Designing Emotionally Adept Social Agents

John U. Balis
balis@wisc.edu
University of Wisconsin-Madison
Madison, Wisconsin

Andrew Geng
ageng@wisc.edu
University of Wisconsin-Madison
Madison, Wisconsin

Bilge Mutlu
bilge@cs.wisc.edu
University of Wisconsin-Madison
Madison, Wisconsin

ABSTRACT

Current state of the art consumer social agents such as Google Home, Siri, and Alexa are not built to take into account user emotional state when deciding how to interact with a user. When interacting with one another, humans employ a variety of strategies to improve the efficiency and subjective emotional cadence of their interaction. Therefore, interaction strategies which lack an awareness of user emotional state may fail to map to real world interactions. It may be possible to benefit from human interaction strategies when designing behaviors for social agents. We propose a scalable system for translating human strategies into a machine learning model for social agent interactions.

KEYWORDS

human-computer interaction, social agents, intelligent agents, emotion classification, machine learning

ACM Reference Format:
John U. Balis, Andrew Geng, and Bilge Mutlu. 2020. Designing Emotionally Adept Social Agents. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Social agents are virtual and physical technologies that interact with their users using human norms of communication. Popular commercial social agents include Google Home, Alexa, and Siri. These social agents are widely used and often present significant utility in day to day activities by allowing users to ask searchable questions, control home automation and media devices, and manage emails, to-do lists, and calendars [8].

Recent studies have shown that one in four households has a home assistant manufactured by Amazon or Google, and that nearly 47.4% of the U.S. populace owns a smartphone with the Siri voice assistant [15, 16]. As the prevalence of social agents continues to increase, it is worthwhile to consider flaws in user experience with the functionality of contemporary social agents.

The primary objective of this paper is to propose a system for designing emotionally adept social agents which recognizes and responds to the user's state, emotional or otherwise, in a manner akin to that of human behavior. By imbuing human-like recognition and response, we hope to be able to improve user experience with the social agent. A secondary objective of this paper is to provide a system that is free from researchers' presuppositions surrounding groupings of emotional states, as well as appropriate responses to each given emotional states. Through this, we hope to provide a basis for analyzing contemporary theories of emotion, as well as to discover novel interactions between humans and social agents.

2 BACKGROUND

There have been numerous studies exploring user experience with social agents. Some publications have claimed that there is a negative user experience when interacting with social agents, citing limitations on fully hands-free interaction, and failures to take into account norms and social cues [1]. Language issues have also been significantly associated with claims of unsatisfactory experience. Users who are not native English speakers have reported difficulty engaging with English-language social agents [18]. Other issues, such as an inability by social agents to deliver on expectations of human-like response during user interaction, have shown that there is room to improve on the current state of the art for social agents [13].

Fundamentally, many state of the art social agents are not equipped to engage with their users' emotional state. In real world interactions, humans employ various tactics in response to the emotional state of the person with whom they are interacting. These phenomenons are well documented by the appraisal theory of emotions [19]. However, modern state of the art social agents do not leverage this theory to dynamically respond to user emotional state. This issue has been well documented and explored in recent literature [2, 4, 7, 9–12, 14, 20]. A relatively recent review on the future of personal assistants proposed by Cohen et al., proposed several questions such as *can the user be characterized as being in multiple emotional states simultaneously, what emotional*

states are important to track, and *how should an assistant react to the emotional state of the user*, are proposed as important questions that needs further exploration [17].

Progress has been made in existing literature, which has attempted to address social agents' inability to dynamically respond to user emotional state [2, 4, 7, 9–12, 14, 20]. Specifically, many of the proposed social agents have showcased effectiveness in classifying user emotional state as well as procuring some form of relevant response to each state. The schema proposed by Jain et al., showcases an agent which categorizes complex emotional states and applies specific treatment for each predetermined category of emotion [9]. Other publications have also similarly based their work on the appraisal theory of emotions[4]. Many of these publications propose a response to a user emotion as a subsequent emotion exhibited by the social agent [11, 14, 20].

Although many publications have shown promising results, most proposed social agents have chosen to follow a pattern of classifying user emotion to assumed categories, each corresponding to a predefined response from the social agent [9, 10, 12]. The available categories of user emotions tend to be predefined by the researchers based on well documented emotional states, such as happiness and anger. The responses assigned to each category also tend to be predefined by the researchers. Due to the constraining nature of categorizing emotions and labeling each state into predesignated categories, more sophisticated emotions, that don't fit in the designated categories, might go unanswered for or, in the worst case, be classified un-optimally. This may result in an overall poorer user experience. Additionally, due to the secondary constraining nature of the predesignated responses to each user emotion category, these responses to the user's state do not cater to the situation and instead are only capable of providing a single response for a broad range of user states. Additionally, these predesignated actions are not applied as treatments to alter the user's emotional state, rather they are only intended to induce the social agent to act in what is assumed to be a contextually appropriate manner [9]. Specifically, the goal of these predesignated actions is not to improve the user experience, but rather to acknowledge a change in the user's state. Thus, it could be valuable to improve user experiences with social agents by applying actions that coincide with human strategies for improving emotional cadence.

3 SYSTEM

Summary

The proposed custom software stack consists of a browser interface, custom backend servers, an Android application, a MySQL database, an altered Mycroft voice assistant, and an Empatica E4 watch. The network layout of these components is showcased in figure 1.

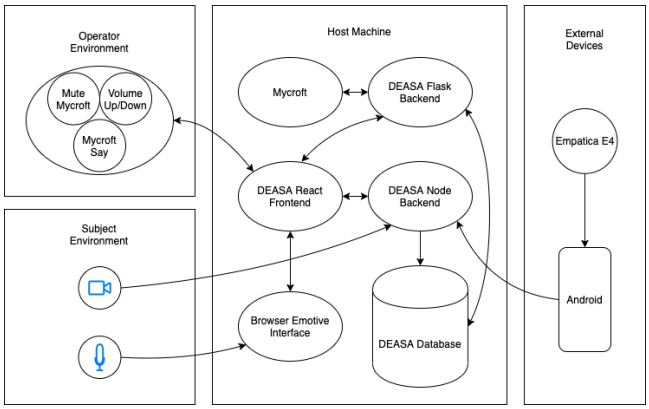

Figure 1: System network and device layout

The system used in this study runs on two machines. The Host machine provides the Browser Emotive Interface, shown in figure 3, to the subject, and the "operator environment" machine provides the Browser Control Interface, shown in figure 2, to the operator.

The Browser Emotive Interface can give emotive feedback to the subject, while the Browser Control Interface allows the operator to control Mycroft, as well as the Browser Emotive Interface. The Empatica E4 wristwatch worn by the subject is connected via Bluetooth to an Android phone which is connected via local network to the Host Machine.

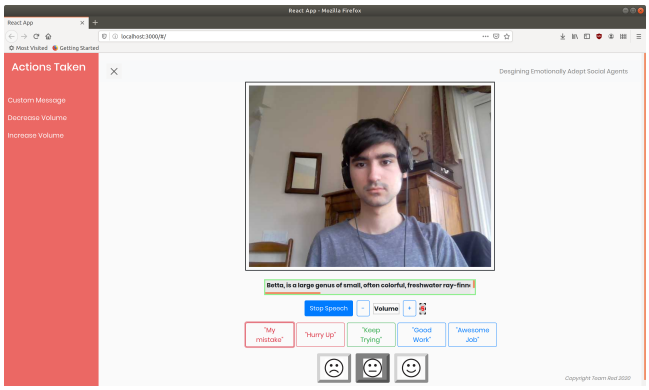

Figure 2: Operator Interface (Browser Control Interface)

The system supports a mode where operator control actions are recorded, in addition to a mode where two machine learning models are applied to an existing data set to automatically perform actions if appropriate, without requiring them to be requested by the operator.

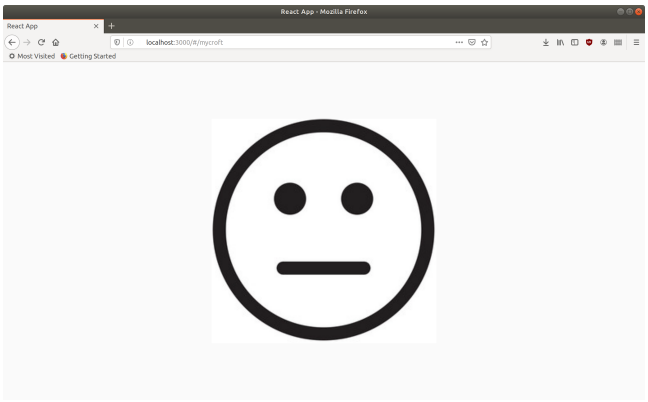

Figure 3: Subject Interface (Browser Emotive Interface)

React Frontend

https://github.com/DEASA-System/DEASA_react_frontend

The DEASA React Frontend is a web interface built utilizing the React framework and contains the Browser Control Interface for the operator, the Browser Emotive Interface for the subject, and a control panel for experimenters. Actions and subject states collected from the DEASA React Frontend are routed to the DEASA Flask Backend and the DEASA Node Backend. Data that the Browser Control Interface and Browser Emotive Interface collects and uses include: subject vocal volumes, operator actions, and the Mycroft utterance feed. The subject's vocal stream is collected from the DEASA React Frontend's Browser Emotive Interface via the machine's default microphone before being parsed for vocal volume in decibels, and sent to the DEASA Node Backend. A simple significance threshold, estimated from standard deviations, is set to 0.05 decibels for filtering out background noise. Samples with recorded volume below 0.05 decibels are not recorded. Operator actions are collected from the DEASA React Frontend's Browser Control Interface and are sent to the DEASA Node Backend and the DEASA Flask Backend. Data from the Mycroft utterance feed is sent to the text to speech system within the DEASA React Frontend's Browser Control Interface. Any utterance from Mycroft will be sent to the DEASA React Frontend as a string before being spoken utilizing the text to speech library *speak-tts*.

Flask Backend

https://github.com/DEASA-System/DEASA_flask_backend

The DEASA Flask Backend is a python Flask server responsible for routing facial landmarking data, a video feed, operator actions, and a Mycroft utterance feed between the DEASA React Frontend, the DEASA Node Backend, Mycroft, and the DEASA Database. Operator responses collected from the Browser Control Interface are sent to the DEASA Flask Backend by the DEASA React Frontend, and then relayed to both Mycroft and the DEASA Database so that they can be carried out by Mycroft and recorded by the database. Additionally, the Flask backend also collects the Mycroft utterance feed issued by Mycroft, and relays this information to the DEASA React Frontend. Additionally, the DEASA Flask Backend records a video focused on the face of the subject and sends the feed to the DEASA React Frontend. The Flask backend extracts a series of facial landmarks from each frame of the video stream, utilizing the *opencv* library. A lightweight 68 point landmarking is applied to each facial frame before the resulting landmarking data is sent to the DEASA Database. Images of the subject's face are collected and sent using code adapted from the companion code to [5]. The core of the system is adapted from [6], but the Vue Frontend suggested by this article is replaced with the DEASA React Frontend. A simple normalization is applied to the facial landmarks to normalize each subject's unique facial features. When in automatic response mode, the DEASA Flask Backend uses a uses a pair of saved models to predict whether it should take an action and what action to take. This decision process is performed at some fixed time interval, and chosen actions are sent to Mycroft and the DEASA React Frontend for execution.

Node Backend

https://github.com/DEASA-System/DEASA_node_backend

The DEASA Node Backend is a server built utilizing *node.js* and contains routes used by the DEASA React Frontend and the Empatica Android Application. The DEASA Node Backend is intended to be used as a server for collecting data from the DEASA React Frontend and Empatica Android Application before sending the data to the DEASA Database. Data that the DEASA Node Backend collects and uses include: subject vocal volume, physiological data, and operator actions. The subject's vocal volumes are obtained from the DEASA React Frontend before being sent into the DEASA Node Backend, and subsequently sent into the DEASA Database. Physiological data are obtained from the Empatica Android Application before being sent into the DEASA Node Backend and subsequently sent into the DEASA Database. The collected physiological data include: electrodermal activity, heart rate, body temperature, and body movements. Operator actions are obtained from the DEASA React Frontend before being sent into the DEASA Node Backend and subsequently sent into the DEASA Database.

MySQL Database

https://github.com/DEASA-System/DEASA_tools

The DEASA Database contains 4 tables which include: FaceTable, ResponderTable, VolumeTable, and DataTable. VolumeTable consists of the subject's timestamped vocal volume, collected from the Host Machine's microphone. FaceTable consists of timestamped facial landmarking values. ResponderTable consists of timestamped operator actions. DataTable consists of timestamped physiological data collected from the subject by the Empatica E4 Wristwatch.

Mycroft

https://github.com/DEASA-System/DEASA_mycroft

A slightly-altered version of Mycroft is used, specifically a modified mycroft-volume skill and a modified Custom research-session skill. All commands are sent to Mycroft from the DEASA Flask Backend through the Mycroft Messagebus. The modified volume-skill sets Mycroft's volume to a specified level by an intent sent through the Mycroft messagebus. The research-session skill issues utterances sent from the DEASA Flask Backend, in addition to sending all Mycroft utterances through a direct websocket to the DEASA Flask Backend.

Android Application

https://github.com/DEASA-System/empatica_app

The Empatica Android Application, shown in figure 4, is an Android application built utilizing E4link, an Empatica-provided developer package. The Empatica Android application is intended to be used for collecting data from the Empatica E4 Wristband, shown in figure 5, through a Bluetooth adapter from a Android device. All collected physiological data are subsequently sent to the DEASA Node Backend. The specific physiological data that the Empatica E4 Wristband collects include: electrodermal activity, heart rate, body temperature, and body movement. These physiological data are routed from the Empatica Android Application before being sent to the DEASA Node Backend.

Learning Model

https://github.com/DEASA-System/DEASA_flask_backend

Data detailing operator actions, subject physiological data, subject vocal volume, and facial landmarking data are stored during training sessions into the DEASA Database. Two training sets are constructed, one to train a model to predict if an action should be taken, given a preceding time window, and one to train a model to predict which action to take, given the same preceding time window. Data is first bucketized into intervals of fixed temporal length; if more than one sample for a given feature falls into a bucket, the sample values will be averaged together. If a temporal range has no values for any feature, a bucket will not be created for that range.

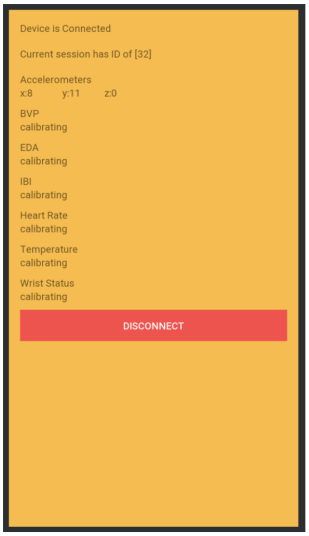

Figure 4: The Android companion app for the Empatica E4

Figure 5: The Empatica E4 watch

Each feature within the bucket table is then standardized by dividing it by the mean value for that feature. Once the temporal buckets have been created, they are assigned a cluster label using an agglomerative clustering process with a euclidean metric. This clustering is used to create the "action time prediction" and "action time classification" training sets.

The "action type prediction" training set is composed of samples corresponding to each operator action, with the sample's label being the type of the action, and the feature values being occurrence counts for each type of cluster in a preceding time window. The "action time prediction" training set is calculated by uniformly randomly sampling the end time of a fixed interval from the total interval encompassing the response times. If one or more response times fall into the sampled range, then a sample is created with an "action taken" label. If no response times fall into the sampled range, then the label is set to "no action taken." In both cases, features are generated corresponding to the number of occurrences of each type of cluster during some fixed preceding window of time. Models are fit to each of the training sets using a keras multi-layer neural network, and subsequently saved alongside process data to be used in real time by the DEASA Flask Backend. Keras is a neural network library, and the code used in this study is adapted from [3].

At runtime, the DEASA Flask Backend collects a fixed window of data trailing the current time (this window is the same length as the windows used to generate the training set samples). It then bucketizes and clusters the data from that range, where cluster *id* is determined by the nearest neighbor in the standardized clustering of the training set. Next, features are generated using the number of clusters in

a fixed preceding time window in the same way the features are generated in the training sets. The resulting sample is then sent to the "action time prediction" model. If the model predicts that an action should be taken, then the sample is given to the "action type prediction" model, which returns a prediction of what action type should be taken. The DEASA Flask Backend then sends a command to either the DEASA React Frontend or Mycroft, triggering the specified action.

Recommended System Specs

Host Machine	Operator Control Machine	Empatica Interface
Debian or Ubuntu Machine	Modern web browsers (Tested with firefox, chrome, and safari)	Android phone with Bluetooth (Android Version 7.0 and 7.1.1)

4 METHODS

Summary

We propose a methodology to learn user emotional states, and appropriate machine response to each emotional state, by translating human strategies during cooperative interactions into a pair of multi-layered neural network models for social agents. The neural network models are trained using a "Wizard of Oz" study structure implemented through a custom software stack.

The proposed "Wizard of Oz" study involves a participant (subject) interacting with a social agent that is believed by the subject to be autonomous. However, actions by the social agent are partially specified by a different participant (operator). Through this setup, the operator's actions inform and train the neural network models on what actions to take based on the state of the subject. The state of the subject is monitored through: vocal volume, facial landmarks, and several physiological data.

Human Subjects Components

Within this proposed methodology, there are two phases of human subject study sessions. The phase 1 study session (Wizard of Oz phase) involves the collection of data through the Wizard of Oz system with the goal training 2 neural network models. Subsequently, the phase 2 study session (evaluation phase) involves utilizing the trained models in a controlled and experimental trial as a means of evaluating their validity.

Phase 1 (the Wizard of Oz phase) is designed to provide the necessary data to form a valid training set for the neural network models. In this phase, each session consists of two participants with one participant (subject) performing a set of tasks using the Mycroft social agent while being monitored for vocal volume, facial landmarks, electrodermal activity, heart rate, body temperature, and body movements. Another participant (operator) will, unbeknownst to the subject, observe the subject as they interact with the social agent. The operator will be able to request, through the control interface, specific actions for the social agent to take. The operator's stated goal is to improve the interaction between the subject and social agent through these interface commands. A baseline compensation is designated, however, additional monetary reward will be given to both the operator and subject if the subject's designated tasks are performed correctly. The subject's tasks consist of a series of tedious math problems and sets of basic information gathering problems. Actions that the operator can take involve muting the social agent, increasing/decreasing social agent's speech volume, changing display expression of social agent, and saying predesignated words such as "great work" or "keep trying". The entire session is limited to 30 minutes, and a debriefing is given to both participants at the end of the session.

Phase 2 (the evaluation phase) is designed to provide a measurement for the effectiveness of the proposed neural network models. In this phase, participants are randomly assigned into control and experimental groups. Both the control and experimental groups are given the same set of tasks to complete (note this task is identical to the subject's task in phase 1). The control group is given a default, unaltered Mycroft social agent while the experimental group is given an altered Mycroft social agent with the neural network models attached. Measurements are collected from both control and experimental groups which consist of: speed of task, accuracy of task, and a Likert scale questionnaire.

Participant Specifics

(Disclaimer: no participants have been recruited due to COVID-19, the following is meant as a template!)

A total of $n = 72$ participants were hired for this study with 36 males and 36 females. There were k participants in total to which their data was discarded. All participants were between the ages of 18 and 55 with k participants being University of Wisconsin-Madison students. All participants were required to be native English speakers. A total of 36 participants were involved in the phase 1 study session and a total of 36 participants were involved in the phase 2 study session.

5 PRELIMINARY RESULTS

(Disclaimer: this study has not been run with participants sampled from the general population. Rather, all insofar collected data is from example runs by the researchers.)

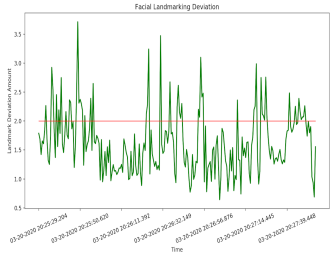

Figure 6: Example Facial Landmark Deviation

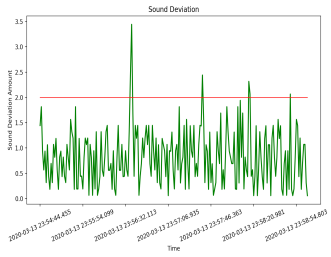

Figure 7: Subject Microphone Volume Deviation

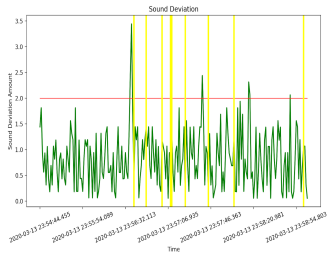

Figure 8: Volume Deviation annotated with Mute Actions

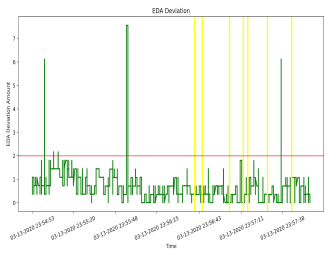

Figure 9: EDA Deviation, annotated with Mute Actions

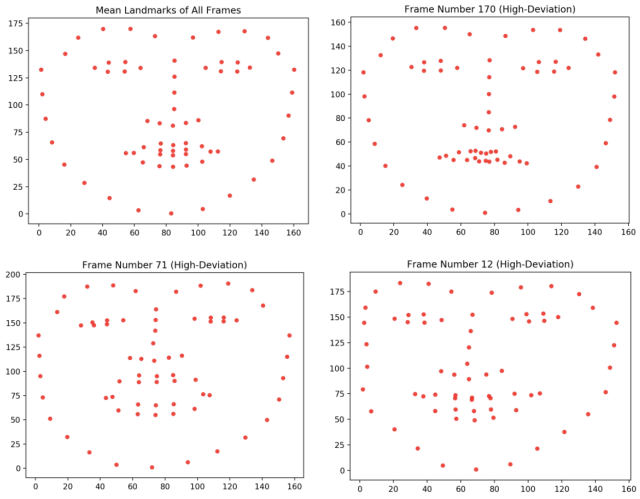

Figure 10: Mean Deviation of Each Facial Landmark, as well as plots of landmarks from three Notable High-Deviation time frames

Observational Results

Significant frames of the subject's facial features are extracted over the duration of the study. Frames were determined as significant under the assumption of a multivariate Gaussian distribution where each individual facial landmark was expected to follow that of an independent Gaussian distribution. Any frames with facial landmarks that exceed 2 standard deviations are considered significant. In figure 6 we see a time series graph of the greatest facial landmarks within each frame. Some individual time frames that were shown to be significant are plotted in figure 10, as well as the mean of each individual landmark.

Significant frames of the subject's vocal volume are extracted over the duration of the study. Frames were determined as significant under the assumption of a Gaussian distribution where any frames that have a vocal volume exceeding that of 2 standard deviations away from the mean are considered significant. In figure 7 we see a graph of the vocal volume, in terms of deviations from the mean, over time. Subsequently, in figure 8, we show a comparison of the significant frames in correlation to the times of when "mute" actions were taken by the operator.

Significant frames of the subject's electrodermal activity data are extracted over the duration of the study. Frames were determined to be significant under the assumption of a Gaussian distribution where any frames that had a electrodermal activity value exceeding that of 2 standard deviations from the mean are considered significant. In figure 9 we see a graph of the electrodermal activities in terms of deviations from the mean over time annotated with the times of operator mute actions.

Cross Validation of Model

(Disclaimer: these results are unsafe to generalize, as they were arrived at using example data)

Provisional cross validation results have yielded an "action time prediction" model validation accuracy of approximately 0.9 and an "action time classification" model validation accuracy of 0 or 0.0769. The training set size of "action time prediction" model was 240 with a validation set size of 60. Additionally, the training set size of "action time classification" model was approximately 52 with a validation set size of approximately 13.

Empirical Results from Phase 2

(Disclaimer: Phase 2 has not been completed due to COVID-19 and subsequent limitations on human subjects research)

6 DISCUSSION

This paper showcases the collection of several types of data which are subsequently used to train two response model. Each type of collected data comes with its own set of advantages and disadvantages. Tracking facial landmarks provides larger sets of features and has been shown to correlate well with explicitly labeled emotions [22]. However, a significant downside of facial landmarking is that there are many situations where it may not be realistic or permissible for a social agent to use facial landmarks. While it may be possible on a

car-mounted social agent or one posted in a public space to access facial landmarks, a social agent running from a smart watch or a phone may not always have a camera angled towards the face of the user. Additionally, concerns regarding privacy may result in difficulty obtaining permissions to access its host device's camera. Facial landmarking may also be too computationally expensive depending on the platform. Many standalone social agents platforms are not designed with cameras. Ultimately, facial landmarking may not be a feasible data to collect for all commercial social agents.

Another type of data considered in this study is the physiological data, such as electrodermal activity and body temperature, collected from the Empatica E4 wristwatch. A significant downside is that it is unlikely a social agent, running on a consumer smartwatch, would have the ability to gather electrodermal activity data. Furthermore, it is unlikely a social agent running on a standalone device, would have any access to any physiological data. Additionally, smart watches may have Software Development Kits which do not support real time data extraction.

Lastly, the proposed model incorporates user vocal volume. User vocal volume will likely be available in systems where users issue voice commands to a social agent. Therefore, this data should be widely accessible across most social agent platforms, making it very feasible for a commercial social agent to incorporate user volume in its response model.

Additional possible features outside of those considered in the proposed model include the 'sentiment' of a command issued by the user based on the words comprising the command. A numeric sentiment feature can be calculated using a numeric sentiment heuristic such as the approach outlined in Taddy et al. [21].

One possible direction of research, using the architecture presented in this paper, is to determine which combination of features yields the highest cross validation accuracy for the "action type classification" model and the "action time classification" models. It is additionally worthwhile to evaluate which resulting pair of predictive models leads to the highest user satisfaction. Another consideration for further research in this space is which treatments should be available for the system to perform. The system presented in this paper has the following treatments available: increment or decrement the volume of the social agent's voice, share one of five pre-selected messages with the user, prematurely terminate whatever the social agent is currently saying, and change the avatar image being displayed to the user on the visual interface page. Assuming they are permitted by the software development kit of that particular platform, all of these treatments, except for the avatar image, would be available to a system operating on a standalone social agent or one operating from a phone. There are additional possible treatments which are not considered in this implementation such as: playing background music, changing the sound played by the social agent in response to a wakeword, or changing the wakeword sensitivity. In the context of social agents which are hosted on physical robots, physically gesturing or altering face/body position can also be applied as a treatment.

	Android/EmpaticaE4	IPhone/Apple Watch	Android/Empatica E4/Laptop(Ubuntu)	Android/Empatica E4/Laptop(Ubuntu)/ OpenBCI EEG setup
Face Landmarks	Partial: Google Mobile ML Kit FaceDetector Warning: Face may not be in view of phone	Partial: Dlib Warning: Face may not be in view of phone	Yes: python OpenCV	Yes: python OpenCV
Volume	Yes: built in microphone	Yes: built in microphone	Yes: built in microphone	Yes: built in microphone
Physiological Data	Yes: EDA, Heart rate, Skin Temperature, Acceleration(localized to watch)	Yes: Heart rate, Acceleration(local to watch)	Yes: EDA, Heart rate, Skin Temperature, Acceleration(localized to watch)	Yes: EDA, Heart rate, Skin Temperature, Acceleration(localized to watch), EEG
Tokenized User Speech	Yes	Yes	Yes	Yes

Figure 11: A table of possible device setups for implementing an emotion-aware emotional agent system. The Android/Empatica E4/Laptop(Ubuntu) system was implemented in this study

7 ACKNOWLEDGMENTS

Funded by the University of Wisconsin-Madison LS Honors Program through a Trewartha Senior Thesis Research Grant awarded by the LS Honors Program.

Special thanks to the Wisconsin HCI laboratory and Andrew Schoen for their guidance throughout this project.

REFERENCES

[1] David Coyle Kellie Morrissey Peter Clarke Sara Al-Shehri David Earley Benjamin Cowan, Nadia Pantidi and Natasha Bandeira. 2017. "What can i help you with?": infrequent users' experiences of intelligent personal assistants. 1–12. https://doi.org/10.1145/3098279.3098539

[2] Cynthia Breazeal. 2003. Emotion and Sociable Humanoid Robots. *International Journal of Human-Computer Studies* 59 (07 2003), 119–155. https://doi.org/10.1016/S1071-5819(03)00018-1

[3] Jason Brownlee. 2019. Multi-Class Classification Tutorial with the Keras Deep Learning Library. https://machinelearningmastery.com/multi-class-classification-tutorial-keras-deep-learning-library/

[4] Jonathan Gratch and Stacy Marsella. 2004. A domain-independent framework for modeling emotion. *Cognitive Systems Research* 5 (12 2004), 269–306. https://doi.org/10.1016/j.cogsys.2004.02.002

[5] Miguel Grinberg. 2017. Flask Video Streaming Revisited. https://blog.miguelgrinberg.com/post/flask-video-streaming-revisited

[6] Michael Herman. 2019. Developing a Single Page App with Flask and Vue.js. https://testdriven.io/blog/developing-a-single-page-app-with-flask-and-vuejs/

[7] José Vidal Hong Jiang and Michael Huhns. 2007. EBDI: an architecture for emotional agents. 11. https://doi.org/10.1145/1329125.1329139

[8] Matthew Hoy. 2018. Alexa, Siri, Cortana, and More: An Introduction to Voice Assistants. *Medical Reference Services Quarterly* 37 (01 2018), 81–88. https://doi.org/10.1080/02763869.2018.1404391

[9] Shikha Jain and Krishna Asawa. 2015. EMIA: Emotion Model for Intelligent Agent. *Journal of Intelligent Systems* 0 (01 2015). https://doi.org/10.1515/jisys-2014-0071

[10] Shikha Jain and Krishna Asawa. 2015. Programming an Expressive Autonomous Agent. *Expert Systems with Applications* 43 (08 2015). https://doi.org/10.1016/j.eswa.2015.08.037

[11] Hong Jiang and José Vidal. 2006. From rational to emotional agents. (01 2006).

[12] A. Loyall Joseph Bates and Reilly Neal. 1994. Integrating Reactivity, Goals, and Emotion in a Broad Agent. (11 1994).

[13] Ewa Luger and Abigail Sellen. 2016. "Like Having a Really Bad PA": The Gulf between User Expectation and Experience of Conversational Agents. 5286–5297. https://doi.org/10.1145/2858036.2858288

[14] John Yen Magy El-Nasr and Thomas Ioerger. 2000. FLAME Fuzzy logic adaptive model of emotions. *Autonomous Agents and Multi-Agent Systems* 3 (09 2000), 219–257. https://doi.org/10.1023/A:1010030809960

[15] Nielsen Media. 2018. *(SMART) SPEAKING MY LANGUAGE: DESPITE THEIR VAST CAPABILITIES, SMART SPEAKERS ARE ALL ABOUT THE MUSIC.* Retrieved April 26, 2020 from https://www.nielsen.com/us/en/insights/article/2018/smart-speaking-my-language-despite-their-vast-capabilities-smart-speakers-all-about-the-music/

[16] S. O'Dea. 2020. *Subscriber share held by smartphone operating systems in the United States from 2012 to 2019.* Retrieved April 26, 2020 from https://www.statista.com/statistics/266572/market-share-held-by-smartphone-platforms-in-the-united-states/

[17] Eric Horvitz Rana Kaliouby Phil Cohen, Adam Cheyer and Steve Whittaker. 2016. On the Future of Personal Assistants. 1032–1037. https://doi.org/10.1145/2851581.2886425

[18] Aung Pyae and Paul Scifleet. 2018. Investigating differences between native english and non-native english speakers in interacting with a voice user interface: a case of google home. 548–553. https://doi.org/10.1145/3292147.3292236

[19] Klaus Scherer. 2005. *Appraisal Theory.* 637 – 663. https://doi.org/10.1002/0470013494.ch30

[20] Syed Ali Raza Richard Billingsley Suman Ojha, Jonathan Vitale and Mary Anne Williams. 2019. Integrating Personality and Mood with Agent Emotions. *International Conference on Autonomous Agents and Multiagent Systems* (05 2019), 2147–2149.

[21] Matt Taddy. 2013. Multinomial Inverse Regression for Text Analysis. *J. Amer. Statist. Assoc.* 108, 503 (2013), 755–770. http://www.jstor.org/stable/24246859

[22] Ivona Tautkute, Tomasz Trzcinski, and Adam Bielski. 2018. I Know How You Feel: Emotion Recognition with Facial Landmarks.