graphics Applicat 2006;26(3):42–52.
16. Endrass B, Rehm M, André E. Planning small talk behavior with cultural influences for multiagent systems. Comput Speech Lang. 2011;25(2):158–74.
17. Liu H, Singh P, ConceptNet. A practical commonsense reasoning toolkit. BT Technology Journal, Volume 22, Kluwer Academic Publishers; 2004.
18. Neviarouskaya A, Prendinger H, Ishizuka M. Recognition of affect, judgment, and appreciation in text. In: Proceedings of the 23rd international conference on computational linguistics, Beijing, China; 2010:806–814.
19. Mateas M. Ph.D. thesis. Interactive drama, art and artificial intelligence. School of Computer Science, Carnegie Mellon University; 2002.
20. Zhe X, Boucouvalas AC. Text-to-emotion engine for real time internet communication. In: Proceedings of international symposium on communication systems, Networks and DSPs, Staffordshire University, UK; 2002:164–168.
21. Ptaszynski M, Dybala P, Shi W, Rzepka, R, Araki K. Towards context aware emotional intelligence in machines: computing contextual appropriateness of affective states. In: Proceeding of IJCAI; 2009.
22. Craggs R, Wood M. A two dimensional annotation scheme for emotion in dialogue. In: Proceedings of AAAI spring symposium: exploring attitude and affect in text; 2004.
23. Werry C. Linguistic and interactional features of internet relay chat. In computer-mediated communication: linguistic: social and cross-cultural perspectives. Pragmatics and beyond new series 39. Amsterdam: John Benjamins; 1996:47–64.
24. Bouckaert RR, Frank E, Hall M, Holmes G, Pfahringer B, Reutemann P, Witten IH. WEKA-experiences with a java open-source project. J Mach Leran Res. 2010;11:2533–41.
25. Landauer TK, Dumais S. Latent semantic analysis. Scholarpedia. 2008;3(11):4356.
26. Widdows D, Cohen T. The semantic vectors package: new algorithms and public tools for distributional semantics. The fourth ieee international conference on semantic computing (IEEE ICSC2010); 2010.
27. Zhang L, Barnden JA. Affect and Metaphor Sensing in Virtual Drama. Int J Comput Games Technol. 2010; 2010 (article ID 512563).
28. ATT-Meta project databank. http://www.cs.bham.ac.uk/~jab/ATT-Meta/Databank/. Assessed March 2012.
29. Rabiner LR. A tutorial on hidden markov models and selected applications in speech recognition. Proc IEEE. 1989;77(2):257–86.
30. Wang Z, Lee J, Marsella S. Towards More comprehensive listening behavior: beyond the bobble head. In: Proceedings of the 10th international conference on intelligent virtual agents. Iceland; 2011.