A Sequential Bayesian Dialog Agent for Computational Ethnography

Abe Kazemzadeh, James Gibson, Juanchen Li, Sungbok Lee,
Panayiotis Georgiou, Shrikanth Narayanan

Signal Analysis and Interpretation Lab, University of Southern California
Los Angeles, CA, USA
{kazemzad@,jjgibson@,juanchel@,sungbokl@,georgiou@sipi.,shri@sipi.}usc.edu

Abstract

We present a sequential Bayesian belief update algorithm for an emotional dialog agent's inference and behavior. This agent's purpose is to collect usage patterns of natural language description of emotions among a community of speakers, a task which can be seen as a type of computational ethnography. We describe our target application, an emotionally-intelligent agent that can ask questions and learn about emotions through playing the emotion twenty questions (EMO20Q) game. We formalize the agent's algorithms mathematically and algorithmically and test our model experimentally in an experiment of 45 human-computer dialogs with a range of emotional words as the independent variable. We found that 44% of these human-computer dialog games are completed successfully, in comparison with earlier work in which human-human dialogs resulted in 85% successful completion on average. Despite being lower than this upper-bound of human performance, especially on difficult emotion words, the subjects rated that the agent's humanity was 6.1 on a 0 to 10 scale. This indicates that the algorithm we present produces realistic behavior, but that issues of data sparsity may remain.

Index Terms: dialog agents, emotion recognition, chatbot, EMO20Q,

1. Introduction

The evolution of language in our primitive ancestors endowed early humans with the ability to communicate about things beyond their immediate perception. In addition to feeling and expressing emotions in response to current situations [1], we are also able to talk about emotions that may have occurred in the past or that may be altogether hypothetical. Non-human primates and many other animals can display their emotions through social signals, but only humans can communicate about their emotions symbolically.

Some aspects of human emotions are social signals, e.g. facial features, that show stability across cultures and even across species. However, some key aspects of emotional behavior in humans is symbolic: the words that designate emotions in natural languages are conventional symbols. The relation of emotions as social and biological signals with the symbolic processes of natural language is what we call *natural language description of emotions*.

Whereas humans have both emotions and the ability to manipulate symbols using language, computers are only able to manipulate symbols and lack biological emotion pathways. To what extent these biological processes of emotion affect natural language description of emotion, and to what extent this language can be learned and simulated by an "emotionless" computer, is currently an open question and the topic of this paper.

In the field of affective computing, it is common to rely on a limited set of basic emotions to be used for labeling emotional data. However, when considering natural language descriptions of emotions there are a practically limitless number of ways to name and describe emotions. As long as a community of speakers decides, by customs or convention, to designate an emotion using an arbitrary sequence of phonemes (c.f. *duality of patterning* [2]) , we must, at some point, deal with emotion labels that cannot be simply enumerated theoretically. Instead, we propose an agent that will carry out the laborious task of eliciting natural language descriptions of emotion in the populations that we wish to study. We feel that this elicitation process can be thought of as a type of computational ethnography [3] for a linguistic community of practice [4].

In previous work [5], we collected natural language descriptions of emotions from a target population using a game called emotion twenty questions (EMO20Q), which is the familiar game of twenty questions played with emotions as the unknown objects. Based on the data we collected of humans playing EMO20Q, we created a statistical model for an agent who plays EMO20Q using a sequential Bayesian update algorithm. This algorithm can be seen as a simplification of the partially observable Markov decision process for dialog systems [6, 7]. The simplification is made possible due to the constrained interaction of the twenty questions task. For elicitation of human knowledge, the goal for our agent is to play realistically and to approach human performance in successfully completing the EMO20Q task. The data we used for training and testing is described in Section 2. The model, algorithm, and implementation for our agent is described in 3 and the results are presented in Section 4 and discussed in Section 5.

2. Data

We trained our agent on data from both human-human and human-computer EMO20Q matches. A match in the context of EMO20Q is a game-instance where one player assumes the role of the answerer and the other player, the role of the questioner. Typically, in the human-human matches, the players will alternately switch roles. In the case of human-computer matches, the computer always plays the questioner role.

Using the methodology described in [5], we collected 110 human-human dialogs, and using an earlier version of our system [8], we collected 131 human-computer dialogs. In the 110 matches that we collected between two humans, we found that the matches where the target emotion word was guessed 85% of the time after approximately 11.4 turns. In the early versions of the dialog agent, not those described in the current paper, the performance was much lower than the human-human case. However, due to using a computer agent, we were able to col-

lect more data with less effort because the agent was available to play online.

Our current system is an offline desktop application that has an asynchronous chat-based design and a new algorithm, described in the next section. To test our algorithm we collected a new set of data using the implementation of the new algorithm. The 15 subjects were told that they would play EMO20Q with a computer and they were asked to pick 3 emotion words, one easy word, one medium word, and one difficult one, based on how difficult they thought it would be to guess. They were also asked to rate the naturalness of the agent on a 0-10 scale, and were given an opportunity for open-ended comments.

3. Methodology

3.1. Framework for Sequential Bayesian Belief Update

The model we use for the agent is a sequential Bayesian belief update algorithm that starts with a conditional probability distribution estimated from a corpus of human-human and human-computer EMO20Q matches, as described in Section 2. In this data, there was a set of 105 emotion words that were observed. Let E be this set of 105 emotion words and let $\varepsilon \in E$ be a categorical, Bayesian (i.e., unobserved) random variable distributed over this set. Each question-answer pair from the match of EMO20Q is considered as an observed feature of the emotion being predicted. Thus, if Q is the set of questions and A is the set of answers, then a question $q \in Q$ and an answer $a \in A$ together compose the feature $f = (q, a)$, where $f \in Q \times A$. The conditional probability distribution, $P(f|\varepsilon)$, is estimated from the training data using a smoothing factor of 0.5 to deal with sparsity.

In our model we stipulate that the set of answers A are four discrete cases: "yes", "no", "other", and "none". When the answer either contains forms of 'yes' or 'no', it is labeled accordingly. Otherwise it is labeled 'other'. The forms of 'yes' are 'yes', 'yeah', 'yea', 'yep', and 'aye'[1], and the forms of 'no' are 'no' and 'nope'. The feature value 'none' is assigned to all the questions that were not asked in a given dialog. 'None' can be seen as a missing feature when the absence of a feature may be important. For example, the fact that a certain question was not asked about a particular emotion may be due to the fact that that question was not relevant at a given point in a dialog.

Similarly, we stipulate that the questions can be classified into some discrete class that is specified through a semantic expression as described in [5]. For example, the question "is it a positive emotion?" is represented as the semantic expression "e.valence==positive". If the answer to this question was "maybe", the resulting feature would be represented as `('e.valence==positive','other')`.

Using Bayes' rule and the independence assumption of the naïve Bayes model, we can formulate the agent's belief about the emotion vector ε after observing features $f_1 ... f_t$ as

$$P(\varepsilon|f_1, ..., f_t) = \frac{\prod_{i=1}^{t} [P(f_i|\varepsilon)]\, P(\varepsilon)}{\prod_{i=1}^{t} P(f_i)}. \quad (1)$$

When the game begins the agent can start with a uniform prior on its belief of which emotion is likely or it can use information obtained in previously played games. In the experiment of this paper, we use a uniform prior, $P(\varepsilon = e_k) = 1/|E|,\ \forall k = 1...|E|$. We chose to use the uniform prior to start with because our training data contains many single count training instances and because we want to examine how the system performs with less constraints.

In (1), the posterior belief of the agent of emotion e_k at time t, $P(\varepsilon = e_k|f_1, ..., f_t)$ is computed only after the agent has asked the t questions. In contrast the formulation we use is dynamic in that the agent updates its belief at each time point based on the posterior probability of the previous step, i.e., at time t:

$$P(\varepsilon|f_1, ..., f_t) = \frac{P(f_t|\varepsilon)P(\varepsilon|f_1, ..., f_{t-1})}{P(f_t)} \quad (2)$$

We introduce a new variable $\beta_{t,k} = P(\varepsilon = e_k|f_1, ..., f_t)$ for the agent's belief about emotion k at time t and postulate that the agent's current prior belief is the posterior belief of the previous step. Then, the agent's belief unfolds according to the formula:

$$\begin{aligned} \beta_{0,k} &= P(\varepsilon = e_k) = 1/|E| \\ \beta_{1,k} &= \frac{P(f_1|\varepsilon = e_k)}{P(f_1)}\beta_{0,k} \\ &\vdots \\ \beta_{t,k} &= \frac{P(f_t|\varepsilon = e_k)}{P(f_t)}\beta_{t-1,k}. \end{aligned} \quad (3)$$

Decomposing the computation of the posterior belief allows the agent to choose the best question to ask the user at each turn, rather than having a fixed battery of questions. In this case, we define "best" as the question that is most likely to have a 'yes' answer given ε. This criterion indicates how often the question was observed in the training data in the context of emotions as they are currently weighted by $P(\varepsilon|f_1, ..., f_{t-1})$. The agent asks the best question and takes the user's response as input. It then parses the input to classify it into one of {yes, no, other}. This information is then used to update the agent's belief as to which emotion in E in most likely.

Identity questions are a special type of question where the agent makes a guess about the emotion. An affirmative answer to an identity question (e.g., "is it happy?") means that the agent successfully identified the user's chosen emotion. Any other answer to an identity question will set the posterior probability of that emotion to zero because the agent can be sure it is not the emotion of interest. Also, because it is playing a *twenty* questions game d is set to 20, but this could be changed for the agent to generalize to different question-asking tasks. The pseudo-code for the main loop of the adaptive Bayesian agent is shown in Algorithm 1.

3.2. EMO20Q Agent Implementation

The EMO20Q questioner agent is implemented as a state machine as seen in Figure 1. From the start state, the agent welcomes the user and waits until they are ready. When the user is ready, the agent enters the question asking state. From the question asking state, the agent can transition to a confirmation state when the question being asked is a guess (e.g., "is the emotion happiness?"). If an affirmative answer to a guess is confirmed, the emotion has been guessed successfully and the agent enters an intermediate state for asking if the user wants to play again. If not, the agent exits, but if the user wants to play again, the agent will reset its prior and start a new match.

[1] These forms of 'yes' and 'no' were determined from the data. The case of 'aye' is an example of how some users have tried trick the agent, in this case by talking like a pirate. From the agent's point of view, it will be difficult to distinguish the language of population that actually includes pirate demographics from language containing experimental artifacts like this.

Algorithm 1 adaptive Bayesian emo20q agent

Input: $F = Q \times A$, E, and $P(f|\varepsilon)$
$\beta_{0,k} \leftarrow 1/|E|, \forall k = 1...|E|$
for $i = 1$ **to** d **do**
  $q^{(i)} = \underset{q \in Q}{\operatorname{argmax}} P((q, \text{'yes'})|\varepsilon)$
  Print $q^{(i)}$
  $a^{(i)} \leftarrow$ user's input answer
  $f_i \leftarrow (q^{(i)}, a^{(i)})$
  $\beta_{i,k} \leftarrow \beta_{i-1,k} \cdot P(f_i|\varepsilon = e_k)/P(f_i), \forall k = 1...|E|$
  if ($q^{(i)}$ is identity question for $e_k \wedge a^{(i)} =$ 'yes') **then**
    Return: $e^* = e_k$
  end if
  if ($q^{(i)}$ is identity question for $e_k \wedge a^{(i)} =$ 'no') **then**
    $\beta_{i,k} \leftarrow 0$
  end if
end for
$k^* \leftarrow \underset{k \in 1...|E|}{\operatorname{argmax}}[\beta_{i,k}]$
$e^* \leftarrow e_{k^*}$
Return: most likely emotion given observations: e^*

The input and output behavior of the agent is implemented as a monkey patched read-evaluate-print loop. Read-evaluate-print loops are simply interactive, shell-like applications. *Monkey patching* refers to the technique whereby objects in dynamically interpreted languages like Python can have member functions reassigned at run-time. In this case, it is the read, evaluate, and print functions that are reassigned to implement state-specific behavior. Figure 1 shows the states and transitions that define the agent's general behavior.

The probablistic reasoning described in the previous section was implemented with the Natural Language Toolkit (NLTK) [9].

4. Results

The results of our usability experiments on fifteen subjects are summarized in Table 4 . To compare the agent's performance with human performance from previous studies [5], we used two objective measures and one subjective measure. The success rate, shown in column two Table 4, is an objective measure of how often the EMO20Q matches ended with the agent successfully guessing the user's emotion. The number of turns it took for the agent to guess the emotion is the other objective measure. The last column, naturalness, is a subjective measure where users rated how human-like the agent was, on a 0-10 scale.

The emotion words chosen by the subjects as "easy" were recognized by the agent with similar success rate and number of required turns as human-human matches. Some examples of "easy" emotions are anger, happiness, and sadness. However, successful outcomes were fewer in emotions chosen as "medium" and "difficult". Some examples of "medium" emotions are contentment, curiosity, love, and tiredness. Pride, frustration, vindication, and zealousness are examples of "difficult" emotions. The different classes of words were not disjoint: some words like anger, disgust, and confusion spanned several categories. A complete listing of the words chosen by the subjects of the experiment is given in Table 4.

The results in terms of successful outcomes and number of turns required to guess the emotion word are roughly reflected in the percent of words that are in vocabulary. Despite the low performance on emotion words deemed "medium" and "difficult", there was not a corresponding decrease in the perceived naturalness of the questioner agent.

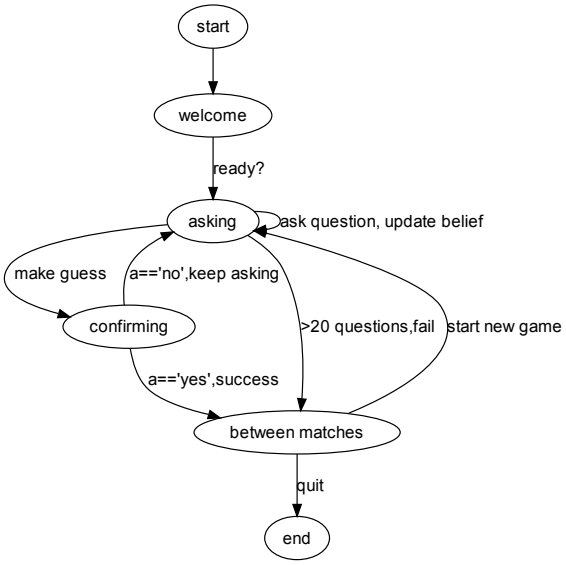

Figure 1: EMO20Q questioner agent state machine.

Table 1: Experimental results.

difficulty	% success	avg. turns	% in vocab.	naturalness
easy	73%	11.4	100%	6.9
medium	46%	17.3	93%	5.5
difficult	13%	18.2	60%	5.8
total	44%	15.6	84%	6.1

5. Discussion

The results we obtained show that this model provides an agent that users find to be human-like despite having fewer successful outcomes than in human-human matches. The naturalness of the agent's behavior will make it useful as a tool to obtain more ethnographic data about how people describe emotions. The out-of-vocabulary rate observed in Table 4 indicates that the agent is useful for eliciting new emotion words. Collecting more data would continue to improve the agent's coverage and performance. Apart from collecting more data, here are also algorithmic and practical improvements that can be made.

Algorithmically, the area that we could improve the most is the selection of the next question. Currently, the naive Bayes

Table 2: Observed emotion words by difficulty.

difficulty	examples
easy	happiness, anger, sadness, calm, confusion
medium	anger, confusion, contentment, curiosity, depression, disgust, excitement, fear, hate, irritation, love, melancholy, sorrow, surprise, tiredness
difficult	devastation, disgust, ecstasy, ennui, frustration, guilt, hope, irritation, jealousy, morose, proud, remorse, vindication, zealousness

assumption of independence results in similar questions being asked, especially questions that are semantically opposite (e.g., "is it a positive emotion?" and "is it a negative emotion?"), so incorporating semantic relations between features will help. Also, the selection of the next question based on the maximum probability of the answer being "yes" tends to ask the same questions in each match. This repetitive behavior is not ideal if the goal is to collect a wide variety of human knowledge, so this is another area for improvement. In an earlier paper [10], we describe a methodology for exploring questions that are not well represented in the training data in order to create an agent that is an effective learner. Other possible approaches include choosing questions that minimize the expected entropy of the posterior [11, 12].

Another algorithmic issue for future work is that the EMO20Q agent learns in batch-mode, meaning that it will make the same mistakes in consecutive dialogs unless there is an explicit retraining. Moreover, our current algorithm does not react differently to incongruous information. Besides learning statistically from new data, it would be ideal to learn more from data that appears incongruous or salient. Another technical issue is the quantization of answers into $\{$yes, no, other$\}$ categories. This is a coarse categorization, but it helps to decrease sparsity in the feature space. Having a continuous or fuzzy scale between "yes" and "no" is an alternative representation that could provide a more informative representation while also avoiding sparsity issues.

One practical improvement is to remove turns from the training data that contain anaphoric references. Because the agent automatically selects turns from previous human-human matches for use in human-computer matches based only on questions and answers, there are anaphora which are unresolved that appear as non sequiturs to the users. The presence of these non sequiturs led to less informative responses and (based on the user comments) led to lower naturalness scores.

The collection and representation of world knowledge is frequently a bottleneck in both artificial intelligence and natural language processing. Designing agents that can automatically collect this information is therefore an important possibility in overcoming these limitations. Many times, the world knowledge problem is posed as a task of collecting facts. However, natural language descriptions of emotions may vary between cultures and over time. Therefore, rather than collecting facts to form a fixed ontology, we feel that the idea of ethnography is more applicable for this kind of subjective cultural data. The question-asking framework to do this has been theoretically described as a Socratic epistemology [13]. The quantitative, computational approach to social sciences that is enabled by natural language processing and the prevalence of online communication can be seen in emerging fields such as sentiment analysis and crowd-sourcing.

The emotions recognized by the EMO20Q questioner agent are more conceptual than the typical emotion recognition task. Being able to integrate traditional emotion recognition, i.e., the recognition of social signals, with the understanding of natural language descriptions of emotions is an open issue with many implications. The task of annotation of emotional data is one area where applying linguistic descriptions to emotions in social interactions could be fruitful. The same agent playing the role of an interlocutor as well as an evaluator is an intriguing and important possibility with practical applications to more ecologically-valid interaction based evaluation and derivation of human informatics for clinicians and other analysts [14].

The code and data for the agent described in this paper can be found at http://sail.usc.edu/emo20q .

6. Acknowledgments

The authors would like to thank the EMO20Q players, Nassos Katsamanis, Kartik Audhkhasi, Jeremy Chi-Chun Lee, the reviewers of ACII2011, Ben Alderson-Day, and Carbon Five Hack Night.

7. References

[1] S. C. Marsella and J. Gratch, "Ema: A process model of appraisal dynamics," *Journal of Cognitive Systems Research*, vol. 10, pp. 70–90, 2009.

[2] C. F. Hockett and S. Altmann, *A note on design features*. Indiana University Press, 1968, pp. 61–72.

[3] M. Arnold, D. Shenviwagle, and L. Yilmaz, "Scibrowser: a computational ethnography tool to explore open source science communities," in *Proceedings of the 48th Annual Southeast Regional Conference (ACMSE 2010)*, Oxford, MS, April 2010.

[4] E. C. Wenger, *Communities of practice: learning, meaning, and identity*. Cambridge University Press, 1998.

[5] A. Kazemzadeh, P. G. Georgiou, S. Lee, and S. Narayanan, "Emotion twenty questions: Toward a crowd-sourced theory of emotions," in *Proceedings of ACII'11*, 2011.

[6] J. D. Williams and S. Young, "Partially observable markov decision processes for spoken dialog systems," *Computer Speech and Language*, vol. 21, no. 2, pp. 393–422, April 2007.

[7] B. Thomson, J. Schatzmann, and S. Young, "Bayesian update of dialogue state for robust dialogue systems," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2008.

[8] A. Kazemzadeh, J. Gibson, P. Georgiou, S. Lee, and S. Narayanan, "Emo20q questioner agent," in *Proceedings of ACII (Interactive Event)*, 2011, the interactive demo is available at http://sail.usc.edu/emo20q/questioner/questioner.cgi.

[9] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*. O'Reilly Media, 2009.

[10] A. Kazemzadeh, S. Lee, P. G. Georgiou, and S. Narayanan, "Determining what questions to ask, with the help of spectral graph theory," in *Proceedings of Interspeech*, 2011.

[11] D. Geman and B. Jedynak, "Shape recognition and twenty questions," IN PROC. RECONNAISSANCE DES FORMES ET INTELLIGENCE ARTIFICIELLE (RFIA), Tech. Rep., 1993.

[12] B. Jedynak, P. I. Frasier, and R. Sznitman, "Twenty questions with noise: Bayes optimal policies for entropy loss," *Journal of Applied Probability*, vol. 49, no. 1, pp. 114–136, March 2012.

[13] J. Hintikka, *Socratic Epistemology: Explorations of Knowledge-Seeking by Questioning*. Cambridge University Press, 2007.

[14] B. Alderson-Day, "Verbal problem-solving in autism spectrum disorders: A problem of plan construction?" *Autism Research*, vol. 4, no. 6, pp. 401–411, December 2011.