MIT Media Laboratory Software Agents Group Technical Report. November 2002.

Automatic Affective Feedback in an Email Browser

Hugo Liu
Software Agents Group
MIT Media Laboratory
Cambridge, MA 02139
+1 617 253 5334
hugo@media.mit.edu

Henry Lieberman
Software Agents Group
MIT Media Laboratory
Cambridge, MA 02139
+1 617 253 0315
lieber@media.mit.edu

Ted Selker
Context-Aware Computing Group
MIT Media Laboratory
Cambridge, MA 02139
+1 617 253 6968
selker@media.mit.edu

ABSTRACT
This paper demonstrates a new approach to recognizing and presenting the affect of text. The approach starts with a corpus of 400,000 responses to questions about everyday life in Open Mind Common Sense. This so-called commonsense knowledge is the basis of a textual affect sensing engine. The engine dynamically analyzes a user's text and senses broad affective qualities of the story at the sentence level. This paper shows how a commonsense affect model was constructed and incorporated into Chernov face style feedback in an affectively responsive email browser called EmpathyBuddy. This experimental system reacts to sentences as they are typed. It is robust enough that it is being used to send email. The response of the few dozen people that have typed into it is dramatically enthusiastic.

This paper debuts a new style of user interface technique for creating intelligent responses. Instead of relying on specialized handcrafted knowledge bases this approach relies on a generic commonsense repository. Instead of relying on linguistic or statistical analysis alone to "understand" the affect of text, it relies on a small society of approaches based on the commonsense repository.

Keywords
Emotion and Affective UI, Agents and Intelligent Systems, Context-Aware Computing, User and Cognitive models.

INTRODUCTION
One of the impressive triumphs of the computer revolution is that it has given us more effective tools for personal and social expression. Through emails, weblogs, instant messages, and web pages, we are able to share our experiences with friends, family, co-workers, or anyone else in the world who will listen. We use these mediums on a daily basis to share stories about our daily lives. However, as useful as these tools have become, they still lack the highly treasured social interactivity of an in-person conversation. Much as we desire to relate stories of experiences that have saddened, angered, frustrated, and delighted us, the text sits unmoved in cold, square boxes on the computer screen. Nass et al.'s study of human-computer social interaction reveals that people naturally expect their interactions with computers to be social and affective, just as with other people! [20],[21]. Sadly though, people have been so conditioned to expect so little from the user interfaces of today that we are not even bothered by their inability to affectively respond to us like a friend or family member might do.

This shortcoming in current user interfaces hinders progress in the bigger picture too. If software is to transform successfully into intelligent software agents, a social-affective connection between the user and computer must be established because the capacity for affective interaction plays a vital role in making agents believable [2],[27]. Without it, it will be hard to build trust and credibility in the human-computer relationship.

All of this gives rise to the question: Can a user interface react affectively with useful and believable responsiveness to a user engaged in a storytelling task like email or weblogging? We argue that the answer is yes! In this paper, we present a commonsense-based textual analysis technology for sensing the broad affective qualities of everyday stories, told line-by-line. We then demonstrate this technology in an affectively responsive email browser called EmpathyBuddy. EmpathyBuddy gives the user automatic affective feedback by putting on different Chernov-style emotion faces to match the affective context of the story being told through the user's email. We evaluate the impact of the system's interactive affective response on the user, and on the user's perception of the system.

Paper's Organization
This paper is structured as follows: First, we put our approach into perspective discussing existing approaches to textual affect sensing and other related work. Second we motivate our commonsense treatment of emotions with research from the cognitive psychology and artificial intelligence literature. Third, we discuss methods for constructing and applying an commonsense affect model. Fourth, we discuss how a textual affect sensing engine was incorporated into Chernov face style feedback in an affectively responsive email browser called EmpathyBuddy, and we examine a user scenario for our system. Sixth, we present the results of a user evaluation of EmpathyBuddy. The paper concludes with a summary of contributions, and plans for further research.

THE APPROACH IN PERSPECTIVE

Affective behavior is not only an important part of human-human social communication [20], but researchers like Picard have also recognized its potential and importance to human-computer social interaction. [25], [21]. In order for computers to make use of user affect, the user's affective state must invariably first be recognized or sensed. Researchers have tried detecting the user's affective state in many ways, such as, inter alia, through facial expressions [20],[1], speech [11], physiological phenomena [29], and text [3],[10]. This paper addresses textual affect sensing. In the following subsections, we review existing approaches to textual sensing and compare these to our approach.

Existing Approaches

Existing approaches to textual affect sensing generally fall into one of two categories: keyword spotting, and statistical modeling.

Keyword spotting for automatic textual affect sensing is a very popular approach and many of these keyword models of affect have gotten quite elaborate. Elliott's Affective Reasoner [10], for example, watches for 198 affect keywords (e.g. distressed, enraged), plus affect intensity modifiers (e.g. extremely, somewhat, mildly), plus a handful of cue phrases (e.g. "did that", "wanted to"). Ortony's Affective Lexicon [23] provides an often-used source of affect words grouped into affective categories. Even with all its popularity, keyword spotting is not very robust in practice because it is sensing aspects of the prose rather than of the semantic content of a text. Affect vocabulary may vary greatly from text to text, or may be absent altogether. For example, the text: "My husband just filed for divorce and he wants to take custody of my children away from me," certainly evokes strong emotions, but lack affect keywords. A lot of affective communication in a text is done without explicit emotion words, and in these cases, keyword spotting would inevitably fail.

Statistical methods can sometimes do better. By feeding a machine learning algorithm a large training corpus of affectively annotated texts, it is possible for the system to not only learn the affective valence of affect keywords as in the previous approach, but such a system can also take into account the valence of other arbitrary keywords, punctuation, and word co-occurrence frequencies. Statistical methods such as latent semantic analysis (LSA) [8] have been popular for affect classification of texts, and have been used by researchers on projects such as Goertzel's Webmind [13]. However, statistical methods are generally semantically weak, meaning that, with the exception of obvious affect keywords, other lexical or co-occurrence elements in a statistical model have little predictive value individually. As a result, statistical text classifiers only work with acceptable accuracy given a sufficiently large text input. So while these methods may be able to affectively classify the user's text on the page or paragraph-level, they will not work on smaller text units such as sentences.

While page or paragraph-level sensing has its applications, there will be many applications for which this is not granular enough. Synthetic agents may demand the higher level of interactivity that can only be met by sensing the affect of individual sentences. [17],[2],[10]. Affective speech synthesis will benefit from affect annotations of text at the sentence-level. [5]. Some context-aware systems will want to be able to react to the affective state of the user as captured in a single sentence or command. [16].

A Commonsense Knowledge Based Approach

Our proposed approach uses a large-scale (on the order of ½ million facts) knowledge base filled with commonsense about the everyday world, including affective commonsense, to construct the user's "commonsense affect model." This model is premised on the observation that people within the same population tend to have somewhat similar affective attitudes toward everyday situations like getting into a car accident, have a baby, falling in love, having a lot of work, etc. One likely explanation for this is that these attitudes are part of our commonsense knowledge and intuition about the world, which is shared across people within a cultural population.

A textual affect sensing engine uses the constructed commonsense affect model to try to sense broad affective qualities of the text at the sentence level. We believe that this approach addresses many of the limitations of the existing approaches.

Whereas keyword spotting senses only affective keywords in the prose, commonsense knowledge lets us reason about the affective implications of the *underlying semantic content.* For example, while the affect-keyword-based approach might work for the sentence "I was badly injured in a scary car accident," only the commonsense knowledge-based approach would work when the affect words are removed: "I was injured in a car accident."

Whereas semantically weaker statistical methods require larger inputs, semantically stronger commonsense knowledge can sense emotions on the sentence-level, and thereby enable many interesting applications in synthetic agents, affective speech synthesis, and context-aware systems mentioned above.

Having put our approach into proper perspective, in the next section, we motivate our commonsense treatment of emotions with literature in cognitive psychology.

COMMONALITY OF AFFECTIVE ATTITUDES TOWARD THE EVERYDAY WORLD

The idea put forth in this paper is that there is some user-independent commonality in people's affective knowledge of and attitudes toward everyday situations and the everyday world which is somehow connected to people's commonsense about the world. It is the presence of this shared knowledge and attitude, which enables a person to recognize and feel empathy for another person's situation. Without shared affective knowledge and attitudes within cultural populations, social communication would be very difficult between people. Though we know of no direct research on the commonality of affective knowledge and attitudes and its linkages to commonsense, there is much indirect support from the psychology literature.

As far back as Aristotle's Rhetoric [6], and as recently as Damasio [7], Ortony [23], and Minsky [18] emotions have been identified as being an integral part of human cognition, and researchers acknowledge that affective expression is greatly influenced by cognition. On the other hand, people's commonsense knowledge about the world provides an important context for human cognition with which people interpret the world [18]. Furthermore, psychologist William James noted that the recognition of emotion in language depends on traditions and cultures, so people may not understand necessarily understand the emotions of other cultures [14]. Though no explicit experiments have been performed, it seems a reasonable conclusion to draw that James's thesis really alludes to how the perception and expression of emotion finds its roots into the affective commonsense knowledge and attitudes indigenous to a culture. In fact, Minsky's Emotion Machine [18] seems to imply just this – that much of people's affective attitudes and knowledge is an integral part of their commonsense knowledge.

Powerful is the result that much of people's affective attitudes and responses to everyday situations lies in commonsense knowledge. It allows for the possibility that generic commonsense knowledge about the everyday world might be used to help us create a commonsense model of human affective response to everyday situations. Of course, we do not presume that such a user-independent (within a culture) model will always be right, (because situational context also plays a role), or that it will allow for very fine grained discernment of a user's affective state; our hope is that this kind of model will allow us to bootstrap the affective intelligence of any user interface or agent in which it appears. In a later section, the evaluation of our prototype affective sensing system confirms this.

In the next section, we go in-depth into the methods with which we construct a commonsense affect model and apply it to build a textual affect sensing engine.

METHODS FOR CONSTRUCTING AND APPLYING A COMMONSENSE AFFECT MODEL

The goal of our enterprise is to sense broad affective qualities in story text based on large-scale affective commonsense knowledge of the everyday world.

Broadly, our approach can be decomposed into the following phases: 1) mine affective commonsense out of a generic commonsense knowledge base called Open Mind; 2) build a commonsense affect model by calculating mappings of everyday situations, things, people, and places into some combination of six "basic" emotions; 3) and use this constructed model to analyze and affectively annotate story text.

The following subsections gives a more detailed treatment of each of the three phases.

Mining Affective Commonsense out of a Generic Commonsense Knowledge Base

Our approach relies on having broad knowledge about people's common affective attitudes toward situations, things, people, and actions. If we want our affective sensing engine to be robust, we will have to supply it with a great breadth of knowledge that reflects the immensity and diversity of everyday knowledge.

From three large-scale generic knowledge bases of commonsense: Cyc [15] (2 million assertions), Open Mind Common Sense (OMCS) [28] (1/2 million sentences), and ThoughtTreasure [19] (100,000 assertions), we chose OMCS because its English-sentence representation of knowledge is rather easy to manipulate and analyze using language parsers. In the future, we expect to also incorporate knowledge from the other two commonsense knowledge sources as well.

In OMCS, commonsense is represented by English sentences that fit into 20 or so sentence patterns expressing a variety of different relations between concepts. An example of a sentence from OMCS is: (Sentence pattern words are italized). "*A consequence of* getting into a fight *is* someone will get hurt." OMCS also contains affective commonsense like "*Some people find* ghosts *to be* scary."

From OMCS, we first extract a subset of the sentences which contain affective commonsense. This represents approximately 10% of the whole OMCS corpus. The identification of these sentences is heuristic, accomplished mainly through keyword spotting. These affect keywords serve as "emotion grounds" in sentences, because their affective valences are already known.

Building a Commonsense Affect Model

After identifying a subset of the commonsense knowledge that pertains to emotions, we build a commonsense affect model with which we can analyze the affective qualities of a user's text. In truth such a model is a society of different models that compete with and complement one another. All of the models have homogeneously structured entries, each of which have a value of the form:

[a happy, b sad, c anger, d fear, e disgust, f surprise]

In each tuple, a-f are scalars greater than 0.0, representing the magnitude of the valence of the entry with respect to a particular emotion.

Why six "basic" emotions? In our implementation we have chosen to work with the six so-called "basic" emotions enumerated above, which were proposed by Ekman based on his research into universal facial expressions [9]. This choice seemed appropriate considering that our prototype application would be displaying Chernov-style faces. It should be noted that our approach can be grounded in any set of "basic emotions" which can be discerned through affect keywords, which include, most prominently, sets proposed by Ekman [9], Frijda [12], James [14], and Plutchik [26]. For a complete review of proposals for "basic emotions", see [22].

A Society of Models. Having established the similarities of the models, we go on to briefly explain each of the models used in our current implementation.

Subject-Verb-Object-Object Model. This model represents a declarative sentence as a subject-verb-object-object frame. For example, the sentence "Getting into a car accident can be scary," would be represented by the frame: [<Subject>: ep_person_class*, <Verb>: get_into, <Object1>: car accident, <Object2>:] whose value is: [0,0,0,1,0,0] (fear).

The strength of this model is accuracy. SVOO is the most specific of our models, and best preserves the accuracy of the affective knowledge. Proper handling of negations prevents opposite examples from triggering an entry. The limitation of SVOO however, is that because it is rather specific, it will not always be applicable.

Concept-level Unigram Model. For this model, concepts such as verbs, noun phrases, and adjective phrases are extracted from each sentence. Affectively neutral concepts/words (e.g. "get," "have") are excluded using a stop list. For example, in the sentence: "Car accidents can be scary," the following concept is extracted: [<Concept>: "car accident"] and is given the value: [0,0,0,1,0,0] (fear).

Concept-level unigrams are not as accurate as SVOOs, but experiences better coverage.

Concept-level Valence Model. This model defers from the above-mentioned concept-level unigram model in the value. Rather than the usual six-element tuple, the value indicates that a word has positive or negative connotations. Associated with this model is hand-coded meta-knowledge about how to reason about affect using valence. This model is useful in disambiguating a sentence's affect when it falls on the cusp between a positive emotion and negation emotion.

Modifier Unigram model. This model assigns six-emotion tuple values to the verb and adverbial modifiers found in a sentence. The motivation behind this is that sometimes modifiers are wholly responsible for the emotion of a verb or noun phrase, like in the sentences,

"Moldy bread is disgusting", "Fresh bread is delicious"

In constructing each of the aforementioned models, we first choose a bag of affect keywords, pre-classified into the six basic emotions, to act as "emotion grounds." To build up the models, we twice propagate the affective valence from the grounds to the connected concepts in OMCS and from those concepts to yet other concepts. After each propagation, the affect value is discounted by a factor *d*. With the completed commonsense affect model, we can evaluate story texts and on the sentence-level, we can sense its commonsense affective quality. This is discussed in the next subsection.

Applying Commonsense Affect Models to Analyze Story Text

In analyzing story text, we center our discussion around three key issues: choosing basic story units for affect annotation; model-driven scoring and disambiguation; and inter-sentence smoothing.

Choosing basic story units. Our analysis of story text is performed on the sentence-level. It might be more accurate to say independent-clause level rather than sentence-level because our analysis first splits sentences, which contain multiple independent clauses, at their clausal break. Independent clauses are clauses joined by coordinating conjunctions like "but," and "however." Independent clauses are the smallest meaningful units which can represent an event.

Model-driven scoring and disambiguation. Our goal here is to use our models to evaluate each sentence, and annotate it with one of the six emotions, or "neutral." For each sentence, we apply each of our models, which return a six-tuple emotion score. The scores are weighted by model predictiveness to form a total. If the magnitude of the total score does not exceed a certain confidence threshold, then our models do not know enough to analyze that sentence, and we annotate that sentence as being neutral. To disambiguate between two likely emotions, we prefer the emotion of the previous sentence, and also prefer a positive or negative emotion based on the concept-level valence model.

Inter-sentence smoothing and meta-emotions. After sentences have been annotated with one of the six basic emotions or "neutral," we apply various techniques aimed at smoothing the transition of emotions from one sentence to the next. There are four smoothing techniques: decay, interpolation, global mood, and meta-emotions. For brevity we only give an example of each. Capitalized words represent annotated sentences.

- (Decay): SAD NEUTRAL NEUTRAL → SAD SAD50% NEUTRAL
- (Interpolation): SAD NEUTRAL SAD → SAD SAD50% SAD
- (Global mood, e.g. sad): ANGER → ANGER+SAD20%
- (Meta-emotion): FEAR HAPPY → FEAR RELIEF

Having discussed in detail our commonsense approach to affect sensing in text, we go on to present an experimental application that integrates the affect sensing engine into an affectively responsive user interface.

AUTOMATIC AFFECTIVE FEEDBACK IN AN EMAIL BROWSER

The first version of the system described the affective quality of the text a person wrote by displaying text in another window. The system was interesting but felt a bit unmotivated. The term flame has become a verb. Email has a reputation of accentuating affective responses. By thinking about what application would be improved by evaluating emotion sentence by sentence, we came up with composing email as a first experiment for our textual emotion sensing engine.

The issue of paying attention to the text you are writing makes it difficult to pay attention to other text. The Chernov face was chosen as a glanceable interface that might be less intusive to to the typist.

This section is subdivided into three sections: 1) an overview of the EmpathyBuddy user interface; 2) a discussion of fail-soft interface features; and 3) a walkthrough of a user scenario.

User Interface Overview

EmpathyBuddy is an email browser with a Chernov face embedded in its window that emotes in sync with the affective quality of text being typed by the user (shown in Figure 1). The cartoon facial expressions were taken from the University of Central Florida's E-FERET emotion image database [4]. EmpathyBuddy's faces express the six basic Ekman emotions plus decayed versions of the six basic emotions, and also four transitory meta-emotions. As of the writing of this paper, there is no animation.

The layout of the email browser is meant to be familiar to the user. At the upper left corner are email header fields. In the lower left is the email body text box. An affect demon frequently polls the text in the body and analyzes the text using the emotion sensing engine. The avatar in the upper left corner changes faces to try to match the dynamic affective context of the story. Because of the limitations of pre-drawn facial expressions, the cartoon character cannot fully express the affect annotations outputted by the sensing engine. In particular, the pre-drawn facial expressions cannot account for global mood's secondary influence on a primary emotion.

Figure 1. EmpathyBuddy Email Agent

Fail-soft interface features

In preliminary usability experience with three users, we discovered a few interface aspects that frustrated a couple of the users. In response we have incorporated a few features into the interface to make it more fail-soft.

One user stated that although he was delighted by most of the faces and how they seemed to match the affective content of his story, he thought that anger was used in the wrong context and that when used inappropriately it made the email browser seem hostile. He stated his belief that "the agent getting angry at me for no good reason makes me not want to use it." We believe that the user would have felt the same about the emotion disgust, though none of the users encountered that emotion during the preliminary usability test. In response, we have increased the confidence threshold for anger and disgust so that they will only be displayed in extreme cases. Also, if anger or disgust participates in an ambiguity with another emotion, it will always lose. This feature helps make the agent more fail-soft by instructing it to observe that there is a higher cost to the user when mistakes are made with certain strong emotions.

Two of the three users pointed out that sometimes the agent got some emotion wrong, and they wished there was some way to change the emotion of the agent without having to wait until another full sentence was typed to change the emotion. In response, we have added handling of text emoticons. Whenever a text emoticon is typed, the emotion of the agent will immediately update. Therefore, if a user feels that the agent has misinterpreted a sentence, he/she can type any of over 100 emoticons and immediately correct the facial expression of the emoticon. The emoticon handling feature adds fail-softness to the user by letting the user quickly correct the mood of the agent.

Walkthrough of a User Scenario

We walk through a scenario in which the user writes an email to her mom telling her about he buys a new car but uneventfully wrecks it. Thankfully though, she was not hurt. Figure 2 gives a time-lapse walkthrough of this scenario. This is a useful scenario because it highlights some of the more advanced features of the affect sensing engine.

In the sentence, "it's a gorgeous new sports car!" the engine's models are not certain about the affect of sports cars. They show that this sentence is ambiguous and that it could be one of two emotions: surprise or anger. Three disambiguation features all concluded that the correct emotion was surprise. First, according to the conceptual valence model, this sentence was characterized by positive emotion, and since surprise is positive whereas anger is not, this feature chose surprise. Second, the previous sentence disambiguation feature prefers surprise because that emotion also occurred in the previous sentence. Third, according to the fail-soft strategy of only showing anger and disgust in extreme cases, anger would have also been disallowed from occurring here.

Figure 2: User Scenario Walkthrough

The last two sentences are a good illustration of meta-emotion smoothing. The sentence, "I got into an accident and I crashed it" evoked fear, while the sentence "Thankfully, I wasn't hurt" evoked happy. However, it would seem rather unnatural to change suddenly from a frightened expression to a happy expression. Humans don't easy forget the anxiety they held two seconds ago! The affect sensing engine recognizes the pattern of moving from fear to happy as a meta-emotion. It decides that happy should actually be revised into relief (from anxiety). It does so accordingly, and the emotion displayed by the EmpathyBuddy avatar reflects this.

USER TESTING AND SYSTEM EVALUATION

A 20-person user study was conducted to quantitatively measure the "performance" of the EmpathyBuddy email browser. In this section, we first present the experiment design, followed by an analysis of the performance measurement results, and concluded by proposals for further evaluation.

Experiment Design

Twenty people who all describe themselves as regular email users participated in this user study. They were asked to use three email clients to perform the same task. The task they were asked to perform was: "send an email to someone and tell them a brief but interesting story about something you did recently." They were instructed to write about the same basic story using all three interfaces but were told it was okay to tell it differently. To minimize the corrupting effect of natural language parsing errors, participants were further instructed to try to write clearly and without run-on sentences.

The three email clients they were asked to perform this task in are all variations on the EmpathyBuddy email browsers Consequently, they all have the same look and feel with the only difference being the expression of the avatar. The first client has a permanent neutral face. This client is meant to be the control for the overall UI and meant to provide a baseline. The second client displays alternating, randomized faces. To be more specific, at the end of every sentence break, this client will display a random face. This client is meant to control for the deliberate selection of faces by the affect sensing engine. The third client is the real EmpathyBuddy browser.

Users are presented the three clients in random order, and are not told anything about them. Each user was instructed to observe the Think Aloud Protocol and their comments were recorded. Each user participated in the study under supervision. After using each of the clients, users are asked to answer 4 questions in a questionnaire. For each question, users are asked to evaluate how much they agreed with a statement, on a scale from 1 (strongly disagree) to 7 (strongly agree). The four aspects that users were asked to evaluate were: entertainment, interactivity, intelligence, and adoption.

Performance Measurement Results

The quantitative results of the questionnaire are shown in Figure 3. The length of each bar indicates the mean score, which is also written at the right end of each bar. Each line segment bisecting the end of the bars represents one standard deviation above and below the mean.

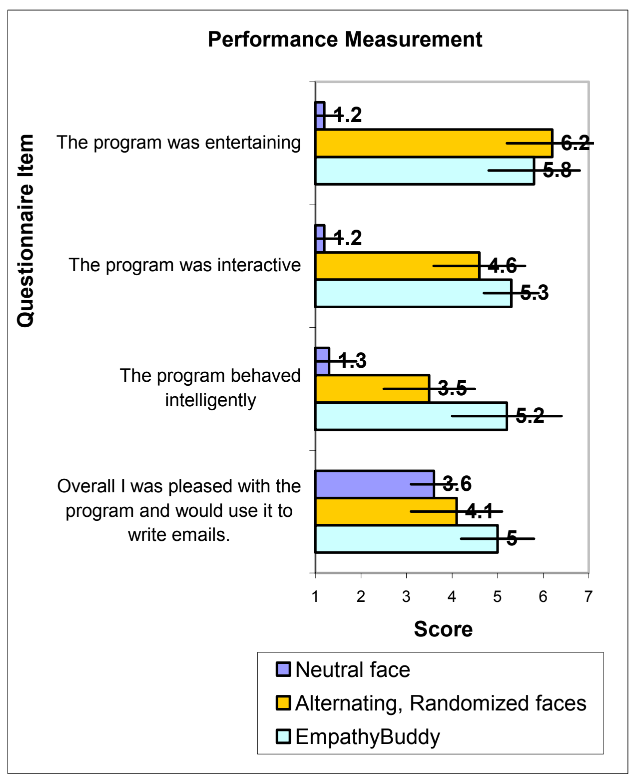

Figure 3. User Testing Questionnaire Results

The results of the performance measurement evaluation were generally positive. On all four aspects, EmpathyBuddy well outperformed the neutral face control. On three of the four aspects, EmpathyBuddy narrowly outperformed the random face control, and virtually tied on the fourth. In general, the results were more or less expected, though we did not anticipate such a close gap between random faces and EmpathyBuddy. Below, we give a question-by-question analysis.

- *"The program was entertaining":* We did not expect random faces to be more entertaining than EmpathyBuddy. In a post-study interview, one participant pointed out that "outrageousness can be very entertaining," referring to how randomizing faces displayed even the most outrageous faces like disgust and anger with equal likelihood.

- *"The program was interactive":* This provides interesting insight into interactivity. Even though randomized faces was more entertaining and changed more frequently than EmpathyBuddy, EmpathyBuddy was still more interactive, and had a smaller variance than with random faces. In post-study interviews, some users remarked that once they suspected randomized faces was displaying odd, out of context faces, they no longer felt like they were interacting with the interface. But how did randomized faces get such a high score? We suspect it is because it appeared to have some of the properties of interactivity including reciprocal action, feedback, and immediacy. Given a longer interaction with the system however, we predict this mirage would break down. EmpathyBuddy prevailed because its changes in faces better met the interactivity criteria of relevancy and synchronicity. We suspect that EmpathyBuddy did not score higher because the face did not change as frequently as it could have. There are also limitations to the static face that can't display all the nuances in the emotion annotations, such as global mood and decay.

- *"The program behaved intelligently":* This was the largest margin by which EmpathyBuddy outscored randomized faces. This particular result is very positive because it lends some partial support to the precision of the affect sensing engine. Though somewhat troubling is the very large variance associated with this score. This is accounted for by the fact that EmpathyBuddy made a few dramatic errors displaying contradictory faces during testing with a few users. A separate evaluation to measure the precision and recall of the backend emotion sensing engine would help further elucidate the results for this question.

- *"Overall, I was pleased with the program and would use it to write emails.":* The results here were also very surprising. We did not anticipate the neutral face control to score so high, but as one user remarked, "the [neutral face] looks like it's sleeping! How cute!" We also did not expect randomized faces to score greater than neutral! Why would people want to switch to a mail client that shows random faces? Users stated that it was more entertaining than their mail clients. We suggest that users are so bored of their static email interfaces and are ready to flock to something – anything – more interactive, entertaining, and emotional. For users to have scored EmpathyBuddy a point above neutral with respect to system adoption is very gratifying. In fact, EmpathyBuddy consistently scored about neutral in all four aspects.

Further Evaluation

We hope to perform two additional evaluations on the work presented in this paper in the near future. One evaluation should compare the performance of our affect sensing engine against keyword spotting statistical language methods. A second evaluation should measure how a user's story authoring behavior might change as a result of affective feedback from an avatar.

CONCLUSION AND FUTURE PROSPECTS

This paper debuts a new user interface technique for the automatic sensing of affect in text using affective commonsense knowledge about the everyday world mined from a repository of ½ million commonsense facts. This approach addresses the limitations of existing approaches like keyword spotting and statistical language modeling. Because affective commonsense is used to reason about the underlying meaning of the text, our method can sense affect in text without the aid of obvious affect keywords, which cannot be relied on to be present in user input. And unlike statistical methods, which require large inputs, our method can successfully sense affect on the sentence-level. This is a granularity level that is particularly important to many user interfaces such as highly interactive applications and affective text-to-speech.

A textual affect sensing engine was built using this approach, and was used in an email browser to give automatic affective feedback to the user through avatar expressions. The results of user testing showed that our textual affect sensing engine significantly improved the user's perception of the system's interactivity and intelligence, and that users were willing to adopt an email interface giving this kind of affective feedback.

Having addressed many of the limitations of existing approaches to textual affect sensing, we believe that our approach can open up many possibilities for how such a technology might be applied to improve user interfaces. Almost any context-aware agent can benefit from understanding the user's affective state, and can use this context to interact with the user with more sensitivity. We can also imagine that textual affect sensing, working in concert with an affective speech synthesizer, might make for a sophisticated, affectively intelligent speech interface. Or in network gaming, avatars can be given affective expressions. And so on.

In future work, we hope to overcome the present limitation of our approach to stories about everyday events, by mining the Web for affective commonsense about a variety of domains, and adding analogical reasoning capabilities. We are also currently working on integrating the sensing engine with affective speech synthesis to produce an affective text-to-speech user interface.

ACKNOWLEDGMENTS

The authors would like to acknowledge the work of our colleague Push Singh at the MIT Media Laboratory for his work in creating the Open Mind Commonsense repository.

REFERENCES

1. Bartlett, M., Hager, J.C., Ekman, P., and Sejnowski, T.J. Measuring facial expressions by computer image analysis. *Psychophysiology, vol. 36*, pp. 253--263, 1999.
2. Bates, J. The Role of Emotion in Believable Agents. *Communications of the ACM, Special Issue on Agents*, July 1994.
3. Batliner, Huber, Niemann, Nöth, Spilker, and Fischer The Recognition of Emotion. In Wahlster, W. (ed.): *Verbmobil: Foundations of Speech-to-Speech Translations*, pages 122-130. Springer, New York, Berlin. 2000.
4. E-FERET Emotion Image Database Search Engine. Available at: https://forloop.cs.ucf.edu/db/
5. Cahn, J.E. The Generation of Affect in Synthesized Speech. *Journal of the American Voice I/O Society, Volume 8*. July, 1990. Pages 1-19.
6. Cope, E.M. *An Introduction to Aristotle's Rhetoric, with Analysis, Notes, and Appendices*. London: MacMillan, 1867.
7. Damasio, A.R. *Descartes' Error: Emotion, Reason, and the Human Brain.* Gosset/Putnam Press, New York, NY 1994.
8. Dumais, S.T., Furnas, G.W., Landauer, T.K., Deerwester, S., and Harshman, R. Using latent semantic analysis to improve access to textual information. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI'88)*, 1988.
9. Ekman, P. Facial expression of emotion. *American Psychologist, 48*, 384-392. 1993.
10. Elliott, C. *The Affective Reasoner: A Process Model of Emotions in a Multi-agent System.* PhD thesis, Northwestern University, May 1992. The Institute for the Learning Sciences, Technical Report No. 32.
11. Fernandez, R. and Picard, R.W. Modeling Driver's Speech under Stress, *Proc. ISCA Workshop on Speech and Emotions*, Belfast, 2000.
12. Frijda, N.H. *The Emotions.* Cambridge University Press, Editions de la Maison des Sciences de l'Homme, Paris, 1986.
13. Goertzel, B., Silverman, K., Hartley, C., Bugaj, S., and Ross, M. The Baby Webmind Project. *Proceedings of AISB 2000, the annual conference of The Society for the Study of Artificial Intelligence and the Simulation of Behaviour*. 2000.
14. James, W. What is an emotion? In C. Calhoun & R. C. Solomon (Eds.), *What is an emotion? Classic readings in philosophical psychology* (pp. 127-141). New York: Oxford University Press. 1984.
15. Lenat, D.B. CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33-38, 1995.
16. Lieberman, H. and Selker, T. Out of Context: Computer Systems that Learn About, and Adapt to, Context, *IBM Systems Journal, Vol 39, Nos 3&4,* pp. 617-631, 2000.
17. Maes, P. Artifical Life meets Entertainment: Lifelike Autonomous Agents. *Communications of the ACM. Special Issue on Novel Applications of AI.* 1995.
18. Minsky, M. *The Emotion Machine*. Pantheon, not yet available in hardcopy. Drafts available: http://web.media.mit.edu/~minsky/.
19. Mueller, E.T. ThoughtTreasure: A natural language/commonsense platform (Online). http://www.signiform.com/tt/htm/overview.htm. 1998.
20. Nass, C.I., Stener, J.S., and Tanber, E. Computers are social actors. In *Proceedings of CHI '94,* (Boston, MA), pp. 72-78, April 1994.
21. Nass, C., Sundar, S.S. Is Human-Computer Interaction Social or Parasocial? Paper submitted to Human Communication Research. 1994.
22. Ortony, A., and Turner, T. J. What's basic about basic emotions? *Psychological Review, 97,* 315-331. 1990.
23. Ortony, A., Clore, G. L., and Foss, M. A. The referential structure of the affective lexicon. *Cognitive Science, 11*, 341-364. 1987.
24. Ortony A., Clore, G. L., and Collins, A. *The cognitive structure of emotions.* New York: Cambridge University Press. 1988
25. Picard, R. W., *Affective Computing*, The MIT Press, Mass., 1997.
26. Plutchik, R. A general psychoevolutionary theory of emotion. In R. Plutchik & H. Kellerman (Eds.), *Emotion: Theory, research, and experience: Vol. 1. Theories of emotion* (pp. 3-33). New York: Academic. 1980.
27. Reilly, S. *Believable Social and Emotional Agents.* Technical Report, CMU-CS-96-138, School of Computer Science, Carnegie Mellon University. 1996
28. Singh, P., The public acquisition of commonsense knowledge. *Proceedings of AAAI Spring Symposium*. Palo Alto, CA, AAAI. 2002.
29. Vyzas, E. and Picard, R.W. Online and Offline Recognition of Emotion Expression from Physiological Data. *Workshop on Emotion-Based Agent Architectures at the Int. Conf. on Autonomous Agents*, Seattle, WA, 1999.