Learned babbling for more emotionally intelligent virtual agents*

Trevor Buckner
Yale University
New Haven, Connecticut
trevor.buckner@yale.edu

Robert Baines
Yale University
New Haven, Connecticut
robert.baines@yale.edu

ABSTRACT

The ability for a human to connect emotionally with a virtual agent could largely be dependent on the agent's ability to express and detect emotion of a user. A large part of emotional expression in humans comes from speech, and underlying tones and cadences within that speech. Current technology is not at a stage where a virtual agent can dynamically generate authentic, intelligible speech beyond pre-selected phrases. We propose a system based on deep neural networks that is trained to recognize and classify the emotion in a person's speech, and then reciprocate that emotion by generating some babbling noises that approximate human speech in the same emotion. This would provide more relate-able emotional feedback to the user, and could perhaps be used for next-generation non-player-characters in video games, or as a practical tool to help people on the Autism spectrum learn how tonality on their voice might impact social perception.

KEYWORDS

Virtual Agent, Convolutional Neural Networks, Learning Emotion, Deep Learning, Autism

ACM Reference Format:
Trevor Buckner and Robert Baines. 2018. Learned babbling for more emotionally intelligent virtual agents. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Figure 1: We present an interactive virtual agent software architecture that considers human emotion, in the form of a user's speech, in formulating babbling responses and generating emotional faces. The work represents a step toward next-generation non-player characters for video games, and potentially could be used as a digital outlet for people with ASD to practice the (customary) social impacts of their speech tonality.

1 INTRODUCTION AND RELATED WORK

Toward the goal of more life-like, sentient virtual agents, there has been extensive work on human speech recognition and parsing. A variety of approaches have been proposed to interpret and represent spoken word. Among these, large successes have been witnessed with deep feed-forward neural networks [1], deep recurrent neural networks [2], and other machine learning approaches like support vector machines [3]. Perhaps due to the massive success and growth of the natural language processing field, there have been occasional research efforts focused on recognizing the emotion embedded in that language. Emotion, aside from its explicit expression in the meaning of words, is a crucial aspect of communication. Studies of human-to-human verbal interaction suggest that tonality, cadence, and frequency of spoken words are significant features that carry as much influence as explicit dialogue. For instance, without any changes in wording, the same sentence can be threatening or ironic, depending on the length of stressed words [4]. Likewise, according to Mortensen, leading expert in communication theory, "...the verbal part of a spoken message has considerably less effect on whether a listener feels liked or disliked than a speaker's facial expression or tone of voice." [5]. The sophisticated nature of inflections in human speech makes deploying a deterministic model for use in virtual agents seemingly intractable. Indeed, previous attempts to detect emotion in speech often rely upon other signals, such as supplementary video data [10, 11] or highly complicated feature tagging, ranking nebulous attributes such as valence, tension, or antagonism [12, 13]. To simplify this process and minimize the amount of data pre-processing by humans, it is meaningful to be able to classify the emotion a person is expressing directly from the raw audio signal alone. By using supervised learning techniques on large banks of tagged human emotional speech examples, the reliance upon supplementary input from video data or human-dependent data labeling might be reduced or completely eliminated.

In a related vein, there has been extensive work on speech synthesis. Modeling speech synthesis as hidden Markov models and

*Conducted in Fall 2018 at Yale University, with Prof. Marynel Vazquez

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

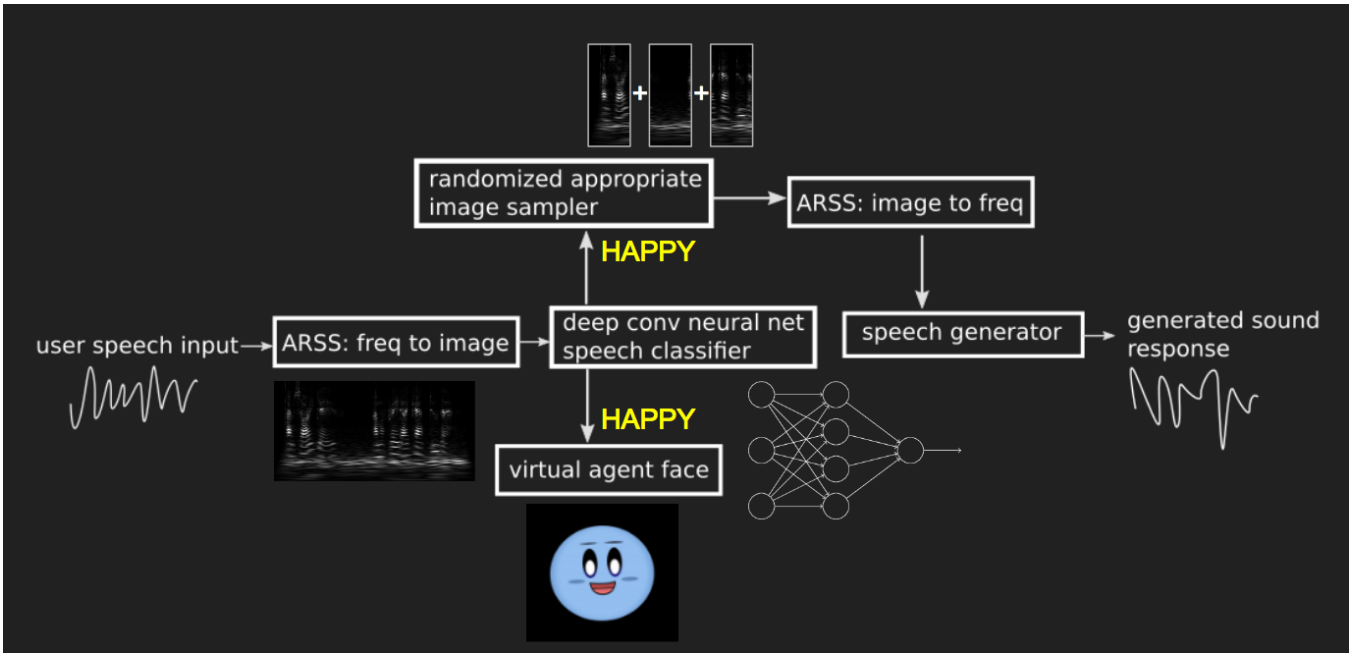

Figure 2: A user speaks, and that is changed via Fourier transform to a spacial image representation, which is then classified by a deep neural net as a designated emotion tag. The classification output, along with a sample of the user speech, is passed to an algorithm which generates a spacial domain sound response, which is then converted via inverse Fourier transform to a sound wave (audible speech). The classification output tag is also passed to the script rendering the virtual agent's face and mouth movements, informing its expression.

performing optimization-based dictionary searches [6], or generating complex phrases using deep neural nets [7] and recurrent neural networks [8] are compelling state of the art techniques. However, little to no work has sought to reconstruct the tone or emotion of synthesized voice in an intelligent, natural manner according to conversational context. Because of this, virtual agents have limited capacity to express human-like emotion or react to the perceived emotions in human partners, which diminishes their interactive efficacy. Human infants mimic their parents' speech, generating simple vocalized primitives (babbling) before they are capable of generating cogent sentences [9]. Yet it is obvious to humans when an infant is upset versus when they are happy. It is an exciting prospect to endow a virtual agent with a similar babbling -type capacity so that it can better express emotion to a user, despite the absence of intelligible language.

Herein, we propose a software architecture to enhance the emotional interactivity of virtual agents (Figure 11). We propose a deep feed-forward neural network which can process incoming human speech, and label it as HAPPY, SAD, NEUTRAL, or ANGRY. The network then passes the emotion label and samples of the user's speech to an algorithm which generates, in response, an audio waveform reflecting the expressed emotion using smoothed, concatenated snippets from a massive data set containing examples of human speech. Finally, the emotion is mirrored via a virtual face (*i.e.* if the user shouts angrily at the agent, it will show an angry face in return, and babble in an angry tone).

2 TECHNICAL APPROACH

2.1 Data collection and pre-processing

As with any supervised machine learning approach, a data set was needed for training data. Human speech datasets are common and readily available online, and some also include relevant labels categorizing the audio into different emotional expressions. The following datasets were considered as candidates for training our deep neural network for emotional dialogue classification:

- LDC2002S28 - Several hours of recorded speech data from various speakers tagged with 15 different emotions [14].
- SEMAINE - Several hours of recorded conversations with virtual agents, tagged with realtime emotional data consisting of several different features [13].
- RAVDESS - Recorded and labeled speech data from 24 professional actors, including emotion intensity [11].
- RML Emotion Database - Recorded speech and video of people in several languages, tagged with 7 different emotions [10].

We ultimately decided to utilize the RAVDESS dataset for the bulk of our training dataset, due to its wide range of voice types, high recording quality, and nicely categorized audio files. In addition to RAVDESS, we included the RML Emotion Database which was of lower audio quality, but helped to increase the variety in our training data and enhance the robustness of the neural network. We also added several minutes of our own voices as training data. From each of these datasets, we extracted the files labeled HAPPY,

Figure 3: Graphical representation of an audio file, a man speaking in a neutral voice. Generated using the ARSS software package.

SAD, NEUTRAL, and ANGRY. Finally, we included several minutes of microphone background noise (breathing, computer fans, silence, etc.) as a fifth sanity-check category, NOISE.

The key step to creating usable training data was to convert the existing audio files into a format that could be fed into a neural network, keeping in mind the primary goal of classifying emotion directly from an audio signal without introducing any supplementary data. We chose to encode our dataset speech samples into a frequency mapping, thereby preserving and exposing the inherent sound patterns in an array that that could be easily read by a machine learning system.

The results are audio snippets which have been transformed into 2D images using the Analysis Resynthesis Sound Spectrograph (ARSS) (http://arss.sourceforge.net/). This package converts WAV files into frequency data in one axis, and time in the other using a Fourier transform (Figure 3). The intensity of individual pixels represents the intensity of that audio frequency at distinct points in time. The resolution of the image is configurable, and can be changed to render higher- or lower-quality sound encoding.

Using FFMpeg (https://www.ffmpeg.org/), we removed any silence at the beginning and end of the audio files, then using SoX (http://sox.sourceforge.net/), we trimmed the files into 1-second snippets. Each of these snippets was fed into the ARSS software, generating an image (Figure 3). After testing several different resolutions and listening to the decoded audio, we settled on an image with 150 pixels in the time dimension, and 81 pixels in the frequency dimension (12 pixels per octave, from 80-8000 Hz). This level of resolution results in an audio encoding that is fairly representative of the original sound. The final, processed dataset includes approximately 2,200 images of size 81x150 pixels in grayscale. Each of these images is then sorted by label (HAPPY, SAD, ANGRY, NEUTRAL, and NOISE), and placed into an array to be used as training data, along with the appropriate 1-hot encoding for the target data.

2.2 Real-time neural network emotion classification

We used a deep feed-forward neural network to classify human speech, based on a face-detection image classifier. This 10-layer convolutional neural network was constructed using Tensorflow with Keras overlay in Python. The architecture consists of several

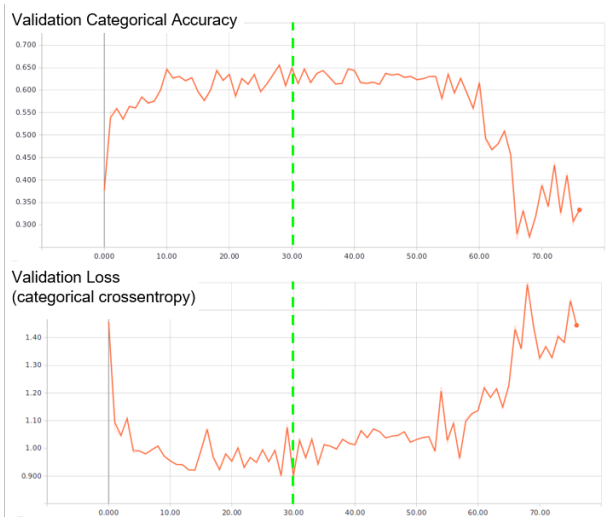

Figure 4: Validation Accuracy and Loss over 70 epochs of training. Minimum loss at 30 epochs: 0.88 Validation Loss, 65% Validation Accuracy. Accuracy on training data reaches 97% accuracy.

fully-connected dense layers, three dropout layers, and two max pooling layers. We used RelU activation functions in each dense layer except on the last two layers, which used sigmoid and softmax activation respectively. The architecture is pictured in Fig. 5.

During initial training steps (using only the RAVDESS dataset) our network quickly reached a categorical accuracy of 97% on the training data (Figure 4), with about 50% on the validation data, after approximately 15 minutes (7 epochs) of training on an Intel core i5-6300HQ (2.30 GHz) CPU. However, at this stage severe overfit became noticeable as validation accuracy began to worsen substantially while training accuracy continued to diminish. Testing against live microphone input resulted in all audio samples being categorized in an apparently random fashion whether the user was speaking or not.

At this point, we introduced the additional datasets (RML Emotional Database, our own voices, and microphone background noise), and the fifth category of NOISE. The neural network was also redesigned to include the several dropout layers mentioned above. This new solution again trained very quickly, but was able to avoid overfitting for much longer (30 epochs) with a categorical accuracy of 65%. Although the classifier is still prone to errors, this value is an enormous enhancement over random guessing (20% for a 5-category system). Testing with live microphone input showed much-improved results: silence was categorized consistently as NOISE, but spoken audio could be classified correctly with some repeatability.

The confusion matrix for our entire dataset is shown in Figure 6. As consistent with live experiments, most misclassifications tend to occur between ANGRY and HAPPY (both louder sound samples). It was also noted that in live testing, many samples are misclassified as SAD, which we suspect is due to the user not speaking loudly enough, as the SAD samples are often quiet and low-pitched.

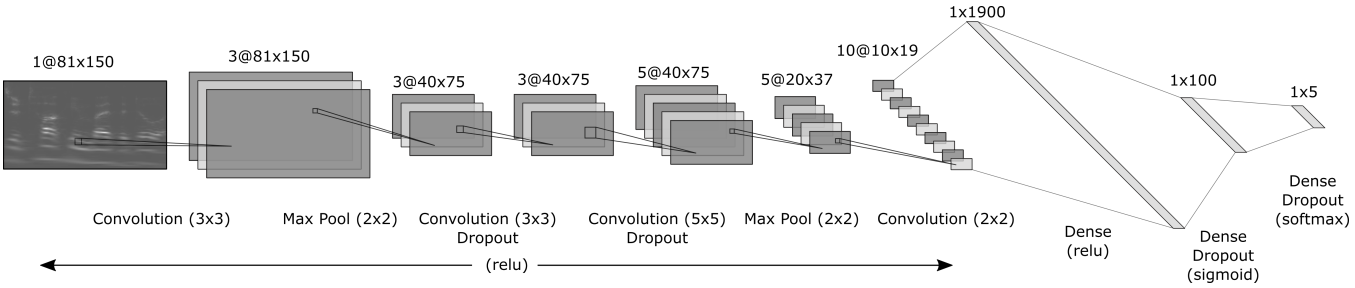

Figure 5: A user speaks, and that is changed via Fourier transform to a spacial image representation, which is then classified by a deep neural net as a designated emotion tag. The classification output, along with a sample of the user speech, is passed to an algorithm which generates a spacial domain sound response, which is then converted via inverse Fourier transform to a sound wave (audible speech). The classification output tag is also passed to the script rendering the virtual agent's face and mouth movements, informing its expression.

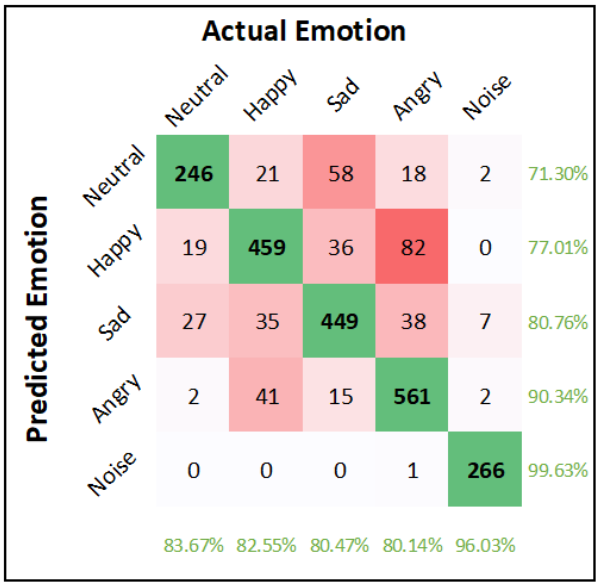

Figure 6: Confusion matrix of the classifier against all sample data. Darker red squares indicate higher misclassifications. Green percentages along the right and bottom edges indicate the percentage of correct classifications for each row and column. Note that the percentages here are rather higher than the validation accuracy. This is due to the inclusion of all training data, which will be biased toward accuracy in the training set.

To handle user input, a Python script continually recorded from the microphone in 1-second increments using the PyAudio package (https://people.csail.mit.edu/hubert/pyaudio/). As each recording was made, it was saved as a .wav file, overwriting the previous recording. The file was then analyzed using FFMPEG to determine the average audio volume in decibels. Any file below a certain thresholding value, chosen experimentally, is treated as total silence. The audio file was then immediately fed into ARSS via a bash script to be converted to frequency data in .bmp format, unless silence was detected, in which case a solid black (all-zero) image is produced.

A second, parallel Python script handles all other functionality. This script reads in any available .bmp files from ARSS and immediately classifies it using the trained neural network. As the user continues talking, the agent will track the number of times each emotion is detected. In this way, the agent is able to select the most common emotion to respond to. For example, the user may be speaking in a happy tone for 5 seconds, but the agent may misclassify one of the 1-second segments as ANGER. However, since the majority of the segments are correctly classified as HAPPY, the agent will determine an overall correct classification of HAPPY. This classification then drives the emotion of the virtual agent face and babbling speech generation. Once the script detects that the user has finished speaking (i.e., waits for an all-black "silent" image), the virtual agent will respond with an appropriate matching facial expression and babbling speech according to the detected emotion as detailed below in sections 2.3 and 2.4.

2.3 Generate waveform from learned speech patterns

Initially we intended to build a second network, in the form of a convolutional recurrent neural network (CRNN), which would generate a sequence of sounds based on the most recently heard speech from the user. After running into scaling problems–dynamically changing the network input size based on the user's duration of speech, and the correct thresholding of this speech–we opted to generate speech-like babbling sound that roughly corresponds to the emotional tone of the user by accessing snippets of pictorially represented sound directly from our entire sorted audio dataset. In particular, we pass the classification from the deep neural net to a script which concatenates 0.3 second snippets of spacial domain-transformed speech into sustained, 5 second phrases. Applying a small amount of Gaussian blur to these concatenated snippets ensures they blend seamlessly.

After we have aggregated a 5-second equivalent pictorial representation of speech, we decode it back into an audio waveform, using ARSS. The resulting WAV file is then output to the computer's

speakers using the pyaudio library. See Algorithm 1 for a more detailed procedure.

```
Data: emotionTag, userIsSpeaking, userWasSpeaking
Result: audioOutput
snipSize ← 0.3 seconds
babbleSize ← 5 seconds
if userWasSpeaking and not userIsSpeaking then
    emotionTag ← most frequent emotion detected
    if emotionTag != NOISE then
        selectedList ← all image files matching emotionTag
        while generatedSize < babbleSize do
            sample ← random image file from selectedList
            snippet ← random part of sample, size snipSize
            babbleSound ← babbleSound + snippet
            generatedSize ← generatedSize + snipSize
        end
        babbleSoundBlur ← GaussianBlur(babbleSound)
        audioOutput ← ConvertToAudio(babbleSoundBlur)
        saveAudioToDirectory(audioOutput)
        play audioOutput
    end
end
```

Algorithm 1: Babble generation

2.4 Virtual agent face rendering for real-time interaction

We used Adobe Character Animator CC and its face tracking utility to render a virtual agent with HAPPY, SAD, ANGRY, and NEUTRAL expressions. The character's features were custom-drawn. Eyebrow, face pose, mouth, and eye position were translated directly from a human actor performing the expressions.

A total of 8 distinct .mp4 clips, each approximately 30 seconds long, were saved into the local directory running the project. These consisted of each combination of emotion and a babble or no-babble condition (Figure 7). The babble conditions included minor alterations to the virtual agent's expression in addition to syllabic lip-sync to try and realistically convey word formation was linked to mouth movements. Using openCV and the imshow function, we were able to cycle through the virtual agent animations based on the classified emotion and talk state. The face was designed to update with each detection of a new emotion, sometimes resulting in frequent changes in expression as the user talks. However, once the user has stopped talking and the agent detects silence, it selects the emotion that was most frequently detected from the user's speech to determine the response.

3 EFFICACY ASSESSMENT OF VIRTUAL AGENT INTERACTION

We asked people to interact with our system to ascertain its efficacy in both correctly classifying a user's emotion based on their voice and reciprocating that emotion appropriately. 9 end-users, 4 female and 5 male with mean age of 26.33 (standard deviation of 2.5), participated in the study. We asked them to rank on a 5-point Likert scale the following questions:

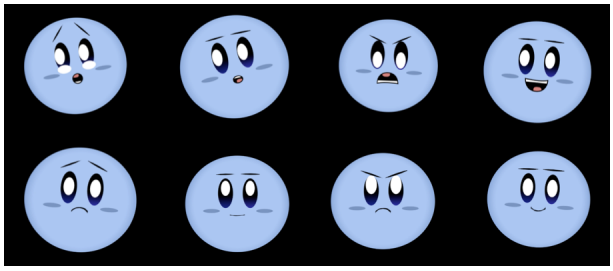

Figure 7: Examples of the expressions, both in talk state and non-talk state, we display to the user based on their classified emotion state. The top row are the talk state faces which are displayed to the user in tandem with the emotional babbling synthesis. The bottom row are the non talk state faces. We developed the virtual agent faces to exhibit dynamism such as blinking and sporadic mouth and head movement in all phases of expression with the goal of generating more empathetic, realistic user interaction.

(1) I felt that the virtual agent's face reflected the emotion I expressed in my voice
(2) I felt like the virtual agent's tone of voice reflected the emotion I expressed in my voice
(3) I felt like the virtual agent was responsive to me
(4) I felt like the virtual agent only talked when I wasn't talking
(5) The virtual agent was fun to talk to

The results to these questions are summarized in Figure 8. Evidently, the virtual agent appeared to be fun and engaging for all end-users. This is shown by the high average response to Q5 and the small confidence bars therein. We note also that the majority of participants thought that the virtual agent only talked when they did not and was quite responsive (Q4 and Q3). This finding testifies to the efficacy of the audio threshold algorithm in distinguishing noise from human voice. There is clearly room for improvement in both emotion recognition and speech synthesis, as average responses for Q2 and Q3 border on neutral to mild disagreement. A noteworthy point is that only two of the participants had little exposure to coding and/or computer science coursework. The remaining participants had all taken programming coures at some point in their education. In replicating this study, we will strive to incorporate individuals from different academic (as well as non-academic) background to gather a better impression of our system's utility in different social spaces.

4 DISCUSSION OF PERFORMANCE

We attribute the mediocre performance of the emotion recognition and corresponding virtual agent face update to foremost a non-optimal speech sample rate. As mentioned, we trained the emotion classifier neural network with snippets of audio data spanning 1 second, and when listening to a user, we also parsed audio input in 1 second increments to remain consistent with this input size. Although the 1 second segments made training the neural network feasible in a short period of time, in retrospect, we believe a shorter duration of speech must be analyzed to discern the emotion of a spoken phrase instead of learning to recognize particular words. By

training on smaller audio segments and sampling the user's speech more frequently, we believe that the neural network could extract emotion from the *sound* of the voice, as opposed to the content of the dialogue.

Along with this, some improved "smoothing" function could be applied to the emotion detection so as create a more consistent agent response. For example, often the system would detect happiness or anger at the beginning of a spoken phrase, but change to sadness by the end. Rapid back-and-forth switching of the virtual agent's expression suggests that our system was classifying intermittent intonations and inflections in user dialogue, rather than the overarching emotion. Considering the user's input across a larger time scale before changing the agent's emotive state would probably rectify this problem.

We ascribe the remainder of difficulty in the virtual agent's perception of human emotion to a variety of tonality and loudness. The training corpus for the emotion classifier consisted of audio samples at a normalized baseline decibel level. However, in practice, many end-users in our study had soft voices that were challenging for our system to interpret, let alone detect. It may be meaningful in future work to normalize user input in some way, thereby more closely approaching volume levels similar to the training data.

Moving forward, we also intend to address the shortcomings in emotional recognition by training our emotion classifier on more examples of tagged human speech. For example, the RAVDESS dataset is entirely composed of two spoken phrases, *"kids are talking by the door"* and *"dogs are sitting by the door"*. This may have had some overfitting effect on training since some phonemes occurred much more frequently than others. We also believe that additional training data recorded on a variety of microphone settings will enhance recognition across platforms. Among user trials, it was found that certain microphone settings and makes elicited better user-agent interaction experiences. This was a phenomenon we were explicitly trying to avoid, and we took precautions to diversify the training data. However, it is clear now, that we did not diversify it enough. Future work may involve duplicating the training information with various noisy elements overlaid to simulate different recording environments.

Lastly, common post-survey comments centered on the virtual agent's unusual voice: "It sounded like an alien," noted one participant. We intend to enhance the legitimacy of babbling by synthesizing noises with a CRNN **in situ** as originally planned, rather than relying on a body of strung-together snippets of audio. By synthesizing the voice directly from the user's speech input and generating a continuous sequence of babbling speech, we envision that our virtual agent will better reflect the user's own notion of emotion and tonality. If so, this approach will boost user ratings of Q2 (reflecting the tone of voice expressed by the user's voice). Additionally, we believe a CRNN will generate more natural-sounding babbling because it does not rely on manually programmed parameters (i.e.,we chose the duration of babbling and the duration of the snippets to be strung together).

5 CONCLUSION

We presented a virtual agent that interacts in a meaningful, emotionally-driven way by adapting its facial expression and generated babbling sounds to reflect a user's vocally expressed emotions. A deep convolutional neural network, receiving as input the user's speech transformed into a pictorial representation, was utilized to classify incoming emotions. An algorithm which drew on a database of sound files, concatenating and smoothing them, served to generate babbling sounds. A virtual agent cartoon with realistic facial motions and lip sync conveyed the audio and visual emotional responses. A thresholding algorithm distinguished when a user was or was not talking, as not to prevent the virtual agent from interrupting them mid-sentence. User trials illustrate that our system is engaging and enjoyable, but could be improved in terms of its ability to correctly place a wide range of user voices into the correct emotion category.

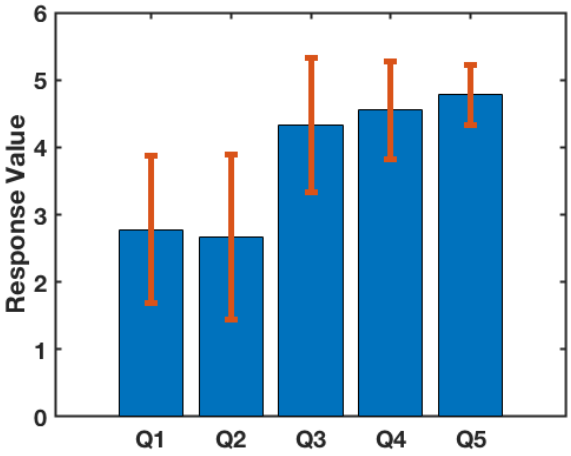

Figure 8: Statistical breakdown of user response to the 5-point Likert scale questionnaire. 5 = strongly agree. 1 = strongly disagree. The bars represent the mean response value, and the error bands show plus and minus 2 standard deviations from the mean.

REFERENCES

[1] G. Hinton et al., "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups," IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 82 to 97, Nov. 2012.
[2] A. Graves, A. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 2013.
[3] E. Cambria and B. White, "Jumping NLP Curves: A Review of Natural Language Processing Research [Review Article]," IEEE Computational Intelligence Magazine, vol. 9, no. 2, pp. 48 to 57, May 2014.
[4] R. J. Kreuz and R. M. Roberts, "Two Cues for Verbal Irony: Hyperbole and the Ironic Tone of Voice," Metaphor and Symbolic Activity, vol. 10, no. 1, pp. 21 to 31, Mar. 1995.
[5] Mortensen, C D. Communication Theory. New Brunswick, N.J: Transaction Publishers, 2008. Print.
[6] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.00CH37100), Istanbul, Turkey, 2000.
[7] H. Ze, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 2013.
[8] H. Zen and H. Sak, "Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis," in 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South

Brisbane, Queensland, Australia, 2015.
[9] J. F. Werker and R. C. Tees, "INFLUENCES ON INFANT SPEECH PROCESSING: Toward a New Synthesis," Annual Review of Psychology, vol. 50, no. 1, pp. 509 to 535, Feb. 1999.
[10] Xie, Zhibing, and Ling Guan. "Multimodal information fusion of audiovisual emotion recognition using novel information theoretic tools." In Multimedia and Expo (ICME), 2013 IEEE International Conference on, pp. 1 to 6. IEEE, 2013.
[11] Livingstone, Steven R., and Frank A. Russo. "The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A Dynamic, Multimodal Set of Facial and Vocal Expressions in North American English." Edited by Joseph Najbauer. PLOS ONE 13, no. 5
[12] Cowie, Roddy, Ellen Douglas-Cowie, Susie Savvidou*, Edelle McMahon, Martin Sawey, and Marc SchrÃűder. "'FEELTRACE': An instrument for recording perceived emotion in real time." In ISCA tutorial and research workshop (ITRW) on speech and emotion. 2000.
[13] McKeown, Gary, Michel Valstar, Roddy Cowie, Maja Pantic, and Marc Schroder. "The semaine database: Annotated multimodal records of emotionally colored conversations between a person and a limited agent." IEEE Transactions on Affective Computing 3, no. 1, 2012.
[14] Linguistic Data Consortium. Emotional prosody speech and transcripts. LDC Catalog No.: LDC2002S28, University of Pennsylvania. 2002.