Eemoji: integrating voice emotion into animated emojis and haptics

Ehsan Jahangirzadeh Soure

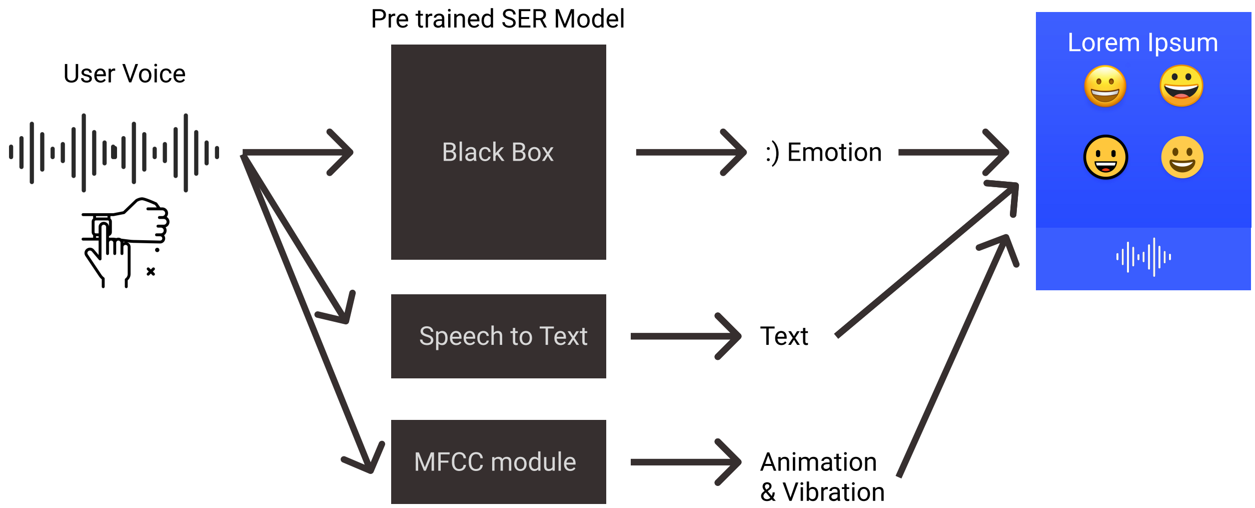

Fig. 1: Eemoji is an animated and vibrated emoji enriched text messaging for smartwatches to convey emotion captured and detected from the user input voice. The user voice will be transformed into text and emotion. The emotion will be used to recommend a set of emojis. The voice features will be used to animate the emoji and vibrate the smartwatch for an elevated experience of emotion-based emojis.

Abstract—Sending text messages and selecting emojis through smartwatches is both time-consuming and demanding as it requires the user to type on a keyboard on a small screen. Users have to interact with their smartwatch's small screen to write down a message with a topmost average of ten words per second. Besides, they have to spend a lot of time choosing from a huge list of emojis to make their message less vague. All these are even harder for visually impaired users. We propose an emotion enriched voice-first text messaging application called Eemoji. To minimize the interaction with the small screen and make the system more accessible, we propose using speech to text. Also, Eemoji utilizes Speech Emotion Recognition (SER) to detect the emotion from the voice, which removes all the extra hassle for selecting emojis through the keyboard. Finally, Eemoji integrates animation and vibration based on the user's voice features to elevate the emotion transformation through adding extra layers on top of static emojis.

Index Terms—Emoji, Emoticons, Emotion, Communication.

1 Introduction

Every day people are using different means of communication through their phones, computers, and smartwatches. A large portion of this communication happens through text [?]. Conveying emotion in text-based communication is a significant challenge because of the lack of nonverbal behaviors that are known as the main reflectors of human emotion [?]. To fix this challenge, emoticons were introduced in 1872, which were simply a group of characters representing an expression. Eventually, these emoticons were evolved into emojis created by Shigetaka Kurita for the first time in 1999 [?]. Emojis helped with a wide range of challenges in text-based communication, one of which is showing and conveying emotion in this means of communication. However, emojis are not effective in expressing deep emotions. They are mostly used to clarify ambiguities in the text, and sometimes they are misused in this concept. However, with all these drawbacks in mind, emojis are still the best solution that we have to convey emotion in text-based communication. Recently smartwatches are becoming a must-have gadget for most people, and more applications are being developed for them. They are not extensively used for messaging, but they can be used in certain situations for this purpose. The studies in this area are focused on improving the speed and accuracy of typing, but non is focused on how to improve emoji selection.

Humans know how to communicate their emotions through speech, which means they know how to convey emotion with voice and how to understand emotions through someone else's voice. Speech emotion recognition (SER) is the main process to extract human emotion or affection from the voice. Recently, it is gaining more and more popularity, and it is being used in different contexts like call centers for classification of customer satisfaction or cars to evaluate drivers' emotional state for further judgment to prevent accidents.

Ehsan Jahangirzadeh Soure is with the University of Waterloo, David R. Cheriton School of Computer Science. E-mail : ejahangi@uwaterloo.ca

Many studies were conducted on facilitating communication through emojis and other means (e.g., haptic). For example, Opico [?] is an emoji-first communication that enables messaging through series of emojis. Also, there are some tools like Face2Emoji [?] that help users convert facial expressions to emojis. Haptics is also studied previously. As an example, [?] focuses on supporting haptic communication in messaging. However, to the best of our knowledge, prior work has not been focused on detecting and embedding emotion from a voice into emojis. Some prior work is also focused on detecting emotions from speech signals with the help of machine learning and artificial intelligence [?, ?].Previous work tried to address text input in smartwatches. ETAO Keyboard [?] and SwipeRing [?] are two methods of typing on smartwatches. Also, different ways of typing on smartwatches while moving are studied in [?]. However, non of the previous work tried to embed emojis in text-based communication through smartwatches. So the main problem here is that we have different approaches in different domains to solve a specific part of our problem, which is conveying emotion through animated emojis and vibration in text-based communication. We want to improve the previous work and merge the solutions to elevate expressing emotion in day-to-day messaging.

We are introducing Eemoji, a smartwatch messaging tool based on SER and speech-to-text, which embeds the emotion into text without any interaction with a keyboard on the watch. Eemoji has three main components: a pre-trained SER model to recognize the emotion from the voice, a speech-to-text library to translate the user's voice into text, and an animation calculation module that generates numeric values based on the audio for animating the emojis and vibration of smartwatch upon receiving a message. And finally, the user interface is designed to represent all these features both on the side of sender and receiver. Different components of the system are designed to address all the aforementioned shortcomings. The speech-to-text components help with most challenges related to messaging on smartwatches, the SER module helps to embed the emotion into the communication without a need to interact with the watch, and finally, the animation and vibration module helps convey an extra layer of information for emotion through different channels.

Our contributions in this paper are in three ways. First of all, we are introducing a new context for and interaction with the SER model by using it in personal day-to-day messaging. Second, we are introducing a new way of embedding emotion into communication in smartwatches through a hands-free interaction. Finally, Eemoji's animation engine elevates conveying emotion with the help of animation and haptics. All in all, the final product of Eemoji is an emotion enriched text-based communication, with high accessibility for all smartwatch users.

2 Related Work

Our work is inspired and informed by related work in three areas: usability testing analysis, machine learning for user experience research, and collaborative visual analysis.

2.1 Emoticons and Emojis in Communication

Although emojis have been around for a long time, their meanings and how to use them or whether or not to use them is still ongoing research. Numerous studies have been done on the meaning of emojis and how to use them in communication. For example, EmoTag [?] focuses on the emotion-based analysis of emojis considering that they are gaining popularity more and more like a standalone language. Wiseman et al. in [?] talk about how people personalize emojis in their communication and how they use them to convey specific personal sentiments that they cannot tell through normal words. On top of all these works, Opico [?] investigates an emoji first communication through mobile phones, which shows how emojis are turning into a ubiquitous language.

On the other hand, a branch of the studies is focused on improving and elevating emojis. As an example, Face2Emoji [?] uses users' facial expressions to filter out the possible emojis for them. Communication through multi-modal signals (e.g., haptic experience) is also extensively studied. As an example, inTouch [?] presents a new approach for haptic-based interpersonal communication. However, to the best of our knowledge non of the previous utilized SER and audio signal analysis algorithms to filter out the emojis and generate animation and vibration based on the actual voice.

2.2 Speech Emotion Recognition

Speech emotion recognition (SER) is being integrated more and more into HCI. Control features have changed from simple text and display-based control to more complex features like voice, vision, and gestures [?]. Early research in this field focused only on simple linguistic features [?]. The human mind processes the other aspect like paralinguistic and phonetic data too. Recent work is more focused on deep learning methods. Abbaschian et al. review some of the recent work on SER that is based on deep learning [?]. Wani et al., in their systematic review of SER [?], mention that recent neural network and deep network models are surpassing the previous models in terms of accuracy, but still, the non-neural network models can be useful in devices with low processing power.

Many datasets in different languages are developed for speech emotion recognition. RAVDESS and SAVEE are datasets including both facial data and voice data. However, TESS is a dataset including only voice for emotional training. Datasets mentioned above are all recorded in English. However, there are many well-prepared datasets in other languages too, which are covered in [?].

Overall, there are a lot of studies on the machine learning side of SER, and it has been improved significantly recently. However, in terms of embedding into HCI, it can be used in many different contexts. To the best of our knowledge, we could not find any prior work embedding the SER into interpersonal communication, and Eemoji is focusing on making it happen.

2.3 Text input in Smartwatches

Smartwatches and wearable devices, in general, are gaining more and more popularity every day. They are used mostly to show simple information and communication with a more powerful device. Smartwatches specifically receive and show messages perfectly, but replying to a message through text entry is an unsolved challenge. Studies like ZoomBoard and DriftBoard [?, ?] focused introduce new types of text entry on ultra-small devices. ETAO [?] Keyboard is also another recent study proposing a new text input technique for smartwatches. However, all these interactions are still performing mediocre, and users can barely reach an average of ten words per minute. Considering the fact that wearable devices, especially smartwatches, are designed to be used more when the user is in the mobile state some of the research focuses on interactions in this specific scenario. Turner et al. cover some of the recent work in their paper [?]. Previous studies are focused on text entry, but none of them investigated how their techniques will perform in selecting emojis through the keyboard. Besides, to the best of our knowledge, no prior work tried to address emoji selection in the smartwatches. Emoji addresses this problem by recommending a set of emojis based on extracted emotion from the user's voice.

3 Design of Eemoji

Our main goal is to facilitate emotion-integrated interpersonal communication in smartwatches. Informed by previous studies and considerations for better accessibility we designed Eemoji with the design considerations listed and explained below.

3.1 Design Considerations

The design considerations discussed here make the outline for implementation of our tool.

D1: Voice over text for emotion, and text over voice for communication. Previous studies suggest that it is easy to infer emotion from text, but it does not represent the actual emotion. They further mention that the actual emotion can be extracted from the voice, and that is what humans do naturally. Therefore in our design, we should always try to capture and reproduce the emotion based on the voice signal, which is voice over text. Getting voice messages is not convenient for everyone. Voice messages are most convenient for the sender. On the side of the receiver, a voice message is not quickly scannable, and the situation and surroundings determine whether one can listen to them or needs specific devices like headphones. With this in mind, we say text over voice for communication.

D2: Prevent overchoice scenarios. As we discussed in the related works section, the research on text entry for smartwatches is still ongoing research. None of the approaches reached an acceptable words per minute average. Therefore, the designed tools should try to minimize the choices to maximize the area for each choice. Based on this rationale for emojis we have to propose a set of emojis instead of showing all of them. A recommendation system can rank all the possible emojis and users can select from those recommended sets. Also, choice overload can be a problem for users in terms of decision-making. Therefore the designed tool should try to minimize this problem to increase users' satisfaction in decision-making.

D3: Simple but accessible. Tigwell et al. in [?] talk about how technology introduces some barriers alongside its benefits for some people. The paper mentions how emoji descriptors can be a problem in communication for visually impaired people. Therefore, they recommend that new features should be implemented to lower this barrier. To reduce this barrier, we believe the interactions should be minimized. Also, the tool should minimize choices and selections as another important factor in improving the user experience for visually impaired users.

4 Eemoji System

Feature	Definition
Zero Crossing Rate	Rate of sign changes in a particular frame.
Energy	Sum of squares of the signal values.
Entropy of Energy	Measure of abrupt changes.
Spectral Centroid	Spectrum's center of gravity.
Spectral Spread	The second central moment of the spectrum.
Spectral Entropy	Measure of spectral power distribution.
Spectral Flux	Measure of power spectrum change rate.
Spectral Rolloff	The frequency below which a specified percentage of the total spectral energy is lied
MFCCs	Mel Frequency Cepstral Coefficients.
Chroma Deviation	The standard deviation of the 12 chroma coefficients.

Table 1: Possible features to be extracted from audio.

4.1 System Overview

We developed the Eemoji system based on the aforementioned design considerations. As shown in Fig. 1, Eemoji consists of a backend for processing and a frontend visual interface.

The backend consists of two main modules for extracting emotion and generating animation and vibration patterns from audio features. Both modules help with D2 and D3 by helping to filter the emojis based on the results. The SER module in the backend extracts emotion from feature which is aligned with D1 to put voice over text for emotion.

The frontend contains simply three different components at most. The tap to talk button, the actual message, and the suggested set of emojis. Based on D3 the interface is designed to be simple and accessible. Also, the speech-to-text happens on the user side and on the frontend. This module helps convert the voice into the actual text to make text over voice for messaging based on D1.

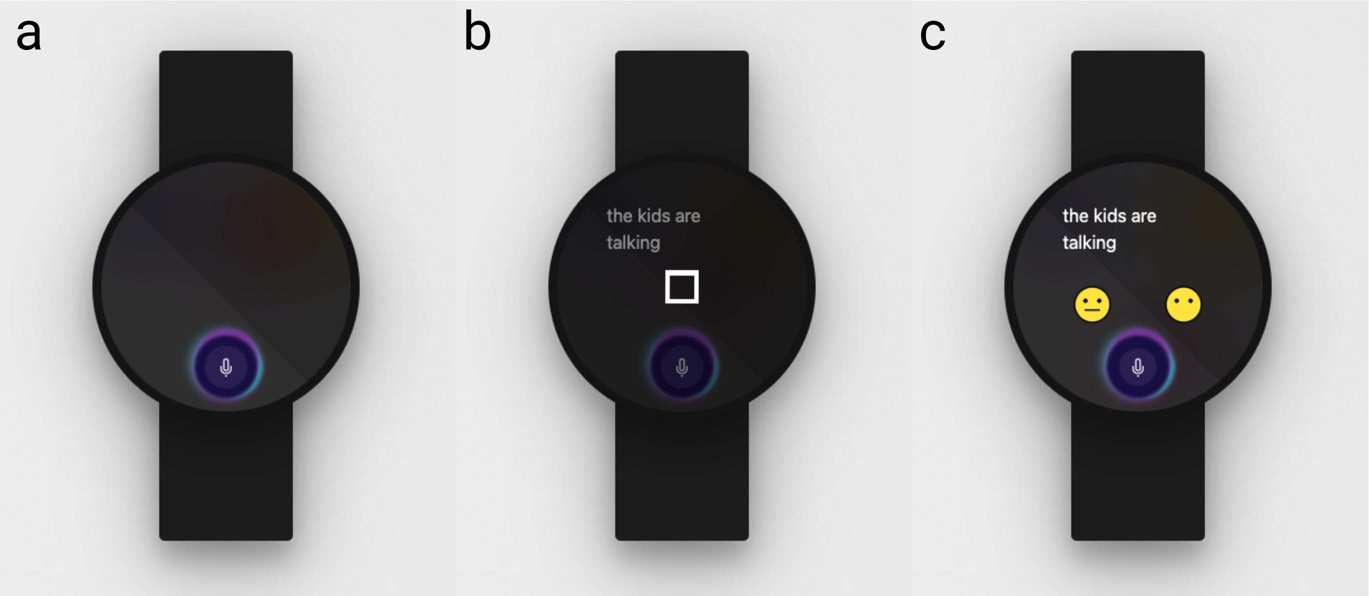

Fig. 2: The Eemoji user interface is simple and accessible. the whole process of sending a emotion-enriched message is: (a) Tap and hold the mic button and start talking. (b) The audio will be converted to text which will be shown on the screen and then the audio will be sent to the backend for further process. Meanwhile user will see a loading indicator. (c) Finally, the results will be shown on the screen as a set of emojis sorted by the intensity from left to right, and the emojis will be animated on the y-axis.

4.2 Speech Emotion Recognition

Eemoji uses a pertained SER model [?] for emotion recognition.

Every SER model has four main components. A database of emotional audios, which this module uses a combination of datasets to decrease the chance of overfitting and also to cover more emotions. Our SER model uses CREMA-D, RAVDESS, SAVEE, and TESS. To use all these datasets, it preprocesses all the data beforehand to blend them in one dataframe to train the model afterward. All the datasets cover eight emotions (neutral, calm, happy, sad, angry, fear, disgust, and surprise).

To further generalize the model, data augmentation is applied to the audio data. This approach helps generate new data points, which help in generalizing the model and preventing the effects of anxiety or any other external event from impacting the results of the model. In the model that we are using noise injection, stretching, and pitching are used for data augmentation.

The next step is feature extraction. This would be covered more in the next section as we need it not only for the SER model but also for our animation and vibration generation engine. However, in general, there are features that can be extracted from the audio, and we covered some of them in Table 1. The highlighted features in the table are the ones used in the model code.

The model is implemented as a convolutional neural network. However, as mentioned it is a pre-trained model and it is not the main contribution of this project, so we are not going to cover the model in detail. The model is using a combination of convolutional layers, max-pooling layers, dropout layers, flatten layers, and dense layers. The pre-trained model has a 61% accuracy which is decent considering that it is for overall eight emotions.

The javascript backend calls the python script for the SER model and passes the information for the audio. The python script extracts the features and then sends the features to the model for emotional evaluation. The results would be sent back to the javascript backend to be further processed and sent back to the frontend.

4.3 Audio Analysis and Feature Extraction

For the feature extraction module, we have a python script that will receive the audio file and then process it. Our script has two main functions. One of the functions will extract features based on the highlighted features in Table 1 and then send them to the SER model. The second function would focus on the MFCCs. It would generate the coefficients from the audio and then produce a 1D array of numbers based on the average of each array of coefficients. Finally, those numbers will be sent back to the frontend for animation and vibration generation. In what follows we will dive deeper into MFCC and why we used it for the animation generation purpose.

4.3.1 MFCC

Mel Frequency Cepstral Coefficients (MFCCs) is a way of extracting features from audio. It is based on the MEL scale. The process starts by dividing the frequency band into sub-bands and then calculating the coefficients using the Discrete Cosine Transform (DCT). Now the question is, why are we focusing on this specific algorithm? The answer is simple MEL scale is similar to the way humans perceive voices and their frequencies which makes it useful in most audio-based analyses.

Humans do not perceive voice frequencies on a linear scale, which means they can detect differences in low frequencies a lot easier than the higher frequencies. The difference between 100Hz and 500Hz is obvious for humans, but it is not valid for 15KHz and 15.5KHz.

With this in mind, Stevens, Volkmann, and Newmann [?] proposed the MEL scale. This scale is based on how the listener perceives different pitches in to be equal distance from one another.The MEL scale is widely used for feature extraction, and many of the libraries like librosa have functions for it implemented in them. After calculating the coefficients the result is a 2D array of twenty arrays of signed numbers. We calculate the average for each of these arrays to finally reach a 1D array. The numbers in this array then will be used in the frontend for animating the emojis.

4.4 Eemoji User Interface

Eemoji is has a very simple UI with one tap and hold button initially aiming to maximize the accessibility.

Eemoji has a very simple UI with one tap and hold button initially aiming to maximize accessibility. The UI will only have three components which are the main components, and nothing extra will be shown to the user based on (D3). The actual interface is not implemented on a smartwatch, but as a proof of concept, we designed and implemented a smartwatch on the web so that readers can better understand how Eemoji will work in action. However, because we do not have a smartwatch, we are also missing the vibration engines, so the user interface cannot mimic the vibrations.

When a user starts using the application, a tap and hold button will be the first thing the user will see. This button is designed based on most voice assistants' and messengers' voice message buttons. They all have the same hold and talk interaction, which is familiar for users. Then the user can start talking, and the voice will be captured and saved as a blob of information on the browser. Meanwhile, the audio will be sent to the react-speech-recognition library to be converted to text. The temporarily recognized text will be shown to the user as most voice assistants do.

When the text is finalized and the user releases the button, the audio blob will be sent to the backend to be processed. The SER will return an emotion keyword, and the animation generation engine will send an array of numbers based on the audio sent to the backend. The received emotion will be turned into a set of emojis. These emojis are preselected based on searching for emotion keywords on the openmoji.org website. The mapping of keywords to emojis can be seen in figure 3. The emojis are ranked based on the exaggeration of the emotion from left to right. Therefore, users can select the exaggeration level by tapping on the emojis.

The emojis will then be shown on the screen, and they will be animated based on the numbers generated by the animation engine in the backend. Eemoji only supports animation on the y-axis, for now, so the animations look simple. To make the animations smooth and natural we have used react-spring, which is a library based on spring physics simulation.

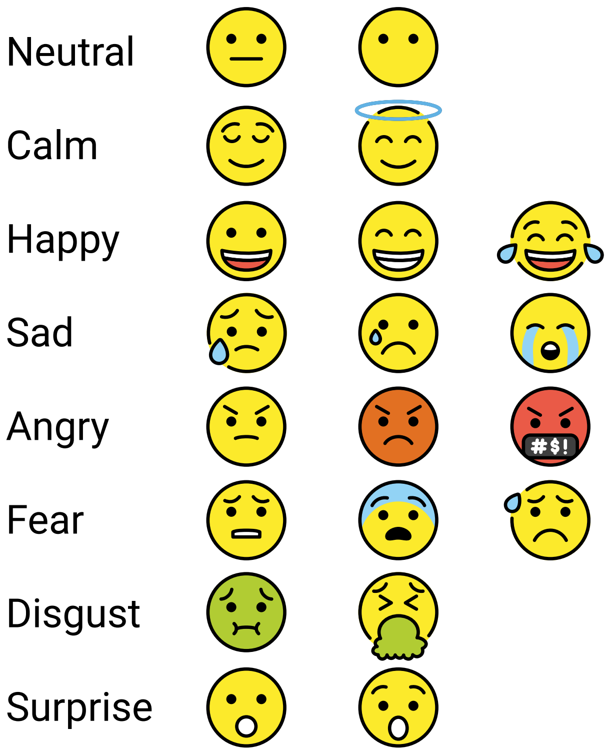

Fig. 3: Emojis used in the system and their mapping to emotions detected by SER.

5 Discussion

In this section, we point out some design implications obtained from the study and potential future directions.

5.1 Limitations

In this section we will go through some of the limitations in our work, and will propose possible improvements on these limitations.

User Study. As an HCI research project, there is a need for human participants to evaluate and validate the hypothesis. In this project, we did not run a user study, so it is just ideation about all these features and interactions. To further validate our hypothesis and find out a clear answer to our research question, a user study should be conducted. In the study, our tool should be compared to a baseline tool which will be texting using the keyboard or using any of the previously mentioned works. It should be a between-subject study to measure users' satisfaction and perceived usefulness.

Vibration. Another feature that we discussed in this project is vibration, which is not implemented in our tool due to not having access to actual smartwatches. However, it is a big limitation of our work as we assume that the animation and vibration generated by the tool are based on how humans perceive voice, and it should trigger the same feeling for users through the vibration on their wrist. Besides, the vibration is not a visual aspect, and it is not completely controllable. Considering its mechanical limitations, we are not sure to what extent this vibration will be felt by the users.

Intensity of emotion. We mentioned that the user interface should be simple. Our D3 talks about why we believe we need a simple UI to achieve higher accessibility. However, in our interface, we show two or three emojis for each emotion, and each emoji represents the intensity of that emotion. However, this is against our accessibility design choice. To fix this, we have to make a final suggestion and show only one emoji. This means we should also filter the emojis based on their intensity. It is not an easy task considering that it is not obvious how the voice signal and paralinguistic features of human speech change to show the intensity. Also, this is relative and subjective, which means no specific threshold can be used to categorize the intensities. Possibly some reinforcement learning can be applied to handle this problem.

SER Model. We used a pre-trained SER model in our project. This means we are tied to this model's accuracy and shortcomings. The model focuses on eight emotions, which are considered too many. Therefore the accuracy drops, and we can say it is not necessary to validate the fact that our tool is useful. The high accuracy models in this area are all implemented as neural networks and deep networks that need high computing power and GPUs. In this project, we are using a laptop to represent the actual smartwatch, but in the real world, there would not be the same amount of computing resources. Putting the computation in the backend and in another server is also against accessibility considering that not always users have access to high-speed networks. Thus, SER should be implemented as an SVM for better performance with lower computing power.

5.2 Future Work

In this section, we are going to talk about some of the future work that can improve the Eemoji and is needed to validate some of the hypotheses we mentioned in the paper.

Complex Animations and Vibration. Eemoji supports a simple animation on the y-axis right now. There is no specific rationale on why we made emojis animate on the y-axis and not the x-axis. Future work should investigate the effects of different animations and whether they have any effects on the perceived emotion by the end-user or not. The same applies to vibration. Further studies should be conducted on how people feel emotion through the vibration on their skin. Also, the animation can be even more complex by adding zooming or even morphing effects to change from one specific emoji to another one representing the emotional shifting in a sentence.

Animating the Text. Animating emojis sounds like a good idea, and it may add an extra layer of information to the raw emoji. However, the emotion is actually in the pronunciation of the words. One may put stress on some words or say some words in a specific way. These are all subjective and can change from person to person. Therefore, they are unique, and they can be learned in interpersonal communication by both ends of the communication. As a future direction, we believe animating the words generated by speech to text library can show how the feeling is changing and how the other person is putting stress on some words. This can go even one level deeper and be implemented on a character level. The same considerations that we mentioned for more complex animations apply here too. There is a need for an animation framework that suggests the best matching animation for each emotion.

NLP Emotion Detection for Sarcastic Emojis. Our tool can detect eight emotions at the moment, and it can suggest a set of emojis based on the detected emotions. However, the are so many more emojis, and most of them do not just show the actual emotion and instead represent a sarcastic feeling. To achieve this, future work should add a natural language emotion detection module alongside the speech emotion recognition module. If both emotion detection modules detect the same emotion, then we have more confidence over our results and the detected emotion, and if the detected emotions differ, then we can just translate it to a sarcastic emoji. For example, if our natural language emotion recognition module detects ”sad,” our SER also detects ”sad,” then we will show a sad emoji with even higher confidence, but if one detects ”happy” and the other detects ”sad,” then we will show the sarcastic emoji laughing and crying. The natural language emotion detection can also be useful in intensity recommendation, considering words can be ranked based on their intensity easier than the audio signal.

Embed to Voice Assistants. Our D3 talks about accessibility. Voice commands can always be helpful, and voice assistants are gaining more and more popularity. It is easy to use when one cannot interact physically with the device. Considering that, another possible future direction is embedding Eemoji into voice assistants. The represented Eemoji is a system in smartwatches, but it can be more accessible if one can talk to a voice assistant and ask them to generate a text message, and the assistant can embed the emotion into the text without any extra commands.

6 Conclusion

Informed by the literature, we designed and implemented an emotion-enriched text messaging app, Eemoji, to support texting through small devices like smartwatches without losing the emoji selection. Eemoji extracts emotion from users' voices using a pre-trained AI model and includes a simple user interface to make texting and emoji selection easier and more accessible through smartwatches. In sum, our work has taken the first step to create an emotion enriched text messaging to support people using small gadgets like smartwatches daily and the visually impaired people to be able to text without a need for any text entry.

Acknowledgments

I could not find any type of formatting to cite the datasets, and I did check some of the papers that I had to see how they cite them. However, non of them did cite them, so I believe writing down the name should be enough.