Acta Cybernetica **16** (2004) 625–642.

Extraction of Affective Components from Texts and Their Use in Natural Language Dialogue Systems*

Gábor Tatai† and László Laufer‡

Abstract

We are carrying out a research in the field of Human Computer Interaction and developing a natural language dialogue system in Hungarian. In the beginning chapters we briefly describe the architecture of our dialogue system, BotCom with examples of its semantic processing capabilities. We give examples of how the system is handling the topics of the discussion, how the dialogue history is being used in order to enhance the reply generation. In the subsequent parts we give an overview of the emotional state detecting, processing and generating module, called GALA, which is founded on the grounds of Robert Plutchik's emotional model. We show how BotCom is utilizing the detected emotional loads of the user's messages, therefore enabling the chatterbot to give relevant answers both semantically and affectively. In the final chapter we explain how the database of GALA was filled up with expressions assigned to their emotional loads. We also describe a graphical user interface (GUI) being designed to model the changing emotional loads in dialogues, songs and poems, and how it can be used for the emotional labeling of the phrases.

Keywords: Affective Computing, Emotional Modeling, Dialogue System, ECA, Chatterbot

1 Introduction

Many embodied conversational agents (ECAs) are targeting the Internet. However, systems that are bound to this global network not only benefit from several advantages of the huge amount of accessible information provided by this medium, but inherit its common problems as well. Among those are the difficulties of relevant search, complexity of available information, unstructuredness, lack of navigation support [3], bandwidth limitations and so forth. Another difficulty in order to

*Presented at the 1st Conference on Hungarian Computational Linguistics, December 10–11, 2003, Szeged.

†Department of Computer Science, University College London, Gower Street, WC1E 6BT, London, UK, e-mail: g.tatai@cs.ucl.ac.uk

‡Department of Psychology, ELTE University of Sciences, 1064 Izabella utca 46, Budapest, Hungary, e-mail: laufer@erg.bme.hu

achieve better believability of the agent is to make it capable of detecting the user's emotions and giving emotionally adequate responses [2].

Although many researchers recommended to apply emotions in artificial intelligence systems, not only in conversational systems, but generally in cognitive architectures and decision making systems [11, 6], it is still rare to find applications utilizing this type of cognition in problem solving. Also in modern chatterbot systems the designers do not really pay ample attention to emotional modeling, or the solution they use is far from being sophisticated.

For a human being a communication partner without even the simplest emotions can be very boring. Adequate detection and communication of emotions are essential in every day interactions [4], and human-computer interactions as well [17]. Not mentioning tutoring and general educational systems which are one of the primary potential application areas of the chatterbots. Probably most of us have had unpleasant experiences with poker-faced teachers at oral examinations, when there was no apparent sign of understanding, approval or disapproval. Therefore it is almost a mandatory requirement to equip an ECA with a reasonably refined emotional model. Such a communication partner can be more alive, convincing and entertaining as noted already in earlier experiments [17].

Thus we aim to develop a more sophisticated emotional module, enabling the treatment of the continuously changing emotional status during a conversation

There is no space to discuss all the features and interesting implementation experiences with our ECA system in this paper. Therefore we are only focusing on our system architecture with a special emphasis on the emotional module. We think others might find interesting and get some thought-provoking ideas in our theoretical approach and its practical implementation.

2 Architecture and Technical Considerations

Our ECA system called BotCom is primarily a client-server architecture with a chatterbot development module, BotMaker (Fig. 1.).

The ECA is embedded in web pages and displayed in a web browser. The client visualizes the messages of the chatbot, the animations representing its emotions and the gags, as well as forwarding users' messages to the server. Portal engine integration enables access to the indexed database of keywords and full articles of the site, providing extendibility to the knowledge base of the chatterbot.

On the server side a Java servlet collects the messages and transmits them to a multiplexer which queues or distributes them among the servers for parallel operation. Messages are processed separately, in each logged-in user's context, which are responsible for the communication and stores user preferences and personal details. The dialogue system matches them against the patterns stored in the knowledge base, and the answer, which consists of the status obtained from the GALA emotional model (see below) and the browser controlling commands (if any), are sent back to the client.

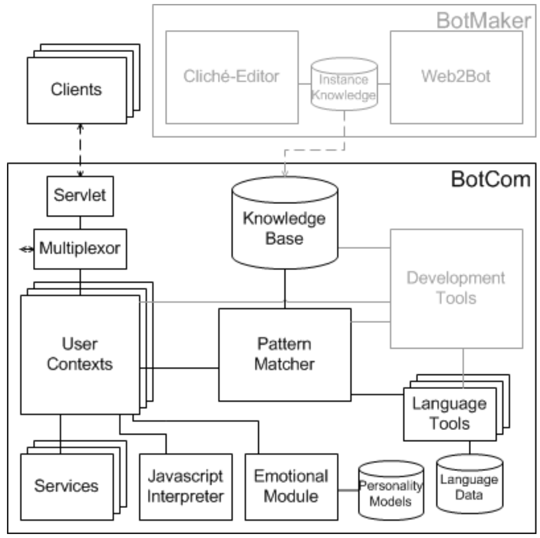

Figure 1: The main components of BotCom and the BotMaker service and development tools

Development tools for uploading and managing the knowledge base of the ECA and adding functionalities include the BotMaker and the Cliché Editor.

3 Message Processing

An ECA have to perform complex procedures, in order to give an adequate answer to the user message. The adequacy of the emotional load of the answer is a separate issue, we are discussing it in the description of the emotional module. On Fig. 2. we tried to give an overview of the response generation procedure.

As it is visible on the flowchart, the core of the system is a pattern matcher working on a knowledgebase of dialogue segments. One of the key issues when creating a communicational agent is to set up a structure and to establish its knowledgebase. On Fig. 3. we are showing the knowledgebase editor interface of BotCom, Botdev. The dialogues segments are structured into topics and subtopics, these are important to determine the semantic difference between to sentences. It was also important to give alternatives to each sentence entered, the meanings of the alternatives are the same, but they can differ in their emotional load. The interface enables its user to write scripts in a special script language to define the circumstances of giving the specific answer, it also gives an opportunity for testing the chatterbot with using the entry.

The incoming message is forwarded to the pattern matcher and also submitted to 3 procedures for further analysis (see Fig. 2.). Syntactic processing (on the right)

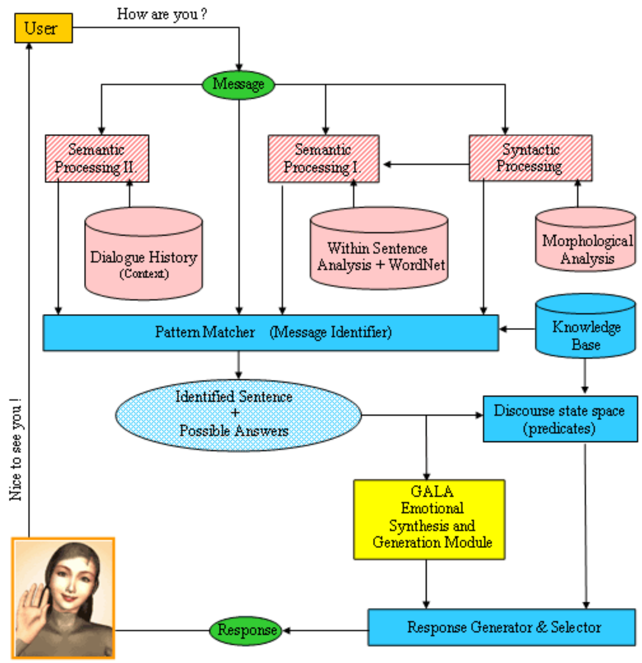

Figure 2: The message processing workflow of our chatbot system BotCom

makes sure, that the spelling mistakes are corrected, also other common mistypes (partly due to the difference between English and Hungarian keyboards) are taken into consideration. This sub-module is using morphological analysis tools available for Hungarian language. The output is forwarded to the first semantic processing sub-module (beside the pattern matcher).

The first semantic processing sub-module (see Fig. 2. upper-middle part) is responsible for semantic processing of words and phrases within the sentence. In order to carry out this task, it utilizes semantic-nets: OSZK's Thesaurus [13], and other free thesaurus dictionaries and WordNet databases. This way if the pattern matcher did not identify the incoming message in the knowledgebase, we are able to provide further suggestions with identical meaning to be matched among the existing dialogue segments.

The second type of semantic processing sub-module (see Fig. 2. upper-left part) is to identify the role of the sentence in the dialogue. We expect the incoming message to have certain attributes in a certain phase of a discussion. We utilize the dialogue history, and conclude which part of the knowledgebase can be excluded and which part should be considered with a higher weight in the pattern matching algorithm.

In the second phase of the response generation, after the pattern matcher identi-

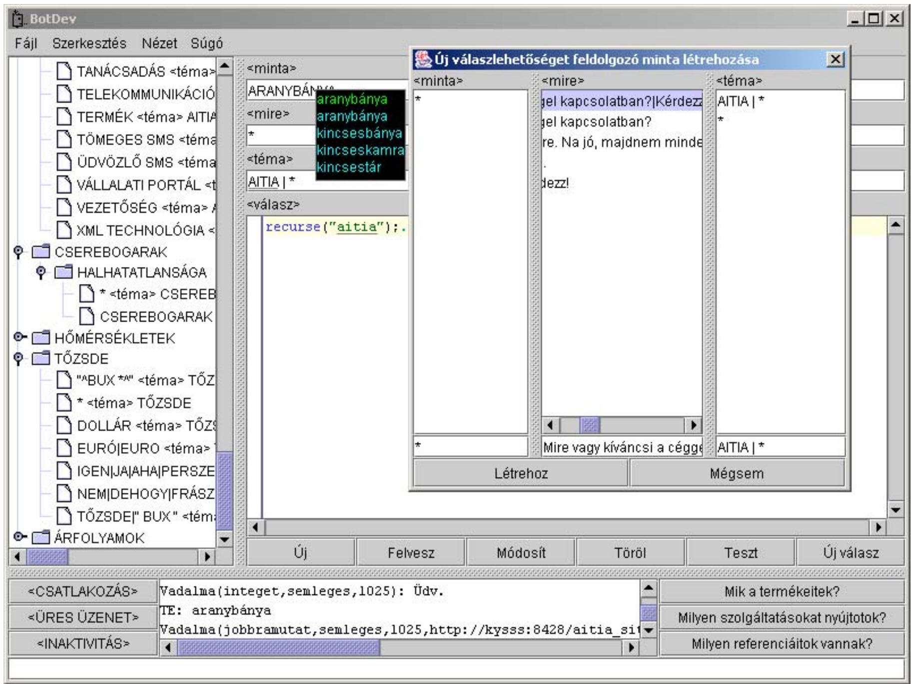

Figure 3: The knowledbase editor of BotCom

fied a sentence in the knowledge base, expectedly with identical semantical content to the incoming message, the pattern matcher gives the possible responses as well. The dialogue as a whole has a position in a state space. This position is determined mainly by different predicates describing the user (his habits, properties, other attributes assumed from this and the previous discussions) and the chatterbot (likes, dislikes, opinions stated previously) and other circumstances of the conversation. These are represented in the discourse space state, and influencing which possible answer is to be chosen from the different possibilities offered by the pattern matcher.

The Gala emotional module, described below, also processes the incoming messages and possible answers and chooses the best answer, which is emotionally adequate for the chatterbot to reply. Gala keeps track of the emotional aspect of the discussion. With its multi-layered architecture it is possible to model the emotional influences of the consequent user messages and to generate emotional moods for the chatbot, and also to create different personalities on the basis of which the ECA system gives different responses in similar situations.

On the basis of these information the response generator select an answer from the possibilities, and the chatterbot system displays it, together with the animation assigned to the message in its user interface.

4 Affective Computing

In this paragraph we would like to describe the emotional module of our system, but first say a few words about it's theoretical basis within artificial intelligence research.

For about a decade, the research area of Affective Computing has been flourishing. This field, as Rosalind Picard, a leading researcher in affective computing defined, deals with problems where "computing is related to, arises from or deliberately influences emotions" [16].

When researching into emotional computing one should certainly determine what is considered as an emotion. Since various recent advances in psychology and neurology are making cognitive scientists and psychologists to rethink the origins and behaviors of various emotions, there is presently no universally accepted emotional model.

On the other hand, there are some theories and foundations which we accept as basis. The early work on emotions by Darwin [5] and others (e.g. [10, 20, 19, 7]) who distinguished discrete categories, so-called basic emotions have resisted the test of time at least for their main categories and the description of these emotions. Others, however, have emphasized the continuous dimension of emotions [18], which is likely to be closer to the real operation of emotions. Nevertheless, it seems that the reflections on the emotions which identify them within the self, thus making the self aware of them, introduce a discrete element into the anyway continuous neurobiological system. The discreteness is even more noticeable when the emotion is translated to language and expressed so [12].

The most widely accepted set of basic emotions seems to be the eight basic emotions described by Tomkins (fear, anger, anguish, joy, disgust, surprise, interest and shame). This set is a reasonable extension of Darwin's first identification of the basic set (joy, anger, fear) [5], and, more or less, others' categories also correspond to it.

Beyond these, there is a well-known and widely accepted classification of emotions. This is the OCC Cognitive Model [15, 21] which provides not only a definition of them, but also a hierarchy based on the target of the emotion (self, other) and other meta-categories such as consequences of events, aspects and actions. This is a strict and complex cognitive model of the key emotions, however, in [14] Ortony himself admitted that the OCC model is far too complicated for the development of emotional characters and proposed a simplified model. In order to use this approach, one does not only need to model the emotions themselves, but it is necessary to adopt the entire cognitive model including all its processes and structures. In that case, we would be able to manage the emotions appropriately, however this would be very cumbersome. Even a simple model implies a complex architecture, as is clearly visible from reports on similar projects (Integrating Models of Personality and Emotions into Lifelike Characters) at DFKI [1].

5 The GALA Hierarchical Emotion Processing Model.

5.1 The Plutchik Model

Our goal was to select a sophisticated emotional model complex enough to provide ample workspace, but not bound to a net of cognitive processes. We have found that the model suggested by the psychologist Robert Plutchik is an appropriate starting point for further description and generation of synthetic emotions [22]. He defined eight basic emotions that are just a little different from those of Tomkins. In addition he suggested another dimension, intensity (with 3 levels). Therefore we get 24 emotional states. A novelty of his proposal is that the emotional space is mapped to an upturned 3D cone (see Fig. 4), where the positioning of a particular emotion reflects psychological distances and intensity differences between states. The origo is in the apex of the cone expressing neutrality ("no emotion"). We have selected this model for implementation and incorporated it as a part of our more complex, layered architecture. This geometrical approach allows easy manipulation of the emotions.

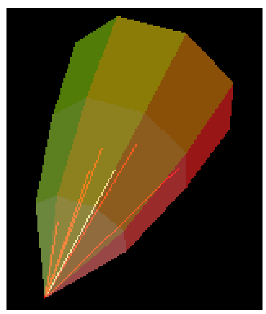

Figure 4: The 3D cone representation of emotions

5.2 The Architecture of GALA

In order to create a usable synthetic emotion model for a chatterbot that communicates mainly via text messages, it is essential to design an appropriate mapping scheme between the emotions and the expressions that the chatbot sends or receives (Fig. 5).

5.3 Layer One – Message Act Processing

The connection between the text messages, better said, message acts and emotions is thoroughly investigated by M. A. Gilbert in his work [9] and later in [8]. This work was particularly useful because it has pointed out that there is a reasonable

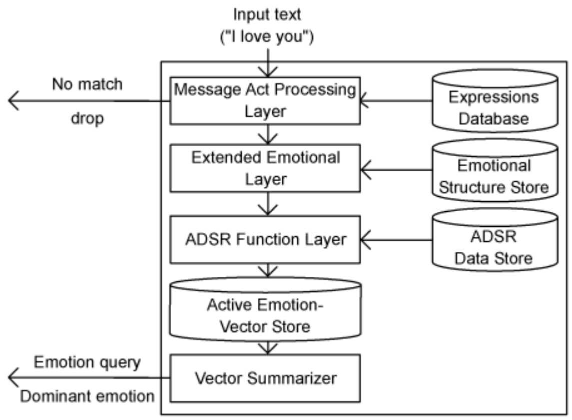

Figure 5: The architecture of the GALA engine and the processing of a message act

difference between the well known concept of a speech act and a message act that the meaning of the latter also carries emotional load. According to Gilbert's definition, a message act is similar to the speech acts; "A message act, being analogous to an utterance act involves an expression of emotion that is identifiable to the recipient or observer." (for example: threat, fear etc.). The speech act is used when analyzing sentences, query semantic information and emphasizes the verbal aspect of the communication. In contrast the message act focuses on various forms of the communication and used to express dialogues and emotional aspects in addition to the semantic, meaning level.

When processing the input expressions, initially they are passed to the GALA processing module. In the first step the conversational elements (words, expressions, sentences) in the dialogue are assigned to the emotional message acts, which are processed at the uppermost layer. If GALA finds a match of the input expression it passes to the second layer, otherwise the element is dropped and does not alter the emotions of the chatbot. The set of recognizable elements can be extended and the system can be trained.

5.4 Layer Two – Extended Emotional Layer

A message act almost never expresses a pure basic emotion. Therefore we have defined a second layer that allows the construction of mixed emotions that are more closely related to the message acts. In this layer one or more basic emotion components can be assigned to the individual message acts. These components can be weighted, so a message act could be represented by a mixture of e.g. 20 % sadness, 40 % anger, 40 % fear. There is a predefined database, the "Emotional Structure Store" (ESS) that allows the storage of these complex emotional structures and their assignment to the appropriate message acts. The basic emotion components are represented by vectors and thus the resulting mixed emotion is also stored and manipulated in a vector. On the 3D surface suggested by Plutchik, the origin of these component vectors is the apex of the cone, and they point towards the center of the particular surface segments that are assigned to that basic emotion. In general we did not introduce a limitation on the number of basic emotions that can build up the resultant emotion which is associated with the message act. However, according to our experiments and anticipated real life requirements, it seems that there is a practical upper limit. It appears that most of the expressions can be realistically covered by a mixture of maximum 4 of the 24 predefined emotions in a given situation.

Summarizing: the function of the second layer is to break up the message acts into basic emotions taken from the Emotion Structure Store and forward them to the third layer.

5.5 Layer Three – ADSR Refinement Function

It is well known that emotions are time dependent, so we had to manage that the amplitude of their synthetic pairs change over time as well, since it is not handled by the original model. In addition, the characteristics of this change vary by emotion. As it was not our goal to replicate the unclear nature of the chemical and neurobiological change of emotions in the brain, we chose a new controllable approach, hoping to achieve a realistic result.

Therefore, in the third layer we used a well known function, called ADSR to describe the change of emotions over time. Signal processing and musical applications widely apply the method of dividing a signal to four main parts. Thus a cover curve of a signal can be defined by the Attack, Decay, Sustain and Release values (see Fig. 6.).

In our model all basic emotions can have their own ADSR functions, which ensure unique time characteristics for each of them. Moreover, the intensity of each emotion can be tuned in order to provide opportunity of the relative scaling of the emotions. So, in a particular situation the amplitude (intensity) of joy can be stronger than the intensity of the rest of the emotions still maintaining the same ADSR function and thus its behavior over time.

5.6 Computing of the Dominant Emotion

The ultimate goal of the process is to determine the dominant emotion since we have a strong constraint – we can play only one animation at a time. The third layer, after the decomposition of the message acts into emotional structures and further into basic emotions, selects the ADSR and intensity values corresponding to time t ("now") for each basic participating emotion. These emotion vectors are kept in the Active Emotion-Vector Store (AEVS). The actual resulting emotion vector is calculated by summing and normalizing these active vectors. This final vector will contain the effect of the actual message act (through its decomposed

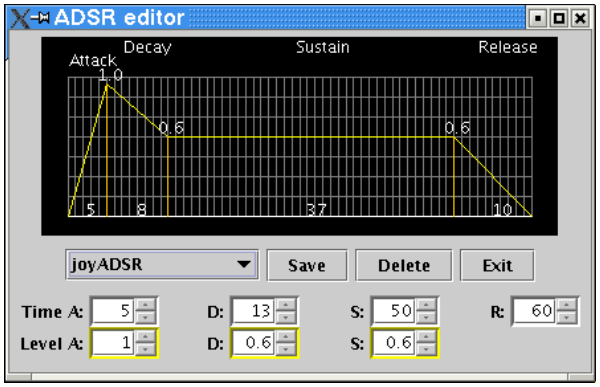

Figure 6: The ADSR function editor (It is available for each emotion or emotion structure).

participating emotions) and the fading effect of previous message acts which are already in the Active Emotion-Vector Store.

The processing of the normalized sum of the vectors produced by the three layers can be expressed in the following closed formula. Please note, that we are not calculating by merely using this formula, since it is a pipeline process and partial results must be calculated previously. The formula is provided for better understanding only.

$$\underline{r}(t) = \sqrt{\frac{\sum_{k=1}^{n} \underline{e}_k^2(t)}{n}} = \sqrt{\frac{\sum_{k=1}^{n} (i_k \underline{E}_k f_k(t - t_{0k}))^2}{n}}$$

$\underline{r}$. : resulting emotional state vector
t : actual time
n : number of active emotion vectors (where "active" indicates a non-zero vector)
$\underline{e}_k$: the k^{th} active emotion vector
i_k : the maximal intensity of $\underline{e}_k$
$\underline{E}_k$: unit vector of $\underline{e}_k$
f_k : ADSR function belonging to $\underline{e}_k$
t_{0k} : the activation time of the k^{th} emotion

The result vector can be drawn and it will point to some location within the cone. The dominant emotion for that t moment will be that basic emotion (out of 24) which is the closest to the endpoint of the vector sum.

As time goes, the system dynamically recalculates the emotions by repeating this decomposition process from layer one to three. Therefore a sequence and pattern will be displayed as emotions that can be traced on the emotion editor (see Fig. 4 later) appear and fade.

One may notice that our original standpoint was to create a fairly straightforward emotional model, however, it may seem to become moderately complicated. In fact, most of the components are quite simple and reused such as the emotion handling and the ADSR functions. The only thing that happened is that we have used them in multiple layers enabling complexity with simple, reused and overrideable components. This approach is similar to the theoretical and experimental results of psychology, where complex memory and emotional models arise from using very simple elements.

6 Administration Components of GALA

For administrating the GALA emotion module we created an application with graphical interface. The interface can load continuous texts, and visualize the emotional labels that are already stored in the emotional expression store assigned to the expressions in the content. This way through the administrator panel we can read out the text, and follow the changes of emotional states in time.

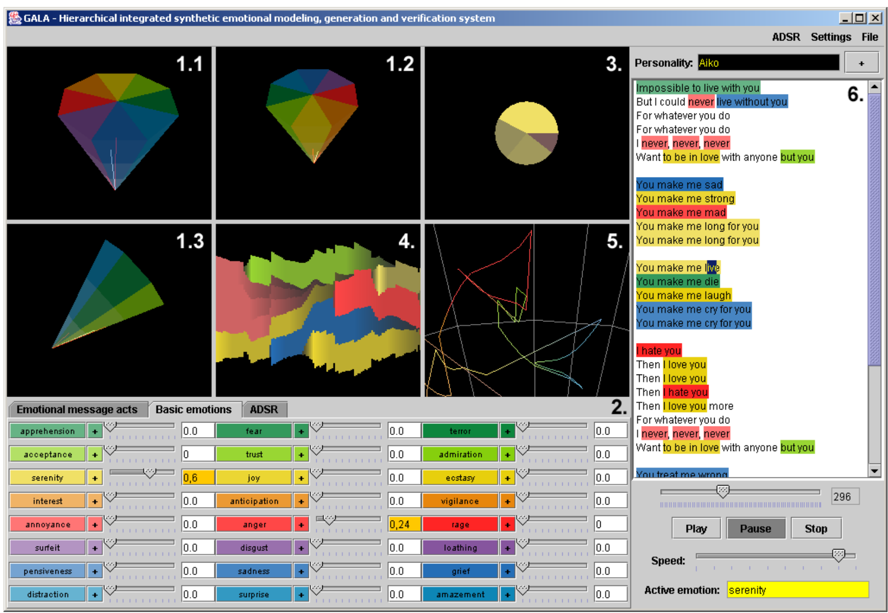

Figure 7: The emotion editor for administrating and testing the GALA model.

6.1 Tracing and Visualization Components of GALA

This interface allows the carrying-out of reproducible experiments on the emotional model, while the various visualization tools allow us to trace the process of emotion generation. In the following paragraphs we briefly describe its components.

6.2 Camera views

The three rotateable and zoomable camera views (see Fig.7. panel 1.1, 1.2, 1.3) display the cone with the vectors (active emotion vectors and the resulting emotion vectors) inside. These vectors are representing the different emotional influences caused by the previous passages, they are constantly changing, and fading away as a result of the ADSR function manipulates their length. The resulting emotional vector is also being displayed with the thicker white vector visible on all three panels.

6.3 Basic emotions monitor and Text panel

The basic emotion monitor shows the intensities of each basic emotion at a given moment (see Fig.7. panel 2.). It provides a great help to refine the basic emotions assigned to each phrase in the text. The cursor on the Text panel (see Fig.7. panel 6.) shows us which part of the text is being currently read. On the Basic emotion monitor it is possible to readjust the intensity of the 24 basic emotions participating in the composition of the complex emotional load of the current phrase. At the button part of the Text panel (see Fig.7. panel 6.) the user can start, stop and choose a position in the reading or adjust the speed for better following the changes in the resulting emotion vector. The dominant emotion (or active emotion) is calculated from the projection of the resulting emotion vector to the surface segments of the cone and being displayed at the button left corner of the panel. This emotional state is the output of the GALA emotional subsystem.

6.4 Active emotion component diagram

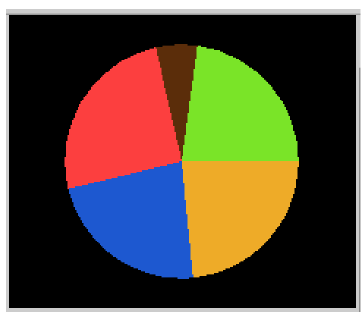

Figure 8: Active emotion component diagram.

Besides the camera views, there is another possible way to track and display active emotions (see Fig. 8). The sectors of the circle symbolize the active emotion components at any given time and their colors correspond to that of the appropriate emotion. The areas of the sectors are directly proportional to the percentage shares of the basic emotions from the current resulting emotion vector, the present composite-emotional state. The length of this vector, the intensity of the composite emotion is represented by the radius of the circle.

6.5 Emotion-history panel

Figure 9: Emotion-history panel.

This panel has an important role in displaying the emotions synthesized throughout the whole conversation. When "reading" the text, in this case a poem, the system generates a colorful pattern that gives us a picture of the poem and enables us to get an overview of the emotional characteristics of the analyzed poem (see Fig. 9).

The X axis of the pattern stands for the time dimension: a vertical slice shows the active emotions (represented by the appropriate colors) at a given time, proportionate to their actual intensity. The height of a vertical slice represents the length of the resulting emotion vector.

As time goes by, new slices ("emotion ensembles") are added to the picture from right, resulting in the continuous condensation of the existing piece.

6.6 Emotion tracing panel

This panel allows us to follow and depict the route of the resulting emotion (the sequence of emotional change) of the conversation, in the function of time. For better visualization, we used only the wire-frame view of the Plutchik cone (see Fig. 10).

This visualization panel can help us in the development of vector summing or norming: it may reveal the possible shortcomings of the computing of the resulting vector (e.g. frequent vector activity in the high intensity area of cone, overly exaggerated movement).

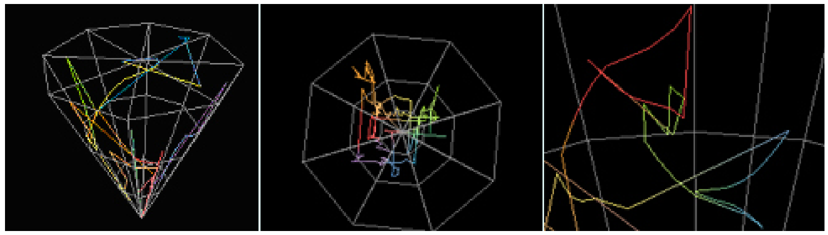

Figure 10: Emotion tracing panel.

7 Testing the GALA Modeler

7.1 Original Transitions

First of all, we have to emphasize that the Plutchik model itself allowed easy transitions in the following cases:

- When the intensity of emotion has changed, it could move into a similar emotional state with increased effect (e.g. from joy to ecstasy)
- An emotion could easily change its immediate neighbor. It was Plutchik's original intention as well to place the similar emotions next to each other. The movement on the surface of the cone realizes this transition.
- When an emotion has a lighter intensity, it could easily switch into another lighter emotion, since their geometrical distance is small (e.g. acceptance to boredom). On the other hand transitions between states that are far away geometrically are less likely (e.g. admiration to boredom).
- A similar category is the switch between very strong emotions. According to the model, a change between strong emotions is less likely than between light emotions. This is also a result of the geometry.
- For the sake of accuracy it is important to note, that in Plutchik's model emotions, moods and cognitive states are mixed together a little (e.g. boredom is more a cognitive state than an emotion etc.). But since we are building an approximate, synthetic model, we can afford this more unified and "standardized" approach. Also, Plutchik's own view strengthens our position as he argues for the unified treatment with good reason. For instance, because of the fact that the difference can be captured by a time component which is handled by the ADSR functions in our extension of the model [12, 23].

7.2 Completely Adequate Behaviors

- Most of the basic observations related to the original Plutchik model were relevant and worked well. Including the change of the length of the emotional vector (intensity) that resulted in realistic emotional status transitions (rage-anger-annoyance etc.).

- Effects similar to each other, resulted in a stronger effect; the contradictory effects (pensiveness - serenity) weakened or neutralized each other.

7.3 Vector Calculation Issues

Although the summing operation of the vectors for obtaining a resultant vector seems to be straightforward and a computationally economical solution, it does not at all provide appropriate behavior self-explanatory. So, we have to ask ourselves, if it is a good idea to "sum up the emotions" into one resultant emotion?

Of course, in everyday life we cannot talk about the "resultant vector of our emotions". However, when the time comes for a reply, humans also act on the basis of a current dominant emotion and they express that with small variations. Therefore the idea of selecting a dominant emotion is not so far from real-life. Either way, we were anyway forced to select an explicit final emotion since we had to use this for choosing the appropriate animation and search for a relevant text reply.

The dominant emotion selection method can be based on a "weighted summing operation", which is present in humans in a special, unconscious form. It adds up all emotions that happened over time and provide a store, or imprint. That emotional memory is query-able, and one can retrieve the dominant emotional state for example by asking - how are you? Thus the synthetic modeling process may approach this calculation problem similarly.

In several cases the sum operation of some emotional effects do not deliver the expected emotional result status (e.g. $\underline{\mathbf{r}}_{pensiveness} + \underline{\mathbf{r}}_{annoyance} =? \underline{\mathbf{r}}_{boredom}$, which is not really what one would expect from a lifelike model). This issue was clearly foreseeable and was caused by the geometrical nature of the emotional space.

7.4 Issues with Setting the ADSR Functions

It is not easy to pre-program the behavior of the ADSR functions. If the sustain duration of the emotions is too long, there will be too many emotional components present in a certain time that may cause inadequate resulting emotion vector, increasing the distortion of the model. On the other hand, if the sustain interval is too short, the effect of the emotion might improperly be insignificant.

The difference between the two cases can be observed on the two Emotion History panels below (Fig. 11.): the second has much more active components in the same time.

There are possibilities to resolve this conflict. One improvement will be the introduction of a relation matrix, where emotions could inhibit or stimulate each other. This way if a new emotion occurs, it could block the effect of long-term, previously arisen emotions. There is an analogue of this effect in real life. It is well known that, the most recently experienced emotion often is a greater determining factor than its original intensity.

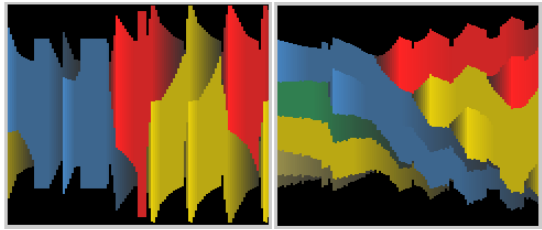

Figure 11: The results of processing the same text with different ADSR settings (different sustain periods).

8 Conclusions and Future Work

We described our attempt to create a chatterbot capable of giving emotionally adequate responses to user utterances. As a starting point we created an architecture for the dialogues system, capable of tracking the current phase of the discussion by analyzing the incoming messages syntactically and semantically, and identifying their role in the dialogue.

Than we created an emotional module to synthesize the incoming user sentences and to generate an appropriate answer to them. We took R. Plutchik theoretical model of emotions and applied it to detect the emotional aspects of natural language communication. We had to make certain amendments to the model itself and ended up developing not "just" a system architecture, but a theoretical model of how communication affects emotional states and how do these states influence communication.

For the proper functioning of our ECA's emotional module, we had to make applications that allow emotional labeling of expressions and sentences and adjusting different attributes (like ADSR data) of them. This also raised interesting theoretical questions, which we tried to test in everyday life and find an answer.

Despite the many questions raised, our emotional model proved to be very practical, useful and a class by itself. It is uniquely synthesizing, mapping and generating emotional responses, decoding the emotional load from sentences and modeling the emotional aspect of the conversation and responding accordingly. We hope these solutions we found are useful for other researchers dealing with dialogue systems, natural language processing, and affective computing.

References

[1] E. André, M. Klesen, P. Gebhard, S. Allen and T. Rist, Integrating models of personality and emotions into lifelike characters, in: A.M. Paiva ed., Affective interactions: Towards a new generation of computer interfaces, Lecture Notes in Artificial Intelligence (Volume 1814) (Springer-Verlag, Berlin, Germany, 2000) 150-165.

[2] J. Bates, The role of emotions in believable agents, Technical report, CMU-CS-94-136, Carnegie Mellon University, 1994.

[3] D. Benyon and K. Höök, Navigation in information spaces: Supporting the individual, in: Proceedings of the IFIP TC13 International Conference on Human-Computer Interaction, 1997) 39-46.

[4] A. Damasio, Descartes' error: Emotion, reason, and the human brain (G. P. Putnam's Sons, New York, USA, 1994.

[5] C. Darwin, The expression of emotions in man and animals (D. Appleton and Company, New York, 1872).

[6] D. Dennett, What is thinking, Think Online (2001).

[7] P. Ekman, An argument for basic emotions, in: N. Stein and K. Oatley eds., Basic emotions, (Lawrence Erlbaum, Hove, UK, 1992) 169-200.

[8] M.A. Gilbert, Language, words and expressive speech acts, in: F. van Eemeren et al. eds., Proceedings of the Fourth International Conference of the International Society for the Study of Argumentation, 1999) 231-234.

[9] M.A. Gilbert, Prolegomenon to a pragmatics of emotion, in: Proceedings of the Ontario Society for the Study of Argumentation, (Brock University, St Catherine's, Ontario, Canada, 1997).

[10] D. Goleman, Emotional intelligence (Bantam Books, 1997).

[11] M. Minsky, The society of mind (Voyageur Pub, 1987).

[12] K. Oatley and J.M. Jenkins, Understanding emotions (Blackwell Publishers, Inc., Oxford, UK, 1996).

[13] Országos Széchényi Könyvtár Tezaurusz: `http://www.oszk.hu/ujdonsag/tezauruj.html`

[14] A. Ortony, On making believable emotional agents believable, in: R.P. Trapple ed., Emotions in humans and artifacts, (MIT Press, Cambridge, USA, 2003).

[15] A. Ortony, G.L. Clore and A. Collins, The cognitive structure of emotions (Cambridge University Press, Cambridge, England, 1988).

[16] R.W. Picard, Affective computing (The MIT Press, Cambridge, Massachusetts, USA, 1997).

[17] P. Rizzo, Why should agents be emotional for entertaining users? A critical analysis, in: A.M. Paiva ed., Affective interactions: Towards a new generation of computer interfaces, Lecture Notes in Artificial Intelligence (Volume 1814) (Springer-Verlag, Berlin, Germany, 2000) 166-181.

[18] H. Schlosberg, Three dimensions of emotions, Psychological Review 61 (1954) 81-88.

[19] S.S. Tomkins, Affect, imagery, consciousness: The negative affects (Springer, New York, 1963).

[20] S.S. Tomkins, Affect, imagery, consciousness: The positive affects (Springer, New York, 1962).

[21] A. Paiva, Affective interactions: Toward a new generation of computer interfaces (Springer-Verlag, 2000).

[22] R. Plutchik, A general psychoevolutionary theory of emotion, in: R. Plutchik and H. Kellerman eds., Emotion theory, research, and experience, (Academic Press, 1980) 3-33.

[23] R. Plutchik, What is mood for? A critique, Psychology 3 (1992).

Received May, 2004