October 2019

MEASURING AND VISUALIZING USER SENTIMENT CHANGES OVER MULTIPLE CHANNELS

Xinjun Zhang

Qihong Shao

Changyong Zhao

Antonio Nucci

MEASURING AND VISUALIZING USER SENTIMENT CHANGES OVER MULTIPLE CHANNELS

AUTHORS:
Xinjun Zhang
Qihong Shao
Changyong Zhao
Antonio Nucci

ABSTRACT

Techniques are provided to simultaneously infer approximately where a speaker is looking and the speaker's emotion during a conversation. Due to privacy concerns, only the speaker's approximate facial features may be estimated. The inferred face may be converted into a cartoon face that retains the main facial features. This may enhance user interaction experience even when a speaker does not turn on video in teleconference.

DETAILED DESCRIPTION

User sentiment over a product is highly correlated with that user's churn probability. To reduce user churn, a company may attempt to monitor user sentiment. This may allow the company to increase user satisfaction by providing a better service, for example.

User sentiment information can be embedded in multiple data sources (e.g., user blog or forum, voice volume or tone changes in online conferences, changes in facial expression in video meetings, etc.). Existing sentiment analysis is only focused on one independent perspective (e.g., text information in forums, blogs, etc.) or in speech recognition (e.g., tone changes, etc.). In very rare cases, those system can operate in real time, but can generally only work with offline data collected over a period of time. Meanwhile, there is a strong correlation among different information channels (e.g., text, speech, video, etc.). For example, when a user becomes angry or feels dissatisfied, the user may write negative feedback and express an angry face during an online video meeting while speaking with a high-pitched voice.

A system is described herein which enables a comprehensive understanding about user sentiment and dynamics over time by integrating data from multiple channels, including text, video and audio sources. Moreover, for audio input scenarios, a live cartoon face may be produced based on sentiment score and voice features using a deep learning

Published by Technical Disclosure Commons, 2019

framework. The purpose of the cartoon face is to mimic user appearance to enhance the experience and make conversation more interactive and enjoyable.

To efficiently integrate these information channels, user emotion is precisely estimated through multiple information channels. For example, the user may make a comment or discuss a product on social media, or communicate with a support engine to resolve a problem. Customer service experience for online video conferencing may also be enhanced when users choose to talk over audio rather than video. This may be accomplished by approximating user appearance as a cartoon face, thereby simulating face-to-face communication.

Figure 1 below illustrates an example sentiment engine configured to integrate information from multiple input channels, such as text, video, and audio.

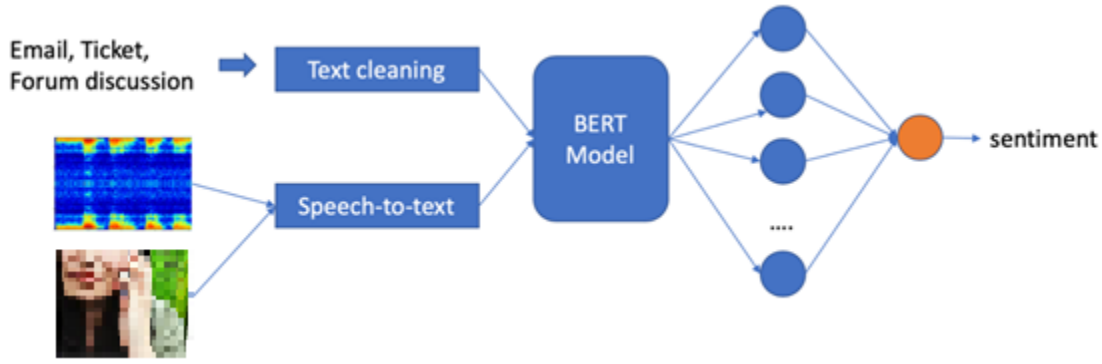

Figure 1

For ordinary text input, the system may apply a text cleaning process and feed the cleaned text directly into a Bidirectional Encoder Representations from Transformers (BERT) model after text encoding. The cleaning process includes removing system log, error information, and other non-free text. Only the sentences that appear to be natural human language are retained. For audio input, the speech wave data is converted to a spectrogram data format for translation from speech to text. For video input, the audio track may be extracted and processed through a similar procedure as the audio input.

When listening to a user speaking on the phone or in an online audio conference without seeing the user's face, the listener often tries to guess at the user's appearance and emotional state and construct a corresponding mental model. It is very important for customer support to understand user intent and emotion changes in order to make communication more effective and efficient. In fact, due to the mechanism of speech

production, there is a strong connection between user voice, appearance, and sentiment changes. User gender, age, ethnicity, and shape of mouth may all impact voice generation, as may user speech (e.g., language, accent, speed, pronunciation, etc.). Voice change is a strong indicator of sentiment changes (e.g., when a user becomes mad, voice pitch grows higher).

The system described herein may infer user appearance and sentiment changes from user speech. An image of the face may be reconstructed canonically from an audio input of an online conference, with real time sentiment changes. Figure 2 below illustrates an overview of an example method for determining face and emotional changes based on voice.

Figure 2

Rather than producing an exact image of the user, the system may recover characteristic physical features that are correlated with user speech as well as user sentiment changes in the conversation in near real time.

The method may be trained in a self-supervised manner using the natural co-occurrence of speech and faces in videos, without requiring additional information such as human annotation.

Figure 3 below illustrates an example system architecture and components for converting audio/video input to a cartoon face.

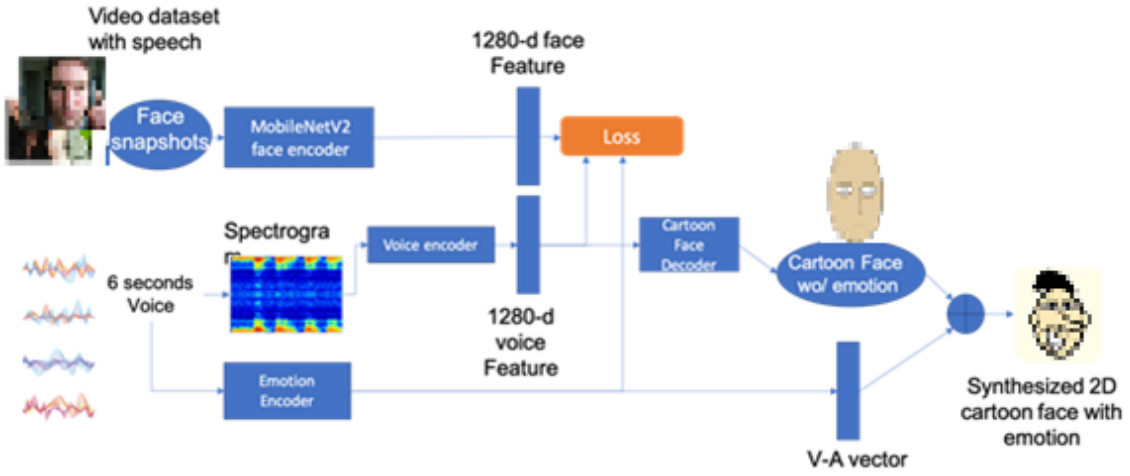

Figure 3

The system may include three main components: a voice encoder, a face decoder, and an emotional change component. The voice encoder takes a complex spectrogram of speech as input, and predicts a low-dimensional face feature that would correspond to the associated face. The face decoder takes as input the face feature and produces an image of the face in a canonical form (e.g., front-facing and with neutral expression). The emotional change component causes the model to automatically generate the next episode and provide near real time sentiment change signals. Each episode may be generated periodically (e.g., every minute) or in any other manner.

The voice encoder module uses a Convolutional Neural Network (CNN) to convert a short input of speech into a vector, and then feeds into the face decoder to reconstruct the cartoon face image. As illustrated in the voice encoder network of Figure 4 below, the CNN may include a convolutional layer, a Rectified Linear Unit (ReLU) layer, a maximum pooling layer, and a batch normalization layer.

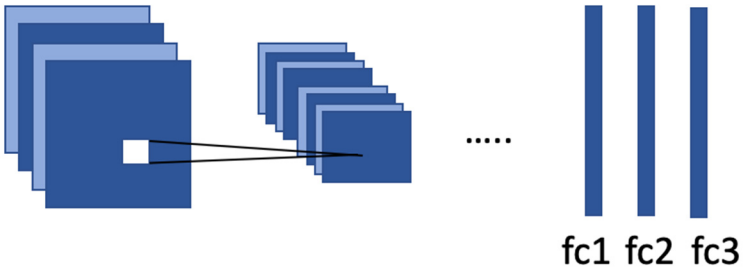

Figure 4

The blocks of the convolution layer, ReLU layer, and batch normalization layer may alternate with the maximum pooling layer(s), which pool along only the temporal dimension of the spectrograms while leaving the frequency information that was carried over. This is to preserve vocal characteristics, which are better contained in the frequency content, whereas linguistic information usually spans a longer time duration.

At the end of these blocks, an average pooling may be applied along the temporal dimension. This enables efficient aggregation of information over time and makes the model applicable to input speech of varying duration.

The pooled features are then fed into two fully connected layers to produce a 1280 dimensional face feature.

A deep neural network model takes the complex spectrogram of a short speech segment as input and predicts a feature vector representing the face.

The face decoder reconstructs the image of a face from a low-dimensional face feature. Any irrelevant variations (e.g., pose, lighting, etc.) may be filtered out while preserving the facial attributes.

The model may be trained with face features extracted from a fast pre-trained face recognition network architecture that enables faster real time processing. The model may be trained separately and kept fixed during the voice encoder training.

The loss function may be based on the L1 distance and/or additional loss terms. For example, the difference in the activation of the last layer of the face encoder may be additionally penalized. Both the predictions and the ground truth face features may be fed into these layers to calculate the losses.

The final loss is:

$$Loss = \left\| f_{MN}(V_f) - f_{MN}(V_S) \right\|_1 + \lambda_1 \left\| \frac{V_f}{\|V_f\|} - \frac{V_S}{\|V_S\|} \right\|_2^2 + \lambda_2 L_{distill}\left(f_{MN}(V_f),\, f_{MN}(V_S)\right) - \lambda_3 \sum_{i=1}^{m} y_i \log^{p_i}$$

f_{MN} is the first layer of the face decoder which connects with the output of a 1280 dimensional voice encoder and converts it to the dimensionality required by the face decoder network. V_f and V_S are the 1280 dimension face feature and 1280 dimension voice feature, respectively. $L_{distill}$ is the knowledge distill loss between two entities, and is

defined as: $L_{distill} = -\sum_i p_{(i)}(a) log p_{(i)}(b)$, and $p_{(i)}(a) = \exp(a_i/T) \Big/ \sum_j \exp(a_j/T)$, where T is the hyperparameter. $\lambda_1, \lambda_2, and\ \lambda_3$ are the coefficients that weight the loss from three factors. The fourth loss factor is the penalty function for misclassified sentiment, $y_{(i)}$ is the true label, and $p_{(i)}$ is the predicted probability for that sample i.

During training, the face decoder is fixed, and only the voice encoder that predicts the face feature is trained. The voice encoder may be trained to reduce the loss between the face encoder layer output.

The generative deep learning model described herein may reconstruct customer appearance from their voice in real time, based on a light and fast deep learning framework. Most existing systems are for research purposes only, are not real time, have slow response times, and cannot be leveraged in industry systems.

Furthermore, a continuously synchronized cartoon face may be updated by detecting sentiment changes in addition to tone and pitch changes. Existing systems can only provide a static emotionless facial profile.

The deep learning framework described herein may significantly increase processing capacity (e.g., by a factor of thirty).

The system may also convert voice to visible appearance and dynamic emotions which can benefit broad scenarios (e.g., improving customer support experience, protecting customer privacy by generating only a cartoon face, etc.).

In summary, techniques are provided to simultaneously infer approximately where a speaker is looking and the speaker's emotion during a conversation. Due to privacy concerns, only the speaker's approximate facial features may be estimated. The inferred face may be converted into a cartoon face that retains the main facial features. This may enhance user interaction experience even when a speaker does not turn on video in teleconference.